

As per the latest curriculum prepared and recommended by the  
Council for the Indian School Certificate Examinations, New Delhi

**Frank**

**EMU BOOKS**

# Computer Applications

## Solution (ICSE)

Complete solution for students  
at [www.frankedu.com/students-zone](http://www.frankedu.com/students-zone)

FRANK EDUCATIONAL AIDS

X

# Chapter 1: Concept of Objects

## Practice Questions

### ----->Objective-Type Questions<-----

#### A. Choose the correct answer:

1. An object is represented by two attributes, out of which one is characteristics and the other one is \_\_\_\_\_.

- a) Behaviour      b) Situation      c) Abstraction      d) Encapsulation

**Ans. a) Behaviour**

2. Name the programming technique that implements programs as an organized collection of interactive objects.

- a) Procedural Programming      b) Modular Programming  
c) Object-Oriented Programming      d) None of these

**Ans. c) Object-Oriented Programming**

3. Name the characteristics of Object-Oriented Programming that hides the complexity and provides a simple interface.

- a) Encapsulation      b) Polymorphism      c) Abstraction      d) Inheritance

**Ans. c) Abstraction**

4. What is the behaviour aspect of an object represented by?

- a) Functions      b) Data Members      c) Both a and b      d) None of these

**Ans. a) Functions**

5. What is the ability of an object to have many methods which functions differently but have the same name?

- a) Polymorphism      b) Encapsulation      c) Abstraction      d) Inheritance

**Ans. a) Polymorphism**

6. Name the object-oriented technique that encompasses the parent class state and behaviours into its child.

- a) Polymorphism      b) Encapsulation      c) Abstraction      d) Inheritance

**Ans. d) Inheritance**

7. Name the art of implementing encapsulation in Object-Oriented Programming.

- a) Polymorphism      b) Encapsulation      c) Abstraction      d) Class

**Ans. d) class**

8. What is meant by state of an object?

- a) Functions of the object      b) Data members of the object  
c) Content of an object      d) All of these

**Ans. c) Content of an object**

9. Name the term used to express the ability of objects to interact with each other.

- a) Message Passing      b) Instantiation      c) Logical construct      d) All of these

**Ans. a) Message Passing**

#### B. Fill in the blanks:

1. An object is an identity with certain characteristics and behaviour.

2. The values/attributes of the characteristics of an object are called the state of an object.

3. All the complexities of a program should be hidden in such a way that abstraction is obtained.
4. Class is a mechanism to implement encapsulation.
5. Objects interact with each other with the help of Message Passing.
6. A huge program divided into several individual segments is called functions/methods.
7. The objects methods/functions are said to access data.
8. A class is a logical construct and an object is a physical reality.
9. All objects have identity and are distinguishable even if the constituent components are same.

----->**Subjective-Type Questions**<-----

**Answer the following questions:**

**1. Give two examples of real-world objects. Also specify their characteristics and behaviour.**

**Ans.** Real Life Objects

Example1:

Object Name: Pen

Characteristics: Company made, Colour, Body Shape and Nip.

Behaviour:

- 1) Used for writing.
- 2) Used for correction
- 3) Used for drawing or sketching

Example 2:

Object Name: Dog

Characteristics: Name, Colour and Breed.

Behaviour:

- 1) Barking
- 2) Wagging its tail
- 3) Running speed

**2. What do you understand by state of an object? Explain with an example.**

**Ans.** State of an object refers to the condition in which an object is in. Thus the values/attributes of its characteristics is represented. For example, a book is an object, which may be either in “open” or “closed” state, similarly a bulb may be in switched on or switched off state.

**3. How are objects implemented in Software?**

**Ans.** Every object in real life has certain characteristics and behavior. It is the characteristic that is represented by data variables and behaviour by methods or functions encapsulated in a class using a software.

**4. What is abstraction? How is encapsulation related to it?**

**Ans.** Abstraction is the process of hiding the complexity and giving a simple interface and encapsulation is the process of implementing abstraction.

Using encapsulation, the data and the code is wrapped round inside a class.

**5. Define encapsulation.**

[ICSE 2006]

**Ans.** Encapsulation is the process of wrapping/combining the characteristics and behavior into a single unit of an object to form a unique identity.

**6. Explain the term object using an example.**

[ICSE 2006]

**Ans.** An object is an entity with certain characteristics and behaviour making it having its own individuality and distinguishable. For example a pen has certain characteristics, for example its colour, shape, size, etc. and behaviour which is used to write, sketch or draw, etc. Even though two pens are identical with the same colour, texture, shape and size, yet they are two distinguishable pens.

**7. What is a class?**

**Ans.** A class is a method to implement encapsulation in programming. It is the class keyword in Java that is used to wrap around the data members and member methods into a single unit. It acts as a template that allows instances of it to be created, which are called objects.

**8. What is Message Passing?**

**Ans.** When the objects need to interact with one another, they pass or request information to or from one another. This interaction is known as message passing.

**9. What is Object-Oriented Programming?**

**Ans.** Object Oriented Programming is a technique of implementing programs which are organized as a co-interactive collection of objects, each of which represents an instance of a class.

**10. State the 4 characteristics/principles of Object-Oriented Programming.** [ICSE 2005]

**Ans.** The 4 characteristics/principles of Object Oriented Programming are:

- i) Abstraction
- ii) Encapsulation
- iii) Polymorphism
- iv) Inheritance

**11. Explain briefly:**

**a) Abstraction**

**b) Inheritance**

**c) Polymorphism**

**Ans. a) Abstraction:** Abstraction refers to the art of hiding the complexities and giving a simple interface. For example in a car, for a layman, it is good enough to know driving. Hardly will he know the intricacies of movement of the engine, the working of the electrical and electronic components; what he will know or will be interested with is the steering for turning, the accelerator for speed and the switches. This is because he has been given a simple interface and the complexity of the car engine movement has been kept completely hidden from him. Thus he doesn't bother about the intricacies of the car's engine movement, but concentrates only on driving.

**b) Inheritance:** Inheritance is the ability to inherit characteristics from another object. In more concrete terms, an object is able to pass on its state and behaviours to its children. For inheritance to work, the objects need to have characteristics in common with each other. It is basically a term that is used to represent hierarchical relationship of generalization.

**c) Polymorphism:** The term Polymorphism is actually a combination of two terms Poly and Morphism. Here the term Poly means many and Morphism means forms. It is the ability of objects to have many methods of the same name, but each one responds to different types of specific behaviour as they have different forms.

**12. How does class acts as a blueprint for an object?**

**Ans.** A class acts as a blueprint or template that defines certain characteristics and behaviour and an object is an instance or copy of a class.

**13. How does encapsulation acts as a protective layer for an object?**

**Ans.** Encapsulation is a technique that binds together function and the data into a single unit. You can imagine it to be as a protective wrapper that prevents the code and data from being accessed by other codes defined outside the wrapper.

**14. How does modularity helps in reducing the complexity of a program?**

**Ans.** Modularity helps in breaking a huge task into smaller units called functions for easy maintainability and reusability thus reducing the complexity of the task at large.

**15. What do you understand by data abstraction? Explain with an example. [ICSE 2006 & 2010]**

**Ans.** Data abstraction refers to, providing only essential information to the outside world and hiding their background details, i.e., to represent the needed information in program without presenting the data.

Let's take one real life example of a TV, which you can turn on and off, change the channel, adjust the volume, and add external components such as speakers, VCRs, and DVD players, BUT you do not know its internal details, that is, you do not know how it receives signals over the air or through a cable, how it translates them, and finally displays them on the screen.

**16. Give a real world explanation about abstraction.**

**Ans.** A switch board is an example of abstraction. Imagine if the concept of switchboard did not exist. You would find a great number of wires hanging round your room. To start with an appliance, you needed to join two wires. But essentially you need to know which group of wires should be joined to start an appliance. Moreover, more the number of wires, more will it be confusing to identify the wires to start an appliance. Thus joining of wires is not only confusing but may also prove fatal as far as safety is concerned. Thus, the electrician installs a switchboard that connects each of the wires to a switch. Hence, it is just enough to know for the user to switch on that switch that starts an appliance.

**17. Why is abstraction often referred to be as relative?**

**Ans.** Abstraction is often referred to as relative because abstraction is the selective examination of certain aspects of a problem. The goal of abstraction is to isolate those aspects that are important for some purpose and suppress or hide those aspects that are unimportant. Abstraction must always be for some purpose, because the purpose determines what is important and what is not. Many different abstractions of the same thing are possible, depending on the purpose for which they are made.

# Chapter 2: The Java Phenomenon

## Practice Questions

### ----->Objective-Type Questions<-----

#### A. Choose the correct answer:

1. What was Java initially called?

- a) Oak                                  b) C                                  c) C++                                  d) None of these

**Ans. a) Oak**

2. What is Java Programming Language?

- a) A run-time system                                  b) A set of development tools  
c) An Application Programming Interface (API)                                  d) All of these

**Ans. d) All of these**

3. Name the process that converts source code to bytecode.

- a) Interpretation                                  b) Compilation                                  c) Both a and b                                  d) None of these

**Ans. b) Compilation**

4. A Virtual Processor that is implemented in software and runs using the capabilities provided by your operating system and computer hardware.

- a) Byte Code                                  b) Compiler                                  c) Interpreter                                  d) Java Virtual Machine (JVM)

**Ans. d) Java Virtual Machine (JVM)**

5. Name the hardware or software environment in which a program executes.

- a) Operating System                                  b) Platform                                  c) Software                                  d) Program

**Ans. b) Platform**

6. Name the programs that can be developed in such a way that it remains embedded in a web page and runs on the viewer's machine in a secured manner by Java-compatible browsers.

- a) Applets                                  b) Applications                                  c) Servlets                                  d) Enterprise Java Beans (EJB)

**Ans. a) Applets**

7. Name the Application program that is written and compiled which may then be executed in any machine provided it contains the JVM.

- a) Applets                                  b) Applications                                  c) Servlets                                  d) Enterprise Java Beans (EJB)

**Ans. b) Applications**

8. What is the type of Java program that allows development and deployment of transactional, distributed object application-based, server-side software components?

- a) Applets                                  b) Applications                                  c) Servlets                                  d) Enterprise Java Beans (EJB)

**Ans. d) Enterprise Java Beans (EJB)**

9. Choose from among the following which is not the characteristics of a Java program.

- a) It is platform-independent.  
b) It helps in developing any kind of applications.  
c) It does not support interactive content on web pages.  
d) It is ideal to make Distributed networking applications.

**Ans. c) It does not support interactive content on web pages.**

10. What is a set of pseudo-machine language instructions that are understood by the Java Virtual Machine and are independent of the underlying hardware called?

- a) JVM                      b) Source Code                      c) Compilation                      d) Bytecode

**Ans. d) Bytecode**

### **B. Fill in the blanks:**

1. Java made Web interesting as it makes it Interactive
2. Java was formerly known as Oak.
3. Bytecode is a set of pseudo-machine language instructions that are understood by the JVM.
4. Java's portability can be attributed to Run-time system and its API.
5. Java programs are Interpreted by the JVM.
6. A Platform is the hardware/software environment in which a program remains.
7. Applets are small programs that remain embedded in a web page and runs on the viewer's machine in a secured manner.

### **----->Subjective-Type Questions<-----**

#### **Answer the following questions:**

##### **1. What is the significance of WWW?**

**Ans.** WWW or World Wide Web allows a huge pool of information (as web pages) to be shared by multiple users over a World Wide Network (Internet) and the introduction of Java makes the web pages dynamic and interactive in a secured manner.

##### **2. What is Browsing?**

**Ans.** Browsing over the internet allows surfing through the internet, thus enabling the users to see information in different pages and interact with it.

##### **3. Name two types of Java programs.**

**Ans.** The two types of Java programs are Applications and Applets.

##### **4. Why is Java a choice for the Web?**

**Ans.** The introduction of active web pages makes Java an attractive addition to the web page. When we click on a web page containing a Java applet, it is not that we just read it, listen to it, or watch it— we can interact with it. The dynamic, interactive content to the web was brought about by Java: So much so web applications made in Java run as if it is a computer program installed in the local computer.

##### **5. What is Java API?**

**Ans.** An application programming interface (API), in the context of Java, is a collection of prewritten packages and classes. Similar to a user interface, which facilitates interaction between humans and computers, an API serves as a software program interface facilitating interaction.

In Java, most basic programming tasks are performed by the API's classes and packages, which are helpful in minimizing the number of lines written within pieces of code.

##### **6. Define the term Bytecode.**

**Ans.** Bytecode is a set of pseudo-machine language instructions that are understood and executed by the Java Virtual Machine and are independent of the underlying hardware.

##### **7. What is JVM? Explain its relationship with bytecode.**

**Ans.** The Java Virtual Machine (JVM) is a microprocessor that is implemented in software and runs using the capabilities provided by your operating system and computer hardware. Since the Java virtual machine is not a real microprocessor, the Java bytecode is interpreted, rather than executed, directly in the native machine instructions of the host computer.

##### **8. What is a platform? How does it differ from a Java platform?**

**Ans.** A platform is the hardware or software environment in which a program remains. Most

platforms may be described as a combination of the operating systems (like Windows 2000/XP, Linux, Solaris, MacOS, etc.) and hardware.

The Java platform differs from most other platforms in that it is a software-only platform that runs on top of other hardware-based platforms. The Java platform has two components:

1. Java Virtual Machine (JVM)
2. Java Application Programming Interface (API)

**9. Why is Java suited for distributed networking applications?**

**Ans.** Java is ideal to make distributed networking applications because of its built-in networking support and the runtime system's capability to dynamically load Java bytecode across the network.

**10. What makes Java suitable to create secured applications?**

**Ans.** It was designed with multiple layers of security countermeasures embedded in the compiler, runtime system, and in browsers that support Java, known as Java-enabled browsers. These countermeasures make Java inherently more secure for developing any type of trusted application software.

**11. What are the features in Java that were added, but wasn't present in C++? What were the features that were discarded?**

**Ans.** The features that were added to Java which were not present in C++ are:

- 1) Java Virtual Machine (JVM)
- 2) Java Application Programming Interface (API)

The features that were dropped are Pointers and Operator Overloading.

**12. What is the Green Project?**

**Ans.** Patrick Naughton, on December 5<sup>th</sup>, 1990, started the Green Project, which he defined as an effort to "do fewer things better".

**13. State the events occurring from the year 1994 onwards with respect to Java.**

**Ans.** In the year 1994 the Web emerged, Oak was renamed Java and was used as the basis for a Web browser, called Web Runner. In 1995 Java, along with its documentation, compiler and runtime engine were made available over the Web, as an alpha version. Initially Java was hosted on SPARC Solaris, and then on Windows NT. In the summer of 1995, Java was ported to Windows 95 and Linux.

In the fall of 1995 the Java Beta 1 version was released through Sun Microsystems Website, and Java support was introduced in the Netscape 2.0 browser. During those days it was a fairly popular browser.

**14. State the areas in which Java can be used.**

**Ans.** Java applications areas are Enterprise Software, Secured Network Applications and interactive web pages.

**15. What is Java platform? In this respect also discuss why Java is called a programming language and a platform.**

**Ans.** The Java platform is a software-only platform that runs on top of other hardware-based platforms. It has two major components:

1. Java Virtual Machine (JVM)
2. Java Application Programming Interface (API)

Java is not just a programming language that is used to develop applications but it also provides a runtime environment that simulates a hardware and software called *platform*.

**16. State the characteristics of Java.**

**Ans.** Characteristics of Java:

- a) Write Once Run Anywhere (WORA)



- b) Light Weight Code
- c) Security
- d) Built-in Graphics
- e) Object Oriented Language
- f) Supports Multimedia
- g) Platform Independent
- h) Open Product

**17. What are the different types of programs that can be used in Java?**

**Ans.** The different type of programs that can be used using Java technology are:

- a) Applets
- b) Applications
- c) Servlets
- d) Enterprise Java Beans (or EJB)

# Chapter 3: The BlueJ Environment

## Practice Questions

### ----->Objective-Type Questions<-----

#### A. Choose the correct answer:

1. Which of the following is an IDE?

- a) NetBeans                      b) BlueJ                      c) Eclipse                      d) All of these

**Ans. d) All of these**

2. Name the tool that is in-built in the BlueJ environment that allows you to instantly evaluate arbitrary Java expressions and statements.

- a) Class                      b) Objects                      c) Code Pad                      d) All of these

**Ans. c) Code Pad**

3. The Show Code Pad option is present in the \_\_\_\_\_ menu of the BlueJ IDE.

- a) Project                      b) Edit                      c) Tools                      d) View

**Ans. d) View**

4. What is the name of the window on which the output as displayed by System.out.println( ) received?

- a) Terminal Window                      b) Main Window                      c) Editor Window                      d) Object Bench

**Ans. a) Terminal Window**

5. On which menu of the Terminal Window is the option to Clear screen at method call?

- a) Option                      b) Edit                      c) Project                      d) None of these

**Ans. a) Option**

6. Which of the following is not a feature of BlueJ?

- a) The concept of class and objects are presented in graphical manner.  
b) BlueJ interface is much simpler compared to contemporary IDEs like NetBeans and Eclipse.  
c) Objects are graphically shown on the Terminal Window.  
d) Code Pad allows to evaluate arbitrary Java expressions and statements.

**Ans. c) Objects are graphically shown on the Terminal Window.**

7. What is the joining of data term in Java called?

- a) Conjunction                      b) AND                      c) Concatenation                      d) None of these

**Ans. c) Concatenation**

8. What is the name of the text file that is automatically created containing a short description about your project, or may contain specifications regarding your project?

- a) AboutMe.Txt                      b) Readme.Txt                      c) AboutZUs.Txt                      d) All of these

**Ans. b) README.TXT**

#### B. State whether True or False:

1. IDE stands for Integrated Development Environment. [True]  
2. The class icon showing stripes indicates that the class is yet to be compiled. [True]  
3. The quick key combination for compilation is Ctrl + L. [False]  
4. The Terminal Window shows the output of a BlueJ program. [True]  
5. The operator + may be used to join two strings. [True]  
6. Creating a project actually forces you to organize the files in BlueJ into a folder. [True]

#### C. Fill in the blanks:

1. **Bluej** is an IDE that was designed in such a way that it supports the learning and teaching of object-oriented programming.
2. The **Code Pad** is a tool that instantly evaluates arbitrary Java expressions and statements.
3. BlueJ **Projects** are directories or folders containing the files that you create.
4. When you create a project a file named **README.TXT** is automatically created that allows you to write all relevant documents related to a BlueJ project.
5. A compiled source code creates a bytecode file recognised by the JVM having the extension **.class**.

## ----->Subjective-Type Questions<-----

### A. Answer the following questions:

#### 1. What is BlueJ? Where was it developed?

**Ans.** BlueJ is a popular IDE which is equipped with tools that support the learning and teaching of object-oriented programming. BlueJ was designed in such a way that it supports the learning and teaching of object-oriented programming thus making its design different from other development environments (like NetBeans/Eclipse) as a result.

BlueJ was developed in the year 1999 by Machael Kolling and Professor John Rosenberg at Monash University in Melbourne.

#### 2. How do you start BlueJ in Windows Operating System?

**Ans.** BlueJ can be started either from the desktop or from the Start Menu.

From the Desktop double-click on the BlueJ icon.

From the Start menu follow these steps:

1. Click on the Start button on the Taskbar.
2. From the Start Menu click on All Programs.
3. From the menu list that appears click on the BlueJ folder and then click on BlueJ icon.

#### 3. What is an IDE? State its advantages.

**Ans.** An IDE stands for Integrated Development Environment. It is a software that has a built-in text editor, with tools for compilation, execution and testing. Its advantages are:

- Write and edit source code
- See errors as you type
- See highlighted code syntax in different colours
- Automate repetitive tasks
- Compile the source code (program)
- Browse class structures
- View JavaDocs
- Use drag-and-drop utilities for easy building of features, such as graphic objects or creating

database connections

#### 4. Why is BlueJ used to teach Java in schools and colleges?

**Ans.** BlueJ is used to teach Java in schools and colleges because of the following features:

- a) Representation of class and objects is done in a graphical manner.
- b) The simplicity of the interface makes it easy to use for first time users.
- c) The Object Bench that allows object interaction explains the concept of classes and objects in a graphical manner.

The Code Pad allows you to instantly evaluate arbitrary Java expressions and statements.

#### 5. State 2 advantages of using BlueJ?

**Ans.** Two advantages of BlueJ are:

- a) There is no need for a main( ) method for execution of a Java code.
- b) Testing of methods can be performed interactively.

#### **6. What is Code Pad?**

**Ans.** Code Pad is used to quickly and easily evaluate an arbitrary snippet of Java expression.

#### **7. How do you start coding in BlueJ?**

**Ans.** To start coding using BlueJ, you need to create a **Project** and start with a new **class** to start coding into it. Once the **class** icon is displayed you may double click on it to start coding

#### **8. What is the significance of README file?**

**Ans.** The README file is used to write all relevant documents related to the project. It is simply a text file containing a short description about your project, or may contain specifications regarding your project.

#### **9. How do you create a class in BlueJ?**

**Ans.** To create a **class** in BlueJ click on the **New Class...** button on the main project window, which in turn displays the **Create New Class** dialog, where you can assign a name for the **class**.

#### **10. How do you start the editor in BlueJ?**

**Ans.** To start the editor in BlueJ, you just need to double click on the **class** icon or right click on the **class** icon and click on **Open Editor** option from the pop-up menu.

#### **11. How do you remove the default code generated when a class is created in BlueJ?**

**Ans.** To remove the default code, you may use your mouse by click and drag to select the entire code or press **Ctrl+A** to select the entire code and press the **Delete** key.

#### **12. How do you compile a program in BlueJ?**

**Ans.** To compile a Java source code simply click on the **Compile** button on the editor window or use the keyboard combination **Ctrl+K**.

#### **13. What is the difference between System.out.print and System.out.println?**

**Ans.** System.out.print() displays or outputs a data item but keeps the cursor in the same line, hence the next output continues to be in the same line. System.out.println() on the other hand display the data item and brings the cursor to the next line, hence the next output is displayed in the next line following the previous output.

#### **14. What is a Terminal Window? How do you automatically clear the Terminal Window screen with every method call?**

**Ans.** The Terminal Window is a window that acts as a console for output of a Java program, when using System.out.print( ) or System.out.println( ) statements.

To automatically clear the Terminal window with every method call use the following in the Terminal Window:

- a) Click on the **Options** menu.
- b) Click on **Clear screen at method call** option from the menu.

#### **15. What is concatenation?**

**Ans.** Joining of two strings or data items is termed as concatenation.

#### **16. What happens when all the data items are numeric and + is used in the System.out.print statement?**

**Ans.** When all data items are numeric and + is used in the System.out.println() statement, it gets evaluated, *i.e.*, it gets added.

#### **17. How do you print different numeric data items in the same System.out.print statement without getting it evaluated?**

**Ans.** This is done by concatenating a blank string in the middle. For example,  
`System.out.println(5 + " " + 6);`

Will give the output as: 5 6

**18. How do you compile and execute a program using Sun's Java compiler?**

**Ans.** To compile a java source code `javac` file for compilation and for execution use the `java` file.

**19. Which method is necessary to execute a Java program in Java run-time environment?**

**Ans.** The method that is necessary to execute a Java program in Java run-time environment is `public static void main(String args[])`.

**20. Why is BlueJ environment considered better for coding programs than conventional Java programming?**

**Ans.** The BlueJ environment is considered better for coding programs than conventional Java programming because:

- a) There is no need for a main method.
- b) Testing can be performed interactively.
- c) It allows the complexity of input to be deferred.
- d) It encourages object-oriented thinking and programming.

**B. Give the output of the following programs:**

**1. public class Trial**

```
{
    static void example( )
    {
        System.out.print("Hello");
        System.out.print("from");
        System.out.print("BlueJ");
    }
}
```

**Ans.** HellofromBlueJ

**2. public class Trial**

```
{
    static void example( )
    {
        System.out.print("Hello");
        System.out.println("from");
        System.out.print("BlueJ");
    }
}
```

**Ans.** Hellofrom  
BlueJ

**3. public class Trial**

```
{
    static void example( )
    {
        System.out.println(27+36);
    }
}
```

```
System.out.println(12*19);
System.out.println(57%12);
```

```
}
}
}
Ans. 63
      228
      6.84
```

#### 4. public class Trial

```
{
    static void example( )
    {
        System.out.print("Calculation=");
        System.out.println((17+48)*12);
    }
}
```

**Ans.** Calculation=780

### C. Find the errors in the following programs:

#### 1. public Class Trial

```
{
    static void example ( )
    {
        System.out.println("Systemetic");
        System.out.println("Programming")
        System.out.println("In Java");
    }
}
```

#### 2. public class Trial

```
{
    Static void example( );
    {
        system.out.println("Systemetic");
        system.out.println("Programming");
        System.out.println("In Java");
    };
}
```

**Ans.** Find the errors in the following programs: (All corrections are underlined)

#### 1. public class Trial

```
{
    static void example ( )
    {
        System.out.println("Systemetic");
        System.out.println("Programming");
    }
}
```

```

        System.out.println("In Java");
    }
}

```

2. public class Trial

```

{
    static void example()
    {
        System.out.println("Systemetic");
        System.out.println("Programming");
        System.out.println("In Java");
    }
}

```

### Lab Exercises

1. Write a program to display your name and your friend's name using `System.out.println( )` in different lines.

**Ans.** public class Question1

```

{
    static void main()
    {
        System.out.println("Devashis Chakraborty");
        System.out.println("Amit Mishra");
    }
}

```

2. Write a program to display the names of five fruits using `System.out.print( )` 5 times in the same line.

**Ans.** public class Question2

```

{
    static void main()
    {
        System.out.print("Apple ");
        System.out.print("Guava ");
        System.out.print("Orange ");
        System.out.print("Grapes ");
        System.out.print("Mango");
    }
}

```

3. Write a program to find the add 5 and 6 and multiply it with 7. (Hint :  $(5+6)*7$ )

**Ans.** public class Question3

```

{
    static void main()

```

```
    {
        System.out.println((5+6)*7);
    }
}
```

**4. Write a program to find the sum and difference between 15 and 8.**

**Ans.** public class Question4

```
{
    static void main()
    {
        System.out.println(15+8);
        System.out.println(15-8);
    }
}
```

**5. Write a program to find the quotient and the remainder, when 123 is divided by 11. (Hint: Use / and % operator).**

**Ans.** public class Question5

```
{
    static void main()
    {
        System.out.println("Quotient="+(123/11));
        System.out.println("Remainder="+(123%11));
    }
}
```

**6. Write a program to find the sum of the cubes of 12, 9 and 4.**

**Ans.** public class Question6

```
{
    static void main()
    {
        System.out.println(12*12+9*9+4*4);
    }
}
```

**7. Write a program to add 56 and the last digit of 768 [Hint whenever you divide a no. by 10 the remainder is always the last digit.]**

**Ans.** public class Question7

```
{
    static void main()
    {
        System.out.println(56+768%10);
    }
}
```



# Chapter 4: Java Fundamentals

## Practice Questions

### ----->Objective-Type Questions<-----

#### Fill in the blanks:

1. Java uses Unicode character set.
2. Unicode is a two bytes character set.
3. Character Set is a set of valid characters that a language can recognize.
4. The smallest individual unit in a program is a token.
5. Identifiers are fundamental building blocks that give names to different parts of a program.
6. Escape sequences/Non graphic characters are characters that cannot be typed directly from the keyboard.
7. Variables represent named storage locations whose values can be manipulated during program execution.
8. Operators that act upon two operands are referred to as Binary operators.
9. The process of converting one predefined type into another is called Type Conversion.
10. The new operator is used to allocate memory for objects and arrays.

### ----->Subjective-Type Questions<-----

#### Answer the following questions:

##### 1. What is Character Set?

**Ans.** Character set is a collection of all the characters that can be used in a Java program. A character represents any letter (whether capital or small), digits (0 to 9) or any symbol (% , & , ^ , @ , \$ , | , etc).

##### 2. What is bytecode?

**Ans.** Bytecode is an intermediate language which is received upon compilation of source code and interpreted by the JVM (Java Virtual Machine).

##### 3. What is indentation in a Java program?

**Ans.** Indentations are whitespaces that are given by programmers to increase the readability of a source code.

##### 4. What are keywords?

**Ans.** Keywords are reserved words in Java and thus provide a special meaning to the Java compiler.

##### 5. What are Identifiers? State the rules while using Identifiers.

**Ans.** Identifiers are the names given to different parts or elements of a program to identify it.

The rules that you should keep in mind to use identifiers are:

- a) It can have any alphabet (capital or small), digits, underscore and dollar sign characters.
- b) It should not begin with digits or should not contain any special character.
- c) It cannot have a space in between.
- d) It must not be a keyword.
- e) It can be of any length.
- f) Since Java is case-sensitive, it treats capital (upper-case) and small (lower-case) characters differently.

## 6. What is a literal? What are the different types of literals available in Java?

**Ans.** Literals are tokens that are used to represent constants that may be assigned to a variable. Integer literal or Fixed point literal, Floating point literal, Boolean literal, Character literal, String literal, Null literal and Class literal are the literals that are used in Java.

## 7. State the difference between a Boolean literal and a character literal.

**Ans.** A character literal is used to enclose a character within single quotes and boolean literal refer to two states i.e. true (to represent success) and false (to represent failure).

## 8. How are Decimal, Octal and Hexadecimal integer literals represented in Java?

**Ans.** Decimal Integer Literals use digits from 0 to 9 (i.e. 10 digits)  
Octal Integer Literals use digits from 0 to 7 (i.e. 8 digits) and should be prefixed with 0.  
Hexadecimal Integer Literals use digits from 0 to 9 and alphabets from A (or a) to F (or f) representing 10 to 15 (i.e. 16 numbers) altogether and should be prefixed with 0x or 0X.

## 9. What are the escape sequences available in Java?

**Ans.** The escape sequences available in Java are:

Escape Sequences	Meaning
<code>'\b'</code>	Backspace
<code>'\f'</code>	Form feed
<code>'\n'</code>	New line
<code>'\r'</code>	Carriage return
<code>'\t'</code>	Horizontal Tab
<code>'\''</code>	Single quote
<code>'\"'</code>	Double quote
<code>'\\'</code>	Backslash
<code>'\0n'</code>	Octal number, where n is the number
<code>'\xHn'</code>	Hexadecimal number, where n is the number
<code>'\uHn'</code>	Unicode character represented through its hexadecimal code n

## 10. What are the different punctuators available in Java?

**Ans.** There are 9 punctuators available in Java. They are: ( ) { } [ ] ; , .

## 11. What are Tokens?

**Ans.** Token is the smallest fundamental unit in a program. They are the valid characters or symbols that can be used in a Java program.

## 12. State the difference between token and identifier.

**Ans.** Token is the smallest fundamental unit in a program, whereas identifier is a token which is used to give names to different parts or elements of a program to identify it.

## 13. State the two kinds of data types.

**Ans.** The two data types are:

- Fundamental Data Type
- Composite Data Type

## 14. State the size of integer data types in Java.

**Ans.** The size of integer data types are:

- byte – 8 bits
- short – 16 bits
- int – 32 bits
- long – 64 bits

## 15. How are floating point numbers represented in Java?

**Ans.** Floating point numbers are used to represent fractional numbers (with decimal point). Java

provides two ways of representing fractional numbers viz. **float** and **double**. Float type numbers are suffixed with F or f and double type numbers are suffixed with D or d, however suffixing with D or d for double data type is optional.

#### 16. What are variables? How are variables initialized?

**Ans.** Variable is a name given to memory location where data is stored or needs to be stored. The general format of variable declaration is as follows:

```
<data-type><variable-name>;
```

For initialization of variables use the general syntax :

```
<data-type> <variable-name>=<constant>;
```

#### 17. How are symbolic constants represented in Java?

**Ans.** Symbolic constants are represented with the **final** keyword before a variable declaration: For example, `final int MAX=100;` declares a symbolic constant MAX with the constant value 100.

#### 18. How are negative numbers represented in Java?

**Ans.** Negative numbers are represented in Java with the – (minus) sign preceding the number. Example -34, -24.34, etc.

#### 19. Differentiate between operator and expression.

**Ans.** An operator is a symbol or command that is used to perform a particular task like evaluating an expression. Expression on the other hand is a combination of an operand and operators that evaluates to a value.

#### 20. Explain the term typecasting.

**Ans.** Typecasting is the process of converting the resultant data type of an expression from one data type to another.

#### 21. State the difference between Unary and Binary Operators?

**Ans.** Unary operators are mathematical operators that works on a single operand (eg. `-b`, `++c`) whereas binary operators are mathematical operators that works on two operands (eg. `a+b`, `c*d`).

#### 22. What is meant by precedence of operators?

**Ans.** Precedence of operators refers to the order in which a mathematical expression with multiple mathematical operators gets evaluated.

#### 23. What are relational operators?

**Ans.** Relational Operators are used to define a conditional expression.

#### 24. What is a compound statement? Give an example.

**Ans.** The group of statements within a block is referred to as a compound statement.

For example,

```
{
    int a=5, b=6;
    System.out.println(a+ “ ”+b);
}
```

#### 25. What are the different logical operators available in Java?

**Ans.** The different logical operators available in Java are AND( `&&`), OR ( `||`) and NOT ( `!`).

**26. State the difference between Shift-right (>>) and Shift-right-zero-fill (>>>) operators.**

**Ans.** Shift right (>>) operator is used to shift bits towards the right and the vacant places after shifting is filled up with 0 if the number is positive and 1 if the number is negative to thus retaining the sign after shifting.

Shift-right-zero-fill shift bits towards the right and the vacant places after shifting is filled up with 0. Thus the number always results to a positive number.

**27. What is type conversion? Under what circumstances is there a "Loss of Information"?**

**Ans.** The process of converting implicitly or explicitly one data type into another compatible data type is termed as type conversion.

In case a variable or constant of higher size is assigned to a variable of smaller size, it results in **possible loss of precision** error.

**28. What are comments? What are the different types of comments that may be used in a Java program?**

**Ans.** Comments are notes in a program to provide an easy understanding of the problem. This ensures proper documentation. But these statements are ignored by the compiler and no question of error may occur within this, whatsoever you may write within it. There are three types of statements available in Java, viz. Single-line Comment, Multiline Comment and Documentation Comment.

**29. What is the purpose of new operator?**

**Ans.** The new operator is used to allocate memory for objects and arrays.

**30. What is a null statement?**

**Ans.** Statement which has only a semicolon is termed as null statement. These statements are generally used where syntaxes of certain commands require a statement whereas the programmer may not want to give it. In that case to abide by the syntax of the command the null statement is used.

**31. What is a compound statement?**

**Ans.** The group of statements within a block is referred to as a compound statement.

For example,

```
{
    int a=5, b=6;
    System.out.println(a+ " "+b);
}
```

**32. What do you understand by type conversion?**

**Ans.** Converting explicitly using type-casting the resultant of a mathematical expression from one data type to another is termed as type conversion.

**33. State the importance of a "class".**

**Ans.** The class acts as a skeleton for an object. The classes are the bases of an object. It is a template upon which an object is built. The class is the mechanism that is used to create objects.

**34. Why is Java called a strongly typed language?**

**Ans.** Java is a strongly-typed or statically-typed language in the sense that every variable has a type, every expression has a type and every type is strictly defined. Also all assignments, whether explicit or via parameters passing in method calls, are checked for type compatibility.

### 35. How do you declare objects?

**Ans.** Objects are declared using the **new** command using the following syntax:

```
<Class-name> <object-name>= new <Class-name>();
```

### 36. State one difference between the floating point literals float and double.

**Ans.** The float data type is of size 32 bits, whereas double data type is of 64 bits.

## Lab Exercises

### A. Write the following programs:

1. Write a program to initialize two integer variables a and b with 5 and 6 respectively and interchange them. Thus after interchanging a and b will be 6 and 5 respectively.

**Ans.** public class Question1

```
{
    static void main()
    {
        int a=5,b=6,t;
        System.out.println("Before Interchanging a="+a+" b="+b);
        //Swapping the values
        t=a;
        a=b;
        b=t;
        System.out.println("After Interchanging a="+a+" b="+b);
    }
}
```

2. Write a program to display the names of five fruits with a single System.out.println( ); statement, but in different lines.

**Ans.** public class Question2

```
{
    static void main()
    {
        System.out.println("Apple\nOrange\nGuava\nMango\nGrapes");
    }
}
```

3. Write a program using float type variables to find the area and perimeter of a square whose side is 12 cm.

**Ans.** public class Question3

```

{
    static void main()
    {
        float s=12,a;
        a=s*s;
        System.out.println("Area of the square="+a+" square cm");
    }
}

```

**4. Write a program using int variables to find the sum of three numbers, say 15, 36 and 45, and subtract the result from 100.**

**Ans.** public class Question4

```

{
    static void main()
    {
        int a=15,b=36,c=45,d=100,e,f;
        e=a+b+c;
        f=d-e;
        System.out.println("Sum="+e);
        System.out.println("Difference="+f);
    }
}

```

**5. Write a program using int variables to find the area and perimeter of a rectangle of length 12 cm and breadth 8 cm.**

**Ans.** public class Question5

```

{
    static void main()
    {
        int length=12,breadth=8,area,perimeter;
        area=length*breadth;
        perimeter=2*(length+breadth);
        System.out.println("Area="+area);
        System.out.println("Perimeter="+perimeter);
    }
}

```

**6. Write a program using variables to find the profit and profit per cent of a certain transaction where S.P.= Rs.10000 and C.P.= Rs.7000.**

**Ans.** public class Question6

```

{
    static void main()
    {
        int SP=10000,CP=7000;
        double profit,profitPercentage;
    }
}

```

```

        profit=SP-CP;
        profitPercentage=(double)profit/CP*100;
        System.out.println("Profit="+profit);
        System.out.println("Perimeter="+profitPercentage);
    }
}

```

**7. Write a program using variables to find the cost of 17 pencils if the cost of one pencil= Rs.2.50. Output should be:**

**Cost of 1 pencil= Rs.2.5**

**Cost of 17 pencils= Rs.42.5**

**Ans.** public class Question7

```

{
    static void main()
    {
        float rate=2.50f, cost;
        int pencils=17;
        cost=pencils*rate;
        System.out.println("Cost of 1 pencil= Rs "+rate);
        System.out.println("Cost of 17 pencils= Rs "+cost);
    }
}

```

**8. Write a program to initialize three int variables a, b and c with 234, 456 and 712 and store the sum of the last digits of the variables into d and display it.**

**Ans.** public class Question8

```

{
    static void main()
    {
        int a=234, b=456, c=712, d;
        d=a%10+b%10+c%10;
        System.out.println("Sum of last digits="+d);
    }
}

```

**9. Write a program to initialize an int variable a with 76498 and from it extract the first digit and store it in f and extract the last digit in l and display both these digits.**

**Ans.** public class Question9

```

{
    static void main()
    {
        int a=76498,f,l;
        f=a/10000;
        l=a%10;
        System.out.println("First digit="+f);
    }
}

```

```

        System.out.println("Last digit="+l);
    }
}

```

**10. Write a program using variables to find the average of 35, 43 and 97.**

**Ans.** public class Question10

```

{
    static void main()
    {
        int a=35,b=43,c=97;
        float avg;
        avg=(float)(a+b+c)/3;
        System.out.println("Average of the numbers="+avg);
    }
}

```

### B. Short-answer-type questions:

**1. Write the corresponding expressions for the following mathematical operations:**

i)  $a^2 + b^2$

ii)  $z = x^3 + y^3 - xy/z$

**Ans.** i)  $a*a + b*b$

ii)  $z=x*3 + y*3 - x*y/z$

**2. What will be the output for the following program segment?**

**int a = 0, b = 30, c = 40;**

**a = --b + c++ + b;**

**System.out.println("a = " + a);**

**Ans.** a =99

**3. if m = 5 and n = 2 output the values of m and n after execution in (i) and (ii):**

i) **m -= n;**

ii) **n = m + m/n;**

**Ans.** i) m=3 and n=2

ii) m=5 and n=7

**4. What will be the output of the following, if x = 5 initially?**

i) **5 \*++x**

ii) **5\*x++**

**Ans.** i) 30

ii) 25

**5. Evaluate the following expressions, if the values of the variables are a = 2, b=3 and c=9**

i) **a - (b++) \* (- -c)**

ii) **a \* (++b) % c**

**Ans.** i) -22

ii) 8



6. If  $a = 5$ ,  $b = 9$  calculate the value of :

$a+ = a++ - ++b + a$

**Ans.**  $a=6$  and  $b=10$

7. What is the result stored in  $x$ , after evaluating the following expression

$\text{int } x = 5;$

$x = x++ * 2 + 3 * -x;$

**Ans.**  $-8$

8. Assign the value of  $\pi$  (i.e. 3.142) to a variable with requisite data type.

**Ans.**  $\text{double } \pi = 3.142;$

9. Which of the following are valid comments?

i) `/* comment * 1`

ii) `/* comment`

iii) `// comment`

iv) `*/ comment */`

**Ans.** iii) `// comment`

10. Name the primitive data type in Java that is:

i) a 64-bit integer and is used when you need a range of values wider than those provided by it.

ii) a single 16 bit Unicode character whose default value is '100000'.

**Ans.** i) `long`

ii) `char`

11. Write a Java statement that assigns the value of  $\pi$  (i.e. 3.142) to a variable with requisite data type.

**Ans.**  $\text{double } \pi = 3.142;$

C. State the value of  $a$ ,  $b$  and  $c$  after the execution of each of the following statements where  $a=12$ ,  $b=13$  and  $c=11$ :

1.  $a = a++ + - -b + c++$

**Ans.**  $a=35$ ,  $b=12$ ,  $c=12$

2.  $b = b++ + ++a * 2$

**Ans.**  $a=13$ ,  $b=39$ ,  $c=11$

3.  $c = c++ + a++ * (++b)$

**Ans.**  $a=13$ ,  $b=14$ ,  $c=179$

4.  $b = b++ + ++b + c++ + ++a$

**Ans.**  $a=13$ ,  $b=52$ ,  $c=12$

5.  $a = -a + b++ + ++a + - -b * c++$

**Ans.**  $a=179$ ,  $b=13$ ,  $c=12$

6.  $a += a++ + ++b - c++$

**Ans.**  $a=27$ ,  $b=14$ ,  $c=12$

7.  $b += - -c + ++c + a++ + (b++)$

**Ans.**  $a=13$ ,  $b=59$ ,  $c=11$

8.  $c += a - - + - -a + (b++) * (++b)$

**Ans.**  $a=10$ ,  $b=15$ ,  $c=228$

9.  $a += a++ + ((++b)/2)$

**Ans.** a=31, b=14, c=11

**D. State the result of c while using the following bitwise operators:**

1.  $c = a \& b$  when a=13 and b=4

**Ans.** c=4

2.  $c = a | b$  when a=22 and b=11

**Ans.** c=31

3.  $c = a \wedge b$  when a=14 and b=32

**Ans.** c=46

4.  $c = \sim a$  when a=36

**Ans.** c=37

5.  $c = a \& b | c$  when a=12, b=13 and c=11

**Ans.** c=15

6.  $c = a | b \& c \wedge a$  when a=15, b=12 and c=10

**Ans.** c=15

7.  $c = a \& (\sim b)$  when a=15 and b=12

**Ans.** c=3

8.  $c = (a | \sim b) \& \sim c$  when a=15, b=12 and c=10

**Ans.** c=11

9.  $c = c \wedge (a \& \sim b)$  when a=15, b=12 and c=10

**Ans.** c=9

10.  $c = a \& b \& c | a | b$  when a=15, b=12 and c=10

**Ans.** c=15

**E. State the result of c while using the following shift operators:**

1.  $c = a \gg 3$  when a=-45

**Ans.** -6

2.  $c = a \ll 2$  when a=-33

**Ans.** -132

3.  $c = a \gg 4$  when a=97

**Ans.** 6

4.  $c = a \gg 26$  when a=-97

**Ans.** 63

5.  $c = a \ll 2$  when a=12

**Ans.** 48

**F. Give the output of the following method:**

```
public static void main (String args [])
```

```
{  
    int a = 5;  
    a++;  
    System.out.println(a);  
    a=(a--)-(-a);  
    System.out.println(a);  
}
```

**Ans.** 6

4

# Chapter 5: Programming in Java

## Practice Questions

### >Objective-Type Questions<

#### State whether True or False:

1. A statement missing a semicolon is an example of logical error. [False]
2. You can input only integers using parameterized inputs. [False]
3. Using nextInt( ) member function of the Scanner class, you can input a string. [False]
4. While you input numbers of float data type using Scanner, succeed the number with f? [False]
5. Parameterised input for String requires you to enter data within quotes. [True]

### >Subjective-Type Questions<

#### Answer the following questions:

1. Rewrite the following program after removing the errors in the program, if any:

```
class Myclass
{
    import java.util.Scanner;
    static void function( )
    {
        int a;
        a=sc.nextInt( );
        Scanner sc=new Scanner(System.in);
        System.out.println(a);
    }
}
```

**Ans.** All corrections made are underlined:

```
import java.util.Scanner;
class Myclass
{
    static void function( )
    {
        int a;
        System.out.println("Enter an integer:");
        a=sc.nextInt( );
        System.out.println(a);
    }
}
```

2. Give the output of the following program:

```
public class Output
{
    static void function( )
```

```

    {
        int a=3,b=14;
        System.out.println(a+" "+b);
        a=a+b;
        b=a-b;
        a=a-b;
        System.out.println(a+" "+b);
    }
}

```

**Ans.** 3 14  
14 3

**3. Give the output of the program:**

```

public class Output
{
    static void input(int a,int b)
    {
        int c,d;
        c=(++a) + (b++);
        d=(a-- -) - (- -b);
        System.out.println(c+" "+d);
    }
}

```

**When:**

- a) a=12 and b=13 is taken as parameters.
- b) a=9 and b=4 is taken as parameters.

**Ans.** a) 26 0  
b) 14 6

**4. State the difference between nextInt( ) and nextDouble( ) functions.**

**Ans.** The nextInt() member function of the Scanner class is used to take a number of int data type as input. The nextDouble() member function of the Scanner class is used to take a floating point number of double data type as input.

**5. What is a bug? Also state what is debugging?**

**Ans.** Bug is an error in a program and the process of removal of the bug or error is called debugging.

**6. What are the different types of errors in Java?**

**Ans.** The different types of errors in Java are:

- i) Compile-time errors
- ii) Logical Errors
- iii) Run-time errors

## Lab Exercises

**Write the following programs:**

**1. Write a program to input two integers using parameters and find their sum and product.**

**Ans.** public class Question1

```

{
    static void main(int a,int b)

```

```

    {
        int s,p;
        s=a+b;
        p=a*b;
        System.out.println("Sum="+s);
        System.out.println("Product="+p);
    }
}

```

**2. Write a program to input 5 integers and find their average.**

**Ans.** public class Question2

```

    {
        static void main(int a,int b,int c,int d,int e)
        {
            float avg;
            avg=(a+b+c+d+e)/5;
            System.out.println("Average="+avg);
        }
    }
}

```

**3. Write a program to input two integers using Scanner and find the product of their sum and difference.**

**Ans.** import java.util.\*;

```

public class Question3
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,s,d,p;
        System.out.println("Enter two integers:");
        a=sc.nextInt();
        b=sc.nextInt();
        s=a+b;
        d=a-b;
        p=s*d;
        System.out.println("Sum="+s);
        System.out.println("Difference="+d);
        System.out.println("Product of Sum and Difference="+p);
    }
}

```

**4. Write a program to input three integers and find the difference between their sum and their average.**

**Ans.** import java.util.\*;

```

public class Question4
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);

```

```

int a,b,c,sum;
float avg,diff;
System.out.println("Enter three integers:");
a=sc.nextInt();
b=sc.nextInt();
c=sc.nextInt();
sum=a+b+c;
avg=(float)sum/3;
diff=sum-avg;
System.out.println("Difference="+diff);
}
}

```

**5. Write a program to input the Principal, Rate and Time for a certain amount of money and print the Simple Interest.**

**Note: Simple Interest =  $\frac{(Principle * Rate * Time)}{100}$**

**Ans.**

```

import java.util.*;
public class Question5
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float principal,rate,time,SI;
        System.out.println("Enter Principal, Rate and Time:");
        principal=sc.nextFloat();
        rate=sc.nextFloat();
        time=sc.nextFloat();
        SI=(principal*rate*time)/3;
        System.out.println("Simple Interest="+SI);
    }
}

```

**6. Write a program to input the length and breadth of a rectangle and find its area and perimeter.**

**Note: Area of a rectangle=length\*breadth**

**Perimeter of a rectangle=2\*(length + breadth)**

**Ans.**

```

import java.util.*;
public class Question6
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float length,breadth,area,perimeter;
        System.out.println("Enter length and breadth of a rectangle:");
        length=sc.nextFloat();
        breadth=sc.nextFloat();
        area=length*breadth;
    }
}

```

```

        perimeter=2*(length+breadth);
        System.out.println("Area="+area);
        System.out.println("Perimeter="+perimeter);
    }
}

```

**7. Write a program to input the radius of a circle and find its area and circumference.**

**Note:** *Area of a circle=22/7 \* radius<sup>2</sup>*

***Circumference of a circle= 2 \* 22/7 \* radius***

**Ans.** import java.util.\*;

```

public class Question7
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float radius,area,circumference;
        System.out.println("Enter radius of a circle:");
        radius=sc.nextFloat();
        area=(float)22/7*radius*radius;
        circumference=2*(float)22/7*radius;
        System.out.println("Area="+area);
        System.out.println("Circumference="+circumference);
    }
}

```

**8. Write a program to input the length, breadth and height of a cuboid and find its Volume and Total Surface Area.**

**Note:** *Volume of a cuboid= length\*breadth\*height*

***Total Surface Area=2\*(length\*breadth+breadth\*height+height\*length)***

**Ans.** import java.util.\*;

```

public class Question8
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float length,breadth,height,vol,area;
        System.out.println("Enter length, breadth and height of a cuboid:");
        length=sc.nextFloat();
        breadth=sc.nextFloat();
        height=sc.nextFloat();
        vol=length*breadth*height;
        area=2*(length*breadth+breadth*height+height*length);
        System.out.println("Volume of cuboid="+vol);
        System.out.println("Total Surface Area="+area);
    }
}

```

**9. Write a program to input the radius and height of a cylinder and find its volume and total surface area.**

**Note: Total Surface Area =  $2\pi r^2 + 2\pi rh$**

**Volume =  $\pi r^2 h$**

**Ans.**

```
import java.util.*;
public class Question9
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float radius,height,vol,area;
        System.out.println("Enter radius and height of a cylinder:");
        radius=sc.nextFloat();
        height=sc.nextFloat();
        vol=(float)22/7*radius*radius*height;
        area=2*(float)22/7*radius*(radius+height);
        System.out.println("Volume of cylinder="+vol);
        System.out.println("Total Surface Area="+area);
    }
}
```

**10. Write a program to input three integers and find the sum of the last digit of the numbers.**

**For example, if the inputs are: 12**

**26**

**35**

**Output:**

**Sum of the last digit of the integers are: 13**

**Ans.**

```
import java.util.*;
public class Question10
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c,s;
        System.out.println("Enter three integers:");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        s=a%10+b%10+c%10;
        System.out.println("Sum of the last digit of the integers are:"+s);
    }
}
```

**11. Write a program to input the temperature in Fahrenheit and change it to Celsius.**

**Note: The relation between Fahrenheit and Celsius is given by the formula:**

$$\frac{C}{5} = \frac{F - 32}{9}$$



**Where C=Celsius and F=Fahrenheit**

```
Ans. import java.util.*;
public class Question11
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float f,c;
        System.out.println("Enter temperature in fahrenheit:");
        f=sc.nextInt();
        c=(f-32)*5/9;
        System.out.println("Temperature in Celcius:"+c);
    }
}
```

**12. Write a program to input time in seconds and display the time broken up as hours, minutes and seconds.**

**For Example:**

**INPUT: Enter Time in Seconds: 4326**

**OUTPUT:**

**Time in hours:1**

**Time in minutes:12**

**Time in seconds:6**

```
Ans. import java.util.*;
public class Question12
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int s,h,m,ss;
        System.out.println("Enter temperature in seconds:");
        s=sc.nextInt();
        h=s/3600;
        m=(s%3600)/60;
        ss=(s%3600)%60;
        System.out.println("Time in hours:"+h);
        System.out.println("Time in minutes:"+m);
        System.out.println("Time in seconds:"+ss);
    }
}
```

**13. Write a program to input a floating point number and round it off to the nearest integer.**

**For Example:**

**INPUT: Enter a floating point number: 12.3**

**OUTPUT: Rounded off to nearest integer:12**

**INPUT: Enter a floating point number: 14.5**

**OUTPUT: Rounded off to nearest integer:15**

```
Ans. import java.util.*;
      public class Question13
      {
          static void main()
          {
              Scanner sc=new Scanner(System.in);
              float f;
              int n;
              System.out.println("Enter a floating point number:");
              f=sc.nextFloat();
              n=(int)(f+0.5);
              System.out.println("Rounded off to nearest integer:"+n);
          }
      }
```

**14. Write a program to input a floating point number and round it off to two places of decimal.**

**For Example:**

**INPUT: Enter a floating point number: 12.367**

**OUTPUT: Rounded off to two places of decimal:12.37**

**INPUT: Enter a floating point number: 14.563**

**OUTPUT: Rounded off to two places of decimal:14.56**

```
Ans. import java.util.*;
      public class Question14
      {
          static void main()
          {
              Scanner sc=new Scanner(System.in);
              float f,n;
              int h;
              System.out.println("Enter a floating point number:");
              f=sc.nextFloat();
              h=(int)((f*100)+0.5);
              n=(float)h/100;
              System.out.println("Rounded off to two places of decimal:"+n);
          }
      }
```

**15. Write a program to input two integers (say a and b) and interchange their values and display the result.**

**For Example:**

**INPUT: Enter two integers: 15 36**

**OUTPUT:**

**Before interchange: a=15 and b=36**

**After interchange: a=36 and b=15**

**Please note that you can take just one variable other than a and b for interchanging.**

```
Ans. import java.util.*;
```

```

public class Question15
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c;
        System.out.println("Enter two integers:");
        a=sc.nextInt();
        b=sc.nextInt();
        System.out.println("Before interchange: a="+a+" and b="+b);
        c=a;
        a=b;
        b=c;
        System.out.println("After interchange: a="+a+" and b="+b);
    }
}

```

**16. Write a program to input the time in hours, minutes and seconds and print it in seconds.**

**For Example:**

**INPUT: Enter time in hours:**

1

**Enter time in minutes:**

12

**Enter time in seconds:**

6

**OUTPUT:**

**Time in seconds:4326**

**Ans.** import java.util.\*;

```

public class Question16
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int h,m,s,ss;
        System.out.println("Enter time in hours:");
        h=sc.nextInt();
        System.out.println("Enter time in minutes:");
        m=sc.nextInt();
        System.out.println("Enter time in seconds:");
        s=sc.nextInt();
        ss=h*3600+m*60+s;
        System.out.println("Time in seconds:"+ss);
    }
}

```

**17. Write a program to input three integers and find their sum, without using the mathematical operator +.**

**Ans.**

```
import java.util.*;
public class Question17
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c,s;
        System.out.println("Enter three integers:");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        s=a-(-b)-(-c);
        System.out.println("Sum:"+s);
    }
}
```

**18. Write a program to enter the perimeter of a square and find its perimeter.**

*For Example,*

**INPUT:**

*Enter the perimeter of a square: 64*

**OUTPUT:**

*Area of the square is: 256*

**Ans.**

```
import java.util.*;
public class Question18
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float p,s,a;
        System.out.println("Enter the perimeter of a square:");
        p=sc.nextInt();
        s=p/4;
        a=s*s;
        System.out.println("Area of the square:"+a);
    }
}
```

**19. Write a program to enter the length and area of a rectangle and find its perimeter.**

*For Example,*

**INPUT:**

*Enter the length of the rectangle:12*

*Enter the area of the rectangle: 72*

**OUTPUT:**

*Perimeter of the rectangle:36.0*

**Ans.**

```
import java.util.*;
public class Question19
```

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float l,b,a,p;
        System.out.println("Enter the length of the rectangle:");
        l=sc.nextInt();
        System.out.println("Enter the area of the rectangle:");
        a=sc.nextInt();
        b=a/l;
        p=2*(l+b);
        System.out.println("Perimeter of the rectangle:"+p);
    }
}

```

**20. Write a program to input the basic pay of an employee and find the gross pay of the employee for the following allowances and deductions:**

**Dearness Allowance = 25% of Basic Pay**

**House Rent Allowance=15% of Basic Pay**

**Provident Fund=8.33% of Basic Pay**

**Net Pay=Basic Pay + Dearness Allowance + House Rent Allowance**

**Gross Pay= Net Pay – Provident Fund**

**Ans.** import java.util.\*;

```

public class Question20
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float basic,da,hra,pf,np,gp;
        System.out.println("Enter the basic pay of the employee:");
        basic=sc.nextFloat();
        da=(float)25/100*basic;
        hra=(float)15/100*basic;
        pf=(float)8.33/100*basic;
        np=basic+da+hra;
        gp=np-pf;
        System.out.println("Gross Pay:"+gp);
    }
}

```

**21. Write a program to input the dimensions of a rectangle and find the area of that square whose perimeter is equal to the perimeter of the rectangle.**

**Ans.** import java.util.\*;

```

public class Question21
{
    static void main()
    {

```

```

Scanner sc=new Scanner(System.in);
float l,b,p,s,a;
System.out.println("Enter the length & breadth of the rectangle:");
l=sc.nextFloat();
b=sc.nextFloat();
p=2*(l+b);    //perimeter of the rectangle
s=p/4;        /*side of the square whose perimeter is equal
               to the perimeter of the rectangle*/
a=s*s;        //area of the square
System.out.println("Area of the Square:"+a);
    }
}

```

**22. Write a program to input an integer and if positive change it to negative and if negative change it to positive.**

**INPUT: Enter an integer: 12**

**OUTPUT: Sign changed: -12**

**INPUT: Enter an integer: -14**

**OUTPUT: Sign changed: 14**

**Ans.**

```

import java.util.*;
public class Question22
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b;
        System.out.print("Enter an integer: ");
        a=sc.nextInt();
        b=-a;
        System.out.println("Sign Changed: "+b);
    }
}

```

**23. Write a program to input two integers into two variables (say a and b) and interchange them by:**

**i) using a third variable (say c),**

**ii) without using a third variable.**

**Ans.** (i)

```

import java.util.*;
public class Question23i
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c;
        System.out.println("Enter two integers: ");
        a=sc.nextInt();
        b=sc.nextInt();

```

```

        System.out.println("Before interexchange: a="+a+",b="+b);
        c=a;
        a=b;
        b=c;
        System.out.println("After interexchange: a="+a+",b="+b);
    }
}

```

```

(ii)import java.util.*;
    public class Question23ii
    {
        static void main()
        {
            Scanner sc=new Scanner(System.in);
            int a,b;
            System.out.println("Enter two integers: ");
            a=sc.nextInt();
            b=sc.nextInt();
            System.out.println("Before interexchange: a="+a+",b="+b);
            a=a+b;
            b=a-b;
            a=a-b;
            System.out.println("After interexchange: a="+a+",b="+b);
        }
    }

```

**24. If the marks obtained by a student in five different subjects are input through the keyboard, write a program to find out the aggregate marks and percentage marks obtained by the student. Assume that the maximum marks that can be obtained by a student in each subject is 100.**

**Ans.** import java.util.\*;

```

public class Question24
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c,d,e,agg;
        float perc;
        System.out.println("Enter marks in 5 subjects: ");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        d=sc.nextInt();
        e=sc.nextInt();
        agg=a+b+c+d+e;
        perc=(float)agg/500*100;
        /*Assuming maximum marks
        in each subject is 100*/
    }
}

```

```

        System.out.println("Aggregate Marks="+agg);
        System.out.println("Percentage Marks="+perc);
    }
}

```

**25. If the total selling price of 15 items and the total profit earned on them is input through the keyboard, write a program to find the cost price of one item.**

**Ans.**

```

import java.util.*;
public class Question25
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float sp,profit,cp,cpOfOne;
        System.out.println("Enter total Selling Price of 15 items: ");
        sp=sc.nextFloat();
        System.out.println("Enter total profit of 15 items: ");
        profit=sc.nextFloat();
        cp=sp-profit; //Total Cost Price
        cpOfOne=cp/15; //Cost Price of one item.
        System.out.println("Cost Price Of one Item="+cpOfOne);
    }
}

```

**26. Write a program to input three integers and find their product without the '\*' operator.**

**Ans.**

```

import java.util.*;
public class Question26
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c,p;
        float f,e;
        System.out.println("Enter 3 integers: ");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        f=(float)1/b;
        e=(float)1/c;
        p=(int)(a/f);
        p=(int)(p/e);
        System.out.println("Product="+p);
    }
}

```

**27. Write a program to input an integer and raise it to the power of 2 and display it.**



**INPUT: Enter an integer: 4**

**OUTPUT: Power of 2 is: 16**

**INPUT: Enter an integer: 6**

**OUTPUT: Power of 2 is: 64**

**(Hint: use the shift operator)**

```
Ans. import java.util.*;
      public class Question27
      {
          static void main()
          {
              Scanner sc=new Scanner(System.in);
              int a,b;
              System.out.print("Enter an integer: ");
              a=sc.nextInt();
              b=2<<(a-1);
              System.out.println("Power of 2 is:"+b);
          }
      }
```

# Chapter 6: Conditional Structures

## Practice Questions

Answer the following questions:

**1. What are conditional statements? With which commands do you implement conditional statement in Java?**

**Ans.** Conditional Statements are those statements which contains a *test-expression* which may either result to **true** or **false**. Depending upon the result a statement or groups of statements are executed.

It is implemented through **if-else**, **if-else if-else**, **switch-case** or using ternary operator (?:).

**2. When are braces optional or necessary in conditional statements?**

**Ans.** When there is only one statement to be executed if the result of a *test-expression* is **true** or **false**, giving braces is optional.

**3. What are relational operators?**

**Ans.** **Relational operators** are used for comparison between two data items or quantities that are used in a test-expression/ condition.

**4. State the difference between = and ==.** **[ICSE 2007]**

**Ans.** '=' is used to assign a value to a variable, whereas '==' is used for comparison of two values for equality.

**5. What are logical operators? Explain each of them with the help of an example.**

**Ans.** **Logical operators** are used when there are more than one relational expression needs to be evaluated. It is categorised as AND(&&), OR(||) and NOT(!).

Logical Operator	Name	Behaviour
&&	AND operator	Checks whether the conditions preceding and succeeding it is true or not.
	OR operator	Checks whether either of the conditions preceding and succeeding it is true or not.
!	NOT operator	Just negates the logic of the condition succeeding it to check for its validity.

**6. What is the function of nested if statement?**

**Ans.** An if-else construct within an if or else construct or both is called nested if-else construct.

**7. What is a conditional operator? Explain with the help of an example.**

**Ans.** The Ternary operator is an alternative to if-else statement for a concise clean and compact code.

For example:

```
if(a>10)
    b=10;
else
    b=15;
```

using conditional operator may be written as:

b=(a>10)? 10:15;

### 8. Compare if and ? : .

**Ans.**

If-else	Ternary Operator
It is used when complex logic handling is necessary.	It is used to offer more concise, clean and compact code.
It can have multiple statements, multiple assignments and expressions (more than one statements) in its body.	It produces an expression, and hence a single value can be assigned or incorporated into a larger expression.
<b>Nested if</b> statement is simple and fairly easy to understand.	Ternary operator in its nested form, becomes complex and difficult to understand.

### 9. What is the function of switch statement?

**Ans.** using **if** statements to control the selection is perfectly okay for small programs but it becomes more complex when the number of alternative increases makes the code look more and more complex, at times may even confuse the programmer. Using **switch** statement, it is possible to successively tests the value of an expression against a list of integer or character constants as found in the **case** statement and when a match is found, a block of statements associated with that **case** gets executed.

### 10. State the rules which are to be followed while using switch statement.

**Ans.** Rules for switch-case statements are:

- A **case** statement cannot exist by itself outside of a switch block.
- The **break** statement should appear as the last statement in a **case** block.
- Braces should be used to denote the start and end of the switch block.
- Only character constants and Integer constants are used in the **case** constants but allows only for equality comparisons.
- All case labels in the same **switch** block may have unique values.

### 11. Explain, with the help of an example, the purpose of default in a switch statement. [ICSE 2005]

**Ans.** The *expression* in a **switch** statement is evaluated and its resultant values are matched against the values or constants specified in the **case** statements. When a match is found, the statement sequence associated with that case is executed until the **break** statement or the end of switch statement is reached. The **default** statement gets executed when no match is found. The **default** statement is optional, and if it is missing, no action takes place if all matches fail.

```
For example,  
switch(a)  
{  
    case 1:  
        System.out.println("One");  
        break;  
    case 2:
```

```

        System.out.println("Two");
        break;
    case 3:
        System.out.println("Three");
        break;
    default:
        System.out.println("Not 1,2 or 3");
}

```

In this code if the value of 'a' is neither 1,2 or 3, none of the **case** blocks gets executed and therefore the statement in the **default** block gets executed, which gives the output as:

**Not 1,2 or 3**

**12. Differentiate between if and switch statements.**

**[ICSE 2006]**

**Ans.**

If-else	switch case
It checks for both relational and logical operations.	It checks only for equality.
It may be used with almost any data type.	It is used only with integers and characters.
Comparison may be a variable or an expression or constant with another variable, expression or constant.	Comparison is always with a variable or expression with an integer or character constant.

**13. Explain with an example the if-else-if construct.**

**[ICSE 2007]**

**Ans.** Here is an example of if-else if construct

```

if (marks>=90)
    System.out.println("Grade A");
else if (marks>=70)
    System.out.println("Grade B");
else if (marks>=50)
    System.out.println("Grade C");
else if (marks>=30)
    System.out.println("Grade D");
else
    System.out.println("Grade E");

```

In this code if someone gets 95 the condition marks>=90 gets satisfied and *Grade A* gets printed. Again if marks is 85 the marks>=70, marks>=50 and marks>=30 all get satisfied but only marks>=70 will get executed as this is the first satisfying statement in the **elseif** adder and therefore *Grade B* gets executed. Similarly if marks is 55, *Grade C* gets printed. Now if marks is 15, none of the above conditions gets satisfied and therefore the statement in the else part gets executed, which prints *Grade E*.

## B. Do as directed in the following questions:

1. Using the given values of x, y and z evaluate the following (Answer in true / false).

$(x>=y) \parallel (!(z==y) \&\& (z<x))$

i) x=10, y=5, z=11

ii) x=10, y=10, z=10

iii) x=9, y=10, z=2

**Ans.** i) True

ii) True

iii) True

2. Using conditional operator write equivalent expressions for the following: [ICSE 2007]

i) F=0.5 if x=30 otherwise 0.

ii) Print "Even" if x is an even number otherwise print "Odd".

iii) if (income <= 10000)

tax = 0 ;

else

tax = 12 ;

**Ans.** i)  $F=(x==30)?0.5:0;$

ii)  $\text{System.out.println}((x\%2==0)? \text{"Even"}: \text{"Odd"});$

iii)  $\text{Tax}=(\text{income}\leq 10000)?0:12;$

3. Find the output of the following program segment, when:

[ICSE 2006]

i) val = 500

ii) val = 1600

int val, sum, n = 550;

sum = n + val > 1750? 400 : 200;

System.out.println(sum);

**Ans.** i) 200

ii) 400

4. Write the conditional statement for the following decision-making situation: Provided the value of x is greater than 9, increase the value of y by 15, otherwise decrease the value of y by 36.

**Ans.**  $(x>9)?y=y+15:y=y-36;$

5. Write equivalent Java statements for the following situations:

i) X is greater than equal to Y.

ii) X is not equal to Y.

iii) X is equal to Y.

iv) X is less than Y and X is less than Z.

v) Either X is greater than Y or X is less than Z.

vi) If X has a value greater than 20 then increment the value of J by 1 otherwise decrement the value by 2.

vii) It is not that X is greater than Y.

**Ans.** i)  $X>=Y$

ii)  $X!=Y$

iii)  $X==Y$

iv)  $X<Y \&\& X<Z$

v)  $X>Y \parallel X<Z$

- vi)  $(X > 20) ? J++ : J -= 2;$
- vii)  $!(X > Y)$

6. The following is a segment of a program:

[ICSE 2009]

```
x=1; y=1;
if(n>0)
{
    x=x+1;
    y=y-1;
}
```

What will be the value of x and y, if n assumes a value i) 1 ii) 0?

- Ans.** i) x=2 and y=0  
ii) x=1 and y=1

7. What is the output of the following program:

```
class First
{
    static void check( )
    {
        float num=110;
        System.out.print((num!=-num)?"OK":"NOT");
    }
}
```

**Ans.** ok

8. What is the output of the following program:

```
class Second
{
    static void check( )
    {
        int r, a=50, b=10;
        r=(a>45) ? a : b;
        System.out.print(r);
    }
}
```

**Ans.** 50

9. Rewrite the following set of if-else statement in terms of switch case statement for the following code fragment:

```
int code;
code=sc.nextInt( );
if (code==1)
    System.out.print("Accountant");
else
    if (code==5 || code==6)
        System.out.print("Grade IV");
else
    if (code==3)
        System.out.print("Financial Advisor");
```

**Ans.** int code;

```

code=sc.nextInt();

switch(code)
{
    case 1:
        System.out.print("Accountant");
        break;
    case 5:
    case 6:
        System.out.print("Grade IV");
        break;
    case 3:
        System.out.print("Financial Advisor");
}

```

**10. Find the syntax error(s), if any, in the following program and rewrite it after correction:**

```

Class First{
    public static void main(String args[]){
        int R; W=90;
        R=W-50;
        Switch(W)
        {
            20:system.out.print(Lower Range");
            30:System.Out.print("Middle Range")
            40:System.out.Print("Higher Range);
        }
    }
}

```

**Ans.**

```

class First{
    public static void main(String args[ ]){
        int R, W=90;
        R=W-50;
        switch(W)
        {
            case 20:
                System.out.print("Lower Range");
                break;
            case 30:
                System.out.print("Middle Range");
                break;
            case 40:
                System.out.print("Higher Range");
        }
    }
}

```

*All corrections made are underlined.*

**11. Give the output of the following switch statement:**

```

switch(num)
{
    case 10:
    case 100:
        System.out.println("Ten and hundred");
        break;
    case 1000:
    case 400:
        System.out.println("Thousand and 4 hundred");
        break;
    default:
        System.out.println("Invalid Input");
}

```

a) num=100

b) num=1400

c) num=1000

d) num=400

**Ans.** a) Ten and hundred  
b) Invalid Input  
c) Thousand and 4 hundred  
d) Thousand and 4 hundred

**12. Given the output of the following code fragment: [ICSE 2008]**

**when (i) opn='b' (ii) opn='x' (iii) opn='a'**

```

switch(opn)
{
    case 'a':
        System.out.println("Platform-Independent");
        break;
    case 'b':
        System.out.println("Object-Oriented");
    case 'c':
        System.out.println("Robust and Secure");
        break;
    default:
        System.out.println("Wrong Input");
}

```

**Ans.** i) Object-Oriented  
Robust and Secure  
ii) Wrong Input  
iii) Platform-Independent

**13. Write a program to input an integer and check whether it is greater than 100, or less than 100 or equal to 100. (use if-else if statement)**

**Ans.** import java.util.\*;  
class Question13  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int n;  
 System.out.println("Enter an integer:");



```

        n=sc.nextInt();
        if(n>100)
            System.out.println("Greater than 100");
        else if(n<100)
            System.out.println("Less than 100");
        else
            System.out.println("Equal to 100");
    }
}

```

**14. Write a program to input the Selling Price and Cost Price of a commodity and find the Profit or Loss made upon selling the product.**

**For example,**

**INPUT:**

Enter the Selling Price of the commodity: 456

Enter the Cost Price of the commodity: 400

**OUTPUT:**

Profit: 56

**INPUT:**

Enter the Selling Price of the commodity: 300

Enter the Cost Price of the commodity: 310

**OUTPUT:**

Loss: 10

**Ans.** import java.util.\*;

```
public class Question14
```

```
{
```

```
    static void main()
```

```
    {
```

```
        Scanner sc=new Scanner(System.in);
```

```
        float sp,cp,profit=0,loss=0;
```

```
        System.out.print("Enter the Selling Price of the commodity: ");
```

```
        sp=sc.nextFloat();
```

```
        System.out.print("Enter the Cost Price of the commodity: ");
```

```
        cp=sc.nextFloat();
```

```
        if(sp>cp)
```

```
        {
```

```
            profit=sp-cp;
```

```
            System.out.println("Profit:"+profit);
```

```
        }
```

```
        else if(cp>sp)
```

```
        {
```

```
            loss=cp-sp;
```

```
            System.out.println("Profit:"+loss);
```

```
        }
```

```
        else
```

```
            System.out.println("No profit or gain");
```

```
    }
```

```
}
```

**15. Write a program to check whether a int type number taken as input is a positive, negative or a zero. (Use if-else if statement)**

**Ans.**

```
import java.util.*;
public class Question15
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a;
        System.out.print("Enter an integer: ");
        a=sc.nextInt();
        if(a>0)
            System.out.println("Positive");
        else if(a<0)
            System.out.println("Negative");
        else
            System.out.println("Zero");
    }
}
```

**16. Write a program to check whether a short type number taken as input is an even number or an odd number.**

**Ans.**

```
import java.util.*;
public class Question16
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        short a;
        System.out.print("Enter an integer: ");
        a=sc.nextShort();
        if(a%2==0)
            System.out.println("Even Number");
        else
            System.out.println("Odd Number");
    }
}
```

**17. Write a program to accept an int type value as a parameter and print the square of it, if it is more than 120, otherwise print its cube.**

**Ans.**

```
import java.util.*;
public class Question17
{
    static void check(int n)
    {
        long s;
```

```

        if(n>120)
            s=n*n;
        else
            s=n*n*n;
        System.out.println("Answer="+s);
    }
}

```

**18. Write a program to input 5 real numbers (floating point) and find their average. If the average is greater than 50 print good, otherwise print bad.**

**Ans.** `import java.util.*;`  
`public class Question18`  
`{`  
 `static void main()`  
 `{`  
 `Scanner sc=new Scanner(System.in);`  
 `float a,b,c,d,e,avg;`  
 `System.out.print("Enter 5 real numbers: ");`  
 `a=sc.nextFloat();`  
 `b=sc.nextFloat();`  
 `c=sc.nextFloat();`  
 `d=sc.nextFloat();`  
 `e=sc.nextFloat();`  
 `avg=(a+b+c+d+e)/5;`  
 `if(avg>50)`  
 `System.out.println("GOOD");`  
 `else`  
 `System.out.println("BAD");`  
 `}`  
`}`

**19. Write a program to check whether an int type number (taken as input) is a multiple of 5 or not.**

**Ans.** `import java.util.*;`  
`public class Question19`  
`{`  
 `static void main()`  
 `{`  
 `Scanner sc=new Scanner(System.in);`  
 `int a;`  
 `System.out.print("Enter an integer: ");`  
 `a=sc.nextInt();`  
 `if(a%5==0)`  
 `System.out.println("Multiple of 5");`  
 `else`

```
        System.out.println("Not a multiple of 5");
    }
}
```

**20. Write a program to check whether an int type number (taken as input) is a 2-digit number or not.**

**Ans.**

```
import java.util.*;
public class Question20
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a;
        System.out.print("Enter an integer: ");
        a=sc.nextInt();
        if(a>=10 && a<=99 || a>=-99 && a<=-10)
            System.out.println("2-digit number");
        else
            System.out.println("Not a 2-digit number");
    }
}
```

**21. Write a program to check whether an int type number (taken as input) is a 3-digit number divisible by three or not.**

**Ans.**

```
import java.util.*;
public class Question21
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a;
        System.out.print("Enter an integer: ");
        a=sc.nextInt();
        if(a>=100 && a<=999 || a>=-999 && a<=-100)
        {
            if(a%3==0)
                System.out.println("3-digit number divisible by 3");
            else
                System.out.println("3-digit number but not divisible by 3");
        }
        else
            System.out.println("Not a 3-digit number");
    }
}
```

**22. Write a program to input an integer and check whether it is a 2-digit or a 3-digit number or not. In case it is not a 2-digit or a 3-digit number a relevant message ("Not 2 or 3 digit-number") should be displayed.**

**Ans.**

```
import java.util.*;
public class Question22
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a;
        System.out.print("Enter an integer: ");
        a=sc.nextInt();
        if(a>=10 && a<=99 || a>=-99 && a<=-10)
            System.out.println("2-digit number");
        else if(a>=100 && a<=999 || a>=-999 && a<=-100)
            System.out.println("3-digit number");
        else
            System.out.println("Not 2 or 3 digit number");
    }
}
```

**23. Write a program to input 2 integers and check whether both the numbers are multiples of 7 or not.**

**Ans.**

```
import java.util.*;
public class Question23
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b;
        System.out.print("Enter 2 integers: ");
        a=sc.nextInt();
        b=sc.nextInt();
        if(a%7==0 && b%7==0)
            System.out.println("Both are multiples of 7");
        else
            System.out.println("Both are not multiples of 7");
    }
}
```

**24. Write a program to input three integers and check whether all of them are unique (different) from each other or not.**

**Ans.**

```
import java.util.*;
public class Question24
{
    static void main()
    {
```

```

Scanner sc=new Scanner(System.in);
int a,b,c;
System.out.print("Enter 3 integers: ");
a=sc.nextInt();
b=sc.nextInt();
c=sc.nextInt();
if(a!=b && a!=c && c!=b)
    System.out.println("All are unique");
else
    System.out.println("All are not unique");
}
}

```

**25. Write a program to pass 2 integer numbers as parameters. If either of the two numbers is 0, display "invalid entry" and the program should end; if it is valid entry, divide the larger number with the smaller number and display the result.**

**Ans.** public class Question25

```

{
    static void main(int a,int b)
    {
        float c;
        if(a==0 || b==0)
            System.out.println("Invalid Entry");
        else
        {
            if(a>b)
                c=(float)a/b;
            else
                c=(float)b/a;
            System.out.println("Result="+c);
        }
    }
}
}

```

**26. Write a program to input an integer and check if the number is four-digit long or longer, the program should give a message that the number is a "large number". If it is a two-digit long or shorter, then the computer should provide a message that the number is "too small". Otherwise, print a message WELL DONE, WE THINK ALIKE.**

**Ans.** import java.util.\*;

```

public class Question26
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a;
        System.out.print("Enter an integer: ");
    }
}

```

```

a=sc.nextInt();
if(a>=1000)
    System.out.println("large number");
else if(a<=99)
    System.out.println("too small");
else
    System.out.println("WELL DONE WE THINK ALIKE");
}
}

```

**27. Write a program to input 3 unique integers and print the smallest among them.**

**Ans.** import java.util.\*;

```

public class Question27
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c;
        System.out.print("Enter 3 integerS: ");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        if(a<b && a<c)
            System.out.println("Smallest number:"+a);
        else if(b<a && b<c)
            System.out.println("Smallest number:"+b);
        else
            System.out.println("Smallest number:"+c);
    }
}

```

**28. Write a program to input 3 unique integers and print the smallest among them without using the logical operator && (AND).**

**Ans.** import java.util.\*;

```

public class Question28
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c;
        System.out.print("Enter 3 integerS: ");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        if(a<b)
        {
            if(a<c)

```

```

        System.out.println("Smallest number:"+a);
    else
        System.out.println("Smallest number:"+c);
    }
    else
    {
        if(b<c)
            System.out.println("Smallest number:"+b);
        else
            System.out.println("Smallest number:"+c);
        }
    }
}
}

```

**29. Write a program to accept three integer numbers and check whether all of them are even numbers or not; if they are, it should print "All are even", otherwise check whether all three are odd or not, if they are it should print "All are odd". If none of the above conditions are true, it should print "They are mixed".**

**Ans.** `import java.util.*;`  
`public class Question29`  
`{`  
 `static void main()`  
 `{`  
 `Scanner sc=new Scanner(System.in);`  
 `int a,b,c;`  
 `System.out.print("Enter 3 integerS: ");`  
 `a=sc.nextInt();`  
 `b=sc.nextInt();`  
 `c=sc.nextInt();`  
 `if(a%2==0 && b%2==0 && c%2==0)`  
 `System.out.println("All are even");`  
 `else if(a%2!=0 && b%2!=0 && c%2!=0)`  
 `System.out.println("All are odd");`  
 `else`  
 `System.out.println("They are mixed");`  
 `}`  
 `}`  
`}`

**30. Write a program to accept 3 integer numbers and check whether the last digit of every number is 7 or not. If it is, it should print "The last digit of all the numbers is 7", otherwise it should print, "The last digit is not 7 for every number".**

**Ans.** `import java.util.*;`  
`public class Question30`  
`{`  
 `static void main()`  
 `{`  
 `Scanner sc=new Scanner(System.in);`



```

int a,b,c;
System.out.print("Enter 3 integers: ");
a=sc.nextInt();
b=sc.nextInt();
c=sc.nextInt();
if(a%10==7 && b%10==7 && c%10==7)
    System.out.println("The last digit of all the numbers is 7");
else
    System.out.println("The last digit is not 7 for every number");
}
}

```

**31 Write a program to input a year and check whether it is a leap year or not.**

```

import java.util.*;
public class Question31
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int y;
        System.out.print("Enter a year: ");
        y=sc.nextInt();
        if(y%400==0 || (y%4==0 && y%100!=0))
            System.out.println("Leap Year");
        else
            System.out.println("Not a Leap Year");
    }
}

```

**32. Write a program to accept three angles of a triangle as parameter and check whether it can form a triangle or not. If it forms a triangle, check whether it is an acute-angled, obtuse-angled or right angled triangle.**

**(Hint: To form a triangle, the sum of the angles should be 180 degrees.**

**To form an acute angled triangle every angle should be less than 90 degrees.**

**To form an obtuse angled triangle either of the angle should be more than 90 degrees.**

**To form a right angled triangle either of the angle should be equal to 90 degrees.)**

**Ans.** public class Question32

```

{
    static void main(int a,int b,int c)
    {
        if(a+b+c==180)
        {
            if(a<90 && b<90 && c<90)
                System.out.println("Acute angled triangle");
            else if(a>90 || b>90 || c>90)
                System.out.println("Obtuse angled triangle");
            else

```

```

        System.out.println("Right angled triangle");
    }
    else
        System.out.println("Do not form a triangle");
    }
}

```

**33. Write a program to input the three angles of a triangle and check whether it forms a triangle or not, if it forms a triangle, check whether it is an equilateral, isosceles or a scalene triangle.**

**(Hint: To form a triangle, the sum of the angles should be 180 degrees.**

**To form an equilateral triangle every angle should be equal.**

**To form an isosceles triangle any two angles should be equal.**

**To form a scalene triangle all three angles should be different from each other.)**

**Ans.** import java.util.\*;

```

public class Question33
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c;
        System.out.print("Enter 3 angles: ");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        if(a+b+c==180)
        {
            if(a==b && b==c)
                System.out.println("Equilateral Triangle");
            else if(a==b || b==c || c==a)
                System.out.println("Isosceles Triangle");
            else
                System.out.println("Scalene Triangle");
        }
        else
            System.out.println("Do not form a Triangle");
    }
}

```

**34. Write a program to input the three sides of a triangle and check whether it forms a triangle or not, if it forms a triangle, check whether it is an equilateral, isosceles or a scalene triangle.**

**(Hint: To form a triangle, each side should be less the sum of the other two sides.**

**To form an equilateral triangle every side should be equal.**

**To form an isosceles triangle any two sides should be equal.**

**To form a scalene triangle all three sides should be different from each other.)**

**Ans.** import java.util.\*;

```

public class Question34

```

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c;
        System.out.print("Enter 3 sides: ");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        if(a<b+c && b<c+a && c<a+b)
        {
            if(a==b && b==c)
                System.out.println("Equilateral Triangle");
            else if(a==b || b==c || c==a)
                System.out.println("Isosceles Triangle");
            else
                System.out.println("Scalene Triangle");
        }
        else
            System.out.println("Do not form a Triangle");
    }
}

```

**35. Write a program to accept three sides of a triangle as parameter and check whether it can form a triangle or not. If it forms a triangle, check whether it is an acute-angled, obtuse-angled or right angled triangle.**

**(Hint: To form a triangle, each side should be less the sum of the other two sides.**

**To form an acute-angled triangle the square of every side should be less than the sum of the squares of the other two sides.**

**To form an obtuse-angled triangle the square of any side should be greater than the sum of the squares of the other two sides.**

**To form a right-angled triangle, the square of any side should be equal to the sum of the squares of the other two sides.)**

**Ans.** import java.util.\*;

```

public class Question35
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c;
        System.out.print("Enter 3 sides: ");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        if(a<b+c && b<c+a && c<a+b)
        {
            if(a*a<b*b+c*c && b*b<a*a+c*c && c*c<a*a+b*b)

```

```

        System.out.println("Acute angled Triangle");
    else if(a*a>b*b+c*c || b*b>a*a+c*c || c*c>a*a+b*b)
        System.out.println("Obtuse angled Triangle");
    else
        System.out.println("Right angled Triangle");
    }
else
    System.out.println("Do not form a Triangle");
}
}

```

**36. Write a program to accept a mark obtained by a student in computer science and print the grades accordingly:**

Marks	Grade
Above 90	A
70 to 89	B
50 to 69	C
below 50	D

**Ans.**

```

import java.util.*;
public class Question36
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int m;
        System.out.print("Enter marks in Computer Science: ");
        m=sc.nextInt();
        if(m>=90)
            System.out.println("Grade A");
        else if(m>=70 && m<=89)
            System.out.println("Grade B");
        else if(m>=50 && m<=69)
            System.out.println("Grade C");
        else
            System.out.println("Grade D");
    }
}

```

**37. A bike shop hires bikes by the day at different rates throughout the year. The following table shows how the hire charge varies throughout the year:**

Season	Charge in Rupees
Spring (March May)	13.00
Summer (June August)	19.50
Autumn (September November)	8.75
Winter (December February)	5.00

The proprietor also gives a discount on the number of days a bike is hired for. If the hire period is greater than 7 days, then a reduction of 25% is made. For every bike hired a deposit of rupees 50.00 must be paid. Write a program to input the season and number of days the bike is required for and output the hire charge including the deposit.

```

Ans. import java.util.*;
public class Question37
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int days;
        float rate=0, hireCharge;
        String season;
        System.out.print("Enter the season: ");
        season=sc.nextLine();
        System.out.print("Enter the number of days the bike is hired for: ");
        days=sc.nextInt();
        if(season.equals("Spring")) //equals is used to check equality of strings
            rate=13.00f;
        else if(season.equals("Summer"))
            rate=19.50f;
        else if(season.equals("Autumn"))
            rate=8.75f;
        else if(season.equals("Winter"))
            rate=5.00f;
        hireCharge=rate*days;
        if(days>7)
            hireCharge=hireCharge-25/100f*hireCharge;
        hireCharge=50+hireCharge;
        System.out.println("Hire Charge:"+hireCharge);
    }
}

```

**38.** A cloth showroom has announced the following festival discounts on the purchase of items, based on the total cost of the items purchased:

Total Cost	Discount (in Percentage)
Less than ₹ 2000	5%
₹ 2001 to ₹ 5000	25%
₹ 5001 to ₹ 10000	35%
Above ₹ 10000	50%

**Write a program to input the total cost and compute and display the amount to be paid by the customer after availing the discount. [ICSE 2006]**

```

Ans. import java.util.*;
        public class Question38
        {
            static void main()
            {
                Scanner sc=new Scanner(System.in);
                int total;
                float dis, amt;
                System.out.print("Enter the Total Cost: ");
                total=sc.nextInt();
                if(total<=2000)
                    dis=5/100f*total;
                else if(total>=2001 && total<=5000)
                    dis=25/100f*total;
                else if(total>=5001 && total<=10000)
                    dis=35/100f*total;
                else
                    dis=50/100f*total;
                amt=total-dis;
                System.out.println("Amount Payable:"+amt);
            }
        }

```

**39. An electricity company charges their consumers according to the units consumed per month according to the given tariff:**

Units Consumed	Charges
Up to 100 units	₹ 2 per unit
More than 100 units and up to 200 units	₹ 1.80 per unit
More than 200 units	₹ 1.50 per unit

**In addition to the above, every consumer has to pay Rs.200 as service charge per month. Write a program to input the amount of units consumed and calculate the total charges payable (bill) by the consumer.**

```

Ans. import java.util.*;
        public class Question39

```

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int units;
        float rate, charge;
        System.out.print("Enter the Total Cost: ");
        units=sc.nextInt();
        if(units<=100)
            rate=2;
        else if(units>100 && units<=200)
            rate=1.80f;
        else
            rate=1.50f;
        charge=200+rate*units;
        System.out.println("Amount Payable:Rs "+charge);
    }
}

```

**40. A library charges a fine for books returned late. Following are the fines:**

**First five days : 40 paise per day.**

**Six to ten days : 65 paise per day.**

**Above ten days : 80 paise per day.**

**Design a program to calculate the fine assuming that a book is returned n days late.**

**[ICSE 1996]**

**Ans.** import java.util.\*;  
public class Question40

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        float fine;
        System.out.print("Enter the number of days late: ");
        n=sc.nextInt();
        if(n<=5)
            fine=0.40f*n;
        else if(n>=6 && n<=10)
            fine=0.65f*n;
        else
            fine=0.80f*n;
        System.out.println("Fine Payable:Rs "+fine);
    }
}

```

**41. Write a program to find the result of the following functions:**

**$F(x)=4x+x+y$  if  $x>0$**

**$F(x)= 4x-x-y$  if  $x<0$**

**Both x and y should be of float data type and should be taken as input.**

**Ans.** `import java.util.*;`  
`public class Question41`

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float x,y,f;
        System.out.print("Enter the value of x and y: ");
        x=sc.nextFloat();
        y=sc.nextFloat();
        if(x>0)
            f=4*x+y;
        else if(x<0)
            f=4*x-y;
        else
            f=0;
        System.out.println("Value of the function: "+f);
    }
}
```

**42. Write a program to input an integer and check whether it is a 5-digit number or not. If it is, extract the central digit and print it.**

**Example:**

**INPUT: Enter an integer: 76549**

**OUTPUT: Central digit: 5**

**INPUT: Enter an integer: 12976**

**OUTPUT: Central digit: 9**

**INPUT: Enter an integer: 126**

**OUTPUT: Not a 5-digit number**

**Ans.** `import java.util.*;`  
`public class Question42`

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n,d;
        System.out.print("Enter an integer: ");
        n=sc.nextInt();
        if(n>=10000 && n<=99999)
```



```

    {
        d=(n/100)%10;
        System.out.println("Central digit :"+d);
    }
    else
        System.out.println("Not a 5-digit number");
}
}

```

**43. Write a program to input an integer and check whether it is a 4-digit number or not. If it is, find the sum of the first and last digit.**

**Example**

```

INPUT: Enter an integer: 4765
OUTPUT: Sum of the first and last digit: 9
INPUT: Enter an integer: 7839
OUTPUT: Sum of the first and last digit: 16
INPUT: Enter an integer: 126
OUTPUT: Not a 4-digit number

```

**Ans.** `import java.util.*;`  
`public class Question43`

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n,f,l,s;
        System.out.print("Enter an integer: ");
        n=sc.nextInt();
        if(n>=1000 && n<=9999)
        {
            f=n/1000; //first digit
            l=n%10; //last digit
            s=f+l;
            System.out.println("Sum of first and last digit :"+s);
        }
        else
            System.out.println("Not a 4-digit number");
    }
}

```

**44. Write a program to input an integer and check whether it is a 3-digit number or not. If it is check whether the sum of the first and last digit is an even or an odd number.**

**Example**

```

INPUT: Enter an integer: 476
OUTPUT: Sum of the first and last digit is an even number

```

**INPUT: Enter an integer: 784****OUTPUT: Sum of the first and last digit is an odd number**

**INPUT: Enter an integer: 12**

**OUTPUT: Not a 3-digit number**

**Ans.** import java.util.\*;

```
public class Question44
```

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n,f,l,s;
        System.out.print("Enter an integer: ");
        n=sc.nextInt();
        if(n>=100 && n<=999)
        {
            f=n/100; //first digit
            l=n%10; //last digit
            s=f+l;
            if(s%2==0)
                System.out.println("Sum of the first and last digit is an even number");
            else
                System.out.println("Sum of the first and last digit is an odd number");
        }
        else
            System.out.println("Not a 3-digit number");
    }
}
```

**45. Write a program to input an integer and check whether it is a 4-digit number or not. If it is, divide it into two equal halves and find their sum.**

**Example**

**INPUT: Enter an integer: 4762**

**OUTPUT: Sum of the 2 halves: 109**

**[Hint: 47+62=109]**

**INPUT: Enter an integer: 9863**

**OUTPUT: Sum of the 2 halves: 161**

**[Hint: 98+63=161]**

**INPUT: Enter an integer: 12**

**OUTPUT: Not a 4-digit number**

**Ans.** import java.util.\*;

```
public class Question45
```

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
```

```

int n,f,l,s;
System.out.print("Enter an integer: ");
n=sc.nextInt();
if(n>=1000 && n<=9999)
{
    f=n/100; //first 2 digits
    l=n% 100; //last 2 digits
    s=f+l;
    System.out.println("Sum of the 2 halves:"+s);
}
else
    System.out.println("Not a 4-digit number");
}
}

```

**46. Write a program to input an integer and using ternary operator check whether it is an even or an odd number.**

**Ans.** import java.util.\*;  
public class Question46

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        System.out.print("Enter an integer: ");
        n=sc.nextInt();
        System.out.println((n%2==0)?"Even":"Odd");
    }
}

```

**47. Write a program to input an integer and using ternary operator check whether it is a multiple of 5 or not.**

**Ans.** import java.util.\*;  
public class Question47

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        System.out.print("Enter an integer: ");
        n=sc.nextInt();
        System.out.println((n%5==0)?"Multiple of 5":"Not a multiple of 5");
    }
}

```

**48. Write a program to input an integer and using ternary operator check whether it is a positive number, negative number or a zero or not.**

**Ans.**

```
import java.util.*;
public class Question48
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        System.out.print("Enter an integer: ");
        n=sc.nextInt();
        System.out.println((n>0)?"Positive":(n<0)?"Negative":"Zero");
    }
}
```

**49. Write a program to input two integers and using ternary operator print the larger among the two, in case both are equal it should display "Equal".**

**Ans.**

```
import java.util.*;
public class Question49
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b;
        System.out.print("Enter 2 integers: ");
        a=sc.nextInt();
        b=sc.nextInt();
        System.out.println((a>b)?"Largest="+a:(b>a)?"Largest="+b:"Equal");
    }
}
```

**50. Write a program to input two integers and if both the numbers are even find their sum else find their product, using conditional operator only.**

**Ans.**

```
import java.util.*;
public class Question50
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
```

```

int a,b;
System.out.print("Enter 2 integers: ");
a=sc.nextInt();
b=sc.nextInt();
System.out.println((a%2==0&& b%2==0)?"Sum="+a+b:"Product="+a*b);
}
}

```

**51. Write a program to input three integers and print the largest among the three integers using ternary operator.**

**Ans.**

```

import java.util.*;
public class Question51
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c,l;
        System.out.print("Enter 3 integers: ");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        l=(a>b)?(a>c)?a:c:(b>c)?b:c;
        System.out.println("Largest="+l);
    }
}

```

**52. Using a switch statement, write a menu-driven program to convert a given temperature from Fahrenheit to Celsius and vice versa. For an incorrect choice, an appropriate error message should be displayed.**

(Hint :  $C = \frac{5}{9} \times (F - 32)$  and  $F = 1.8 \times C + 32$ )

[ICSE 2007]

**Ans.**

```

import java.util.*;
public class Question52
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float f,c;
        int ch;
        System.out.println("MENU DRIVEN");
        System.out.println("1. to change temperature from Fahrenheit to Celcius");
        System.out.println("2. to change temperature from Celcius to Fahrenheit");
        System.out.println("Enter your choice:");
    }
}

```

```

ch=sc.nextInt();
switch(ch)
{
    case 1:
        System.out.println("Enter temperature in Fahrenheit:");
        f=sc.nextFloat();
        c=5/9f*(f-32);
        System.out.println("Temperature in Celcius="+c);
        break;
    case 2:
        System.out.println("Enter temperature in Celcius:");
        c=sc.nextFloat();
        f=1.8f*c+32;
        System.out.println("Temperature in Fahrenheit="+f);
        break;
    default:
        System.out.println("Invalid Choice!");
}
}
}

```

**53. Write a program to input two numbers and find their sum, difference or product according to user's choice (menu-driven).**

**Ans.** import java.util.\*;  
public class Question53  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 float a,b,c;  
 int ch;  
 System.out.println("Enter 2 numbers:");  
 a=sc.nextFloat();  
 b=sc.nextFloat();  
 System.out.println("MENU");  
 System.out.println("1. to find the SUM");  
 System.out.println("2. to find the DIFFERENCE");  
 System.out.println("3. to find the PRODUCT");  
 System.out.println("Enter your choice:");  
 ch=sc.nextInt();  
 switch(ch)  
 {  
 case 1:  
 c=a+b;  
 System.out.println("Sum="+c);

```

        break;
        case 2:
        c=a-b;
        System.out.println("Difference="+c);
        break;
        case 3:
        c=a*b;
        System.out.println("Product="+c);
        break;
        default:
        System.out.println("Invalid Choice!");
    }
}
}

```

**54. Write a menu-driven program to find:**

**i) Area of a rectangle, where length and breadth is taken as input. (Area=length \*breadth)**

**ii) Area of a circle, where radius is taken as input. (Area =  $\frac{22}{7}$  \*radius<sup>2</sup>)**

**Ans.**

```

import java.util.*;
public class Question54
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float l,b,r,a;
        int ch;
        System.out.println("MENU");
        System.out.println("1. Area of a Rectangle ");
        System.out.println("2. Area of a Circle");
        System.out.println("Enter your choice:");
        ch=sc.nextInt();
        switch(ch)
        {
            case 1:
            System.out.println("Enter the length and breadth of the rectangle:");
            l=sc.nextFloat();
            b=sc.nextFloat();
            a=l*b;
            System.out.println("Area of the rectangle="+a);
            break;
            case 2:
            System.out.println("Enter the radius of the circle:");
            r=sc.nextFloat();
            a=22/7f*r*r;

```

```

        System.out.println("Area of the circle="+a);
        break;
        default:
        System.out.println("Invalid Choice!");
    }
}
}

```

**55. Write a menu-driven program to input a day number between 1 and 7 and print the corresponding weekday, i.e. for 1 display Sunday, for 2 display Monday for 3 display Tuesday, etc. [ICSE 2002]**

**Ans.** `import java.util.*;`  
`public class Question55`  
`{`  
 `static void main()`  
 `{`  
 `Scanner sc=new Scanner(System.in);`  
 `int n;`  
 `System.out.println("Enter any day number between 1 to 7");`  
 `n=sc.nextInt();`  
 `switch(n)`  
 `{`  
 `case 1:`  
 `System.out.println("Sunday");`  
 `break;`  
 `case 2:`  
 `System.out.println("Monday");`  
 `break;`  
 `case 3:`  
 `System.out.println("Tuesday");`  
 `break;`  
 `case 4:`  
 `System.out.println("Wednesday");`  
 `break;`  
 `case 5:`  
 `System.out.println("Thursday");`  
 `break;`  
 `case 6:`  
 `System.out.println("Friday");`  
 `break;`  
 `case 7:`  
 `System.out.println("Saturday");`  
 `break;`  
 `default:`  
 `System.out.println("Invalid day number");`  
 `}`  
 `}`  
`}`



```
}  
}  
}
```

**56. Write a menu driven program to input a month number between 1 and 12 and print the corresponding month name, i.e. for 1 display January, for 2 display February for 3 display March, etc.**

**Ans.**

```
import java.util.*;  
public class Question56  
{  
    static void main()  
    {  
        Scanner sc=new Scanner(System.in);  
        int n;  
        System.out.println("Enter any month number between 1 to 12");  
        n=sc.nextInt();  
        switch(n)  
        {  
            case 1:  
                System.out.println("January");  
                break;  
            case 2:  
                System.out.println("February");  
                break;  
            case 3:  
                System.out.println("March");  
                break;  
            case 4:  
                System.out.println("April");  
                break;  
            case 5:  
                System.out.println("May");  
                break;  
            case 6:  
                System.out.println("June");  
                break;  
            case 7:  
                System.out.println("July");  
                break;  
            case 8:  
                System.out.println("August");  
                break;  
            case 9:  
                System.out.println("September");  
                break;
```

```

    case 10:
    System.out.println("October");
    break;
    case 12:
    System.out.println("December");
    break;
    default:
    System.out.println("Invalid day number");
    }
}
}

```

**57. Write a program using switch-case to accept an integer number and print the last digit in words.**

**For example**

Input	Output
47	Seven
93	Three
81	One

**Ans.**

```

import java.util.*;
public class Question57
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        System.out.println("Enter a number");
        n=sc.nextInt();
        switch(n%10)
        {
            case 0:
            System.out.println("Zero");
            break;
            case 1:
            System.out.println("One");
            break;
            case 2:
            System.out.println("Two");
            break;
            case 3:
            System.out.println("Three");
            break;

```

```

        case 4:
        System.out.println("Four");
        break;
        case 5:
        System.out.println("Five");
        break;
        case 6:
        System.out.println("Six");
        break;
        case 7:
        System.out.println("Seven");
        break;
        case 8:
        System.out.println("Eight");
        break;
        case 9:
        System.out.println("Nine");
        break;
        default:
        System.out.println("Invalid number");
    }
}
}

```

**58. Write a program to input an integer and check whether it is divisible by 3 or not using switch case only (no if-else statement).**

**Ans.** import java.util.\*;

```

public class Question58
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        System.out.println("Enter a number");
        n=sc.nextInt();
        switch(n%3)
        {
            case 0:
                System.out.println("Divisible by 3");
                break;
            default:
                System.out.println("Not divisible by 3");
        }
    }
}

```

**59.** The 1st day of 2010 was Friday, write a program to input any day number within the month of January and print which day was it. Also check whether the day number entered is a valid date or not.

For example,

**Input:**

**Enter day number: 20**

**Output: Wednesday**

**Input:**

**Enter day number: 30**

**Output: Saturday**

**Input:**

**Enter day number: 34**

**Output: Invalid day number**

**Ans.**

```
import java.util.*;
public class Question59
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        System.out.println("Enter the day number");
        n=sc.nextInt();
        switch(n%7)
        {
            case 1:
                System.out.println("Friday");
                break;
            case 2:
                System.out.println("Saturday");
                break;
            case 3:
                System.out.println("Sunday");
                break;
            case 4:
                System.out.println("Monday");
                break;
            case 5:
                System.out.println("Tuesday");
                break;
            case 6:
                System.out.println("Wednesday");
                break;
            case 0:
                System.out.println("Thursday");
                break;
        }
    }
}
```

```

        default:
        System.out.println("Invalid day number");
    }
}
}

```

**60. Write a program to input an integer and check whether it is a magic number or not. A magic number is such a number whose continuous sum of the digits until it forms a single digit is 1. For example, [55] = 5+5 = [10] = 1+0 = [1], therefore 55 is a magic number. (Remember no loop is to be used).**

**Ans.**

```

import java.util.*;
public class Question60
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        System.out.println("Enter a number");
        n=sc.nextInt();
        if(n%9==1)
            System.out.println("Magic Number");
        else
            System.out.println("Not a Magic Number");
    }
}

```

**61. Write a program to input 3 integers and using conditional operator print the largest among them.**

**Ans.**

```

import java.util.*;
public class Question61
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c,l;
        System.out.print("Enter 3 integers: ");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        l=(a>b)?(a>c)?a:c:(b>c)?b:c;
        System.out.println("Largest="+l);
    }
}

```

**62. Write a program to input an integer and check whether it is a 4-digit, 3-digit, 2-digit or single-digit positive number or not. Do not use any logical operator in the program.**

**Ans.**

```
import java.util.*;
public class Question62
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a;
        System.out.print("Enter an integer: ");
        a=sc.nextInt();
        if(a>=10)
        {
            if(a<=99)
                System.out.print("Two Digit Number");
            else if(a<=999)
                System.out.print("Three Digit Number");
            else if(a<=9999)
                System.out.print("Four Digit Number");
            else
                System.out.print("Not a valid 2-digit, 3-digit or 4-digit Number");
        }
        else
            System.out.print("Not a valid 2-digit, 3-digit or 4-digit Number");
    }
}
```

**63. Write a menu-driven program to check whether a number is i) even or odd, ii) multiple of 17 or not, iii) two-digit number or not.**

**Ans.**

```
import java.util.*;
public class Question63
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a;
        int ch;
        System.out.println("Enter a number:");
        a=sc.nextInt();
        System.out.println("MENU");
        System.out.println("1. to check EVEN or ODD");
```

```

System.out.println("2. to check MULTIPLE OF 17 or not");
System.out.println("3. to check TWO DIGIT NUMBER or not");
System.out.println("Enter your choice:");
ch=sc.nextInt();
switch(ch)
{
    case 1:
        if(a%2==0)
            System.out.println("Even");
        else
            System.out.println("Odd");
        break;
    case 2:
        if(a%17==0)
            System.out.println("Multiple of 17");
        else
            System.out.println("Not Multiple of 17");
        break;
    case 3:
        if(a>=10 && a<=99 || a>=-99 && a<=-10)
            System.out.println("Two digit number");
        else
            System.out.println("Not a two digit number");
        break;
    default:
        System.out.println("Invalid Choice!");
}
}
}

```

**64. Write a program to input an integer and check whether it is positive, negative or a zero, using conditional operator only.**

**Ans.** import java.util.\*;  
public class Question64

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        System.out.print("Enter an integer: ");
        n=sc.nextInt();
        System.out.println((n>0)?"Positive":(n<0)?"Negative":"Zero");
    }
}

```

**65. Write a program to input 5 real numbers and print only those numbers which are less than their average.**

**Ans.** import java.util.\*;  
public class Question65

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float a,b,c,d,e,av;
        System.out.print("Enter 5 real numbers: ");
        a=sc.nextFloat();
        b=sc.nextFloat();
        c=sc.nextFloat();
        d=sc.nextFloat();
        e=sc.nextFloat();
        av=(a+b+c+d+e)/5;
        if(a<av)
            System.out.println(a+" is less than "+av);
        if(b<av)
            System.out.println(b+" is less than "+av);
        if(c<av)
            System.out.println(c+" is less than "+av);
        if(d<av)
            System.out.println(d+" is less than "+av);
        if(e<av)
            System.out.println(e+" is less than "+av);
    }
}
```

**66. Write a program to input three integers and check whether it forms a Pythagorean triplet or not. A set of three integers is said to be a Pythagoren triplet if the sum of the squares of any two integers is equal to square of the third integer. Example, (3, 4, 5), (5, 12, 13), and (7, 24, 25). [ICSE 1994]**

**Ans.** import java.util.\*;  
public class Question66  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int a,b,c;  
 System.out.println("Enter 3 numbers:");



```

a=sc.nextInt();
b=sc.nextInt();
c=sc.nextInt();
if(a*a==b*b+c*c || b*b==a*a+c*c || c*c==a*a+b*b)
    System.out.println("Pythagorean Triplet");
else
    System.out.println("Not a Pythagorean Triplet");
}
}

```

**67. Write a program to find the gross salary of an employee for the following allowances and deductions. Use meaningful variables with the class name Employee and Basic Pay as an input.**

Dearness Allowance	25% of Basic Pay
House Rent Allowance	15% of Basic Pay
Provident Fund	8.33% of Basic Pay
Net Pay	Basic Pay + Dearness Allowance + House Rent Allowance
Gross Pay	Net Pay – Provident Fund

[ICSE 2005]

**Ans.** //Question 67

```

import java.util.*;
public class Employee
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        double basicPay,da,hra,pf,netPay,grossPay;
        System.out.println("Enter the Basic Pay:");
        basicPay=sc.nextDouble();
        da=25/100.0*basicPay;
        hra=15/100.0*basicPay;
        pf=8.33/100*basicPay;
        netPay=basicPay+da+hra;
        grossPay=netPay-pf;
        System.out.println("Gross Pay="+grossPay);
    }
}

```

**68. To foster a sense of water conservation, the water department has an annual water conservation tax policy. The taxes are based on the water consumption of the consumer. The tax rates are as follows:**

Water Consumed (in Gallons)	Tax Rate in ₹ /100 Gallons
Upto 45	No Tax
More than 45 but 75 or less	475.00
More than 75 but 125 or less	750.00
More than 125 but 200 or less	1225.00
More than 200 but 350 or less	1650.00
More than 350	2000.00

**Ans.**

```

import java.util.*;
public class Qusetion68
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int consumed;
        double tax;
        System.out.println("Enter water consumed in Gallons:");
        consumed=sc.nextInt();
        if(consumed<=45)
            tax=0;
        else if(consumed<=75)
            tax=475.00;
        else if(consumed<=125)
            tax=750.00;
        else if(consumed<=200)
            tax=1225.00;
        else if(consumed<=350)
            tax=1650.00;
        else
            tax=2000.00;
        System.out.println("Tax Payable="+tax);
    }
}

```

**69. Write a program to input the water consumed annually in gallons and output the tax payable. [ICSE 2003]**

**Ans.** Do it yourself.

**70. Write a program using a function called area( ) to compute the area of a :**

i) circle( $\pi*r^2$ ) where  $\pi = 3.14$

ii) square(side\*side)

iii) rectangle (length\*breadth)

**Display the menu to output the area as per User's choice.**

**[ICSE 2005]**

```
Ans. import java.util.*;
public class Question70
{
    static void area()
    {
        Scanner sc=new Scanner(System.in);
        int ch;
        float radius,length,breadth,side,area;
        System.out.println("MENU");
        System.out.println("1.To find the area of a circle");
        System.out.println("2.To find the area of a square");
        System.out.println("3.To find the area of a rectangle");
        System.out.println("Enter your choice");
        ch=sc.nextInt();
        switch(ch)
        {
            case 1:
                System.out.println("Enter the radius of the Circle:");
                radius=sc.nextFloat();
                area=3.14f*radius*radius;
                System.out.println("Area="+area);
                break;
            case 2:
                System.out.println("Enter the side of the Square:");
                side=sc.nextFloat();
                area=side*side;
                System.out.println("Area="+area);
                break;
            case 3:
                System.out.println("Enter the length & breadth of the Rectangle:");
                length=sc.nextFloat();
                breadth=sc.nextFloat();
                area=length*breadth;
                System.out.println("Area="+area);
                break;
            default:
                System.out.println("Invalid Choice!");
        }
    }
}
```

**71. Write a program to input an integer and check whether it is a three-digit number or not. If it is print the largest digit in the number.**

**Example**

**INPUT: Enter an integer: 476**

**OUTPUT: Largest digit: 7**

**INPUT: Enter an integer: 986**

**OUTPUT: Largest digit: 9**

**INPUT: Enter an integer: 12**

**OUTPUT: Not a 3 digit-number**

```
Ans. import java.util.*;
public class Question71
{
    static void area()
    {
        Scanner sc=new Scanner(System.in);
        int a,first,middle,last;
        System.out.println("Enter an integer:");
        a=sc.nextInt();
        if (a>=100 && a<=999)
        {
            first=a/100;
            middle=(a/10)% 10;
            last=a% 10;
            if(first>middle && first>last)
                System.out.println("Largest Digit:"+first);
            else if(middle>first && middle>last)
                System.out.println("Largest Digit:"+middle);
            else
                System.out.println("Largest Digit:"+last);
        }
        else
            System.out.println("Not a valid 3-digit number");
    }
}
```

# Chapter 7: Repetitive Structures

## Practice Questions

### A. Answer the following questions:

1. Explain the term loop with an example.

[ICSE 2005]

**Ans.** Loop is a repetitive structure that allows a group of statements to be executed repeatedly. The loop structure executes as long as the given *test-expression* is executed. For example, the following **for** loop will execute 10 times displaying the first 10 natural numbers.

```
int i;
for( i =1 ; i<=10 ; i++)
{
    System.out.print(i+" ");
}
```

The loop starts with the initial value of 1 for 'i' and iterates every time increasing the value of 'i' by 1 as long as the value of 'i' is less than or equal to 10. The loop body in the meanwhile displays each value of 'i', thus giving the output as:

1    2    3    4    5    6    7    8    9    10

### 2. Discuss "Elements That Control a Loop".

**Ans.** Elements that control the loop are:

**1. Initialization Expression(s):** Before entering a loop, the variable that controls the execution of it is called initialization. The initialization of a variable may be inside a loop or outside it. Whatever it may be it is executed only once before the iteration starts.

**2. Test Expression.** The test-expression is a condition, which decides whether the loop body will be executed or not. Till the test-expression evaluates to **true**, the loop body gets executed otherwise the loop is terminated.

**3. Update Expression(s).** The update expression(s) change the value(s) of the control variable(s). The update expression(s) is usually made to be executed; before the next iteration.

**4. The Body of the Loop.** The statements that are executed repeatedly (as long as the, test-expression is true) form the body of the loop.

### 3. What are the three types of looping statements available in Java?

**Ans.** The three types of looping statements in Java are:

a) for

- b) while
- c) do-while

**4. Classify loops depending upon the priority in which a test expression is evaluated.**

**Ans.** Depending upon the priority of a test-expression loops may be categorized as:

- a) Entry-controlled loop
- b) Exit-controlled loop

**5. Give the general syntax of a for loop and discuss each of its components.**

**Ans.** The general syntax of **for** loop is:

```
for (initialization expression(s) ; test-expression ; update expression(s))
{
    //body of the loop;
}
```

- **Initialization Expression(s):** Before entering a loop, the variable that controls the execution of it is called initialization.
- **Test Expression.** The test-expression is a condition, which decides whether the loop body will be executed or not.
- **Update Expression(s).** The update expression(s) change the value(s) of the control variable(s). The update expression(s) is usually made to be executed; before the next iteration.
- **The Body of the Loop.** The statements that are executed repeatedly (as long as the, test-expression is true) form the body of the loop.

**6. Give an example to show how multiple initialization and updation are performed in a for loop.**

**Ans.** For multiple initialization and multiple updation in a **for** loop they are separated by comma (.). The following example shows how multiple initialization and updation is performed in a **for** loop.

```
for(i=1, j=10 ; i<=10 ; j-- , i++)
{
    System.out.println("i=" +i+ " "+"j="+j);
}
```

**7. What is an empty for loop. Give an example.**

**Ans.** If the loop doesn't contain a body, it is called an empty loop. For example,

```
for(i=1;i<=100;i++);
```

Notice the semicolon at the end of the parenthesis, indicating that the **for** loop does not have body and therefore no statement is repeatedly executed again and again.

**8. State one similarity and one difference between while and do-while loops.**

[ICSE 2005]

**Ans. Similarity:** The similarity between **while** and **do-while** loop is that, both the loops executes as long as the given *test-expression* is satisfied.

**Difference:** The difference between **while** and **do-while** loop is that, **while** is an *entry-controlled* loop and **do-while** loop is an *exit-controlled* loop.

**9. State one similarity and one difference between while and for loops. [ICSE 2009]**

**Ans. Similarity:** The similarity between **while** and **for** loop is that, both the loops are entry-controlled loops..

**Difference:** The difference between **while** and **for** loop is that, *initialization* in **while** loop is outside the loop structure, whereas *initialization* of **for** loop is within the loop structure.

**10. What is meant by an infinite loop? Give one example. [ICSE 2008]**

**Ans. Infinite** loop is a loop which executes indefinitely *i.e.* the given *test-expression* is forever evaluated to **true**. For example,

```
for(i=1;i>0;i++)
{
    System.out.println(i);
}
```

The loop executes infinitely because the value of 'i' is always greater than 0.

**11. Give the general syntax of a while loop. How do you create infinite loops using a while loop structure?**

**Ans.** The general syntax of **while** loop is:

```
while (condition or test-expression)
{
    //body of the loop;
}
```

The easiest way of creating an infinite loop using **while** loop is:

```
while(true)
{
    //body of the loop
}
```

**12. Give the general syntax of a do-while loop. How do you create infinite loops using do-while loop structure?**

**Ans.** The general syntax of **do-while** loop is:

```
do
{
    //body of the loop;
} while (condition or test-expression)
```

The easiest way of creating an infinite loop using **while** loop is:

```
do
{
    //body of the loop
} while(true);
```

**13. Compare loops according to its usage.**

**Ans.** The **for** loop is appropriate when you know in advance how many times the loop will be executed i.e. the first and the last limit is already known. The other two loops **while** and **do-while** loops are more suitable in situations when it is not known in advance when the loop will terminate. The **while** should be preferred when you may not want to execute the *body-of-the-loop* even once (in case the test-expression is **false**), and the **do-while** loop should be preferred when you're sure you want to execute the *body-of-the-loop* at least once.

**14. What are nested loops? Give an example of a nested for loop to show its functionality.**

**Ans.** A loop within a loop is called a nested loop. The following example shows how nested **for** loop works:

```
for(i=1;i<=5;i++)
{
    for(j=1;j<=i;j++)
    {
        System.out.print(j);
    }
    System.out.println( );
}
```

The output received for the above program fragment is:

```
1
12
123
1234
12345
```



**15. Name two jump statements and their use.**

[ICSE 2010]

**Ans.** The two jump statements in Java are **break** and **continue**. Whereas **break** statement is used to exit from the loop or **switch** block, the **continue** statement is used to skip the remaining statements in a loop and force the next iteration.

**16. Explain the meaning of break and continue statements.** [ICSE 2005, 2008]

**Ans.** The **break** statement is used to exit from the loop or **switch** block, the **continue** statement is used to skip the remaining statements in a loop and force the next iteration.

**17. Explain how break statement works in a nested loop. Also give an example to show how continue statement works in a nested loop.**

**Ans.** The **break** statement is used to exit from the nested loop in a nested loop of which it is a content.

Example of **continue** statement:

```
for(x=1;x<=2;x++)
{
    for(y=1; y<=4 ; y++)
    {
        if(y==3)
            continue;
        System.out.println(x+" \t "+y);
    }
    System.out.println("x="+x);
}
```

The output for the above code fragment would be,

```
1    1
1    2
1    4
x=1
2    1
2    2
2    4
```

x=2

As you can see whenever the value of **y** reaches **3** it skips the statements only within the inner loop but the execution of the outer loop continues remaining same thus **System.out.println("x="+x);** do get executed as it is actually present only within the outer loop.

**18. With the help of an example, determine the working of**

**i) labelled break and**

**ii) labelled continue.**

**Ans. i)** The following example shows the working of labeled **break** statement:

```
int i,j;
outer: //label for the outer block
for(i=1;i<=3;i++)
{
    inner: //label for the inner block
    for(j=1;j<=5;j++)
    {
        if (j>3)
            break outer;
        System.out.println(i+"\t"+j);
    } //End of the inner block
    System.out.println(j);
}
```

The output would be

```
1    1
1    2
1    3
```

Here you can see that there are two blocks labeled (*outer* and *inner*). Notice that when the value of  $j > 3$  it exits from the block labeled *outer*, which makes it ultimately come out of both the loops.

ii)The following example shows the working of labeled **break** statement:

```
int i,j;
outer:
for (i=0;i<10;i++)
{
    for(j=0;j<10;j++)
    {
        if (j>i)
        {
            System.out.println( );
            continue outer;
        }
        System.out.print(" " + (i*j));
    }
}
```

The output will be

```
0
0 1
0 2 4
0 3 6 9
0 4 8 12 16
0 5 10 15 20 25
0 6 12 18 24 30 36
```

0 7 14 21 28 35 42 49

0 8 16 24 32 40 48 56 64

0 9 18 27 36 45 54 63 72 81

The **continue** statement in this example terminates the loop counting *j* and continues with the next iteration of the loop counting *i*.

### 19. How does the return statement works as a jump statement?

**Ans.** The **return** statement is used to force an exit from the method or function. The following program shows the working of **return** statement:

```
class ReturnDemo
{
    static void exit() //called method
    {
        System.out.println("Before the return");
        if (true)
            return;

        System.out.println("This won't execute");
    }
    static void test( ) //calling method
    {
        exit();
    }
}
```

If the **test( )** method is executed it gives the following output.

Before the return

As you can see that, the final **System.out.println( )** statement is not executed. As soon as return is executed, control passes back to the caller function i.e. **test( )**.

### B. Do as directed in the following questions:

1. Convert the following segment into an equivalent do loop.

[ICSE 2009]

**int x,c;**

**for(x=10,c=20;c>10;c=c-2)**

**x++;**

**Ans.** int x, c;

```

x=10;
c=20;
while(c>10)
{
    x++;
    c=c-2;
}

```

**2. Analyze the following program segment and determine how many times the body of loop will be executed (show the working). [ICSE 2009]**

```

x=5; y=50;
while(x<=y)
{
    y=y/x;
    System.out.println(y);
}

```

**Ans.** Working:

x	y	x<=y	y=y/x	System.out.println(y)
5	50	true	10	10
5	10	true	2	2
5	2	false	---	---

The loop will iterate for 2 times.

**3. Convert the following segment into equivalence for loop. [ICSE 2007]**

```

int i=0;
while(i<=20)
{
    System.out.print(i+" ");
    i++;
}

```

**Ans.** int i;

```

for(i=0;i<=20;i++)
{
    System.out.print(i+" ");
}

```

```
}
```

4. What will be the output of the following code?

[ICSE 2010]

```
int m=2;
int n=15;
for(int i = 1; i<5; i++);
m++; --n;
System.out.println("m=" +m);
System.out.println("n="+n);
```

**Ans.** m=3  
n=14

5. Give the output of the following program:

class Output1

```
{
    static void find( )
    {
        int n=1234, d ,s=0;
        while(n!=0)
        {
            d=n%10;
            s=s*10+d;
            n=n/10;
        }
        System.out.println(s);
    }
}
```

**Ans.** 4321

6. a) Give the output of the following program:

class Output2

```
{
    static void find( )
    {
        int n=1234, d ,s=0,c=0;
        while(n!=0)
        {
            d=n%10;
            s=s+d*(int)Math.pow(10,c++);
            n=n/10;
        }
        System.out.println(s);
    }
}
```

```
}
```

b) Write programs to display the following series:

i) 1, 3, 5, 7, 9, ... 29

ii) 2, 4, 6, 8, 10, ... 30

iii) 100, 90, 80, 70, 60, ... 0

iv) 1, 2, 4, 7, 11, 16, 22, 29, ... nth term. (Take n as input).

v) 2, 4, 8, 14, 22, 32, ... nth terms. (Taken n as input).

**Ans.** a) 1234

b)

```
import java.util.*;

public class Question6b
{
    static void i()
    {
        int i;
        for(i=1;i<=29;i+=2)
        {
            System.out.print(i+" ");
        }
    }

    static void ii()
    {
        int i;
        for(i=2;i<=30;i+=2)
        {
            System.out.print(i+" ");
        }
    }

    static void iii()
    {
```

```
int i;
for(i=100;i>=0;i-=10)
{
    System.out.print(i+" ");
}
}
static void iv()
{
    Scanner sc=new Scanner(System.in);
    int i,n,p=1;
    System.out.println("Enter the value of n:");
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        System.out.print(p+" ");
        p=p+i;
    }
}
static void v()
{
    Scanner sc=new Scanner(System.in);
    int i,n,p=2,c=2;
    System.out.println("Enter the value of n:");
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        System.out.print(p+" ");
```



```
        p=p+c;
        c=c+2;
    }
}
}
```

**7. Write a program to find the sum of first 10 odd natural numbers.**

**Ans.** public class Question7

```
{
    static void main()
    {
        int i,s=0;
        for(i=1;i<=19;i+=2)
        {
            s=s+i;
        }
        System.out.println("Sum of first 10 odd numbers="+s);
    }
}
```

**8. Write a program to find the sum of first 10 even natural numbers.**

**Ans.** public class Question8

```
{
    static void main()
    {
        int i,s=0;
        for(i=2;i<=20;i+=2)
        {
```

```
        s=s+i;
    }
    System.out.println("Sum of first 10 even numbers="+s);
}
}
```

**9. Write a program to find the sum of all 3-digit even natural numbers.**

**Ans.** public class Question9

```
{
    static void main()
    {
        int i,s=0;
        for(i=100;i<=998;i+=2)
        {
            s=s+i;
        }
        System.out.println("Sum of all 3-digit even natural numbers="+s);
    }
}
```

**10. Write a program to find the sum of all 3 digit odd natural numbers, which are multiples of 5.**

**Ans.** public class Question10

```
{
    static void main()
    {
        int i,s=0;
        for(i=101;i<=999;i+=2)
```

```

    {
        if(i%5==0)
            s=s+i;
    }
    System.out.println("Sum="+s);
}
}

```

**11. Write a program to input an integer and find its factorial. Factorial of a number is the product of b) Write programs to display the following series:**

**Ans.** import java.util.\*;

```

public class Question11
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n;
        long f=1;
        System.out.print("Enter an integer:");
        n=sc.nextInt();
        for(i=1;i<=n;i++)
        {
            f=f*i;
        }
        System.out.println("Factorial =" +f);
    }
}

```

**12. Write a program to input an integer and print its factors.**

**For Example,**

**INPUT:**

**Enter an integer:**

**OUTPUT:**

**Factors: 1 2 3 4 6 12**

**INPUT:**

**12 Enter an integer:10**

**OUTPUT:**

**Factors: 1 2 5 10**

**Ans.** import java.util.\*;

```
public class Question12
```

```
{
```

```
    static void main()
```

```
    {
```

```
        Scanner sc=new Scanner(System.in);
```

```
        int i,n;
```

```
        System.out.print("Enter an integer:");
```

```
        n=sc.nextInt();
```

```
        for(i=1;i<=n;i++)
```

```
        {
```

```
            if(n%i==0)
```

```
                System.out.print(i+" ");
```

```
        }
```

```
    }
```

```
}
```

**13. Write a program to input an integer and count the number of factors.**

**Ans.** import java.util.\*;

```
public class Question13
```

```
{
```

```
    static void main()
```

```
    {
```

```
        Scanner sc=new Scanner(System.in);
```

```

int i,n,c=0;
System.out.print("Enter an integer:");
n=sc.nextInt();
for(i=1;i<=n;i++)
{
    if(n%i==0)
        c++;
}
System.out.print("Number of Factors="+c);
}
}

```

**14. Write a program to input an integer and check whether it is a prime number or not. (Remember, prime number has 2 factors).**

**Ans.** import java.util.\*;

```

public class Question14
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,c=0;
        System.out.print("Enter an integer:");
        n=sc.nextInt();
        for(i=1;i<=n;i++)
        {
            if(n%i==0)
                c++;
        }
    }
}

```

```
        if(c==2)
            System.out.print("Prime Number");
        else
            System.out.print("Not a Prime Number");
    }
}
```

**15. Write a program to input 10 integers and find their sum.**

**Ans.** import java.util.\*;

```
public class Question15
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,s=0;
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {
            n=sc.nextInt();
            s=s+n;
        }
        System.out.print("Sum="+s);
    }
}
```

**16. Write a program to input 10 integers and find the sum of even numbers only.**

**Ans.** import java.util.\*;

```
public class Question16
```

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,s=0;
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {
            n=sc.nextInt();
            if(n%2==0)
                s=s+n;
        }
        System.out.print("Sum of even numbers="+s);
    }
}

```

**17. Write a program to input 10 integers and find the sum of two-digit as well as three-digit numbers separately.**

**Ans.** import java.util.\*;

```

public class Question17
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,s1=0,s2=0;
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {

```

```

        n=sc.nextInt();
        if(n>=10 && n<=99)
            s1=s1+n;
        else if(n>=100 && n<=999)
            s2=s2+n;
    }
    System.out.println("Sum of 2-digit numbers="+s1);
    System.out.println("Sum of 3-digit numbers="+s2);
}
}

```

**18. Write a program to input 10 integers and display the largest integer.**

**Ans.** import java.util.\*;

```

public class Question18
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,l=0;
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {
            n=sc.nextInt();
            if(i==1)
                l=n;
            else if(n>l)
                l=n;
        }
    }
}

```



```
    }  
    System.out.println("Largest Number="+l);  
  }  
}
```

**19. Write a program to input 10 integers and display the largest as well as the smallest integer.**

**Ans.** import java.util.\*;

```
public class Question19  
{  
    static void main()  
    {  
        Scanner sc=new Scanner(System.in);  
        int i,n,l=0,s=0;  
        System.out.println("Enter 10 integers:");  
        for(i=1;i<=10;i++)  
        {  
            n=sc.nextInt();  
            if(i==1)  
                s=l=n;  
            if(n>l)  
                l=n;  
            if(n<s)  
                s=n;  
        }  
        System.out.println("Largest Number="+l);  
        System.out.println("Smallest Number="+s);  
    }  
}
```

```
}
```

**20. Write a program to input 10 integers and display the largest even integer. In case there is no even integer, it should print "No even integer found".**

**Ans.** import java.util.\*;

```
public class Question20
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,l=0,f=0;
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {
            n=sc.nextInt();
            if(n%2==0)
            {
                if(f==0)
                    l=n;
                f=1;
                if(n>l)
                    l=n;
            }
        }
        if(f==1)
            System.out.println("Largest Even Number="+l);
        else
            System.out.println("No even integer found");
    }
}
```

```
}  
}
```

**21. Write a program to input 10 integers and display the largest even and smallest odd integer.**

**Ans.** import java.util.\*;

```
public class Question21  
{  
    static void main()  
    {  
        Scanner sc=new Scanner(System.in);  
        int i,n,l=0,f1=0,f2=0,s=0;  
        System.out.println("Enter 10 integers:");  
        for(i=1;i<=10;i++)  
        {  
            n=sc.nextInt();  
            if(n%2==0)  
            {  
                if(f1==0)  
                {  
                    l=n;  
                    f1=1;  
                }  
                if(n>l)  
                    l=n;  
            }  
            else  
            {
```

```

        if(f2==0)
        {
            s=n;
            f2=1;
        }
        if(n<s)
            s=n;
    }
}
if(f1==1)
    System.out.println("Largest Even Number="+1);
else if(f1==0)
    System.out.println("No even integer found");
if(f2==1)
    System.out.println("Smallest Odd Number="+s);
else if(f2==0)
    System.out.println("No odd integer found");

}
}

```

**22. Write a program to input 10 integers and check whether all the entered numbers are even numbers or not.**

**Ans.** import java.util.\*;

```

public class Question22
{
    static void main()
    {

```

```

Scanner sc=new Scanner(System.in);
int i,n,f=0;
System.out.println("Enter 10 integers:");
for(i=1;i<=10;i++)
{
    n=sc.nextInt();
    if(n%2!=0)
        f=1;
}
if(f==0)
    System.out.println("All are Even Numbers");
else
    System.out.println("All are not Even Numbers");
}
}

```

**23. Write a program to input 10 integers and check whether all the entered numbers are odd numbers or not.**

**Ans.** import java.util.\*;

```

public class Question23
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,f=0;
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {

```

```

        n=sc.nextInt();
        if(n%2==0)
            f=1;
    }
    if(f==0)
        System.out.println("All are Odd Numbers");
    else
        System.out.println("All are not Odd Numbers");
    }
}

```

**24. Write a program to input 10 integers and check whether all the entered numbers are same or not.**

**For Example,**

**INPUT:**

**Enter 10 numbers: 10 12 13 234 45 34 67 78 76 12**

**OUTPUT:**

**All numbers are not same.**

**INPUT:**

**Enter 10 numbers: 12 12 12 12 12 12 12 12 12 12**

**OUTPUT:**

**All numbers are same.**

**Ans.** import java.util.\*;

```
public class Question24
```

```
{
```

```
    static void main()
```

```
    {
```

```
        Scanner sc=new Scanner(System.in);
```

```
        int i,n,f=0,l=0;
```

```
        System.out.println("Enter 10 numbers:");
```

```
        for(i=1;i<=10;i++)
```

```

    {
        n=sc.nextInt();
        if(i==1)
            l=n;
        if(n!=l)
            f=1;
    }
    if(f==0)
        System.out.println("All numbers are same");
    else
        System.out.println("All numbers are not same");
    }
}

```

**25. Write a program to input 10 integers and check whether the entered numbers are in ascending order or not.**

**For Example,**

**INPUT:**

**Enter 10 numbers: 10 12 13 25 45 55 67 78 106 122**

**OUTPUT:**

**The numbers are in ascending order.**

**INPUT:**

**Enter 10 numbers: 25 34 56 67 12 32 43 21 23 111**

**OUTPUT:**

**The numbers are not in ascending order.**

**Ans.** import java.util.\*;

public class Question25

{

static void main()

{

Scanner sc=new Scanner(System.in);

```

int i,n,f=0,l=0;
System.out.println("Enter 10 numbers:");
for(i=1;i<=10;i++)
{
    n=sc.nextInt();
    if(i==1)
        l=n;
    else
    {
        if(n<l)
            f=1;
        l=n;
    }
}
if(f==0)
    System.out.println("All numbers are in ascending order");
else
    System.out.println("All numbers are not in ascending order");
}
}

```

**26. Write a program to calculate and print the sum of odd numbers and the sum of even numbers for the first n natural numbers. The integer n is to be entered by the user.**

**[ICSE-2006]**

**Ans.** import java.util.\*;

```

public class Question26
{

```



```

static void main()
{
    Scanner sc=new Scanner(System.in);
    int i,n,s1=0,s2=0;
    System.out.println("Enter a number:");
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        if(i%2==0)
            s1=s1+i;
        else
            s2=s2+i;
    }
    System.out.println("Sum of even numbers="+s1);
    System.out.println("Sum of odd numbers="+s2);
}
}

```

**27. Write programs to find the sum of each of the following series:**

- i)  $s=1*2+2*3+3*4+\dots+9*10$
- ii)  $s=1*2+3*4+5*6+\dots+9*10$
- iii)  $s=1*3+2*4+3*5+\dots+9*11$
- iv)  $s=1-2+3-4+5-6+\dots+19-20$
- v)  $s=1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\frac{1}{5}+\dots+\frac{1}{10}$
- vi)  $s=\frac{1}{2}+\frac{2}{3}+\frac{3}{4}+\frac{4}{5}+\dots+\frac{9}{10}$
- vii)  $s=1-\frac{1}{2}+\frac{1}{3}-\frac{1}{4}+\frac{1}{5}-\dots-\frac{1}{10}$

**Ans.** public class Question27

```

{
    static void i()
    {

```

```
int i,s=0;
for(i=1;i<=9;i++)
{
    s=s+i*(i+1);
}
System.out.println("Sum =" +s);
}
static void ii()
{
    int i,s=0;
    for(i=1;i<=9;i+=2)
    {
        s=s+i*(i+1);
    }
    System.out.println("Sum =" +s);
}
static void iii()
{
    int i,s=0;
    for(i=1;i<=9;i++)
    {
        s=s+i*(i+2);
    }
    System.out.println("Sum =" +s);
}
static void iv()
{
```

```
int i,s=0;
for(i=1;i<=19;i+=2)
{
    s=s+i-(i+1);
}
System.out.println("Sum =" +s);
}

static void v()
{
    int i;
    float s=0;
    for(i=1;i<=10;i++)
    {
        s=s+(float)1/i;
    }
    System.out.println("Sum =" +s);
}

static void vi()
{
    int i;
    float s=0;
    for(i=1;i<=9;i++)
    {
        s=s+(float)i/(i+1);
    }
    System.out.println("Sum =" +s);
}
```

```

static void vii()
{
    int i;
    float s=0;
    for(i=1;i<=9;i++)
    {
        if(i%2==0)
            s=s-(float)i/(i+1);
        else
            s=s+(float)i/(i+1);
    }
    System.out.println("Sum =" +s);
}
}

```

**28. Write a program to find the sum of 1st 10 numbers of Fibonacci series, i.e. 1,1,2,3,5, 8,13.... Lucas series is such a series which starting from 1 and 1, and subsequent numbers are the sum of the previous two numbers.**

**Ans.** public class Question28

```

{
    static void main()
    {
        int i,a=1,b=0,c;
        System.out.println("Fibonacci series =");
        for(i=1;i<=10;i++)
        {
            c=a+b;
            System.out.print(c+" ");

```

```

        a=b;
        b=c;
    }
}
}

```

**29. Write a program to print the first 15 numbers of the Pell series. Pell series is such a series which starts from 1 and 2, and subsequent numbers are the sum of twice the previous number and the number previous to the previous number. Pell series: 1, 2, 5, 12, 29, 70, 169, 408, 985, 2378, 5741, 13860, ...**

**Ans.** public class Question29

```

{
    static void main()
    {
        int i,a=1,b=0,c;
        System.out.println("Pell series =");
        for(i=1;i<=15;i++)
        {
            c=a+2*b;
            System.out.print(c+" ");
            a=b;
            b=c;
        }
    }
}
}

```

**30. Write a program to find the sum of 1st 10 numbers of Lucas series, i.e. 2,1,3,4,7,11, 18,...Lucas series is such a series which starting from 2 and 1, and subsequent numbers are the sum of the previous two numbers.**

**Ans.** public class Question30

```

{
    static void main()
    {
        int i,a=2,b=1,c;
        System.out.println("Pell series =");
        System.out.print(a+" "+b+" ");
        for(i=1;i<=8;i++)
        {
            c=a+b;
            System.out.print(c+" ");
            a=b;
            b=c;
        }
    }
}

```

**31. Write a program to input an integer and check whether it is perfect, abundant or deficient number. If the sum of the factors excluding itself is equal to that number it is perfect, if greater than that number it is abundant and if less than that number it is deficient number.**

**Ans.** import java.util.\*;

```

public class Question31
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,s=0;
        System.out.println("Enter an integer:");
    }
}

```

```

n=sc.nextInt();
for(i=1;i<n;i++)
{
    if(n%i==0)
        s+=i;
}
if(s==n)
    System.out.println("Perfect Number");
else if(s<n)
    System.out.println("Deficient Number");
else
    System.out.println("Abundant Number");
}
}

```

**32. Write a program to input two integers and check whether it forms an amicable pair or not. An amicable pair is such that the sum of the factors excluding itself of one number is the other number and sum of the factors excluding itself of the other number is this number.**

**Example, (220, 284). Since sum of factors excluding itself of:**

$$220 = 1+2+4+5+10+11+20+22+ 44+55+110=284$$

$$284 = 1+ 2+4+71+142=220.$$

**Ans.** import java.util.\*;

```

public class Question32
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,a,b,s1=0,s2=0;

```

```

System.out.println("Enter 2 integers:");
a=sc.nextInt();
b=sc.nextInt();
for(i=1;i<a;i++)
{
    if(a%i==0)
        s1+=i;
}
for(i=1;i<b;i++)
{
    if(b%i==0)
        s2+=i;
}
if(s1==b && s2==a)
    System.out.println("Amicable Pair");
else
    System.out.println("Not an amicable pair");
}
}

```

**33. Write a program to pass an integer as argument and find the sum of its digits.**

**Ans.** public class Question33

```

{
    static void main(int n)
    {
        int d,s=0;
        while(n!=0)

```



```

    {
        d=n%10;
        s=s+d;
        n=n/10;
    }
    System.out.println("Sum of the digits="+s);
}
}

```

**34. Write a program to pass an integer as argument and find the sum of odd digits and even digits separately.**

**Ans.** public class Question34

```

{
    static void main(int n)
    {
        int d,se=0,so=0;
        while(n!=0)
        {
            d=n%10;
            if(d%2==0)
                se=se+d;
            else
                so=so+d;
            n=n/10;
        }
        System.out.println("Sum of the even digits="+se);
        System.out.println("Sum of the odd digits="+so);
    }
}
}

```

**35. Write a program to pass an integer as argument and find the average of its digits.**

**Ans.** public class Question35

```
{
    static void main(int n)
    {
        int d,s=0,c=0;
        float av;
        while(n!=0)
        {
            d=n%10;
            s=s+d;
            c++;
            n=n/10;
        }
        av=(float)s/c;
        System.out.println("Average of its digits="+av);
    }
}
```

**36. Write a program to pass an integer as argument and print the largest as well as smallest digit.**

**Ans.** public class Question36

```
{
    static void main(int n)
    {
        int d,s=0,l=0,f=0;
        while(n!=0)
```

```

    {
        d=n%10;
        if(f==0)
        {
            l=s=d;
            f=1;
        }
        if (d>l)
            l=d;
        if(d<s)
            s=d;
        n=n/10;
    }
    System.out.println("Largest digit="+l);
    System.out.println("Smallest digit="+s);
}
}

```

**37. Write a program to pass an integer as argument and print the second largest digit.**

**Ans.** public class Question37

```

{
    static void main(int n)
    {
        int d,s=0,l=0,f=0,sl=0,t=n;
        while(n!=0)
        {

```

```
d=n%10;
if(f==0)
{
    l=d;
    f=1;
}
if (d>l)
    l=d;
n=n/10;
}
f=0;
while(t!=0)
{
    d=t%10;
    if(d!=1)
    {
        if(f==0)
        {
            sl=d;
            f=1;
        }
        if(d>sl)
            sl=d;
    }
    t=t/10;
}
System.out.println("Second Largest digit="+sl);
```

```
}  
}
```

**38. Write a program to input an integer and remove all the even digits from it.**

**Ans.** import java.util.\*;

```
public class Question38  
{  
    static void main()  
    {  
        Scanner sc=new Scanner(System.in);  
        int i,n,d,c=0,s=0;  
        System.out.println("Enter an integer:");  
        n=sc.nextInt();  
        while(n!=0)  
        {  
            d=n%10;  
            if(d%2!=0)  
                s=s+d*(int)Math.pow(10,c++);  
            n=n/10;  
        }  
        System.out.println("New Number="+s);  
    }  
}
```

**39. Write a program to input an integer and remove all the even digits from it.**

**For Example,**

**INPUT: Enter an integer: 1234**

**OUPUT:**

**13**

**Ans.** Same as question38.

**40. Write a program to pass an integer as argument and form a new number by having the digits arranged in ascending order.**

**Ans.** public class Question40

```
{
    static void main(int n)
    {
        int d,s=0,c=0,i,t;
        for(i=9;i>=0;i--)
        {
            t=n;
            while(t!=0)
            {
                d=t%10;
                if(d==i)
                    s=s+d*(int)Math.pow(10,c++);
                t=t/10;
            }
        }
        System.out.println("Digits in ascending order="+s);
    }
}
```

**41. Write a program to input an integer and check whether all digits in it are same or not.**

**Ans.** import java.util.\*;

public class Question41

```
{
```

```

static void main()
{
    Scanner sc=new Scanner(System.in);
    int d,n,last,f=0;
    System.out.println("Enter an integer:");
    n=sc.nextInt();
    last=n%10;
    while(n!=0)
    {
        d=n%10;
        if(d!=last)
            f=1;
        n=n/10;
    }
    if(f==0)
        System.out.println("All digits are same");
    else
        System.out.println("All digits are not same");
    }
}

```

**42. Write a program to input an integer and check whether all digits in it are even digits or not.**

**Ans.** import java.util.\*;

```

public class Question42
{
    static void main()
    {

```

```

Scanner sc=new Scanner(System.in);
int d,n,f=0;
System.out.println("Enter an integer:");
n=sc.nextInt();
while(n!=0)
{
    d=n%10;
    if(d%2!=0)
        f=1;
    n=n/10;
}
if(f==0)
    System.out.println("All digits are even");
else
    System.out.println("All digits are not even");
}
}

```

**43. Write a program to input an integer and check whether all digits are unique or not.**

**Ans.** import java.util.\*;

```

public class Question43
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int d1,d2,n,t,f=0;
        System.out.println("Enter an integer:");

```



```

n=sc.nextInt();
while(n!=0)
{
    d1=n%10;
    t=n/10;
    while(t!=0)
    {
        d2=t%10;
        if(d1==d2)
            f=1;
        t=t/10;
    }
    n=n/10;
}
if(f==0)
    System.out.println("All digits are unique");
else
    System.out.println("All digits are not unique");
}
}

```

**44. Write a program to input an integer and check whether all are prime digits or not.**

**Ans.** import java.util.\*;

```

public class Question44
{
    static void main()
    {

```

```

Scanner sc=new Scanner(System.in);
int d,n,f=0;
System.out.println("Enter an integer:");
n=sc.nextInt();
while(n!=0)
{
    d=n%10;
    if(d!=2 && d!=3 && d!=5 && d!=7)
        f=1;
    n=n/10;
}
if(f==0)
    System.out.println("All digits are prime");
else
    System.out.println("All digits are not prime");
}
}

```

**45. Write a program to input an integer and check whether all digits are in ascending order or not.**

**Ans.** import java.util.\*;

```

public class Question45
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int d1,d2,n,f=0;
        System.out.println("Enter an integer:");

```

```

n=sc.nextInt();
while(n>9)
{
    d2=n%10;
    d1=(n/10)%10;
    if(d1>d2)
        f=1;
    n=n/10;
}
if(f==0)
    System.out.println("All digits are in ascending order");
else
    System.out.println("All digits are not in ascending order");
}
}

```

**46. Write a program to input an integer and check whether all digits are in descending order or not.**

**Ans.** import java.util.\*;

```

public class Question46
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int d1,d2,n,f=0;
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        while(n>9)

```

```

    {
        d2=n%10;
        d1=(n/10)%10;
        if(d1<d2)
            f=1;
        n=n/10;
    }
    if(f==0)
        System.out.println("All digits are in descending order");
    else
        System.out.println("All digits are not in descending order");
    }
}

```

**47. Write a program to pass an integer as argument and check whether it is armstrong number or not. Numbers whose sum of the cube of its digit is equal to the number itself is called Armstrong numbers. Example  $153=1^3+5^3+3^2$ .**

**Ans.** import java.util.\*;  
public class Question47

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int t,d,s=0,n,f=0;
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        t=n;
        while(t!=0)
        {

```

```

        d=t%10;
        s=s+d*d*d;
        t=t/10;
    }
    if(s==n)
        System.out.println("Armstrong Number");
    else
        System.out.println("Not an Armstrong Number");
    }
}

```

**48.** Write a program to input an integer and check whether it is an automorphic, trimorphic or tri-automorphic number or not. A number  $n$  is said to be automorphic if its square ends in  $n$ . For instance 5 is automorphic, because  $5^2 = 25$ , which ends in 5, 25 is automorphic, because  $25^2 = 625$ , which ends in 25. A number  $n$  is called trimorphic if  $n^3$  ends in  $n$ . For example,  $49^3 = 117649$ , is trimorphic. A number  $n$  is called tri-automorphic if  $3n^2$  ends in  $n$ ; for example, 667 is tri-automorphic because  $3 \times 667^2 = 1334667$ , ends in 667.

**Ans.** import java.util.\*;

```

public class Question48
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int c=0,s,t,n,f=0;
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        t=n;
        while(t!=0)

```

```

    {
        c++;
        t=t/10;
    }

    if((n*n)%(int)Math.pow(10,c)==n)
        System.out.println("Automorphic Number");
    else if((n*n*n)%(int)Math.pow(10,c)==n)
        System.out.println("Trimorphic Number");
    else if((3*n*n)%(int)Math.pow(10,c)==n)
        System.out.println("Tri-automorphic Number");
    }
}

```

**49. Write a program to input an integer and check whether it is a happy number or not. If you iterate the process of summing the squares of the digits of a number and if this process terminates in 1, then the original number is called a happy number. For example  $7 \Rightarrow (7^2) = 49 \Rightarrow (4^2 + 9^2) = 97 \Rightarrow (9^2 + 7^2) = 130 \Rightarrow (1^2 + 3^2 + 0^2) = 10 \Rightarrow (1^2 + 0^2) = 1$ .**

**Ans.** import java.util.\*;

```

public class Question49
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int d,s,n;
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        do

```

```

    {
        s=0;
        while(n!=0)
        {
            d=n%10;
            s=s+d*d;
            n=n/10;
        }
        n=s;
    }while(n>9);

    if(n==1)
        System.out.println("Happy Number");
    else
        System.out.println("Not a Happy Number");
}
}

```

**50. Write a program to input an integer and check whether it is a magic number or not. If you iterate the process of summing the squares of the digits of a number and if this process terminates in 1, then the original number is called a magic number.**

**For example 55=> (5+5)=10=>(1+0)=1.**

**Ans.** import java.util.\*;

```

public class Question50
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int d,s,n;
    }
}

```

```

System.out.println("Enter an integer:");
n=sc.nextInt();
do
{
    s=0;
    while(n!=0)
    {
        d=n%10;
        s=s+d;
        n=n/10;
    }
    n=s;
}while(n>9);

if(n==1)
    System.out.println("Magic Number");
else
    System.out.println("Not a Magic Number");
}
}

```

**51. Write a program to input an integer and check whether it is Harshad or Niven number or not. A number is said to be Harshad if it is divisible by the sum of the digits of that number, example 126 and 1729.**

**Ans.** import java.util.\*;

```

public class Question51
{
    static void main()

```



```

    {
        Scanner sc=new Scanner(System.in);
        int d,s=0,n,t;
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        t=n;
        while(t!=0)
        {
            d=t%10;
            s=s+d;
            t=t/10;
        }
        if(n%s==0)
            System.out.println("Harshad Number");
        else
            System.out.println("Not a Harshad Number");
        }
    }

```

**52.** Write a program to input a number and check whether it is a Kaprekar number or not. Take a positive whole number  $n$  that has  $d$  number of digits. Take the square  $n$  and separate the result into two pieces: a right-hand piece that has  $d$  digits and a left-hand piece that has either  $d$  or  $d-1$  digits. Add these two pieces together. If the result is  $n$ , then  $n$  is a Kaprekar number. Examples are: 9 ( $9^2 = 81$ ,  $8 + 1 = 9$ ), 45 ( $45^2 = 2025$ ,  $20 + 25 = 45$ ) and 297 ( $297^2 = 88209$ ,  $88 + 209 = 297$ ).

**Ans.** import java.util.\*;

```

public class Question52
{

```

```

static void main()
{
    Scanner sc=new Scanner(System.in);
    int c=0,n,t,first,last;
    System.out.println("Enter an integer:");
    n=sc.nextInt();
    t=n;
    while(t!=0)
    {
        c++;
        t=t/10;
    }
    first=(n*n)/(int)Math.pow(10,c);
    last=(n*n)%(int)Math.pow(10,c);
    if(first+last==n)
        System.out.println("Kaprekar Number");
    else
        System.out.println("Not a Kaprekar Number");
    }
}

```

**53. Write a program to input two integers and find their Least Common Multiple (LCM).**

**For Example,**

**INPUT:**

**Enter 2 integers:**

**12**

**8**

**OUTPUT:**

**LCM = 24**

```
Ans. import java.util.*;

public class Question53
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);

        int i,a,b;

        System.out.println("Enter 2 integers:");

        a=sc.nextInt();
        b=sc.nextInt();

        for(i=a;i<=a*b;i++)
        {
            if(i%a==0 && i%b==0)
                break;
        }

        System.out.println("L.C.M.=" +i);
    }
}
```

**54. Write a program to input two integers and find their Highest Common Factor(HCF).**

**For Example,**

**INPUT:**

**Enter 2 integers:**

**12**

**8**

**OUTPUT:**

**HCF = 4**

```
Ans. import java.util.*;

public class Question54
```

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,a,b;
        System.out.println("Enter 2 integers:");
        a=sc.nextInt();
        b=sc.nextInt();
        for(i=a;i>=1;i--)
        {
            if(a%i==0 && b%i==0)
                break;
        }
        System.out.println("H.C.F.="+i);
    }
}

```

**55.** An interesting method of multiplication of integers (not very large numbers) is illustrated by the following example. If the numbers 17 and 19 are to be multiplied, they are put at the top of two columns as shown below:

<b>17</b>	<b>19</b>
-----------	-----------

The numbers at the left-hand side is successively divided by 2 (integer division) while the other is successively multiplied by 2. The results are written one below the other in their respective columns. The process is repeated till the column containing the division results reached 1. At this stage all the numbers in the right-hand column are struck off where numbers corresponding to them on the left-hand column are even.

<b>17</b>	<b>19</b>
<b>8 (even)</b>	<b>38</b>
<b>4 (even)</b>	<b>76</b>
<b>2 (even)</b>	<b>152</b>

1

304

Now the remaining numbers on the right-hand side are added:  $19+304=323$  which is equal to the product of 17 and 19.

Write a program to input 2 integers and using the above method, find their product.

**Ans.** import java.util.\*;

```
public class Question55
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,s=0;
        System.out.println("Enter 2 integers:");
        a=sc.nextInt();
        b=sc.nextInt();
        while(a>=1)
        {
            if(a%2!=0)
                s=s+b;
            a=a/2;
            b=b*2;
        }
        System.out.println("Product="+s);
    }
}
```

**56.** Write a program to calculate and print the sum of odd numbers and the sum of even numbers for the first n natural numbers. The integer n is to be entered by the user.

[ICSE 2006]

**Ans.** import java.util.\*;

```

public class Question56
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,s1=0,s2=0;
        System.out.println("Enter a number:");
        n=sc.nextInt();
        for(i=1;i<=n;i++)
        {
            if(i%2==0)
                s1=s1+i;
            else
                s2=s2+i;
        }
        System.out.println("Sum of even numbers="+s1);
        System.out.println("Sum of odd numbers="+s2);
    }
}

```

**57. Write a menu-driven class to accept a number from the user and check whether it is a palindrome or a perfect number. [ICSE 2008]**

**i) Palindrome number– (a number is palindrome which when read in reverse order is same as read in the right order). Example: 11, 101, 151, etc.**

**ii) Perfect number– (a number is called Perfect if it is equal to the sum of its factors other than the number itself). Example: 6=1+2+3 [ICSE 2009]**

**Ans.** import java.util.\*;

```

public class Question57
{

```

```
static void main()
{
    Scanner sc=new Scanner(System.in);
    int d,n,ch,t,s=0,i;
    System.out.println("Enter a number:");
    n=sc.nextInt();
    System.out.println("MENU");
    System.out.println("1. to check whether it is a Palindrome Number");
    System.out.println("2. to check whether it is a Perfect Number");
    System.out.println("Enter your choice:");
    ch=sc.nextInt();
    switch(ch)
    {
        case 1:
            t=n;
            while(t!=0)
            {
                d=t%10;
                s=s*10+d;
                t=t/10;
            }
            if(s==n)
                System.out.println("Palindrome Number");
            else
                System.out.println("Not a Palindrome Number");
            break;
        case 2:
```

```

        for(i=1;i<n;i++)
        {
            if(n%i==0)
                s+=i;
        }
        if(s==n)
            System.out.println("Perfect Number");
        else
            System.out.println("Not a Perfect Number");
        break;
    default:
        System.out.println("Invalid Choice!");
    }
}
}

```

**58.** Write a menu-driven program to accept a number from the user and check whether it is 'BUZZ' number or to accept any two numbers and print the 'GCD' of them. [ICSE 2009]

i) A BUZZ number is the number which either ends with 7 or divisible by 7.

ii) GCD (Greatest Common Divisor) of two integers is calculated by continued division method.

Divide the larger number by the smaller; the remainder then divides the previous divisor. The process is repeated till the remainder is zero. The divisor then results the GCD.

**Ans.** import java.util.\*;

```

public class Question58
{
    static void main()

```



```
{
Scanner sc=new Scanner(System.in);
int d,n,ch,r;
System.out.println("MENU");
System.out.println("1. to check for BUZZ Number");
System.out.println("2. to find the GCD of two numbers");
System.out.println("Enter your choice:");
ch=sc.nextInt();
switch(ch)
{
case 1:
System.out.println("Enter a number:");
n=sc.nextInt();
if(n%10==7 || n%7==0)
System.out.println("BUZZ Number");
else
System.out.println("Not a BUZZ Number");
break;
case 2:
System.out.println("Enter 2 numbers:");
n=sc.nextInt();
d=sc.nextInt();
do
{
r=n%d;
if(r!=0)
{
```

```

        n=d;
        d=r;
    }
    }while(r!=0);
    System.out.println("GCD="+d);
break;
default:
    System.out.println("Invalid Choice!");
}
}
}

```

**59. Write a menu-driven program to accept a number and check and display whether it is a prime number or not OR an automorphic number or not (Use switch-case statement).**

**i) Prime number: A number is said to be a prime number if it is divisible only by 1 and itself and not by any other number. Example: 3,5,7,11,13, etc.**

**ii) Automorphic number: An automorphic number is the number which is contained in the last digit(s) of its square.**

**Example: 25 is an automorphic number as its square is 625 and 25 is present as the last two digits.**

**Ans.** import java.util.\*;

```

public class Question59
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,t,c=0,n,ch,r;
        System.out.println("MENU");
    }
}

```

```
System.out.println("1. to check for Prime Number");
System.out.println("2. to check for Automorphic Number");
System.out.println("Enter your choice:");
ch=sc.nextInt();
switch(ch)
{
    case 1:
        System.out.println("Enter a number:");
        n=sc.nextInt();
        for(i=1;i<=n;i++)
        {
            if(n%i==0)
                c++;
        }
        if(c==2)
            System.out.println("Prime Number");
        else
            System.out.println("Not a Prime Number");
        break;
    case 2:
        System.out.println("Enter a number:");
        n=sc.nextInt();
        t=n;
        while(t!=0)
        {
            c++;
            t=t/10;
        }
    }
}
```

```

    }
    if((n*n)%(int)Math.pow(10,c)==n)
        System.out.println("Automorphic Number");
    else
        System.out.println("Not an Automorphic Number");
    break;
default:
    System.out.println("Invalid Choice!");
}
}
}

```

**60. Write a program to input 10 integers and print the second largest number. Assume that there is at least one second largest number in the given set of integers. For example,**

**INPUT:**

**Enter 10 integers:  
12 35 46 22 34 56 78 89 23 21**

**OUTPUT:**

**Second Largest Integer: 78**

**Ans.** import java.util.\*;

```

public class Question60
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i,n,l=0,sl=0;
        System.out.println("Enter 10 numbers:");
        for(i=1;i<=10;i++)

```

```

    {
        n=sc.nextInt();
        if(i==1)
            l=n;
        else
        {
            if(n>l)
            {
                sl=l;
                l=n;
            }
            else if(sl==0)
                sl=n;
            else if(n>sl)
                sl=n;
        }
    }
    System.out.println("Second Largest Number="+sl);
}
}

```

**61. Write a program to find the sum of the following series:**

**$S=1+(1+2) + (1+2+3) + (1+2+3+4) + (1+2+3+4+5) + \dots+(1+2+3+4+\dots+10)$**

**[ICSE 2003]**

**Ans.** public class Question61

```

    {
        static void main()
        {

```

```

int i,p=0,s=0;
for(i=1;i<=10;i++)
{
    p=p+i;
    s=s+p;
}
System.out.println("Sum="+s);
}
}

```

**62. Write a program to calculate and print the sum of each of the following series:**

**i) Sum (S) = 2 - 4 + 6 - 8 + ..... -20**

**ii) Sum (S) =  $x/2 + x/5 + x/8 + x/11 + \dots + x/20$  where  $x$  is to be taken as input.**

**Ans.** import java.util.\*;

```

public class Question62
{
    static void sumi()
    {
        int i,s=0;
        for(i=2;i<=20;i+=2)
        {
            if(i%4==0)
                s=s-i;
            else
                s=s+i;
        }
        System.out.println("Sum="+s);
    }
}

```

```

static void sumii()
{
    int i,x;
    float s=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter the value of x:");
    x=sc.nextInt();
    for(i=2;i<=20;i+=3)
    {
        s=s+(float)x/i;
    }
    System.out.println("Sum="+s);
}
}

```

**63. Write a program to find the sum of series, taking the value of ‘a’ and ‘n’ from the user.**

$$s = a/2 + a/3 + a/4 + \dots + a/n$$

**Ans.** import java.util.\*;

```

public class Question63
{
    static void main()
    {
        int i,n,a;
        float s=0;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the value of a and n:");
    }
}

```

```

a=sc.nextInt();
n=sc.nextInt();
for(i=2;i<=n;i++)
{
    s=s+(float)a/i;
}
System.out.println("Sum="+s);
}
}

```

**64. Write a program to find the sum of series, taking the value of ‘a’ and ‘n’ from the user.**

$s = a + \frac{1}{2}a + \frac{3}{4}a + \frac{5}{6}a + \frac{7}{8}a + \dots$  up to  $n$  terms

**Ans.** import java.util.\*;

```

public class Question64
{
    static void main()
    {
        int i,n,a,p=1;
        float s=0;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the value of a and n:");
        a=sc.nextInt();
        n=sc.nextInt();
        for(i=1;i<=n;i++)
        {
            s=s+(float)(a+p)/(p+1);
            p=p+2;
        }
    }
}

```



```

    }
    System.out.println("Sum="+s);
}
}

```

**65. Write a program to find the sum of series, taking the value of 'n' from the user.**

$s = 1 + \frac{2}{2} * 3 + 2 + \frac{3}{3} * 4 + 3 + \frac{4}{4} * 5 + 4 + \frac{5}{5} * 6 + \dots$  up to  $n$  terms

**Ans.** import java.util.\*;

```

public class Question65
{
    static void main()
    {
        int i,n;
        float s=0;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the value of n:");
        n=sc.nextInt();
        for(i=1;i<=n;i++)
        {
            s=s+(float)(i+(i+1))/((i+1)*(i+2));
        }
        System.out.println("Sum="+s);
    }
}

```

**66. Write a program to compute and display the sum of the following series:**

[ICSE 2007]

$s = 1 + \frac{2}{1} * 2 + 1 + 2 + \frac{3}{1} * 2 * 3 + 1 + 2 + 3 + \frac{4}{1} * 2 * 3 * 4 + 1 + 2 + 3 + 4 + \frac{5}{1} * 2 * 3 * 4 * 5 + \dots$  up to  $n$  terms

where n is to be taken as input from the user.

**Ans.** import java.util.\*;

```
public class Question66
{
    static void main ()
    {
        int i,n,p=1,q=1;
        float s=0;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the value of n:");
        n=sc.nextInt();
        for(i=1;i<=n;i++)
        {
            p=p+(i+1);
            q=q*(i+1);
            s=s+(float)p/q;
        }
        System.out.println("Sum="+s);
    }
}
```

**67. Write a program to print the sum of negative numbers, sum of positive even numbers and sum of positive odd numbers from a list of numbers (N) entered by the user. The list terminates when the user enters a zero. [ICSE 2005]**

**Ans.** import java.util.\*;

```
public class Question67
{
```

```

static void main()
{
    int N,sn=0,spe=0,spo=0;
    Scanner sc=new Scanner(System.in);
    do
    {
        System.out.println("Enter the value of N:");
        N=sc.nextInt();
        if(N<0)
            sn+=N;
        else if(N>0)
        {
            if(N%2==0)
                spe+=N;
            else
                spo+=N;
        }
    }while(N!=0);
    System.out.println("Sum of negative numbers="+sn);
    System.out.println("Sum of positive even numbers="+spe);
    System.out.println("Sum of positive odd numbers="+spo);
}
}

```

**68. Shasha Travels Pvt. Ltd. gives the following discount to its customers:**

Ticket amount	Discount
Above ₹ 70000	18%
₹ 55001 to ₹ 70000	16%
₹ 35001 to ₹ 55000	12%
₹ 25001 to ₹ 35000	10%
less than ₹ 25001	2%

Write a program to input the name and ticket amount for the customer and calculate the discount amount and net amount to be paid. Display the output in the following format for each customer:

(Assume that there are 15 customers, first customer is given the serial no (SL. No. 1), next customer 2 ..... and so on.

```

Ans. import java.util.*;

public class Question68
{
    static void main()
    {
        int ta,i;
        float disc,net;
        String n;
        Scanner sc=new Scanner(System.in);
        for(i=1;i<=15;i++)
        {
            System.out.println("Enter the name of the customer:");
            n=sc.nextLine();
            System.out.println("Enter the ticket amount:");
            ta=sc.nextInt();
            sc.nextLine();//dummy input
            if(ta>70000)

```

```
        disc=18/100f*ta;
else if(ta>=55001)
        disc=16/100f*ta;
else if(ta>=35001)
        disc=12/100f*ta;
else if(ta>=25001)
        disc=16/100f*ta;
else
        disc=2/100f*ta;
net=ta-disc;
System.out.println("Serial Number="+i);
System.out.println("Customer Name="+n);
System.out.println("Ticket Amount="+ta);
System.out.println("Discount="+disc);
System.out.println("Net Amount="+net);
    }
}
}
```

**69. Using nested loops try producing the following patterns on the screen:**

i) 1  
12  
123  
1234  
12345

ii) 5  
54  
543  
5432  
54321

iii) 1  
21  
321  
4321  
54321

iv) 54321  
5432  
543  
54  
5

v) 54321  
4321  
321  
21  
1

vi) 1  
22  
333  
4444  
55555

vii) 55555  
4444  
333  
22  
1

viii) 11111  
2222  
333  
44  
5

ix) 5  
44  
333  
2222  
11111

x) \*  
\*\*  
\*\*\*  
\*\*\*\*  
\*\*\*\*\*

xi) \*\*\*\*\*  
\*\*\*\*  
\*\*\*  
\*\*  
\*

xii) \* \* \*  
\* \*  
\*  
\* \*  
\* \* \*

xiii) 1  
01  
101  
0101  
10101

xiv) 12345  
2345  
345  
45  
5

xv) 5  
45  
345  
2345  
12345

xvi) 1  
 21  
 321  
 4321  
 54321  
 654321

xvii) 1  
 11  
 101  
 1001  
 10001  
 100001

xviii) 1  
 11  
 112  
 1123  
 11235  
 112358

xix) 1  
 121  
 12321  
 1234321  
 123454321  
 12345654321

xx) 65432123456  
 543212345  
 4321321  
 32123  
 212  
 1

xxi) 12345  
 2345  
 345  
 45  
 5

xxii) 101010  
 010101  
 101010  
 010101

xxiii) 99999  
 7777  
 555  
 33  
 1

xxiv) 1  
 33  
 555  
 7777  
 99999

xxv) 12345  
 34512  
 45123  
 51234

xxvi) 54321  
 32154  
 21543  
 15432

xxvii) 12345  
 51234  
 45123  
 51234

**Ans.** class Question69

```
{
    static void pattern_i()
    {
        int i,j;
        for(i=1;i<=5;i++)
        {
            for(j=1;j<=i;j++)
            {
```

```
        System.out.print(j);
    }
    System.out.println();
}
}
static void pattern_ii()
{
    int i,j;
    for(i=5;i>=1;i--)
    {
        for(j=5;j>=i;j--)
        {
            System.out.print(j);
        }
        System.out.println();
    }
}
static void pattern_iii()
{
    int i,j;
    for(i=1;i<=5;i++)
    {
        for(j=i;j>=1;j--)
        {
            System.out.print(j);
        }
        System.out.println();
    }
}
```



```
    }  
}  
static void pattern_iv()  
{  
    int i,j;  
    for(i=1;i<=5;i++)  
    {  
        for(j=5;j>=i;j--)  
        {  
            System.out.print(j);  
        }  
        System.out.println();  
    }  
}  
static void pattern_v()  
{  
    int i,j;  
    for(i=5;i>=1;i--)  
    {  
        for(j=i;j>=1;j--)  
        {  
            System.out.print(j);  
        }  
        System.out.println();  
    }  
}  
static void pattern_vi()
```

```
{
    int i,j;
    for(i=1;i<=5;i++)
    {
        for(j=1;j<=i;j++)
        {
            System.out.print(i);
        }
        System.out.println();
    }
}

static void pattern_vii()
{
    int i,j;
    for(i=5;i>=1;i--)
    {
        for(j=1;j<=i;j++)
        {
            System.out.print(i);
        }
        System.out.println();
    }
}

static void pattern_viii()
{
    int i,j;
    for(i=1;i<=5;i++)
```

```
{
    for(j=5;j>=i;j--)
    {
        System.out.print(i);
    }
    System.out.println();
}
}
static void pattern_ix()
{
    int i,j;
    for(i=5;i>=1;i--)
    {
        for(j=5;j>=i;j--)
        {
            System.out.print(i);
        }
        System.out.println();
    }
}
static void pattern_x()
{
    int i,j;
    for(i=1;i<=5;i++)
    {
        for(j=1;j<=i;j++)
        {
```

```
        System.out.print("*");
    }
    System.out.println();
}
}
static void pattern_xi()
{
    int i,j;
    for(i=5;i>=1;i--)
    {
        for(j=1;j<=i;j++)
        {
            System.out.print("*");
        }
        System.out.println();
    }
}
static void pattern_xii()
{
    int i,j;
    for(i=1;i<=5;i++)
    {
        for(j=1;j<=5;j++)
        {
            if(i==j || i+j==6)
                System.out.print("*");
            else
```

```

        System.out.print(" ");
    }
    System.out.println();
}
}
static void pattern_xiii()
{
    int i,j;
    for(i=1;i<=5;i++)
    {
        for(j=1;j<=i;j++)
        {
            if((i+j)%2!=0)
                System.out.print(0);
            else
                System.out.print(1);
        }
        System.out.println();
    }
}
static void pattern_xiv()
{
    int i,j;
    for(i=1;i<=5;i++)
    {
        for(j=i;j<=5;j++)
        {

```

```

        System.out.print(j);
    }
    System.out.println();
}
}
static void pattern_xv()
{
    int i,j;
    for(i=1;i<=5;i++)
    {
        for(j=1;j<=5;j++)
        {
            if(i+j>=6)
                System.out.print(j);
            else
                System.out.print(" ");
        }
        System.out.println();
    }
}
static void pattern_xvi()
{
    int i,j;
    for(i=1;i<=6;i++)
    {
        for(j=i;j>=1;j--)
        {

```

```

        System.out.print(j);
    }
    System.out.println();
}
}
static void pattern_xvii()
{
    int i,j;
    for(i=1;i<=6;i++)
    {
        for(j=1;j<=i;j++)
        {
            if(i==j || j==1)
                System.out.print(1);
            else
                System.out.print(0);
        }
        System.out.println();
    }
}
static void pattern_xviii()
{
    int i,j,a,b,c;
    for(i=1;i<=6;i++)
    {
        a=1;b=0;
        for(j=1;j<=i;j++)

```

```
    {
        c=a+b;
        System.out.print(c);
        a=b;
        b=c;
    }
    System.out.println();
}
}
static void pattern_xix()
{
    int i,j;
    for(i=1;i<=6;i++)
    {
        for(j=i;j<=5;j++)
        {
            System.out.print(" ");
        }
        for(j=1;j<=i;j++)
        {
            System.out.print(j);
        }
        for(j=i-1;j>=1;j--)
        {
            System.out.print(j);
        }
        System.out.println();
    }
}
```



```

    }
}
static void pattern_xx()
{
    int i,j;
    for(i=6;i>=1;i--)
    {
        for(j=i;j<6;j++)
        {
            System.out.print(" ");
        }
        for(j=i;j>=1;j--)
        {
            System.out.print(j);
        }
        for(j=2;j<=i;j++)
        {
            System.out.print(j);
        }
        System.out.println();
    }
}
static void pattern_xxi()
{
    int i,j;
    for(i=1;i<=5;i++)
    {

```

```

        for(j=1;j<=5;j++)
        {
            if(j>=i)
                System.out.print(j);
            else
                System.out.print(" ");
        }
        System.out.println();
    }
}

static void pattern_xxii()
{
    int i,j;
    for(i=1;i<=5;i++)
    {
        for(j=1;j<=5;j++)
        {
            if((j+i)%2==0)
                System.out.print(1);
            else
                System.out.print(0);
        }
        System.out.println();
    }
}

static void pattern_xxiii()
{

```

```

int i,j;
for(i=1;i<=9;i+=2)
{
    for(j=1;j<=9;j+=2)
    {
        if(j>=i)
            System.out.print(10-i);
        else
            System.out.print(" ");
    }
    System.out.println();
}
}

static void pattern_xxiv()
{
    int i,j;
    for(i=1;i<=9;i+=2)
    {
        for(j=1;j<=9;j+=2)
        {
            if(j+i>=10)
                System.out.print(i);
            else
                System.out.print(" ");
        }
        System.out.println();
    }
}

```

```

}
static void pattern_xxv()
{
    /*
    12345
    23451
    34512
    45123
    51234
    */
    int i,j;
    for(i=1;i<=5;i++)
    {
        for(j=i;j<=5;j++)
        {
            System.out.print(j);
        }
        for(j=1;j<=i-1;j++)
        {
            System.out.print(j);
        }
        System.out.println();
    }
}

static void pattern_xxvi()
{

```

```
/*
54321
43215
32154
21543
15432
*/
int i,j;
for(i=5;i>=1;i--)
{
    for(j=i;j>=1;j--)
    {
        System.out.print(j);
    }
    for(j=5;j>=i+1;j--)
    {
        System.out.print(j);
    }
    System.out.println();
}
}
static void pattern_xxvii()
{
    /*
12345
51234
45123
```

```

34512
23451
*/
int i,j;
for(i=5;i>=1;i--)
{
    for(j=i+1;j<=5;j++)
    {
        System.out.print(j);
    }
    for(j=1;j<=i;j++)
    {
        System.out.print(j);
    }
    System.out.println();
}
}
}

```

**70. Write a program to input a number and print all its prime factors using prime factorization method.**

**For Example,**

**INPUT: Enter an integer: 24**

**OUTPUT: Prime factors using prime factorisation are:**

**2**  
**2**  
**2**  
**3**

**Ans.** import java.util.\*;

```

public class Question70
{
    static void main()
    {
        int n,p=2;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        System.out.println("Prime factors using prime factorisation are:");
        while(n>1)
        {
            if(n%p==0)
            {
                System.out.println(p);
                n=n/p;
            }
            else
                p++;
        }
    }
}

```

**71. Write a program to input a number and check whether it is a Smith number or not. Smith number is such a number the sum of whose digits equals the sum of the digits of its prime factors.**

**Smith number is a composite number in which the sum of its digits is equal to the sum of the digits of all its prime factors.**

**For example, 378 is a Smith number as the sum of the digits of 378 are:  $3+7+8 = 18$ .**

**The prime factors of 378 are: 2, 3, 3, 3, 7 (sum =  $2+3+3+3+7 = 18$ ).**

Similarly, 22 is a Smith number as the sum of the digits are:  $2+2=4$ . The prime factors of 22 are: 2 and 11 (sum =  $2+(1+1) = 4$ )

Other examples include 27, 58, 85, 94, 121, 166, 202, 265, etc.

**Ans.** import java.util.\*;

```
public class Question71
{
    static void main()
    {
        int n,d,s1=0,t,t1,s2=0,p=2;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        t=n;
        while(t!=0)
        {
            d=t%10;
            s1=s1+d;
            t=t/10;
        }
        t=n;
        while(t>1)
        {
            if(t%p==0)
            {
                t1=p;;
                while(t1!=0)
                {
                    d=t1%10;
```



```
        s2=s2+d;
        t1=t1/10;
    }
    t=t/p;
}
else
    p++;
}
if(s1==s2)
    System.out.println("Smith Number");
else
    System.out.println("Not Smith Number");
}
}
```

# Chapter 8: The Math Class

## Practice Questions

### ----->Objective-Type Questions<-----

#### State True or False:

1. It is possible to find the square root of a number using the `Math.pow()` function. [True]
2. `Math.sqrt()` function is used to find the square of a number. [True]
3. All trigonometric functions work with angles specified in radians. [True]
4. `Math.abs()` function is used to round off a real number to the nearest integer. [False]
5. `Math rint(12.5)` will return 12.0. [True]
6. `Math.ceil()` and `Math.floor()` are same. [False]
7. NaN stands for Not a Number, which is received when you pass a negative number to the `Math.sqrt()` function. [True]
8. `Math.cosec()` function is the reciprocal of `Math.sin()` function. [False]
9. `Math.cot(x)` function is used as a reciprocal of `Math.tan(x)` function. [False]
10. `Math.round()` function returns an int value if the parameter is float data type and long value if the parameter is of double data type. [True]

### ----->Subjective-Type Questions<-----

#### A. Answer the following questions:

1. Name the class that is used for different mathematical functions. Give an example of a mathematical function. [ICSE 2007]

**Ans.** Math class. `Math.sqrt()`

2. Give the output of the following expressions: [ICSE 2008]

i) If  $x = -9.99$ , calculate `Math.abs(x)`;

ii) If  $x = 9.0$ , calculate `Math.sqrt(x)`;

**Ans.** i) 9.99

ii) 3.0

3. Give the output of the following functions:

a) `Math.floor(-126.349)`

b) `Math.max(45.6,17.3)`

c) `Math.min(-0.0,0.0)`

d) `Math.pow(4,3)`

e) `Math.sqrt(625)`

f) `Math.cbrt(125)`

g) `Math.log10(1000)`

h) `Math rint(14.5)`

i) `Math.ceil(-12.56)`

j) `Math.floor(15.36)`

k) `Math.round(146.5)`

l) `Math.max(11,11)`

m) `Math.min(14.3,14.3)`

n) `Math rint(14.562)`

o) `Math.ceil(-14.0)`

**Ans.** a) -127.0

b) 45.6

c) -0.0

d) 64.0

e) 25.0

f) 5.0

g) 3.0

h) 14.0

i) -12.0

j) 15.0

k) 147

l) 11

m) 14.3

n) 15.0

o) -14.0

**4. Write equivalent Java expressions for the following:**

- |  |   |   |
|--|---|---|
| a) $\sqrt{a+b}$  | b) $\frac{1}{3}a^3 + \frac{1}{4}b^3$                    | c) $s = ut + \frac{1}{2}at^2$                                 |
| d) $d = \sqrt{l^2 + b^2}$                                    | e) $\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$                 | f) $\sin^2 x + \cos^2 y$                                      |
| g) $\frac{\tan a + \tan b}{1 - \tan a \cdot \tan b}$         | h) $\frac{1 - \log_{10} x}{1 + \log_{10} x}$            | i) $\frac{\sqrt[3]{5x^2 + \sin y}}{\sqrt[4]{\cos x + 11y^3}}$ |
| j) $\frac{\sin^{-1} x + \cos^{-1} y}{\sqrt{\log(\cos^2 x)}}$ | k) $\tan^{-1} x + \cos^2 y - 2x^2 y^3 - \sqrt{4e^{2x}}$ |   |
| l) $\sqrt{2as + u^2}$  |   |   |

- Ans.**
- a) Math.sqrt(a+b)
  - b) 1/3.0\*a\*a\*a+1/4.0\*b\*b\*b
  - c) s=u\*t+1/2.0\*a\*t\*t
  - d) d=Math.sqrt(l\*l+b\*b)
  - e) -b+Math.sqrt(b\*b-4\*a\*c)/(2\*a)  
-b-Math.sqrt(b\*b-4\*a\*c)/(2\*a)
  - f) Math.pow(Math.sin(x),2)+Math.pow(Math.cos(y),2)
  - g) (Math.tan(a)+Math.tan(b))/(1-Math.tan(a)\*Math.tan(b))
  - h) (1-Math.log10(x))/(1+Math.log10(x))
  - i) Math.cbrt(5\*x\*x+Math.sin(y))/Math.pow(Math.cos(x)+11\*y\*y\*y,1/4.0)
  - j) (Math.asin(x)+Math.acos(y))/Math.sqrt(Math.log(Math.cos(x)\*Math.cos(x)))
  - k) Math.atan(x)+Math.cos(y)\*Math.cos(y)-2\*x\*x\*y\*y\*y-  
Math.sqrt(4\*Math.exp(2\*x))
  - l) Math.sqrt(2\*a\*s+u\*u)

**B. Write programs for the following:**

**1. Write a program to input the area of a square and find its perimeter.**

**Ans.**

```
import java.util.*;
class Question1
{
    static void main()
    {
        double a,s,p;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the area of a square:");
        a=sc.nextDouble();
        s=Math.sqrt(a);
        p=4*s;
        System.out.println("Perimeter="+p);
    }
}
```

**2. Write a program that outputs the results of the following evaluations based on the number entered by the user:**

- i) Natural logarithm of the number
- ii) Absolute value of the number
- iii) Square root of the number
- iv) Random numbers between 0 and 1.

[ICSE 2006]

**Ans.** import java.util.\*;  
class Question2  
{  
static void main()  
{  
double n,l,a,s,r;  
Scanner sc=new Scanner(System.in);  
System.out.println("Enter a number:");  
n=sc.nextDouble();  
l=Math.log(n);  
a=Math.abs(n);  
s=Math.sqrt(n);  
r=Math.random();  
System.out.println("Natural Logarithm="+l);  
System.out.println("Absolute Value="+a);  
System.out.println("Square Root="+s);  
System.out.println("Random Number between 0 and 1="+r);  
}  
}

**3. Write a program to calculate the interest and total amount to be paid by entering the amount of loan and the number of years, either by simple interest method or by compound interest method, as per the choice of user.**

**For Simple Interest,**

**Interest=(Amount\*Rate \* Number of Years) / 100 and Total amount=Amount + Interest**

**For Compound Interest,**

$$\text{Total amount} = \text{Amount} \left( 1 + \frac{\text{Rate}}{100} \right)^{\text{No. of years}}$$

**Ans.** import java.util.\*;  
class Question3  
{  
static void main()  
{  
double p,r,t,i,a;  
int ch;  
Scanner sc=new Scanner(System.in);  
System.out.println("Enter the principle:");  
p=sc.nextDouble();  
System.out.println("Enter the rate:");  
r=sc.nextDouble();  
System.out.println("Enter the time:");  
t=sc.nextDouble();  
System.out.println("Enter 1 to find the Simple Interest");

```

System.out.println("Enter 2 to find the Compound Interest");
System.out.print("Enter your choice:");
ch=sc.nextInt();
switch(ch)
{
    case 1:
        i=(p*r*t)/100;
        a=p+i;
        System.out.println("Interest:"+i);
        System.out.println("Amount Payable:"+a);
        break;
    case 2:
        a=p*Math.pow((1+r/100),t);
        i=a-p;
        System.out.println("Interest:"+i);
        System.out.println("Amount Payable:"+a);
        break;
    default:
        System.out.println("Invalid Choice!");
}
}
}

```

**4. The following formulae describe the properties of portion of circle with radius 'r' and central angle 'x' in degrees:**

$$\text{Area} = \pi r^2$$

$$\text{Length of arc} = \pi r \frac{x}{180^\circ}$$

$$\text{Length of chord} = 2r \sin\left(\frac{x}{2}\right)$$

$$\text{Area of segment} = \frac{\pi r^2 x}{360} - \frac{r^2 \sin x}{2}$$

**If the values of r and x are entered through the keyboard. Write a program to calculate the above values of the circle.**

**Ans.** import java.util.\*;  
class Question4  
{  
 static void main()  
 {  
 double r,x,area,arc,chord,segment,y;  
 Scanner sc=new Scanner(System.in);  
 System.out.println("Enter the value of r and x:");  
 r=sc.nextDouble();  
 x=sc.nextDouble();

```

y=Math.toRadians(x); //Convert degrees to radians to use with Math.sin()
area=Math.PI*r*r;
arc=Math.PI*r*x/180;
chord=2*r*Math.sin(y/2);
segment=(Math.PI*r*r*x)/360-(r*r*Math.sin(y))/2;
System.out.println("Area="+area);
System.out.println("Length of arc="+arc);
System.out.println("Length of chord="+chord);
System.out.println("Area of segment="+segment);
    }
}

```

**5. Write a program to input the length of the 3 sides of a triangle (say a, b and c) and calculate the area depending upon the following criteria:**

- i) If the sum of any two sides of the triangle is lesser than the third side then print "A triangle cannot be formed"
- ii) If all the 3 sides are equal then print "It is an equilateral triangle" and calculate the area as

$$\text{area} = \frac{\sqrt{3}}{2} a^2$$

- iii) If the sum of the squares of any two sides is equal to the square of the third side then print "It is a right-angled triangle" and calculate the area as  $\text{area} = \frac{1}{2} \times \text{product of the perpendicular sides}$  otherwise calculate area as

$$\text{area} = \sqrt{s(s-a)(s-b)(s-c)} \text{ where } s = \frac{a+b+c}{2}$$

**Ans.** import java.util.\*;

```

class Question5
{
    static void main()
    {
        int a,b,c;
        double area,s;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the three sides of a triangle:");
        a=sc.nextInt();
        b=sc.nextInt();
        c=sc.nextInt();
        if(a<b+c && b<a+c && c<a+b)
        {
            if(a==b && b==c)
            {
                System.out.println("It is an equilateral triangle");
                area=Math.sqrt(3)/2*a*a;
            }
            else if(a*a==b*b+c*c || b*b==a*a+c*c || c*c==a*a+b*b)
            {

```

```

        if(a*a==b*b+c*c)
            area=1/2.0*b*c;
        else if(b*b==a*a+c*c)
            area=1/2.0*a*c;
        else
            area=1/2.0*a*b;
        System.out.println("It is a right-angled triangle");
    }
    else
    {
        s=(a+b+c)/2.0;
        area=Math.sqrt(s*(s-a)*(s-b)*(s-c));
        System.out.println("It is a scalene triangle");
    }
    System.out.println("Area of the triangle="+area);
}
else
    System.out.println("Triangle cannot be formed");
}
}

```

**6. Write a program to calculate the values of X, where**

$$X = + \frac{\tan(a) + \tan(b)}{1 + \tan(a) \tan(b)}$$

Where the value of  $a$  ranges from  $0^\circ$  to  $90^\circ$  and the corresponding value of  $b$  is  $90^\circ - a$ .

**Ans.** class Question6

```

{
    static void main()
    {
        double a,b,x,numerator,denominator;
        for(a=0;a<=90;a++)
        {
            b=90-a;
            numerator=Math.tan(Math.toRadians(a))+Math.tan(Math.toRadians(b));

            denominator=1+Math.tan(Math.toRadians(a))*Math.tan(Math.toRadians(b));
            x=numerator/denominator;
            System.out.println(x);
        }
    }
}

```

**7. Write a program to calculate and print the values of:**

$$Z = \frac{x^2 + y^2}{x + y}$$

Where,  $x$  ranges from 0 to 50 and  $y$  remains constant at 5. Use symbolic constant for  $y$ .

**Ans.** class Question7

```
{
    static void main()
    {
        float x,Z;
        final int y=5;
        for(x=0;x<=50;x++)
        {
            Z=(x*x+y*y)/(x+y);
            System.out.println(Z);
        }
    }
}
```

**8. Write a program using conditional operator to input two integers and if the difference is positive, find the square root of its difference, otherwise find its cube root.**

**Ans.** import java.util.\*;

```
class Question8
{
    static void main()
    {
        int a,b,d;
        double ans;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter two integers:");
        a=sc.nextInt();
        b=sc.nextInt();
        d=a-b;
        ans=(d>0)?Math.sqrt(d):Math.cbrt(d);
        System.out.println("Answer="+ans);
    }
}
```



9. Write a program to input two integers, say x and n, and find the sum of the following series:

$$i) x + \frac{x^2}{2} + \frac{x^3}{3} + \frac{x^4}{4} + \frac{x^5}{5} + \frac{x^6}{6} + \frac{x^7}{7} + \dots + \frac{x^n}{n}$$

$$ii) \frac{x^2}{2} + \frac{x^3}{3} + \frac{x^4}{4} + \frac{x^5}{5} + \frac{x^6}{6} + \dots + \frac{x^n}{n+1}$$

$$iii) x^2 + \frac{x^3}{2} + \frac{x^4}{3} + \frac{x^5}{4} + \frac{x^6}{5} + \frac{x^7}{6} + \dots + \frac{x^{n+1}}{n}$$

$$iv) x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \frac{x^5}{5} - \frac{x^6}{6} + \frac{x^7}{7} + \dots \pm \frac{x^n}{n}$$

$$v) \frac{x}{2} - \frac{x^2}{3} + \frac{x^3}{4} - \frac{x^4}{5} + \frac{x^5}{6} - \frac{x^6}{7} + \dots \pm \frac{x^n}{n+1}$$

$$vi) x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \frac{x^5}{5!} + \frac{x^6}{6!} + \frac{x^7}{7!} + \dots + \frac{x^n}{n!}$$

[where ! stands for factor, which is the product of all natural no.s till that number.

For e.g.  $5! = 1 \times 2 \times 3 \times 4 \times 5$ , similarly  $7! = 1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7$ ]

$$vii) \frac{x}{2!} + \frac{x^2}{3!} + \frac{x^3}{4!} + \frac{x^4}{5!} + \frac{x^5}{6!} + \frac{x^6}{7!} + \dots + \frac{x^n}{(n+1)!}$$

$$viii) x^2 + \frac{x^3}{2!} + \frac{x^4}{3!} + \frac{x^5}{4!} + \frac{x^6}{5!} + \frac{x^7}{6!} + \dots + \frac{x^{n+1}}{n!}$$

$$ix) x - \frac{x^2}{2!} + \frac{x^3}{3!} - \frac{x^4}{4!} + \frac{x^5}{5!} - \frac{x^6}{6!} + \frac{x^7}{7!} + \dots \pm \frac{x^n}{n!}$$

$$x) \frac{x}{2!} - \frac{x^2}{3!} + \frac{x^3}{4!} - \frac{x^4}{5!} + \frac{x^5}{6!} - \frac{x^6}{7!} + \dots \pm \frac{x^n}{(n+1)!}$$

$$xi) x^2 - \frac{x^3}{2!} + \frac{x^4}{3!} - \frac{x^5}{4!} + \frac{x^6}{5!} - \frac{x^7}{6!} + \dots \pm \frac{x^{n+1}}{n!}$$

**Ans.**

```
import java.util.*;
class Question9
{
    static void series_i()
    {
        int x,n,i;
        double sum=0;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter two integers:");
        x=sc.nextInt();
        n=sc.nextInt();
```

```
    for(i=1;i<=n;i++)
    {
        sum+=Math.pow(x,i)/i;
    }
    System.out.println("Answer="+sum);
}
```

```
static void series_ii()
{
    int x,n,i;
    double sum=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter two integers:");
    x=sc.nextInt();
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        sum+=Math.pow(x,i)/(i+1);
    }
    System.out.println("Answer="+sum);
}
```

```
static void series_iii()
{
    int x,n,i;
    double sum=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter two integers:");
    x=sc.nextInt();
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        sum+=Math.pow(x,i+1)/i;
    }
    System.out.println("Answer="+sum);
}
```

```
static void series_iv()
{
    int x,n,i;
    double sum=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter two integers:");
    x=sc.nextInt();
    n=sc.nextInt();
    for(i=1;i<=n;i++)
```

```

    {
        if(i%2==0)
            sum-=Math.pow(x,i)/i;
        else
            sum+=Math.pow(x,i)/i;
    }
    System.out.println("Answer="+sum);
}

static void series_v()
{
    int x,n,i;
    double sum=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter two integers:");
    x=sc.nextInt();
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        if(i%2==0)
            sum-=Math.pow(x,i)/(i+1);
        else
            sum+=Math.pow(x,i)/(i+1);
    }
    System.out.println("Answer="+sum);
}

static void series_vi()
{
    int i,j,x,n;
    long f;
    double sum=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter two integers:");
    x=sc.nextInt();
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        f=1;
        for(j=1;j<=i;j++)
            f=f*j;

        sum+=Math.pow(x,i)/f;
    }
    System.out.println("Answer="+sum);
}

```

```

static void series_vii()
{
    int i,j,x,n;
    long f;
    double sum=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter two integers:");
    x=sc.nextInt();
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        f=1;
        for(j=1;j<=i+1;j++)
            f=f*j;

        sum+=Math.pow(x,i)/f;
    }
    System.out.println("Answer="+sum);
}

```

```

static void series_viii()
{
    int i,j,x,n;
    long f;
    double sum=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter two integers:");
    x=sc.nextInt();
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        f=1;
        for(j=1;j<=i;j++)
            f=f*j;

        sum+=Math.pow(x,i+1)/f;
    }
    System.out.println("Answer="+sum);
}

```

```

static void series_ix()
{
    int i,j,x,n;
    long f;
    double sum=0;
    Scanner sc=new Scanner(System.in);

```

```

System.out.println("Enter two integers:");
x=sc.nextInt();
n=sc.nextInt();
for(i=1;i<=n;i++)
{
    f=1;
    for(j=1;j<=i;j++)
        f=f*j;

    if(i%2==0)
        sum-=Math.pow(x,i)/f;
    else
        sum+=Math.pow(x,i)/f;
}
System.out.println("Answer="+sum);
}

```

```

static void series_x()
{
    int i,j,x,n;
    long f;
    double sum=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter two integers:");
    x=sc.nextInt();
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        f=1;
        for(j=1;j<=i+1;j++)
            f=f*j;

        if(i%2==0)
            sum-=Math.pow(x,i)/f;
        else
            sum+=Math.pow(x,i)/f;
    }
    System.out.println("Answer="+sum);
}

```

```

static void series_xi()
{
    int i,j,x,n;
    long f;
    double sum=0;
    Scanner sc=new Scanner(System.in);

```

```

System.out.println("Enter two integers:");
x=sc.nextInt();
n=sc.nextInt();
for(i=1;i<=n;i++)
{
    f=1;
    for(j=1;j<=i;j++)
        f=f*j;

    if(i%2==0)
        sum-=Math.pow(x,i+1)/f;
    else
        sum+=Math.pow(x,i+1)/f;
}
System.out.println("Answer="+sum);
}
}

```

**10. Write a program to find the sum of the following logarithmic series:**

i)  $\frac{\log 0.1}{1} + \frac{\log 0.2}{2} + \frac{\log 0.3}{3} + \frac{\log 0.4}{4} + \dots + \frac{\log 0.n}{n}$

ii)  $\log_{10} 1 - \log_{10} 2 + \log_{10} 3 - \log_{10} 4 + \dots + \log_{10} n$

**Ans.**

```

import java.util.*;
class Question10
{
    static void series_i()
    {
        int n,i,t,c;
        double s=0;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the the value of n:");
        n=sc.nextInt();
        for(i=1;i<=n;i++)
        {
            t=i;
            c=0;
            while(t!=0)
            {
                c++;
                t=t/10;
            }
            s=s+Math.log(i/Math.pow(10,c))/i;
        }
        System.out.println("Sum="+s);
    }
}

```

```

static void series_ii()
{
    int n,i,t,c;
    double s=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter the the value of n:");
    n=sc.nextInt();
    for(i=1;i<=n;i++)
    {
        if(i%2==0)
            s=s-Math.log10(i);
        else
            s=s+Math.log10(i);
    }
    System.out.println("Sum="+s);
}
}

```

**11. A projectile fired at an angle  $\theta$ , given an initial velocity of  $v_0$ , will travel a distance**

$d = \frac{v_0^2}{g} \sin 2\theta$  where  $g$  is the acceleration constant of  $9.8/\text{sec}^2$ . It will be in motion for time  $t$  given by  $t = \frac{2v_0 \sin \theta}{g}$  and reach a maximum height  $h$  of  $\frac{v_0^2}{g} \sin^2 \theta$ . Write a program which

**produces a table of the following form for all angles from 30 to degrees in 5 degree increments and initial velocities from 500 to 1000m/sec in 100m/sec increments:**

**(INITIAL VELOCITY=500m/sec)**

ANGLE	DISTANCE	TIME	HEIGHT
-------	----------	------	--------

**Ans.** class Question11

```

{
    static void main()
    {
        double vo, theta, d,t,h;
        final double g=9.8;
        vo=500; //initial velocity
        System.out.println("ANGLE\t\tDISTANCE\t\tTIME\t\tHEIGHT");
        for(theta=30;theta<=90;theta+=5)
        {
            d=(vo*vo)/g*Math.sin(Math.toRadians(2*theta));
            t=2*vo*Math.sin(Math.toRadians(theta))/g;
            h=vo*vo/g*Math.sin(Math.toRadians(theta));
            System.out.println(theta+"\t\t"+d+"\t\t"+t+"\t\t"+h);
        }
    }
}
}

```

**12. A game of dice is to be simulated for two players, each player gets a chance to throw his dice, and the value is added to his points, this process continues alternately until for the player whose added points equal to 20 and is declared the winner. Write a program to simulate this process using the *random()* function.**

**Ans.** class Question12

```
{
    static void main()
    {
        int d1,d2,s1=0,s2=0;
        do
        {
            d1=1+(int)(Math.random()*6); //dice throw for first player
            s1+=d1;//Add the value of the dice to the first players total
            if(s1>=20)
                break;
            d2=1+(int)(Math.random()*6); //dice throw for second player
            s2+=d2;//Add the value of the dice to the second players total
            if(s2>=20)
                break;
        }while(s1<20 && s2<20);
        if(s1>=20)
            System.out.println("First player is the winner");
        else
            System.out.println("Second player is the winner");
    }
}
```

**13. A number  $n$  is said to be a sunny number if  $\sqrt{n+1}$  is equal to an integer. For example, 8 is sunny as  $\sqrt{8+1} = 3$  which is an integer. Write a program to print all sunny numbers between 1 to100.**

**Ans.** class Question13

```
{
    static void main()
    {
        int i;
        for(i=1;i<=100;i++)
        {
            if(Math.sqrt(i+1)==(int)Math.sqrt(i+1))
                System.out.println(i);
        }
    }
}
```

**14. Write a program to input an angle (say  $n$ ) and find its:**

- i) TAN( $n$ )
- ii) SIN( $n$ )
- iii) COS( $n$ )



according to the user's choice.

```
Ans. import java.util.*;
class Question14
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n,ch;
        System.out.println("Enter the value of n:");
        n=sc.nextInt();
        System.out.println("Enter 1 to find TAN(n)");
        System.out.println("Enter 2 to find SIN(n)");
        System.out.println("Enter 3 to find COS(n)");
        System.out.println("Enter your choice:");
        ch=sc.nextInt();
        switch(ch)
        {
            case 1:
                System.out.println("TAN(n)="+Math.tan(n));
                break;
            case 2:
                System.out.println("SIN(n)="+Math.sin(n));
                break;
            case 3:
                System.out.println("COS(n)="+Math.cos(n));
                break;
            default:
                System.out.println("Wrong Choice!");
        }
    }
}
```

**15. Write a function that simulates rolling a pair of dice until the total on the dice comes up to be a given number. The number that you are rolling for is a parameter to the function. The number of times you have to roll the dice is the return value of the function. You can assume that the parameter is one of the possible totals: 2, 3, ..., 12. Use your function in a program that computes and prints the number of rolls it takes to get snake eyes. (Snake eyes means that the total showing on the dice is 2.)**

```
Ans. class Question15
{
    static int rollingPair(int n)
    {
        int d1,d2,c=0;
        do
        {
            c++;
```

```

        d1=1+(int)(Math.random()*6);
        d2=1+(int)(Math.random()*6);
        if(d1+d2==n)
            break;

    }while(true);
    return c;
}
static void main()
{
    int s;
    s=rollingPair(2);
    System.out.println("Number of rollings for Snake eyes:"+s);
}
}

```

**16.** The roots (say  $x_1$  and  $x_2$ ) of a quadratic equation  $ax^2+bx+c=0$  can be determined by its discriminant ( $d$ ), where  $d = b^2 - 4ac$ . Depending upon its value the nature of the roots and the value of the roots can be determined.

i) If  $d=0$  then the roots are *Real* and *Equal* and the roots can be found using the formula:

$$\frac{-b}{2a}$$

ii) If  $d>0$  then the roots are *Real* and *Distinct* and the roots can be found using the formula:

$$\frac{-b \pm \sqrt{d}}{2a}$$

iii) If  $d<0$  then the roots are *Imaginary* and therefore roots are not found.

Write a program to input  $a, b$  and  $c$  the coefficient of the quadratic equation and find its roots.

**Ans.** import java.util.\*;

```
class Question16
```

```
{
```

```
    static void main()
```

```
    {
```

```
        Scanner sc=new Scanner(System.in);
```

```
        int a,b,c,d;
```

```
        double r1,r2;
```

```
        System.out.println("Enter the coefficients of the quadratic equation:");
```

```
        a=sc.nextInt();
```

```
        b=sc.nextInt();
```

```
        c=sc.nextInt();
```

```
        d=b*b-4*a*c;
```

```
        if(d==0)
```

```
        {
```

```
            System.out.println("Roots are Real and Equal");
```

```
            r1=r2=-b/(2*a);
```

```

        System.out.println("Roots are:"+r1+"\t"+r2);
    }
    else if(d>0)
    {
        System.out.println("Roots are Real and Distinct");
        r1=(-b+Math.sqrt(d))/(2*a);
        r2=(-b-Math.sqrt(d))/(2*a);
        System.out.println("Roots are:"+r1+"\t"+r2);
    }
    else
    {
        System.out.println("Roots are Imaginary");
        System.out.println("Roots not found");
    }
}
}
}

```

**17. Write a program for all four operations (addition, subtraction, multiplication and division according to the user's choice). Each of the problem is to be random. Also if the user gets a problem incorrect, give him/her a second try at the problem. If the problem is correct display "Right", otherwise display "Wrong" after the second try.**

Sample Output 1:	Sample Output 2:
MENU	MENU
1. Addition	1. Addition
2. Subtraction	2. Subtraction
3. Multiplication	3. Multiplication
4. Division	4. Division
Enter your choice: 1	Enter your choice: 3
What is 5 +9=? 13	What is 7 x 8=? 56
Incorrect - please try again.	Right
What is 5 +9=? 11	
Wrong 5+9=14	

**Ans.**

```
import java.util.*;
class Question17
{
```

```

static void main()
{
    Scanner sc=new Scanner(System.in);
    int ch,a,b,c;
    float d;
    System.out.println("MENU");
    System.out.println("1.Addition");
    System.out.println("2.Subtraction");
    System.out.println("3.Multiplication");
    System.out.println("4.Division");
    System.out.print("Enter your choice:");
    ch=sc.nextInt();
    switch(ch)
    {
        case 1:
            a=1+(int)(Math.random()*100); //generating a random number between 1 to 100
            b=1+(int)(Math.random()*100); //generating a random number between 1 to 100
            System.out.print("What is "+a+"+"+b+"=?");
            c=sc.nextInt();
            if(c==a+b)
                System.out.print("Right");
            else
            {
                System.out.println("Incorrect-please try again");
                System.out.print("What is "+a+"+"+b+"=?");
                c=sc.nextInt();
                if(c==a+b)
                    System.out.print("Right");
                else
                    System.out.print("Wrong "+a+"+"+b+"="+a+b);
            }
            break;
        case 2:
            a=1+(int)(Math.random()*100); //generating a random number between 1 to 100
            b=1+(int)(Math.random()*100); //generating a random number between 1 to 100
            System.out.print("What is "+a+"-"+b+"=?");
            c=sc.nextInt();
            if(c==a-b)
                System.out.print("Right");
            else
            {
                System.out.println("Incorrect-please try again");
                System.out.print("What is "+a+"-"+b+"=?");
                c=sc.nextInt();
                if(c==a-b)
                    System.out.print("Right");
                else
                    System.out.print("Wrong "+a+"-"+b+"="+a-b);
            }
            break;
        case 3:
            a=1+(int)(Math.random()*100); //generating a random number between 1 to 100
            b=1+(int)(Math.random()*100); //generating a random number between 1 to 100
            System.out.print("What is "+a+"*"+b+"=?");
            c=sc.nextInt();
            if(c==a*b)

```

```

        System.out.print("Right");
    else
    {
        System.out.println("Incorrect-please try again");
        System.out.print("What is "+a+"*"+b+"=?");
        c=sc.nextInt();
        if(c==a+b)
            System.out.print("Right");
        else
            System.out.print("Wrong "+a+"*"+b+"="+a*b);
    }
    case 4:
        a=1+(int)(Math.random()*100); //generating a random number between 1 to 100
        b=1+(int)(Math.random()*100); //generating a random number between 1 to 100
        System.out.print("What is "+a+"/"+b+"=?");
        c=sc.nextInt();
        if(c==(float)a/b)
            System.out.print("Right");
        else
        {
            System.out.println("Incorrect-please try again");
            System.out.print("What is "+a+"/"+b+"=?");
            c=sc.nextInt();
            if(c==(float)a/b)
                System.out.print("Right");
            else
                System.out.print("Wrong "+a+"/"+b+"="+a/b);
        }
        break;
    default:
        System.out.println("Wrong Choice!");
    }
}
}
}
}
}

```

**18. Write a program to input a real number (floating point number) and round off each number to 2 places of decimal and display the answer.**

For example, if

**INPUT: 12.3867**

**OUTPUT: 12.39**

similarly, if

**INPUT: 73.2846**

**OUTPUT: 73.28**

**Ans.** import java.util.\*;

```
class Question18
```

```
{
```

```
    static void main()
```

```
    {
```

```
        float n,f;
```

```
        int r;
```

```
        Scanner sc=new Scanner(System.in);
```

```
        System.out.println("Enter a real number:");
```

```
        n=sc.nextFloat();
```

```

        r=Math.round(n*100);//Move the decimal place 2 digits to the right
        f=(float)r/100;
        System.out.println(f);
    }
}

```

**19. Write a program to input 10 numbers and find their sum after converting each number to its equivalent positive value.**

**Ans.** import java.util.\*;  
class Question19  
{  
 static void main()  
 {  
 float n,s=0;  
 int i;  
 Scanner sc=new Scanner(System.in);  
 System.out.println("Enter 10 numbers:");  
 for(i=1;i<=10;i++)  
 {  
 n=sc.nextFloat();  
 s=s+Math.abs(n);  
 }  
 System.out.println("Sum="+s);  
 }  
}

**20. Write a program to input a binary number (consisting of only 1's and 0's) and convert it to its equivalent decimal.**

For example, if

**INPUT: 110**

**OUTPUT: 6**

similarly if

**INPUT: 1002**

**OUTPUT: No a proper binary number.**

**Ans.** import java.util.\*;  
class Question20  
{  
 static void main()  
 {  
 int n,s=0,c=0,f=0,d;  
 Scanner sc=new Scanner(System.in);  
 System.out.println("Enter a binary number:");  
 n=sc.nextInt();  
 while(n!=0)  
 {  
 d=n%10;  
 if(d!=1 && d!=0)

```
    {
        f=1;
        break;
    }
    s=s+d*(int)Math.pow(2,c++);
    n=n/10;
}
if(f==0)
    System.out.println("Decimal Equivalent="+s);
else
    System.out.println("Not a proper binary number");
}
}
```

# Chapter 9: Functions/Methods

## Practice Questions

### ----->Objective-Type Questions<-----

#### A. Fill in the blanks:

1. In Java functions are known as Methods.
2. Methods are contained in class.
3. The function name and the parameter list together is known as Function Signature.
4. The access specifier, return type and the function signature together is known as Function prototype.
5. The arguments of the function given in function prototype are called Formal Parameters.
6. The arguments of the function given in statement that calls the function are called Actual Parameters.
7. If a function does not return any value, the returning type in the function prototype will be void.
8. When a function is called by Value, the values of actual parameters are copied into separate memory locations as allocated by the formal parameters.
9. Impure functions use call by Reference.
10. One of the practical implementations of polymorphism is Method Overloading.

#### B. Write function prototypes for the following:

1. **private access method sum which accepts three int type variables as parameters and return a float type.**

**Ans.** private float sum(int a, int b, int c)

2. **default access method Length which accepts a String type variable as parameters and return an int type.**

**Ans.** int Length(String s)

3. **public access method increment which accepts an object of Myclass type as parameters and do not return any value.**

**Ans.** public void increment(Myclass ob)

4. **protected access method largest which accepts a float type, int type and double type data as parameters and have a byte type as return type.**

**Ans.** protected byte largest(float a, int b, double c)

5. **public access static method calculate which accepts a byte and int type data type as parameters and return float type.**

**Ans.** public static float calculate(byte a, int b)

6. **Write the function prototype for the function "sum" that takes an integer variable (x) as its argument and returns a value of float data type.** [ICSE 2009]

**Ans.** float sum(int x)

7. **Write the prototype of a function which takes in 2 integers and 1 String arguments and returns a value which is either 'true' or false'. [ICSE 2010]**

**Ans.** boolean function (int a, int b, String s)

### ----->Subjective-Type Questions<-----

#### A. Answer the following questions:



**1. What is a method?**

**Ans.** Function or Method is a block of code containing executable lines of code that represents the behavioural aspect of an object.

**2. Write two advantages of using functions in a program.**

[ICSE 2010]

**Ans.** The Advantages of using functions are:

- a) **Reduces Complexity:** Breaking a large task into smaller tasks automatically reduces complexity, thereby increasing the maintainability of the program.
- b) **Reusability:** Once a method is defined, it can be invoked again and again from different segments of a program thus reusing the method whenever required.

**3. Explain the function of a return statement.**

[ICSE 2006]

**Ans.** The **return** statement is used to return-back or exit from a function. It comes in two variants:

1. **return;** It is generally used with function having **void** *return-type* and is used to explicitly force an exit from the function.
2. **return <value>;** It is used with function which have to return a value to the place where it is called from. Thus it is used with functions having *return-type* anything other than **void**. This statement not only forces an exit from the function but also returns a value.

**4. If a function contains several return statements, how many of them will be executed?**

[ICSE 2007]

**Ans.** If a function contains several **return** statements, only the first one which JVM comes across is executed. Thus forcing an exit from the function and therefore the remaining **return** statements are not executed.

**5. Name the keyword that causes the control to transfer back to the method call.**

[ICSE 2009]

**Ans.** The keyword that causes the control to transfer back to the method call is **return**.

**6. What is the role of the keyword void in declaring functions?**

[ICSE 2007]

**Ans.** The **void** *return-type* of a function is used to ensure that a function do not return any value.

**7. Classify functions depending upon the value it returns.**

**Ans.** Depending on the value that is returned by a function, it is classified as:

- a) Computational Function
- b) Manipulative Function
- c) Procedural Function

**8. Differentiate between Formal Parameter and Actual Parameter.**

[ICSE 2007, 2008]

**Ans.** The difference between Actual Parameters and Formal Parameters are:

Actual Parameters	Formal Parameters
The parameter list representing the actual value that is given during function invocation.	The parameter list as given in the function prototype that accepts the values from the calling function.
The parameter list may be variables with actual values or constants directly passed to a function.	The parameter list is always variables that accepts the values.

**9. State the difference between function prototype and function signature.**

**Ans.** The difference between function prototype and function signature are:

Function Prototype	Function Signature
Function prototype refers to the return-type, function-name and the parameter-list.	Function signature on the other hand refers to the parameter list, i.e. the data types that comprise the parameter-list, the number of variables and their order.

**10. How are functions called? How does it return a value?**

**Ans.** A function may be called or executed in Java by simply writing the function name followed by the parameter-list within parenthesis. For example, if a function prototype contains the following:

```
void totalSurfaceArea(float l, float b, float h);
```

when calling the function it should be specified as:

```
totalsurfaceArea(x,y,z);
```

where x, y and z are variables of float type and are assigned to l, b and h respectively.

After a function computes a value, it can be returned to the place where it is invoked using the general syntax:

```
return <value>;
```

**11. What is call by value?**

[ICSE 2005]

**Ans.** During call by value a copy of the actual parameters is made by the formal parameters and therefore any changes made to the formal parameters is not reflected by the actual parameters.

**12. State the difference between Call by Value and Call by Reference.**

**Ans.** The difference between Call by Value and Call by Reference is that:

Call by Value	Call by Reference
Using this technique a copy of the values in the actual parameter is made in the formal parameters.	Using this technique a reference to the values in the actual parameter is made by the formal parameters.
Any change made to the formal parameters is not reflected back in the actual parameters.	Any change made to the formal parameters is reflected back in the actual parameters.
Usually primitive data type is used during call by value.	Usually arrays and objects are used during call by reference.

**13. How are the following passed?**

[ICSE 2005]

i) Primitive types

ii) Reference types

**Ans.** i) Call by Value

ii) Call by Reference

**14. Classify functions which accepts objects as parameters.**

**Ans.** i) Pure Functions

ii) Impure Functions

**15. Define an impure function.**

[ICSE 2006]

**Ans.** Impure functions are such methods which change the state of an object.

**16. Differentiate between pure and impure functions.**

[ICSE 2009]

**Ans.** Difference between Pure and Impure functions:

Pure Functions	Impure Functions
Pure functions are such functions which do not change the state of an object passed as parameter.	Impure functions are such functions which change the state of an object passed as parameter.
It doesn't have any side effects as the state of the object is not changed, rather only accessed.	It does have side effects as the state of the object is changed, and therefore one should be careful using it.

**17. Explain function overloading with an example.**

[ICSE 2006]

**Ans.** Multiple functions with the same name but different parameter list is termed as function overloading. When an overloaded function is invoked, Java uses the type and/or number of arguments as its guide to determine which version of the overloaded method to actually call.

Example,

// Demonstrate method overloading.

```
public class OverloadDemo
```

```
{
```

```
    void test()
```

```
    {
```

```
        System.out.println("No parameters");
```

```
    }
```

```
    //Overload test for one integer parameter.
```

```
    void test(int a)
```

```

{
    System.out.println("a:" + a);
}
//Overload test for two integer parameters.
void test(int a, int b)
{
    System.out.println("a and b: " + a + " " + b);
}
//Overload test for a double parameter
double test(double a)
{
    System.out.println("double a: " + a );
    return a*a;
}
static void call()
{
    OverloadDemo ob=new OverloadDemo();
    double result;
    ob.test();
    ob.test(5);
    ob.test(7,9);
    result=ob.test(12.73);
    System.out.println("Result of ob.test(12.73): "+result);
}
}

```

**18. Which OOP principle implements function overloading? [ICSE 2007]**

**Ans.** The principle of Polymorphism is implemented using Function Overloading.

**19. When there are multiple definitions with the same function name, what makes them different from each other? [ICSE 2009]**

**Ans.** When there are multiple definitions with the same name, the parameter-list makes them different from each other.

**20. What are the different access specifiers available in Java?**

**Ans.** The different access specifiers in Java are: default, public, protected and private.

**21. What is the function of main( ) method?**

**Ans.** In conventional Java the main( ) function is from where the program execution begins and is therefore also called the driver function. In BlueJ environment main() do not have any relevance at all it behaves exactly like any other function in Java.

**22. How are static methods of one class called by methods in other classes?**

**Ans.** Methods which are **static** are called by other classes using the **class** name of which it is a content followed by the method call statement. For example,

```

class Myclass
{
    static void trial()
    {

    }
}

```

```

class Anyclass

```

```

    {
        static void call()
        {
            MyClass.trial();//static method of MyClass is invoked
        }
    }

```

## B. Answer as directed:

1. In the program given below:

[ICSE 2012]

```

class MyClass
{
    static int x = 7;
    int y = 2;
    public static void main(String args[ ])
    {
        MyClass obj = new MyClass( );
        System.out.println(x);
        obj.sampleMethod(5);
        int a= 6;
        System.out.println(a);
    }

    void sampleMethod(int n)
    {
        System.out.println(n);
        System.out.println(y);
    }
}

```

State the name and value of the:

- i) method argument or argument variable.
- ii) class variable.
- iii) local variable.
- iv) instance variable .

**Ans.** i) int n value= 5  
 ii) int x value=7  
 iii) int a value 6  
 iv) int y value 2

## Lab Exercises

Write programs for the following:

1. Create a method which accepts two int type variable a and b as parameter and evaluate the following expression:

$$\frac{4.5a + b^2}{a - b}$$

and return it.

**Ans.** class Question1

```

{
    static float calc(int a,int b)

```

```

    {
        float c;
        c=(float)(4.5*a+b*b)/(a-b);
        return c;
    }
}

```

**2. Create a function which accepts an integer as parameter and return true if it is a prime number otherwise return false. In the main() method input an integer and using the above method check whether it is a prime number or not.**

**Ans.** import java.util.\*;

```

class Question2
{
    static boolean isPrime(int n)
    {
        int i,c=0;
        for(i=1;i<=n;i++)
        {
            if(n%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    static void main()
    {
        int n;
        boolean p;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        p=isPrime(n);
        if(p==true)
            System.out.println("Prime Number");
        else
            System.out.println("Not a Prime Number");
    }
}

```

**3. Create a function which accepts an integer as parameter and return true if it is a perfect number otherwise return false. In the main() method input two integers and check whether both the numbers are perfect numbers or not.**

**Ans.** import java.util.\*;

```

class Question3

```

```

{
    static boolean isPerfect(int n)
    {
        int i,s=0;
        for(i=1;i<n;i++)
        {
            if(n%i==0)
                s+=i;
        }
        if(s==n)
            return true;
        else
            return false;
    }

    static void main()
    {
        int n,m;
        boolean p,q;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter 2 integers:");
        n=sc.nextInt();
        m=sc.nextInt();
        p=isPerfect(n);
        q=isPerfect(m);
        if(p==true && q==true)
            System.out.println("Both are Perfect Numbers");
        else
            System.out.println("Both are not Perfect Numbers");
    }
}

```

**4. Create a function which accepts an integer as parameter and return the sum of the square of its digits. In the main( ) method display all three-digit Armstrong numbers.**

**Ans.** class Question4

```

{
    static int sum(int n)
    {
        int d,s=0;
        while(n!=0)
        {
            d=n%10;
            s+=d*d*d;
            n=n/10;
        }
        return s;
    }
}

```

```

static void main()
{
    int i,s;
    System.out.println("All 3 digit armstrong numbers:");
    for(i=100;i<=999;i++)
    {
        s=sum(i);
        if(i==s)
            System.out.println(i+" ");
    }
}
}

```

**5. Create a function which accepts an integer as parameter and return true if it is a palindrome number or not. In the main( ) method input 10 integers and print the largest palindrome number if any.**

**Ans.** import java.util.\*;

```

class Question5
{
    static boolean isPalindrome(int n)
    {
        int d,r=0,t=n;
        while(t!=0)
        {
            d=t%10;
            r=r*10+d;
            t=t/10;
        }
        if(r==n)
            return true;
        else
            return false;
    }

    static void main()
    {
        int i,n,f=0,l=0;
        boolean p;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {
            n=sc.nextInt();
            p=isPalindrome(n);
            if(p==true)
            {

```



```

        if(f==0)
        {
            l=n;
            f=1;
        }
        if(n>l)
            l=n;
    }
}
if(f==1)
    System.out.println("Largest Palindrome Number:"+l);
else
    System.out.println("No Palindrome Number present");
}
}

```

**6. Create a function which accepts an integer as parameter and return the sum of its digits. Create another function to input 10 integers and find the sum of the digits of each number.**

**Ans.** import java.util.\*;

```

class Question6
{
    static int sum(int n)
    {
        int d,s=0;
        while(n!=0)
        {
            d=n%10;
            s=s+d;
            n=n/10;
        }
        return s;
    }

    static void main()
    {
        int i,n,s;
        boolean p;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {
            n=sc.nextInt();
            s=sum(n);
            System.out.println("Sum of the digits in "+n+" is="+s);
        }
    }
}
}

```

**7. Create a function which accepts an integer as parameter and return the largest digit. Create another function to input 10 integers and find the sum of the largest digit of each number.**

**Ans.**

```
import java.util.*;
class Question7
{
    static int largest(int n)
    {
        int d,l;
        l=n%10;
        while(n!=0)
        {
            d=n%10;
            if(d>l)
                l=d;
            n=n/10;
        }
        return l;
    }

    static void main()
    {
        int i,n,s=0,l;
        boolean p;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {
            n=sc.nextInt();
            l=largest(n);
            s=s+l;
        }
        System.out.println("Sum of the largest digit is="+s);
    }
}
```

**8. Create a method which accepts temperature in Celsius and return its Fahrenheit equivalent. Create another method which accepts temperature in Fahrenheit and return its Celsius equivalent. Also create a method to invoke the above methods.**

**Ans.**

```
import java.util.*;
class Question8
{
    static float celciusToFarenheit(float c)
    {
        float f=(float)(9*c)/5+32;
        return f;
    }
}
```

```

    }

    static float fahrenheitToCelsius(float f)
    {
        float c=(float)(5*(f-32))/9;
        return c;
    }
    static void main()
    {
        float f,c;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter temperature in Farenheit:");
        f=sc.nextFloat();
        c=fahrenheitToCelsius(f);
        System.out.println("Temperature in Celcius:"+c);
        System.out.println("Enter temperature in Celcius:");
        c=sc.nextFloat();
        f=celsiusToFahrenheit(c);
        System.out.println("Temperature in Farenheit:"+f);
    }
}

```

**9. Create a class with the following methods:**

**i) int sum(int n), which finds the sum of the digits in n and returns it.**

**ii) void call() to input an integer using scanner and find the sum of its digits using the above method.**

**Ans.** import java.util.\*;

```

class Question9
{
    static int sum(int n)
    {
        int d,s=0;
        while(n!=0)
        {
            d=n%10;
            s=s+d;
            n=n/10;
        }
        return s;
    }

    static void call()
    {
        int n,s;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter an integer:");
        n=sc.nextInt();
    }
}

```

```

s=sum(n);
System.out.println("Sum of the digits in "+n+" is="+s);
    }
}

```

**10. Create a class with the following methods:**

**i) int sum(int n), which finds the sum of the digits in n and returns it.**

**ii) void call() to input 10 integers using scanner and find the sum of its digits of each integer value entered by the user using the above method.**

**Ans.** import java.util.\*;

```

class Question10
{
    static int sum(int n)
    {
        int d,s=0;
        while(n!=0)
        {
            d=n%10;
            s=s+d;
            n=n/10;
        }
        return s;
    }

    static void call()
    {
        int i,n,s;
        boolean p;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {
            n=sc.nextInt();
            s=sum(n);
            System.out.println("Sum of the digits in "+n+" is="+s);
        }
    }
}

```

**11. Create a class with the following methods:**

**i) boolean prime(int n), which returns true if n is prime otherwise returns false.**

**ii) void call() to input 10 integers using scanner and find the sum of those integers which are prime numbers using the above method.**

**Ans.** import java.util.\*;

```

class Question11
{
    static boolean isPrime(int n)

```

```

    {
        int i,c=0;
        for(i=1;i<=n;i++)
        {
            if(n%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    static void call()
    {
        int i,n,s=0;
        boolean p;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter 10 integers:");
        for(i=1;i<=10;i++)
        {
            n=sc.nextInt();
            p=isPrime(n);
            if(p==true)
                s=s+n;
        }
        System.out.println("Sum of the numbers which are prime="+s);
    }
}

```

**12. Create a class with the following methods:**

**i) boolean isArmstrong(int n), which returns true if n is Armstrong number otherwise returns false. Armstrong numbers are those numbers whose sum of the cube of its digits is equal to the number. E.g.  $153=1^3+5^3+3^3$**

**ii) void call( ) to input 10 integers using scanner and find largest Armstrong number if any.**

**Ans.** import java.util.\*;

```

class Question12
{
    static boolean isArmstrong(int n)
    {
        int d,t=n,s=0;
        while(t!=0)
        {
            d=t%10;
            s=s+d*d*d;
            t=t/10;
        }
    }
}

```

```

    }
    if(s==n)
        return true;
    else
        return false;
}

static void call()
{
    int i,n,l=0,f=0;
    boolean p;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter 10 integers:");
    for(i=1;i<=10;i++)
    {
        n=sc.nextInt();
        p=isArmstrong(n);
        if(p==true)
        {
            if(f==0)
            {
                l=n;
                f=1;
            }
            if(n>l)
                l=n;
        }
    }
    if(f==1)
        System.out.println("Largest Armstrong number="+l);
    else
        System.out.println("No Armstrong number present");
}
}

```

**13. Create a class with the following methods:**

**i) boolean prime(int n), which returns true if n is prime otherwise returns false.**

**ii) void call() to input 2 integers using scanner and print only the prime numbers between the given integers.**

**Ans.** import java.util.\*;  
class Question13  
{  
 static boolean isPrime(int n)  
 {  
 int i,c=0;

```

        for(i=1;i<=n;i++)
        {
            if(n%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    static void call()
    {
        int i,n,m,s=0,l=0;
        boolean p;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter 2 integers:");
        n=sc.nextInt();
        m=sc.nextInt();
        if(n<m)
        {
            s=n;
            l=m;
        }
        else
        {
            s=m;
            l=n;
        }
        for(i=s;i<=l;i++)
        {
            p=isPrime(i);
            if(p==true)
                System.out.println(i+" ");
        }
    }
}
}

```

**14. Create a class with the following methods:**

**i) boolean prime(int n), which returns true if n is prime otherwise returns false.**

**ii) int sum(int n), which finds the sum of the digits in n and returns it.**

**iii) void call( ) to display all such three-digit numbers whose sum of the digits is a prime number.**

**Ans.** class Question14

```

{
    static boolean prime(int n)

```

```

    {
        int i,c=0;
        for(i=1;i<=n;i++)
        {
            if(n%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    static int sum(int n)
    {
        int d,s=0;
        while(n!=0)
        {
            d=n%10;
            s+=d;
            n=n/10;
        }
        return s;
    }

    static void call()
    {
        int i,s=0;
        boolean p;
        for(i=100;i<=999;i++)
        {
            s=sum(i);
            p=prime(s);
            if(p==true)
                System.out.println(i+" ");
        }
    }
}

```

**15. Create a class with the following methods:**

- i) boolean prime(int n), which returns true if n is prime otherwise returns false.**
- ii) boolean palindrome(int n), which returns true if n is palindrome otherwise returns false.**
- iii) void call() to display all three digit pal-prime numbers. Pal-prime numbers are those numbers which are both palindrome as well as prime.**

**Ans.** class Question15

```
{
```



```

static boolean prime(int n)
{
    int i,c=0;
    for(i=1;i<=n;i++)
    {
        if(n%i==0)
            c++;
    }
    if(c==2)
        return true;
    else
        return false;
}

static boolean palindrome(int n)
{
    int d,s=0,t=n;
    while(n!=0)
    {
        d=n%10;
        s=s*10+d;
        n=n/10;
    }
    if(s==t)
        return true;
    else
        return false;
}

static void call()
{
    int i;
    boolean p,q;
    for(i=100;i<=999;i++)
    {
        p=prime(i);
        q=palindrome(i);
        if(p==true && q==true)
            System.out.println(i+" ");
    }
}
}

```

**16. Create a class with the following methods:**

**i) boolean prime(int n), which returns true if n is prime otherwise returns false.**

ii) void call() to display all two-digit twin-prime numbers. Twin-prime numbers are those pair of numbers which are both prime as well as whose difference is 2. For example, 11 and 13, 17 and 19, etc.

**Ans.** class Question16

```
{
    static boolean prime(int n)
    {
        int i,c=0;
        for(i=1;i<=n;i++)
        {
            if(n%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    static void call()
    {
        int i;
        boolean p,q;
        for(i=10;i<=97;i++)
        {
            p=prime(i);
            q=prime(i+2);
            if(p==true && q==true)
                System.out.println(i+" "+(i+2));
        }
    }
}
```

**17. Create a class with the following methods:**

i) boolean prime(int n), which returns true if n is prime otherwise returns false.

ii) int sumPrime(int n), which returns the sum of prime digits present in integer n.

iii) void call() to input an integer and check whether the sum of prime digits is also a prime number or not.

**Ans.** import java.util.\*;

```
class Question17
{
    static boolean prime(int n)
    {
        int i,c=0;
        for(i=1;i<=n;i++)
```

```

        {
            if(n%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    static int sumPrime(int n)
    {
        int d,s=0;
        while(n!=0)
        {
            d=n%10;
            if(prime(d)==true)
                s=s+d;
            n=n/10;
        }
        return s;
    }

    static void call()
    {
        int n,s;
        boolean p;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        s=sumPrime(n);
        p=prime(s);
        if(p==true)
            System.out.println("Sum of prime digits is also a prime number");
        else
            System.out.println("Sum of prime digits is not a prime number");
    }
}

```

**18. Create a class with the following methods:**

**i) int largest(int a, int b), which returns largest among the two integers a and b and return it.**

**iii) void call() to input 10 integers and using the above function find the largest among the 10 integers.**

**Ans.** import java.util.\*;  
class Question18  
{

```

static int largest(int a,int b)
{
    if(a>b)
        return a;
    else
        return b;
}

static void call()
{
    int i,n,l=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter 10 integers:");
    for(i=1;i<=10;i++)
    {
        n=sc.nextInt();
        if(i==1)
            l=n;
        else
            l=largest(l,n);
    }
    System.out.println("Largest Number="+l);
}
}

```

**19. Create a class with the following methods:**

**i) int LCM(int a, int b), which returns the least common multiple (LCM) among the two integers a and b and return it.**

**ii) void call( ) to input 10 integers and using the above function find the LCM among the 10 integers.**

**Ans.** import java.util.\*;  
class Question19  
{  
 static int LCM(int a,int b)  
 {  
 int i;  
 for(i=a;i<=a\*b;i++)  
 {  
 if(i%a==0 && i%b==0)  
 break;  
 }  
 return i;  
 }  
}

static void call()  
{  
 int i,n,l=0;

```

Scanner sc=new Scanner(System.in);
System.out.println("Enter 10 integers:");
for(i=1;i<=10;i++)
{
    n=sc.nextInt();
    if(i==1)
        l=n;
    else
        l=LCM(l,n);
}
System.out.println("LCM of the Numbers="+l);
}
}

```

**20. Create a class with the following methods:**

**i) int HCF(int a, int b), which returns the highest common factor (HCF) among the two integers a and b and return it.**

**ii) void call() to input 10 integers and using the above function find the HCF among the 10 integers.**

**Ans.** import java.util.\*;

```

class Question20

```

```

{
    static int HCF(int a,int b)
    {
        int i;
        for(i=a;i>=1;i--)
        {
            if(a%i==0 && b%i==0)
                break;
        }
        return i;
    }
}

```

```

static void call()

```

```

{
    int i,n,l=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter 10 integers:");
    for(i=1;i<=10;i++)
    {
        n=sc.nextInt();
        if(i==1)
            l=n;
        else
            l=HCF(l,n);
    }
    System.out.println("HCF of the Numbers="+l);
}

```

```
    }  
}
```

**21. Create a class called GeneratePrime which will be used to generate n number of prime numbers.**

**The class should have the following methods:**

**i) Method called isPrime() which accepts an integer as a parameter and return true if it is a prime number otherwise return false.**

**ii) Method called display() which accepts an integer n as Scanner input and display the first n prime number by calling the above function.**

**Ans.** import java.util.\*;

```
//QUESTION 21
```

```
class GeneratePrime
```

```
{  
    static boolean isPrime(int n)  
    {  
        int i,c=0;  
        for(i=1;i<=n;i++)  
        {  
            if(n%i==0)  
                c++;  
        }  
        if(c==2)  
            return true;  
        else  
            return false;  
    }  
}
```

```
static void display()
```

```
{  
    int i=2,n,c=0;  
    Scanner sc=new Scanner(System.in);  
    System.out.println("Enter an integer:");  
    n=sc.nextInt();  
    System.out.println("First "+n+" prime numbers are:");  
    do  
    {  
        if(isPrime(i)==true)  
        {  
            System.out.println(i+" ");  
            c++;  
        }  
        i++;  
    }while(c<n);  
}
```

**22. Create a class with the following Member Functions:**

**i) Method called isPrime() which accepts an integer as a parameter and return true if it is a prime number otherwise return false.**

**ii) Method called sumOfPrimeDigits() which accepts an integer as a parameter and return the sum of prime digits only.**

**iii) Method called check() which accepts an integer as parameter and checks whether the sum of the prime digits is also a prime number or not.**

**Ans.** class Question22

```
{
    static boolean isPrime(int n)
    {
        int i,c=0;
        for(i=1;i<=n;i++)
        {
            if(n%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    static int sumOfPrimeDigits(int n)
    {
        int d,s=0;
        while(n!=0)
        {
            d=n%10;
            if(isPrime(d)==true)
                s=s+d;
            n=n/10;
        }
        return s;
    }

    static void check(int n)
    {
        int s;
        boolean p;

        s=sumOfPrimeDigits(n);
        p=isPrime(s);
        if(p==true)
            System.out.println("Sum of prime digits is also a prime number");
        else
            System.out.println("Sum of prime digits is not a prime number");
    }
}
```

```
}
```

**23. Create a class called Series which will contain the following members function:**

**i) long fact(int f) to find the factorial of f and return it.**

**ii) void sumSeries1(int x,int n) to calculate and print the sum of the following series:**

$x + x/2! + x/3! + x/4! + \dots + x/n!$

**iii) void sumSeries2(int x,int n) to calculate and print the sum of the following series:**

$x/2! + x^2/3! + x^3/4! + x^4/5! + \dots + x^n/(n+1)!$

**iv) void sumSeries3(int x,int n) to calculate and print the sum of the following series:**

$x/2! - x^2/3! + x^3/4! - x^4/5! + \dots \pm x^n/(n+1)!$

**Ans. //QUESTION 23**

```
class Series
{
    static long fact(int f)
    {
        int i;
        long p=1;
        for(i=1;i<=f;i++)
            p=p*i;
        return p;
    }

    static void sumSeries1(int x, int n)
    {
        int i;
        float s=0;
        for(i=1;i<=n;i++)
        {
            s+=(float)x/fact(i);
        }
        System.out.println("Sum="+s);
    }

    static void sumSeries2(int x, int n)
    {
        int i;
        float s=0;
        for(i=1;i<=n;i++)
        {
            s+=(float)Math.pow(x,i)/fact(i+1);
        }
        System.out.println("Sum="+s);
    }

    static void sumSeries3(int x, int n)
    {
        int i;
```



```

float s=0;
for(i=1;i<=n;i++)
{
    if(i%2==0)
        s-=(float)Math.pow(x,i)/fact(i+1);
    else
        s+=(float)Math.pow(x,i)/fact(i+1);
}
System.out.println("Sum="+s);
}
}

```

**24.** Where the symbol ! stands for factorial, e.g.  $5!=5*4*3*2*1$ ,  $3!=3*2*1$

• Using overloading technique, write methods to:

- accept two int type data as parameters and return their sum.
- accept three int type data as parameters and return their sum.
- accept two double type data as parameters and return their sum.
- accept a double type and int type as parameters and return their sum.

**Ans.** class Question24

```

{
    static int sum(int a,int b)
    {
        int c;
        c=a+b;
        return c;
    }

    static int sum(int a,int b,int c)
    {
        int d;
        d=a+b+c;
        return d;
    }

    static double sum(double a,double b)
    {
        double c;
        c=a+b;
        return c;
    }

    static double sum(double a,int b)
    {
        double c;
        c=a+b;
        return c;
    }
}

```

```
}
```

**25. Create a function which accepts an integer as parameter and return true if it is a prime number otherwise return false. Create another function which accepts an integer as parameter and return true if it is palindrome otherwise return false. In the main() method display all three-digit pal-prime number. Pal-prime numbers are such numbers which are both palindrome as well as prime numbers.**

**For example, 101, 131, 151, 181, 191, 313, 353, 373, 383, 727, 757, 787, 797, 919 and 929 are all three digit pal-prime numbers.**

**Ans.** class Question25

```
{
    static boolean prime(int n)
    {
        int i,c=0;
        for(i=1;i<=n;i++)
        {
            if(n%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    static boolean palindrome(int n)
    {
        int d,s=0,t=n;
        while(n!=0)
        {
            d=n%10;
            s=s*10+d;
            n=n/10;
        }
        if(s==t)
            return true;
        else
            return false;
    }

    static void main()
    {
        int i;
        boolean p,q;
        for(i=100;i<=999;i++)
        {
            p=prime(i);
```

```

        q=palindrome(i);
        if(p==true && q==true)
            System.out.println(i+" ");
    }

}
}

```

**26. Create a function which accepts an integer as parameter and return the sum of its digits. Create another function which accepts an integer as parameter and return true if it is a magic number otherwise return false. In the main input an integer and check whether it is a magic number or not.**

**If you iterate the process of summing the squares of the decimal digits of a number and if this process terminates in 1, then the original number is called a magic number. For example,  $55 \Rightarrow (5+5)=10 \Rightarrow (1+0)=1$ .**

**Ans.**

```

import java.util.*;
class Question26
{
    static int sum(int n)
    {
        int d,s=0;
        while(n!=0)
        {
            d=n%10;
            s=s+d;
            n=n/10;
        }
        return s;
    }

    static boolean isMagic(int n)
    {
        do
        {
            n=sum(n);
        }while(n>9);
        if(n==1)
            return true;
        else
            return false;
    }

    static void main()
    {
        int n;
        boolean p;
        Scanner sc=new Scanner(System.in);

```

```

System.out.println("Enter a number:");
n=sc.nextInt();
p=isMagic(n);
if(p==true)
    System.out.println("It is a Magic Number");
else
    System.out.println("It is not a Magic Number");
}
}

```

**27. Create a function which accepts an integer as parameter and return the sum of the square of its digits. Create another function which accepts an integer as parameter and return true if it is happy number otherwise return false. In the main input an integer and check whether it is a happy number or not. For example,  $7 \Rightarrow (7^2) = 49 \Rightarrow (4^2 + 9^2) = 97 \Rightarrow (9^2 + 7^2) = 130 \Rightarrow (1^2 + 3^2 + 0^2) = 10 \Rightarrow (1^2 + 0^2) = 1$ .**

**Ans.** import java.util.\*;

```

class Question27
{
    static int sum(int n)
    {
        int d,s=0;
        while(n!=0)
        {
            d=n%10;
            s=s+d*d;
            n=n/10;
        }
        return s;
    }

    static boolean isHappy(int n)
    {
        do
        {
            n=sum(n);
        }while(n>9);
        if(n==1)
            return true;
        else
            return false;
    }

    static void main()
    {
        int n;
        boolean p;
        Scanner sc=new Scanner(System.in);

```

```

        System.out.println("Enter a number:");
        n=sc.nextInt();
        p=isHappy(n);
        if(p==true)
            System.out.println("It is a Happy Number");
        else
            System.out.println("It is not a Happy Number");
    }
}

```

**28. Create a class with the following functions:**

**i) Boolean isBinary( )** which accepts a binary number as parameter and return true if it is a valid binary number or not. A binary number consists of only two digits 0 and 1.

**ii) int binToDecimal(int b)** which accepts a binary number as parameter and return its decimal equivalent.

**iii) int deciToBinary(int d)** which accepts a decimal number as parameter and return its binary equivalent.

**iv) void sum( )** where you input two binary numbers and if they are valid binary numbers find their sum.

**Sample input and output when the sum( ) function is executed:**

**INPUT**

Enter two valid binary numbers: 1011 11011

**OUTPUT**

Sum of the two given binary numbers is: 100110

**INPUT**

Enter two valid binary numbers: 12300 11011

**OUTPUT**

The binary numbers may not be valid!

**Ans.** import java.util.\*;

class Question28

```

{
    static boolean isBinary(int b)
    {
        int d,f=0;
        while(b!=0)
        {
            d=b%10;
            if(d!=1 && d!=0)
                f=1;
            b=b/10;
        }
        if(f==0)
            return true;
        else
            return false;
    }
}

```

static int binToDecimal(int b)

```

    {
        int s=0,d,c=0;
        while(b!=0)
        {
            d=b%10;
            s=s+d*(int)Math.pow(2,c++);
            b=b/10;
        }
        return s;
    }

static int deciToBinary(int d)
{
    int s=0,b,c=0;
    while(d!=0)
    {
        b=d%2;
        s=s+b*(int)Math.pow(10,c++);
        d=d/2;
    }
    return s;
}

static void sum()
{
    Scanner sc=new Scanner(System.in);
    int b1,b2,d1,d2,d,b;
    System.out.println("Enter two valid binary numbers:");
    b1=sc.nextInt();
    b2=sc.nextInt();
    if(isBinary(b1)==true && isBinary(b2)==true)
    {
        d1=binToDecimal(b1);
        d2=binToDecimal(b2);
        d=d1+d2;
        b=deciToBinary(d);
        System.out.println("Sum of the binary numbers:"+b);
    }
    else
        System.out.println("The binary numbers may not be valid!");
}
}

```

**29. Write a class with the name volume using function overloading that computes the volume of a cube, a sphere and a cuboid. [ICSE 2008]**

**Formula:**

**volume of a cube (vc) = s\*s\*s**

**volume of a sphere (vs) =  $\frac{4}{3} * \pi * r * r * r$  (where pi = 3.14 or 22/7)**

**Volume of a cuboid (vcd) =  $l * b * h$**

**Ans.** //Question 29

```
class Volume
{
    void function(int s)
    {
        int v;
        v=s*s*s;
        System.out.println("Volume="+v);
    }

    void function(float r)
    {
        float v;
        v=4/3f*3.14f*r*r*r;
        System.out.println("Volume="+v);
    }

    void function(int l,int b, int h)
    {
        int v;
        v=l*b*h;
        System.out.println("Volume="+v);
    }
}
```

**30. Design a class to overload a function num\_calc() as follows: [ICSE 2009]**

**i) void num\_calc(int num,char ch) with one integer argument and one character argument, computes the square of integer argument if choice ch is 's' otherwise find its cube.**

**ii) void num\_calc(int a, int b,char ch) with two integer arguments if ch is 'p' else adds the integers.**

**iii) void num\_calc(String s1,String s2) with two String arguments, which prints whether the strings are equal or not.**

**Ans.** class Question30

```
{
    void num_calc(int num, char ch)
    {
        if(ch=='s' || ch=='S')
            System.out.println("Square="+num*num);
        else
            System.out.println("Cube="+num*num*num);
    }

    void num_calc(int a,int b, char ch)
    {
        if(ch=='p' || ch=='P')
```

```

        System.out.println("Product="+(a*b));
    else
        System.out.println("Sum="+(a+b));
    }

    void num_calc(String a,String b)
    {
        if(a.equals(b))
            System.out.println("Same");
        else
            System.out.println("Not Same");
    }
}

```

**31. Design a class to overload a function compare( ) as follows: [ICSE 2011]**

- i) void compare(int,int) – to compare two integer values and print the greater of the two integers.**
- ii) void compare(char,char) – to compare the numeric values of two characters and print the character with higher numeric value.**
- iii) void compare (String, String) – to compare the length of the two strings and print the longer of the two.**

**Ans.** class Question31

```

{
    static void compare(int a, int b)
    {
        if(a>b)
            System.out.println("Largest:"+a);
        else
            System.out.println("Largest:"+b);
    }

    static void compare(char a, char b)
    {
        if((int)a>(int)b)
            System.out.println("Largest:"+a);
        else
            System.out.println("Largest:"+b);
    }

    static void compare(String a, String b)
    {
        if(a.length()>b.length())
            System.out.println("Largest:"+a);
        else
            System.out.println("Largest:"+b);
    }
}

```



**32. Design a class to overload a function polygon( ) as follows: [ICSE 2012]**

**i) void polygon(int n, char ch) : with one integer argument and one character type argument that draws a filled square of side n using the character stored in ch.**

**ii) void polygon(int x, int y) : with two integer arguments that draws a filled rectangle of length x and breadth y, using the symbol '@'**

**iii) void polygon( ) : with no argument that draws a filled triangle shown below:**

```
*  
**  
***
```

**Ans.** class Question32

```
{  
    static void polygon(int n,char ch)  
    {  
        int i,j;  
        for(i=1;i<=n;i++)  
        {  
            for(j=1;j<=n;j++)  
            {  
                System.out.print(ch);  
            }  
            System.out.println();  
        }  
    }  
  
    static void polygon(int x,int y)  
    {  
        int i,j;  
        for(i=1;i<=y;i++)  
        {  
            for(j=1;j<=x;j++)  
            {  
                System.out.print("@");  
            }  
            System.out.println();  
        }  
    }  
  
    static void polygon()  
    {  
        int i,j;  
        for(i=1;i<=3;i++)  
        {  
            for(j=1;j<=i;j++)  
            {  
                System.out.print("*");  
            }  
        }  
    }  
}
```

```

        System.out.println();
    }
}
}

```

**33. Design a class to overload a function compute( ) as follows:**

**i) void compute(int,char): to compute the square of the integer argument if the given character argument is 's' otherwise find its cube.**

**ii) void compute(double char): to compute volume of a cube if the given character argument is 'v' otherwise find its diagonal.**

**iii) void compute(int,int,char): to compute area of a rectangle if the given character argument is 'a' otherwise finds its perimeter.**

**Volume of cube=side<sup>3</sup>**

**Area of rectangle=length\*breadth**

**Diagonal of cube= $a\sqrt{3}$**

**Perimeter of rectangle=2\*(length+breadth)**

**Ans.** class Question33

```

{
    static void compute(int a,char c)
    {
        if(c=='s' || c=='S')
            System.out.println("Square="+a*a);
        else
            System.out.println("Cube="+a*a*a);
    }

    static void compute(double a,char c)
    {
        if(c=='v' || c=='V')
            System.out.println("Volume="+a*a*a);
        else
            System.out.println("Diagonal="+a*Math.sqrt(a));
    }

    static void compute(int l,int b,char c)
    {
        if(c=='a' || c=='A')
            System.out.println("Volume="+l*b);
        else
            System.out.println("Perimeter="+2*(l+b));
    }
}
}

```

**34. Design a class to overload a function series( ) as follows:**

**[ICSE 2013]**

**i) double series(double n) with one double argument and returns the sum of the series.**

**sum = 1/1 + 1/2 + 1/3 + ..... 1/n**

**ii) double series(double a, double n) with two double arguments and returns the sum of the series.**

**sum =  $1/a^2 + 4/a^5 + 7/a^8 + 10/a^{11} \dots$  to n terms**

**Ans.** class Question34

```

{
    double series(double n)
    {
        double s=0;
        int i;
        for(i=1;i<=n;i++)
        {
            s=s+1.0/i;
        }
        return s;
    }

    double series(double a,double n)
    {
        double s=0,p=1;
        int i;
        for(i=1;i<=n;i++)
        {
            s=s+p/Math.pow(a,p+1);
            p=p+3;
        }
        return s;
    }
}

```

**35. Design a class to overload a function series( ) as follows:**

**i) double series(double n) with one double argument and returns the sum of the series:**

$s=1+1/2! +1/3! +1/4! +\dots+1/n!$

**ii) double series(double x,double n) with two double arguments and returns the sum of the series,**

$s=x+x^2/2! +x^3/3! +x^4/4! +\dots+x^n/n!$

**Ans.** class Question35

```

{
    double series(double n)
    {
        double i,j,s=0,p;
        for(i=1;i<=n;i++)
        {
            p=1;
            for(j=1;j<=i;j++)
                p=p*j;
            s=s+1/p;
        }
        return s;
    }
}

```

```

double series(double x,double n)
{
    double s=0,j,i,p;
    for(i=1;i<=n;i++)
    {
        p=1;
        for(j=1;j<=i;j++)
            p=p*j;
        s=s+Math.pow(x,i)/p;
    }
    return s;
}
}

```

**36. Write a menu-driven program to accept a number from the user and check whether is a 'BUZZ' number or to accept any two numbers and print 'GCD' of them. [ICSE 2009]**

**i) A BUZZ number is the number which either ends with 7 or divisible by 7.**

**ii) GCD (Greatest Common Divisor) of two integers is calculated by continued division method. Divide the larger number by the smaller, the remainder then divides the previous divisor. The process is repeated till the remainder is zero. The divisor then results the GCD.**

**Ans.** import java.util.\*;

```

class Question36
{
    static void isBuzz(int n)
    {
        if(n%7==0 || n%10==7)
            System.out.println("Buzz Number");
        else
            System.out.println("Not a Buzz Number");
    }

    static void GCD(int n,int d)
    {
        int r;
        do
        {
            r=n%d;
            n=d;
            d=r;
        }while(r!=0);
        System.out.println("GCD="+n);
    }

    static void menu()
    {
        Scanner sc=new Scanner(System.in);
        int n,ch,a,b;
    }
}

```

```

System.out.println("Enter 1 to check for BUZZ number");
System.out.println("Enter 2 to find the GCD of 2 numbers");
System.out.println("Enter your choice:");
ch=sc.nextInt();
switch(ch)
{
    case 1:
        System.out.println("Enter a number:");
        n=sc.nextInt();
        isBuzz(n);
        break;
    case 2:
        System.out.println("Enter 2 numbers:");
        a=sc.nextInt();
        b=sc.nextInt();
        GCD(a,b);
        break;
    default:
        System.out.println("Invalid Entry!");
}
}
}

```

**37. Write a menu-driven program to accept a number and check and display whether it is a Prime Number or not OR an Automorphic Number or not (Use switch-case statement).**

[ICSE 2010]

**i) Prime number:** A number is said to be a prime number if it is divisible only by 1 and itself and not by any other number. Example: 3, 5, 7, 11, 13, etc.

**ii) Automorphic number:** An automorphic number is the number which is contained in the last digit(s) of its square. Example: 25 is an automorphic number as its square is 625 and 25 is present as the last two digits.

**Ans.** import java.util.\*;  
class Question37  
{  
 static void isPrime(int n)  
 {  
 int i,c=0;  
 for(i=1;i<=n;i++)  
 {  
 if(n%i==0)  
 c++;  
 }  
 if(c==2)  
 System.out.println("Prime Number");  
 else  
 System.out.println("Not a Prime Number");  
 }  
}

```

static void isAutomorphic(int n)
{
    int s,c=0,t=n;
    s=n*n;
    while(n!=0)
    {
        c++;
        n/=10;
    }
    if(s%(int)Math.pow(10,c)==t)
        System.out.println("Automorphic Number");
    else
        System.out.println("Not an Automorphic Number");
}

static void menu()
{
    Scanner sc=new Scanner(System.in);
    int n,ch,a,b;
    System.out.println("Enter a number:");
    n=sc.nextInt();
    System.out.println("Enter 1 to check for Prime number");
    System.out.println("Enter 2 to check for Automorphic number");
    System.out.println("Enter your choice:");
    ch=sc.nextInt();
    switch(ch)
    {
        case 1:
            isPrime(n);
            break;
        case 2:
            isAutomorphic(n);
            break;
        default:
            System.out.println("Invalid Entry!");
    }
}
}
}

```

**38. Write a menu-driven program to perform the following: (Use switch-case statement)**

**i) To print the series 0, 3, 8, 15, 24... n terms(value of 'n' is to be an input by the user).**

**ii) To find the sum of the series given below:**

**S=  $\frac{1}{2} + \frac{3}{4} + \frac{5}{6} + \frac{7}{8} \dots \frac{19}{20}$ .**

**[ICSE 2011]**

**Ans.** import java.util.\*;  
class Question38  
{

```

static void series1(int n)
{
    int i,c=3,s=0;
    for(i=1;i<=n;i++)
    {
        System.out.print(s+" ");
        s=s+c;
        c=c+2;
    }
}

static void series2()
{
    int i;
    float s=0;
    for(i=1;i<=19;i++)
    {
        s=s+(float)i/(i+1);
    }
    System.out.println("Sum="+s);
}

static void menu()
{
    Scanner sc=new Scanner(System.in);
    int n,ch;
    System.out.println("Enter 1 to display the first series");
    System.out.println("Enter 2 to display the second series");
    System.out.println("Enter your choice:");
    ch=sc.nextInt();
    switch(ch)
    {
        case 1:
            System.out.println("Enter a number:");
            n=sc.nextInt();
            series1(n);
            break;
        case 2:
            series2();
            break;
        default:
            System.out.println("Invalid Entry!");
    }
}
}

```

**39. Using the switch statement, write a menu-driven program to:**

**[ICSE 2012]**

i) Generate and display the first 10 terms of the Fibonacci series 0, 1, 1, 2, 3, 5, ....

ii) find the sum of the digits of an integer that is input. E.g. 15390=18

For an incorrect choice, appropriate error message should be displayed.

**Ans.** import java.util.\*;

```
class Question39
```

```
{
```

```
    static void fibonacci()
```

```
    {
```

```
        int i,a=0,b=1,c;
```

```
        System.out.print("Fibonacci Series:"+a+" "+b+" ");
```

```
        for(i=1;i<=8;i++)
```

```
        {
```

```
            c=a+b;
```

```
            System.out.print(c+" ");
```

```
            a=b;
```

```
            b=c;;
```

```
        }
```

```
    }
```

```
    static void sum(int n)
```

```
    {
```

```
        int d,s=0;
```

```
        while(n!=0)
```

```
        {
```

```
            d=n%10;
```

```
            s+=d;
```

```
            n=n/10;
```

```
        }
```

```
        System.out.println("Sum of the digits="+s);
```

```
    }
```

```
    static void menu()
```

```
    {
```

```
        Scanner sc=new Scanner(System.in);
```

```
        int n,ch;
```

```
        System.out.println("Enter 1 to display the first 10 no.s of the fibonacci series");
```

```
        System.out.println("Enter 2 to display the sum of the digits of a given integer");
```

```
        System.out.println("Enter your choice:");
```

```
        ch=sc.nextInt();
```

```
        switch(ch)
```

```
        {
```

```
            case 1:
```

```
                fibonacci();
```

```
                break;
```

```
            case 2:
```



```

        System.out.println("Enter a number:");
        n=sc.nextInt();
        sum(n);
        break;

    default:
        System.out.println("Invalid Entry!");
    }
}
}

```

**40. Using the switch statement, write a menu-driven program: [ICSE 2013]**

**i) To check and display whether a number input by the user is a composite number or not (A number is said to be a composite, if it has one or more than one factors excluding 1 and the number itself).**

**Example: 4, 6, 8, 9...**

**ii) To find the smallest digit of an integer that is input:**

**Sample input: 6524**

**Sample output: Smallest digit is 2**

**For an incorrect choice, an appropriate error message should be displayed.**

**Ans.**

```

import java.util.*;
class Question40
{
    static void isComposite(int n)
    {
        int i,c=0;
        for(i=2;i<n;i++)
        {
            if(n%i==0)
                c++;
        }
        if(c>=1)
            System.out.println("Composite Number");
        else
            System.out.println("Not a Composite Number");
    }

    static void smallest(int n)
    {
        int d,s=n%10;
        while(n!=0)
        {
            d=n%10;
            if(d<s)
                s=d;
            n=n/10;
        }
    }
}

```

```

        System.out.println("Smallest digit is="+s);
    }

    static void menu()
    {
        Scanner sc=new Scanner(System.in);
        int n,ch;
        System.out.println("Enter 1 to check whether composite or not");
        System.out.println("Enter 2 to display the smallest digit in a given integer");
        System.out.println("Enter your choice:");
        ch=sc.nextInt();
        switch(ch)
        {
            case 1:
                System.out.println("Enter a number:");
                n=sc.nextInt();
                isComposite(n);
                break;
            case 2:
                System.out.println("Enter a number:");
                n=sc.nextInt();
                smallest(n);
                break;

            default:
                System.out.println("Invalid Entry!");
        }
    }
}

```

**41. Create a function which accepts an integer as parameter and return true if it is a Fibonacci number otherwise return false. A number is said to be Fibonacci if it comes under the Fibonacci series (1, 1, 2, 3, 5, 8, 13, 21, 34 ...). In the main() input a number (say n) and check whether it is a Fibonacci number or not.**

**Sample Input/Output**

**INPUT**

Enter an integer:13

**OUTPUT**

It is a valid Fibonacci number

**INPUT**

Enter an integer: 12

**OUTPUT**

Not a valid Fibonacci number

**Ans.** import java.util.\*;

class Question41

{

static boolean isFibonacci(int n)

```

    {
        int a=1,b=0,c=0;
        while(c<n)
        {
            c=a+b;
            a=b;
            b=c;
        }
        if(c==n)
            return true;
        else
            return false;
    }

    static void main()
    {
        int n;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter a number:");
        n=sc.nextInt();
        if(isFibonacci(n)==true)
            System.out.println("It is a valid Fibonacci number");
        else
            System.out.println("Not a valid Fibonacci number");
    }
}

```

**42. Create a function which accepts an integer as parameter and return true if it is a Fibonacci number otherwise return false. In the main input 10 integers and print the largest Fibonacci number if any.**

**Note that the number which will be entered by the user may not be ordered.**

**Sample Input/Output**

**INPUT**

**Enter 10 integers:  
5 16 21 9 15 4 8 16 11 10**

**OUTPUT**

**Largest Fibonacci number:21**

**INPUT**

**Enter 10 integers:  
25 10 7 6 4 12 18 10 9 14**

**OUTPUT**

**No Fibonacci number present**

**Ans.**

```
import java.util.*;
class Question42
{
    static boolean isFibonacci(int n)
    {
        int a=1,b=0,c=0;
```

```

while(c<n)
{
    c=a+b;
    a=b;
    b=c;
}
if(c==n)
    return true;
else
    return false;
}

static void main()
{
    int n,i,l=0,f=0;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter 10 integers:");
    for(i=1;i<=10;i++)
    {
        n=sc.nextInt();
        if(isFibonacci(n)==true)
        {
            if(f==0)
            {
                l=n;
                f=1;
            }
            if(n>l)
                l=n;
        }
    }
    if(f==1)
        System.out.println("Largest Fibonacci number:"+l);
    else
        System.out.println("No Fibonacci number present");
}
}

```

# Chapter 10: The Class

## Practice Questions

### A. Answer the following questions:

**1. What is a class?**

**Ans.** The class defines the characteristics and behaviour that may implemented by the objects, which are instantiated. It is actually the method of implementing encapsulation in Java.

**2. Mention any two attributes required for class declaration. [ICSE 2008]**

**Ans.** Method name and class name

**3. Why is a class called an object factory? [ICSE 2009]**

**Ans.** Factory in general refers to the process of generating some kind of products meant for it. A class is a blue-print of an object and therefore produces objects when instantiated. The characteristics and behaviour defined by the class is reflected in each and every object that it instantiates. That is why a class is called an object factory.

**4. Define Instance Variable. Give an example of the same. [ICSE 2007]**

**Ans.** Instance variables refers to the data members declared within a class, that represents the characteristics of an object. Each instance variable declared within a class becomes a part of an object that it instantiates. For example,

```
class Myclass
{
    int a,b; //Instance Variable
    static void main( )
    {
        Myclass ob1=new Myclass();
        Myclass ob2=new Myclass();
        ob1.a=5;ob1.b=6;
        ob2.a=7;ob2.b=8;
    }
}
```

In the main( ) function, there are two objects that are instantiated ob1 and ob2. Each object i.e. ob1 will have a set of instance variable a and b (which are assigned 5 and 6 respectively) similarly ob2 will have another set of instance variable a and b (which are assigned 7 and 8 respectively).

**5. What does the following mean? [ICSE 2008]**

**Employee staff = new Employee( );**

**Ans.** An object named **staff** is being allocated memory using the **new** command of the **Employee** class.

**6. What is the use of the keyword this? [ICSE 2009]**

**Ans.** This keyword is used within a member function that acts as a reference to the current object.

**7. What is a composite data type? In this respect also explain why a class is called composite data type?**

**Ans.** Composite data type refers to a data type that is composed of multiple predefined discrete data types. The class acts as a composite data type as it may contain data members that are composed of varied data types kept as a single unit.

**8. How do you declare objects? Show with the help of an example.**

**Ans.** An object is declared using either of the general syntax:

1. <Class-name> <object-name> = new <Class-name>( <Parameter-List>);
2. <Class-name> <object-name>;  
<object-name> = new <Class-name>( <Parameter-List>);

For example, for a class named **Coordinate** to declare an object of **Coordinate** type you may use either of the syntax:

1. Coordinate obj=new Coordinate( ); or
2. Coordinate obj; //declare an object  
obj=new Coordinate( ); // Allocate memory space

**9. How are objects used to access data-members?**

**Ans.** The dot operator is used to access the individual members of a class using an object. Data members can be accessed within the functions of a class using the dot operator as shown in the following syntax,  
<objectname>.<data-member>

**10. Differentiate the following statements :**

i) student Amit = new student( );

ii) student Sumit = new student(84,98,91);

**Ans.** i) An object named Amit is being created of **student** type using default constructor or a non-parameterised constructor.

ii) An object named Sumit is being created of **student** type using a parameterised constructor that accepts three integers as parameters.

**11. Explain, with an example, how to return objects.**

**Ans.** The following program demonstrates how an object may be returned:

```
public class ReturnObject
{
    int a,b;
    void getval(int x,int y)
    {
        a=x;
        b=y;
    }
    ReturnObject increment( )
    {
        ReturnObject temp=new ReturnObject( ); /*a new
                                                object being created*/
        temp.getval(a,b); /*initialize the newly created
                           with values of the current
                           object.*/
        /*The statements below increases the attributes of the newly created
        object.*/
    }
}
```

```

        temp.a++;
        temp.b++;
        return temp;
    }
    void show()
    {
        System.out.println(a+"\t"+b);
    }
    public static void main(String args[ ])
    {
        ReturnObject obj1=new ReturnObject();
        obj1.getval(5,6);
        ReturnObject obj2;    /*Declaration of the object
                               obj2*/

        obj2=obj1.increment( );
        obj1.show( );
        obj2.show( );
    }
}

```

**12. Can an object be an attribute of another class, if so how?**

**Ans.** The following example shows how an object may be an attribute of another class:

```

class Date
{
    int dd,mm,yy;
}

public class Student
{
    int roll;
    String name;
    Date dob=new Date(); //variable dob of Date type
    static void test( )
    {
        Student obj=new Student( );
        obj.roll=1;
        obj.name="Amit";
        obj.dob.dd=12; /*To hold the day within the dd part of object dob of the
        object obj.*/
        obj.dob.mm=9; /*To hold the day within the mm part of object dob of
        the object obj.*/
        obj.dob.yy=90; /*To hold the day within the yy part of object dob of the
        object obj.*/
    }
}

```

**13. What do you understand by Garbage Collection?**

**Ans.** Garbage collection refers to automatic deallocation of the memory which was allocated using the new command when no references to an object exist.

**14. What are class variables and class methods?**

**Ans.** The class variables and class methods are static methods within a class, that are not instantiated with every instance of an object but rather only one copy is created.

**15. What are the rules you should keep in mind while accessing static members?**

**Ans.** The static members follows all rules of the access specifiers along with that it has the following restrictions:

- a) They can only call other static methods. However, objects can be created in it.
- b) They can access only static data (class variables).
- c) They cannot refer to this or super in any way. The usage of super keyword is beyond the scope of this book.

**16. Explain in brief about the static block.**

**Ans.** Like instance variables, which is initialized through constructors, static variables are initialized through static block, it is a block which gets executed exactly once, when the class is first loaded. Since only one copy of static variables exists within a class, it needs to be initialized only once.

**17. Differentiate between static and non-static data members.**

[ICSE 2010]

**Ans.**

Static data-members	Non static data members
Multiple instances are not created with every creation of an object.	Multiple instances are created with every creation of an object.
Static members are called by static methods without using any object.	Non-static members may be called by static methods using an object.

**18. What is a Singleton class?**

**Ans.** A Singleton class is such a class, whose just one and only one instance can be created.

**B. Consider the following class:**

**class IdentifyMyParts**

```
{  
    static int x = 7;  
    int y = 3;  
}
```

i) What are the class variables?

ii) What are the instance variables?

iii) What is the output from the following code:

```
IdentifyMyParts a = new IdentifyMyParts( );  
IdentifyMyParts b = new IdentifyMyParts( );  
a.y = 5;  
b.y = 6;  
a.x = 1;  
b.x = 2;  
System.out.println("a.y = " + a.y);  
System.out.println("b.y = " + b.y);  
System.out.println("a.x = " + a.x);  
System.out.println("b.x = " + b.x);  
System.out.println("IdentifyMyParts.x = " + IdentifyMyParts.x);
```

**Ans.**

- i) x is the class variable.
- ii) y is the instance variable.
- iii) a.y=5  
b.y=6



a.x=2  
b.x=2  
IdentifyMyParts.x=2

**C. Consider the following code and answer the questions that follow:**

```
class academic
{
    int x,y;
    void access()
    {
        int a,b;
        academic student=new academic();
        System.out.println("Object Created");
    }
}
```

- i) What is the object name of the class?
- ii) Name the instance variables used in the class.
- iii) Write the name of local variables used in the program.
- iv) Give the type of function being used and its name.

**Ans.** i) student  
ii) int x and y  
iii) int a and b  
iv) Procedural function type, Name: access

**D. Consider the following code and answer the following questions:**

[ICSE 2009]

```
class academic
{
    int x,y;
    void access()
    {
        int a,b;
        academic student=new academic();
        System.out.println("Object created");
    }
}
```

- i) What is the object name of class academic?
- ii) Name the class variables used in the program.
- iii) Write the local variables used in the program.
- iv) Give the type of function used and its name.

**Ans.** i) student  
ii) none  
iii) int a, b  
iv) Type: procedural function  
Name: access()

**E. Consider the following code and answer the questions that follow:**

```
class vxl
{
    int x,y;
    void init()
```

```

        {
            x=5;
            y=10;
        }
protected void access()
    {
        int a=50, b=100;
        vxl vin=new vxl();
        vin.int();
        System.out.println("Object created");
        System.out.println("I am X="+vin.x);
        System.out.println("I am Y="+vin.y);
    }
}

```

- i) What is the object name of the class vxl?
- ii) Name the local variables of class.
- iii) What is the access specifier of method access()?
- iv) Write the output of above program.

**Ans.** i) vin  
 ii) a and b of int data type  
 iii) protected  
 iv) Object Created  
 I am X=5  
 I am Y=10

**F. Find the errors in the program given below and rewrite the corrected form:**

```

My class
{
    int a;
    static int b;
    void My class(int x, int y)
    void display1()
    {
        System.out.println(a+" "+b);
    }
    static void display2()
    {
        System.out.println(a+" "+b);
    }
    public static void main(String args[ ])
    {
        My class ob1=new My class();
        ob1.My class(5,6);
        ob1.display1();
        ob2.display2();
    }
}

```

**Ans.** class Myclass  
 {

```

    int a;
    static int b;

```

```

Myclass(int x, int y)
{
    a=x;
    b=y;
}

void display1( )
{
    System.out.println(a+" "+b);
}
static void display2( )
{
    System.out.println(b);
}
public static void main(String args[ ])
{
    Myclass ob1=new Myclass(5,6 );
    ob1.display1( );
    display2( );
}
}

```

### G. Write programs for the following:

1. Write a class with name Employee and basic as its data member, to find the gross pay of an employee for the following allowances and deduction. Use meaningful variables.

[ICSE 2005]

**Dearness Allowance = 25% of Basic Pay**

**House Rent Allowance = 15% of Basic Pay**

**Provident Fund = 8.33% of Basic Pay**

**Net Pay = Basic Pay + Dearness Allowance + House Rent Allowance**

**Gross Pay = Net Pay – Provident Fund**

**Ans.** import java.util.\*;  
//Question 1  
class Employee  
{  
float basic;  
void getvalue()  
{  
Scanner sc=new Scanner(System.in);  
System.out.println("Enter the basic pay:");  
basic=sc.nextFloat();  
}  
  
void calculate()  
{  
float da,hra,pf,netPay,grossPay;  
da=25/100f\*basic;  
hra=15/100f\*basic;

```

        pf=8.33f/100*basic;
        netPay=basic+da+hra;
        grossPay=netPay-pf;
        System.out.println("Dearness Allowance:"+da);
        System.out.println("House Rent Allowance:"+hra);
        System.out.println("Provident Fund:"+pf);
        System.out.println("Net Pay:"+netPay);
        System.out.println("Gross Pay:"+grossPay);
    }
}

```

**2. Define a class Salary described as below:**

**[ICSE 2007]**

**Data Members: Name, Address, Phone, Subject Specialization, Monthly Salary, Income Tax.**

**Member methods:**

- i) To accept the details of a teacher including the monthly salary.**
- ii) To display the details of the teacher.**
- iii) To compute the annual Income Tax as 5% of the annual salary above Rs.1,75,000/-.**

**Write a main method to create object of the class and call the above member method.**

**Ans.** import java.util.\*;

//Question 2

class Salary

```

{
    String name,address,subject;
    long phone;
    float monthlySalary,incomeTax;
    void accept()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the name:");
        name=sc.nextLine();
        System.out.println("Enter the address:");
        address=sc.nextLine();
        System.out.println("Enter the phone number:");
        phone=sc.nextLong();
        sc.nextLine();//dummy input
        System.out.println("Enter the Subject Specialization:");
        subject=sc.nextLine();
        System.out.println("Enter the monthly salary:");
        monthlySalary=sc.nextFloat();
    }
}

```

void display()

```

{
    System.out.println("Name:"+name);
    System.out.println("Address:"+address);
}

```

```

        System.out.println("Phone number:"+phone);
        System.out.println("Subject Specialization:"+subject);
        System.out.println("Monthly Salary:"+monthlySalary);
        System.out.println("Income Tax="+incomeTax);
    }

    void compute()
    {
        float annualSalary;
        annualSalary=12*monthlySalary;
        if(annualSalary>175000)
            incomeTax=5/100f*(annualSalary-175000);
        else
            incomeTax=0;
    }

    public static void main(String args[])
    {
        Salary ob=new Salary();
        ob.accept();
        ob.compute();
        ob.display();
    }
}

```

**3. Define a class Employee having the following description: [ICSE 2008]**

**Instance variables:**

**int pan**                      to store personal account number  
**String name**                to store name  
**double tax income**        to store annual taxable income  
**double tax**                   to store tax that is calculated

**Member functions:**

**input ( )**                    Store the pan number, name, taxable income  
**calc( )**                      Calculate tax for an employee  
**display ( )**                  Output details of an employee

Write a program to compute the tax according to the given conditions and display the output as per given format.

**Total Annual Taxable Income Tax Rate**

<b>Upto Rs.1,00,000</b>	<b>No tax</b>
<b>From 1,00,001 to 1,50,000</b>	<b>10% of the income exceeding Rs.1,00,000</b>
<b>From 1,50,001 to 2,50,000</b>	<b>Rs.5000 + 20% of the income exceeding Rs.1,50,000</b>
<b>Above Rs.2,50,000</b>	<b>Rs. 25,000 + 30% of the income exceeding Rs.2,50,000</b>

**Output:**

<b>Pan Number</b>	<b>Name</b>	<b>Tax-income</b>	<b>Tax</b>
—	—	—	—

```

Ans. import java.util.*;
        class Employee //Question 3
        {
            int pan;

```

```

String name;
double tax_income,tax;
void accept()
{
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter the pan number:");
    pan=sc.nextInt();
    sc.nextLine();//dummy input
    System.out.println("Enter the name:");
    name=sc.nextLine();
    System.out.println("Enter the taxable income:");
    tax_income=sc.nextDouble();
}

void display()
{
    System.out.println("Pan Number\tName\tTax-income\tTax");
    System.out.println(pan+"\t"+name+"\t"+tax_income+"\t"+tax);
}

void calc()
{
    if(tax_income<=100000)
        tax=0;
    else if(tax<=150000)
        tax=10/100.0*(tax_income-100000);
    else if(tax<=250000)
        tax=5000+20/100.0*(tax_income-150000);
    else
        tax=25000+30/100.0*(tax_income-250000);
}

public static void main(String args[])
{
    Employee ob=new Employee();
    ob.accept();
    ob.calc();
    ob.display();
}
}

```

**4. Define a class called Mobike with the following description:**

**Instance variables/ Data members:**

<b>Bno</b>	<b>: to store the bike's number</b>
<b>phno</b>	<b>: to store the phone number of the customer</b>
<b>name</b>	<b>: to store the name of the customer</b>
<b>days</b>	<b>: to store the number of days the bike is taken on rent</b>

**charge** : to calculate and store the rental charge

**Member methods:**

**void input ( )** : to input and store the detail of the customer

**void compute ( )** : to compute the rental charge. The rent for a Mobile is charged on the following basis:

**First five days** : Rs.500 per day

**Next five days** : Rs.400 per day

**Rest of the days** : Rs.200 per day

**void display ( )** : to display the details in the following format:

Bike No.	Phone No.	Name No.	of days	Charge
_____	_____	_____	_____	_____

**Ans.** import java.util.\*;

//Question 4

```
public class Mobike
{
    int bno,days;
    long phno;
    String name;
    long charge;

    void input()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the bike's number:");
        bno=sc.nextInt();
        System.out.println("Enter the phone number:");
        phno=sc.nextInt();
        sc.nextLine();//dummy input
        System.out.println("Enter the name:");
        name=sc.nextLine();
        System.out.println("Enter the no. of days taken for rent :");
        days=sc.nextInt();
    }

    void compute()
    {
        if(days<=5)
            charge=500*days;
        else if(days<=10)
            charge=500*5+(days-5)*400;
        else
            charge=500*5+400*5+(days-10)*200;
    }

    void display()
    {
        System.out.println("Bike No.\tPhone No.\tName\t\tNo. of dsys\t\tCharge");
    }
}
```

```

        System.out.println(bno+"\t\t"+phno+"\t\t"+name+"\t\t"+days+"\t\t"+charge);
    }
}

```

**5. Write a program with the following specifications:**

**Class name: Student**

**Data members:**

**Name** : to store the name of a student  
**Hindi** : to store the marks in hindi subject  
**English** : to store the marks in english subject  
**Maths** : to store the marks in mathematics  
**Computer** : to store the marks in computer  
**Average** : to store the average of the marks obtained  
**Grade** : to store the grade depending upon the average.

**Member methods**

**void accept()** : to accept name and marks in the 4 subjects.

**void calcavg()** : to calculate and store the grade according to the following slabs:

<b>Average marks</b>	<b>Grade obtained</b>
<b>90 and above</b>	<b>A++</b>
<b>Between 75 to 89 (both inclusive)</b>	<b>A</b>
<b>Between 60 to 75 (both inclusive)</b>	<b>B</b>
<b>Less than 60</b>	<b>C</b>

**Write the main method to create the object of the class and call the above method.**

**Ans.** import java.util.\*;

class Student //Question 5

```

{
    String name;
    int hindi,english,maths,computer;
    float average;
    String grade;

    void accept()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the name:");
        name=sc.nextLine();
        System.out.println("Enter marks in hindi:");
        hindi=sc.nextInt();
        System.out.println("Enter marks in english:");
        english=sc.nextInt();
        System.out.println("Enter marks in maths:");
        maths=sc.nextInt();
        System.out.println("Enter marks in computer:");
        computer=sc.nextInt();
    }

    void calcavg()
    {

```



```

        average=(float)(hindi+english+maths+computer)/4;
        if(average>=90)
            grade="A++";
        else if(average>=75 && average<=89)
            grade="A";
        else if(average>=60 && average<=74)
            grade="B";
        else if(average<60)
            grade="C";
        System.out.println("Grade="+grade);
    }

    public static void main(String args[])
    {
        Student ob=new Student();
        ob.accept();
        ob.calcavg();
    }
}

```

**6. Design class called Bank with the following descriptions:**

**Data members:**

**Name** : to store the name of the depositor  
**Acno** : to store the account number  
**Type** : to store type of the account  
**Bal** : to store the balance amount in the account

**Member functions:**

**initialise()** : to assign the data members with any value.  
**depo(int a)** : where 'a' is the amount to be deposited and the variable bal is to be updated.  
**withdraw( int a)** : where 'a' is the amount to be withdrawn after checking the balance (Minimum balance should be `1000 ) and the variable 'bal' is to be updated.  
**print()** : to print all the details.

Write the main method to create the object of the class and call the above method.

**Ans.** import java.util.\*;

//Question 6

class Bank

```

{
    String name;
    int acno;
    String type;
    float bal;

    void initialise()
    {
        name="Amit";
        acno=1234673;
        type="Savings";
    }
}

```

```

    bal=5000;
}

void depo(int a)
{
    bal+=a;
}

void withdraw(int a)
{
    if(bal-a<1000)
        System.out.println("Minimum balance should be Rs 1000");
    else
        bal-=a;
}

void print()
{
    System.out.println("Name:"+name);
    System.out.println("Account Number:"+acno);
    System.out.println("Account Type:"+type);
    System.out.println("Balance:"+bal);
}

public static void main(String args[])
{
    Scanner sc=new Scanner(System.in);
    int ch,n;
    Bank ob=new Bank();
    ob.initialise();
    System.out.println("Enter 1 to deposit, 2 to withdraw and 3 for baaccount details:");
    ch=sc.nextInt();
    switch(ch)
    {
        case 1:
            System.out.println("Enter amount to be deposited:");
            n=sc.nextInt();
            ob.depo(n);
            break;
        case 2:
            System.out.println("Enter amount to withdraw:");
            n=sc.nextInt();
            ob.withdraw(n);
            break;
        case 3:
            ob.print();
    }
}

```

```

        break;
    default:
        System.out.println("Invalid Entry!");
    }
}
}

```

**7. Define a class Bill as described below:**

**Data members are:**

**Name** : to store the name of the consumer  
**Consumerno** : to store the consumer number  
**Unitconsumed** : to store the unit consumed

**Member methods are:**

**datainput()** : to read the data of a person  
**compute()** : to calculate the bill amount as per criteria

Units Consumed	Rate
Up to 100 units	1.20
More than 100 and up to 200 units	2.20
More than 200 and up to 300 units	3.20
Above 300 units	4.00

**Display() – To display the output as per the format:**

Consumer Name	Consumer No	Unit Consumed	Bill Amount
---------------	-------------	---------------	-------------

**Ans.** import java.util.\*;

class Bill //Question 7

```

{
    String name;
    int consumerno;
    int unitconsumed;
    void datainput()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the name:");
        name=sc.nextLine();
        System.out.println("Enter the Consumer Number:");
        consumerno=sc.nextInt();
        System.out.println("Enter the Units Consumed:");
        unitconsumed=sc.nextInt();
    }
}

```

```

float compute()
{
    if(consumerno<=100)
        return unitconsumed*1.20f;
    else if(consumerno>100 && consumerno<=200)
        return unitconsumed*2.20f;
    else if(consumerno>200 && consumerno<=300)
        return unitconsumed*3.20f;
    else

```

```

        return unitconsumed*4.00f;
    }

    void display()
    {
        System.out.println("Consumer Name\tConsumer No\tUnit Consumed\tBill
Amount");
        System.out.println(name+"\t"+consumerno+"\t\t"+unitconsumed+"\t\t"+compute());
    }
}

```

**8. Write a program with the following specifications:**

**Class : Empl**

**Data Members:**

- Emp\_No** : To store the employee number
- Name** : To store the name of the employee
- Basic** : To store the basic salary of an employee
- DA** : To store the dearness allowance of an employee
- HRA** : To store the House Rent Allowance of an employee
- TA** : To store the Travelling Allowance of an employee
- PF** : To store the Provident Fund of an employee
- Gross** : To store the Gross Salary

**Member Methods:**

- get ( )** : To accept Employee No., Name and Basic Salary of the employees
- calcu ( )** : To calculate the Gross Salary based on the following condition:

Basic Salary	DA(%)	TA(%)	HRA(%)	PF(%)
>= 20,000	53	12	10	8
>=10,000 to <20,000	45	10	12	7.5
< 10,000	40	8	14	7

**Gross Salary = ( Basic Salary + DA + TA + HRA ) – PF**

**display ( )** : To display the following data in given format :

**EMPLOYEE No. NAME GROSS SALARY PF**

Write a main method to create the object of the above class and call the above method to calculate and print the Employee No. Name, Gross Salary and PF of an employee.

**Ans.** import java.util.\*;

class Empl //Question 8

```

{
    int emp_no;
    String name;
    float basic,da,hra,ta,pf,gross;
    void get()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the Employee No:");
        emp_no=sc.nextInt();
        sc.nextLine();//dummy input
    }
}

```

```

        System.out.println("Enter the Employee Name:");
        name=sc.nextLine();
        System.out.println("Enter the Basic Salary:");
        basic=sc.nextFloat();
    }

    void calcul()
    {
        if(basic>=20000)
        {
            da=53/100f*basic;
            ta=12/100f*basic;
            hra=10/100f*basic;
            pf=8/100f*basic;
        }
        else if(basic>=10000 && basic<20000)
        {
            da=45/100f*basic;
            ta=10/100f*basic;
            hra=12/100f*basic;
            pf=7.5f/100*basic;
        }
        else
        {
            da=40/100f*basic;
            ta=8/100f*basic;
            hra=14/100f*basic;
            pf=7f/100*basic;
        }
        gross=(basic+da+ta+hra)-pf;
    }

    void display()
    {
        System.out.println("Employee No.\tName\tGross Salary\tPF");
        System.out.println(emp_no+"\t\t"+name+"\t\t"+gross+"\t\t"+pf);
    }

    public static void main(String args[])
    {
        Empl ob=new Empl();
        ob.get();
        ob.calcul();
        ob.display();
    }
}

```

**9. Define a class called FruitJuice with the following description:**

### Instance Variables/Data Members:

**int product\_code** – stores the product code number  
**String flavor** – stores the flavour of the juice(orange, apple, etc)  
**String pack\_type** – stores the type of packaging (tetra-pack, bottle, etc)  
**int pack\_size** – stores package size (200 ml, 400 ml etc)  
**int product\_price** – stores the price of the product

### Member Methods:

**void input()** – to input and store the product code, flavour, pack type, pack size and product price.  
**void discount()** – to reduce the product price by 10.  
**void display()** – to display the product code, flavour, pack type, pack size and product price.

Write the main method to create an object of the class and call the above member methods.

**Ans.** import java.util.\*;

```
class FruitJuice //Question 9
```

```
{
    int product_code;
    String flavour,pack_type;
    int pack_size;
    int product_price;

    void input()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the product code:");
        product_code=sc.nextInt();
        sc.nextLine();//dummy input
        System.out.println("Enter the flavour:");
        flavour=sc.nextLine();
        System.out.println("Enter the pack type:");
        pack_type=sc.nextLine();
        System.out.println("Enter the pack size:");
        pack_size=sc.nextInt();
        System.out.println("Enter the product price:");
        product_price=sc.nextInt();
    }

    void discount()
    {
        product_price=product_price-10;
    }

    void display()
    {
        System.out.println("Product Code\tFlavour\tPack Type\tPack Size\tProduct
Price");
    }
}
```

```

System.out.println(product_code+"\t\t"+flavour+"\t\t"+pack_type+"\t\t"+pack_size+"\t\t"
+product_price);
}

```

```

public static void main(String args[])
{
    FruitJuice ob=new FruitJuice();
    ob.input();
    ob.discount();
    ob.display();
}
}

```

**10. Define a class Book with the following specifications:**

**Instance Variables/Data Members:**

**BOOK\_NO** : int type to store the book number  
**BOOK\_TITLE** : String type to store the title of the book  
**PRICE** : float type to store the price per copy

**Member Methods:**

**TOTAL\_COST()** : to calculate the total cost for N number of copies,  
where N is passed to the function as argument  
**INPUT()** : to read **BOOK\_NO, BOOK\_TITLE, PRICE**  
**PURCHASE()** : to ask the user to input the number of copies to be  
purchased. It invokes **TOTAL\_COST()** and prints  
the total cost to be paid by the user.

**Write the main method to create an object of the class and call the above member methods.**

**Ans.** import java.util.\*;  
class Book //Question 10  
{  
 int book\_no;  
 String book\_title;  
 float price;  
  
 float total\_cost(int N)  
 {  
 return N\*price;  
 }  
  
 void input()  
 {  
 Scanner sc=new Scanner(System.in);  
 System.out.println("Enter the BOOK NO.:");  
 book\_no=sc.nextInt();  
 sc.nextLine();//dummy input  
 System.out.println("Enter the BOOK TITLE:");  
 book\_title=sc.nextLine();  
 System.out.println("Enter the BOOK PRICE:");  
 price=sc.nextFloat();  
 }  
}

```

    }

    void purchase()
    {
        Scanner sc=new Scanner(System.in);
        int p;
        System.out.println("Enter the number of copies to be purchased:");
        p=sc.nextInt();
        System.out.println("TOTAL COST="+total_cost(p));
    }

    public static void main(String args[])
    {
        Book ob=new Book();
        ob.input();
        ob.purchase();
    }
}

```

**11. Define a class Flight with following description:**

**Instance Variables/Data Members:**

**fl\_no** : to store the flight number of int type  
**dest** : to store the destination of the flight of String type  
**dist** : to store the distance of the flight of float type  
**fuel** : to store the fuel required by the flight of float type

**Member Methods:**

**i) calfuel( )** : to calculate the value of fuel as per the following criteria

Distance	Fuel
<=1000	500
>1000 and <=2000	1100
>2000	2200

**ii) feedinfo( )** to allow user to enter values for Flight Number, Destination, Distance and call function calfuel( ) to calculate the quantity of Fuel.

**iii) showinfo( )** to allow user to view the content of all the data members.

**Write the main method to create an object of the class and call the above member methods.**

**Ans.** import java.util.\*;

class Flight //Question 11

```

{
    int fl_no;
    String dest;
    float dist;
    float fuel;

    void calfuel()
    {
        if(dist<=1000)
            fuel=500;
        else if(dist>1000 && dist<=2000)
            fuel=1100;
    }
}

```



```

        else
            fuel=2200;
    }

    void feedinfo()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the Flight Number:");
        fl_no=sc.nextInt();
        sc.nextLine();//dummy input
        System.out.println("Enter the Destination:");
        dest=sc.nextLine();
        System.out.println("Enter the Distance:");
        dist=sc.nextInt();
        calfuel();
    }

    void showinfo()
    {
        System.out.println("Flight No.:"+fl_no);
        System.out.println("Destination:"+dest);
        System.out.println("Distance:"+dist);
        System.out.println("Fuel:"+fuel);
    }

    public static void main(String args[])
    {
        Flight ob=new Flight();
        ob.feedinfo();
        ob.showinfo();
    }
}

```

**12. Define a class HOTEL with the following description:**

**Instance Variables/Data Members:**

**Rno** : Room No. of int type.  
**Name** : Customer name of String type.  
**Tariff** : stores per day charges of float type.  
**NOD** : no. of days of integer type.

**Member Methods:**

**CALC()** : to calculate and return Amount as  $NOD * Tariff$  and if the as value of  $NOD * Tariff$  is more than 10000 then  $1.05 * NOD * Tariff$

**Checkin()** : to enter the Rno, Name, Tariff and NOD  
**Checkout()** : to display Rno, Name, Tariff, NOD and Amount by calling **CALC()**

**Write the main method to create an object of the class and call the above member methods.**

**Ans.** import java.util.\*;  
class Hotel //Question 12

```

{
    int rno;
    String name;
    float tariff;
    int nod;

    float calc()
    {
        if(nod*tariff<=10000)
            return nod*tariff;
        else
            return 1.05f*nod*tariff;
    }

    void checkin()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the Room Number:");
        rno=sc.nextInt();
        sc.nextLine();//dummy input
        System.out.println("Enter the Name:");
        name=sc.nextLine();
        System.out.println("Enter the Tariff:");
        tariff=sc.nextFloat();
        System.out.println("Enter the No. of days:");
        nod=sc.nextInt();
    }

    void checkout()
    {
        System.out.println("Room Number:"+rno);
        System.out.println("Name:"+name);
        System.out.println("Tariff:"+tariff);
        System.out.println("No. of days:"+nod);
        System.out.println("Amount:"+calc());
    }

    public static void main(String args[])
    {
        Hotel ob=new Hotel();
        ob.checkin();
        ob.checkout();
    }
}

```

**13. Define a class Telephone having the following description:**

**Instance Variables / Data Members:**

**int prv, pre** : to store the previous and present meter reading

**int call** : to store the calls made (i.e. pre – prv)  
**String name** : to store name of the customer  
**double amt** : to store the amount  
**double total** : to store the total amount to be paid  
**Member Methods:**  
**void input ()** : to input the previous reading, present reading and name of the customer  
**void cal ()** : to calculate the amount and total amount to be paid  
**void display ()** : to display the name of the customer, calls made, amount and total amount to be paid in the following format:

**Name            Calls Made            Amount            Total Amount**  
 \_\_\_\_\_            \_\_\_\_\_            \_\_\_\_\_            \_\_\_\_\_

Write a program to compute the monthly bill to be paid according to the given conditions:

Calls Made	Rate
Up to 100 calls	No charge
For the next 100 calls	90 paise per call
For the next 200 calls	80 paise per call
More than 400 calls	70 paise per call

However, every customer has to pay Rs.180 per month as monthly rent for availing the service.

```

Ans. import java.util.*;
class Telephone //Question 13
{
    int prv,pre,call;
    String name;
    double amt,total;

    void input()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the previous meter reading:");
        prv=sc.nextInt();
        System.out.println("Enter the present meter reading:");
        pre=sc.nextInt();
        sc.nextLine();// dummy input
        System.out.println("Enter the name:");
        name=sc.nextLine();
    }

    void cal()
    {
        call=pre-prv;
        if(call<=100)
            amt=0;
        else if(call<=200)
  
```

```

        amt=0.90f*call;
    else if(call<=400)
        amt=0.80f*call;
    else
        amt=0.70f*call;
    total=180+amt;
}

void display()
{
    System.out.println("Name\tCalls Made\tAmount\tTotal Amount");
    System.out.println(name+"\t"+call+"\t"+amt+"\t"+total);
}
}

```

**14. Define a class called Number having the following members:**

**Data Members:**

**lLimit** : Lower limit of int type

**uLimit** : Upper limit of int type

**Member Functions:**

**void getval()** : To accept the lower and upper limit to depict a range.

**boolean isArmstrong(int n)** : Checks and returns true if 'n' is an armstrong number, otherwise return false.

**void display ()** : It displays all armstrong numbers between the given range.

**Ans.** import java.util.\*;  
class Number //Question 14

```

{
    int lLimit,uLimit;
    void getval()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the lower limit:");
        lLimit=sc.nextInt();
        System.out.println("Enter the upper limit:");
        uLimit=sc.nextInt();
    }
}

```

boolean isArmstrong(int n)

```

{
    int t=n,d,s=0;
    while(t!=0)
    {
        d=t%10;
        s=s+d*d*d;
        t=t/10;
    }
    if(s==n)

```

```
        return true;
    else
        return false;
}

void display()
{
    int i;
    for(i=lLimit;i<=uLimit;i++)
    {
        if(isArmstrong(i))
            System.out.println(i);
    }
}
}
```

# Chapter 11: Constructors

## Practice Questions

### ----->Objective-Type Questions<-----

#### Fill in the blanks:

1. A constructor is a Function defined in a class with name same as that of the class.
2. Constructor is invoked when an object is created.
3. A constructor that does not have any argument is called Default Constructor.
4. A constructor that takes the arguments is called Parameterised Constructor.
5. The **class** is the implicit return type of a constructor function.

### ----->Subjective-Type Questions<-----

#### A. Answer the following questions:

##### 1. What is a constructor? Why is it needed in a program?

**Ans.** Constructor is a function that have the same name as the class and is automatically invoked during creation of an object.

It is needed for:

- a) Allocate memory space for an object.
- b) Initialize data members within an object.

##### 2. State the characteristics of a constructor.

**Ans.** Constructor is a member function of a class having the following characteristics:

1. It has same name as the class-name.
2. It cannot have a return type, not even void is used, because the implicit return type of a constructor is the class itself.
3. It follows the usual rules of accessibility as other members of a class and therefore access modifiers can be applied to it.
4. It gets called automatically whenever an object is created.
5. It is usually used to initialize the members of the class with legal initial values.

##### 3. How are constructors invoked?

**Ans.** Constructor function gets invoked automatically whenever an object is created.

##### 4. Why do we need a constructor as a class member? [ICSE 2007]

**Ans.** Whenever an object is created, it must be constructed usually with legal initial values automatically without being specified by the programmer. This will immensely relieve the programmer from initializing values of objects for large programs and hence the need for constructor.

##### 5. State the difference between function and constructor. [ICSE 2005]

**Ans.** Difference between function and constructor:

Function	Constructor
----------	-------------

Its name is different as the name of the class.	Its name is same as the class-name.
Its <i>return-type</i> should be specified.	No <i>return-type</i> needs to be specified.

**6. How are private constructors called?**

**Ans.** Private constructors can be invoked only when an object is being created only within the same class block.

**7. What are the different types of constructors available in Java?**

**Ans.** Parameterised Constructor and Non-Parameterised Constructor.

**8. What is a default constructor?**

[ICSE 2006]

**Ans.** Default Constructor refers to a non-parameterised constructor that initializes data members with default legal values.

**9. Point out the difference between implicit and explicit default constructors.**

**Ans.** Implicit default constructor is a constructor which gets created automatically by the JVM (which initializes the data members with default initial values) incase the class do not have a constructor explicitly defined by the user. Explicit default constructor is a non-parameterised constructor defined by the programmer to initialize the data-members within a class with default initial values.

**10. What are Temporary Objects? How are they created, explain with the help of an example?**

**Ans.** An object that do not have a name and exist as long as the line is being executed is called Temporary or Anonymous object. Temporary or Anonymous objects or instances are the ones that live in the memory as long as it is being used or referenced in an expression and after that it dies. For example,

```
public class temp
{
    int a,b;
    temp(int x, int y)
    {
        a=x;
        b=y;
    }
    void show()
    {
        System.out.println(a+","+b);
    }
    static void test( )
    {
        new temp(1,2).show( ); /*Temporary object is being called to invoke
                                show( ) function.*/
    }
}
```

**11. What is constructor overloading? Illustrate it with the help of an example.**

**Ans.** Multiple constructors of a class having different parameter lists in a class is termed as constructor overloading. For example,

```
class Cuboid
```

```

    {
        double width;
        double height;
        double depth;
        Cuboid( ) //Constructor with no parameter.
        {
            width=0;
            height=0;
            depth=0;
        }
        Cuboid(double n) //Constructor with single parameter
        {
            width=height=depth=n;
        }
        Cuboid(double w, double h,double d)
        //Constructor with 3 parameters
        {
            width=w;
            height=h;
            depth=d;
        }
        void displayVolume()
        {
            double vol;
            //Compute volume
            vol= width*height*depth;
            System.out.println("The Volume is="+vol);
        }
        static void createObjects()
        {
            Cuboid mycuboid1=new Cuboid();
            //Calls the first constructor version.
            Cuboid mycuboid2=new Cuboid(2.0);
            //Calls the 2nd constructor version.
            Cuboid mycuboid3=new Cuboid(1.5,2.5,4.5);
            //Calls the 3rd constructor version.
            mycuboid1.displayVolume( );
            mycuboid2.displayVolume( );
            mycuboid3.displayVolume( );
        }
    }
}

```

**12. What is a destructor? Is destructor function necessary in Java? If no, explain why.**

**Ans.** Destructor is a function whose purpose is to deallocate memory space for an object which was allocated by the constructor. But Java uses a unique automatic method of deallocation of memory called garbage collection. This happens when no references to an object exist, that object is assumed to be no longer needed, and therefore the memory occupied by the object can be reclaimed. Since



garbage collection happens automatically in Java, there is no need of explicitly defining a destructor function in a class.

**13. What is the implicit return type of a constructor function?**

**Ans.** The implicit return type of a constructor is the class type.

**14. Enter any two variables through constructor with parameters and write a program to swap and print the values.** [ICSE 2005]

**Ans.**

```
class Swap
{
    int a,b;
    Swap(int x, int y)
    {
        a=x;
        b=y;
    }
    void change()
    {
        int t;
        t=a;
        a=b;
        b=t;
        System.out.println(a+ " "+b);
    }
}
```

**15. What is the default initial value of a boolean variable datatype?**

[ICSE2009]

**Ans.** The default initial value of boolean variable is false.

**B. Find the errors in the following program and rectify it:**

```
class Myclass
{
    int a,b;
    void Myclass(int x, int y)
    {
        a=x;
        b=y;
    }
    int Myclass(int x)
    {
        a=b=x;
    }
    void show( )
    {
        System.out.println(a+ " "+y);
    }
    public static void main(String args[ ])
    {
        Myclass ob1=new Myclass( );
    }
}
```

```

Myclass ob2=new Myclass(12.3,14.6,15);
Myclass ob3=new Myclass(7);
ob1.ob2.ob3.show( );

```

```

    }
}
Ans. class Myclass
{
    int a,b;
    Myclass(int x, int y)
    {
        a=x;
        b=y;
    }
    Myclass(int x)
    {
        a=b=x;
    }
    void show( )
    {
        System.out.println(a+ " "+y);
    }
    public static void main(String args[ ])
    {
        Myclass ob1=new Myclass(5,6 );
        Myclass ob2=new Myclass(7);
        ob1.show( );
        ob2.show( );
    }
}

```

**C. Consider the following code and answer the questions that follow:**

```

class academic
{
    int x,y;
    void access( )
    {
        int a,b;
        academic student=new academic( );
        System.out.println("Object Created");
    }
}

```

- i) What is the object name of the class?
- ii) Name the instance variables used in the class.
- iii) Write the name of local variables used in the program.
- iv) Give the type of function being used and its name.

**Ans.** i) student

- ii) x and y
- iii) a and b
- iv) Type: Procedural function  
Name: access()

**D. Consider the following code and answer the questions that follow:**

```
class Today
{
    static int a;
    char b;
    void input( )
    {
        a=20;
        b='Z';
    }
    void convert( )
    {
        char c=(char)(a+b);
        System.out.println(c);
    }
    public static void main( )
    {
        Today t=new Today( );
        t.input( );
        t.convert( );
    }
}
```

**Based on the above given piece of code, answer the questions which follow:**

- i) Name the instance, class and local variables.**
- ii) What is the name of the constructor of the above class?**
- iii) Explain the line: Today t=new Today( );**

- Ans.**
- i) Instance Variable: char b  
Class Variable: int a  
Local Variables: char c
  - ii) No explicit constructor is created but the implicit default constructor name is Today( ).
  - iii) An object named t is being created of the class named Today and allocated memory space.

**E. In the program given below, state the name and the value of the following:**

- i) method argument or argument variable**
  - ii) class variable**
  - iii) local variable**
  - iv) instance variable**
- ```
class Myclass
```

```

{
    static int x=7;
    int y=2;
    public static void main(String args[])
    {
        MyClass obj=new MyClass( );
        System.out.println(x);
        obj.sampleMethod(5);
        int a=6;
        System.out.pritnln(a);
    }
    void sampleMethod(int n)
    {
        System.out.println(n);
        System.out.println(y);
    }
}

```

- Ans.** i) In main( ) function argument: String args[ ]  
 In sampleMethod( ) function argument: int n  
 ii) Class Variable: int x  
 iii) int a  
 iv) int y

**F. Write programs for the following:**

**1. Define a class called Box, having members as:**

**Data Members:** length, breadth and height of int type.

**Member Functions:**

**i) Constructor to initialize the data members.**

**ii) To compute and display the volume.**

**Ans.** class Box //Question 1

```

{
    int length,breadth,height;
    Box(int l,int b,int h)
    {
        length=l;
        breadth=b;
        height=h;
    }

    void compute()
    {
        int volume;
        volume=length*breadth*height;
        System.out.println("Volume="+volume);
    }
}

```

**2. Define a class called Friend, having members as:  
Data Members: Name, Address, Favourite hobby**

**Member Functions:**

**i) Constructor to initialize the data members.**

**ii) To display the details.**

**Also create the main( ) and create 2 objects in it and initialize it with information of two of your friends and display it, by calling the above functions.**

**Ans.** class Friend //Question 2

```
{
    String name,address,hobby;
    Friend(String n, String a, String h)
    {
        name=n;
        address=a;
        hobby=h;
    }

    void display()
    {
        System.out.println("Name="+name);
        System.out.println("Address="+address);
        System.out.println("Favourite Hobby="+hobby);
    }

    public static void main(String args[])
    {
        Friend ob1=new Friend("Amit","Kerala","Football");
        Friend ob2=new Friend("Shristi","Karnataka","Badminton");
        ob1.display();
        ob2.display();
    }
}
```

**3. Define a class named fourSide, having members as:**

**Data members: length, breadth**

**Member Functions:**

**i) Overloaded constructor to initialize the dimension of the four-sided figure with a square and a rectangle.**

**ii) Compute the area and display it.**

**Also create the main( ) method to show the implementation of the class's methods.**

**Ans.** class FourSide //Question 3

```
{
    int length,breadth;
    FourSide(int s)//For Square
```

```

    {
        length=breath=s;
    }

    FourSide(int l,int b)
    {
        length=l;
        breath=b;
    }

    void display()
    {
        int area;
        area=length*breadth;
        System.out.println("Area="+area);
    }

    public static void main(String args[])
    {
        FourSide ob1=new FourSide(4); //Object for a square
        FourSide ob2=new FourSide(7,5); //Object for a rectangle
        ob1.display();
        ob2.display();
    }
}

```

**4. The sum of n natural numbers is defined by the following formula:**

$$n * (n + 1) / 2$$

**Create a class named Natural, which will contain the following members:**

**Data members: n and s of int data type.**

**Member functions:**

**i) Parametrized constructor to initialize n.**

**ii) void compute( ) to calculate the sum of first n natural numbers using the above formula and store it in s.**

**iii) void display( ) to show the sum.**

**Ans.** class Natural //Question 4

```

{
    int n,s;
    Natural(int p)
    {
        n=p;
    }

    void compute()
    {
        s=(n*(n+1))/2;
    }
}

```

```

    }

    void display()
    {
        System.out.println("Sum="+s);
    }
}

```

**5. Define methods in a class named Algebra to find the difference between the following equations:**

$$f1(x) = x^2 + 2x + 15$$

$$f2(y) = -y^2 - 13y + 12$$

$$f3(x,y) = f1(x) - f2(y)$$

**where x and y are both of double data type.**

**Ans.** class Algebra //Question 5

```

{
    static double f1(double x)
    {
        return x*x+2*x+15;
    }

    static double f2(double y)
    {
        return -(y*y)-13*y+12;
    }

    static void f3(double x,double y)
    {
        System.out.println(f1(x)-f2(y));
    }
}

```

**6. Define a class named Conversion having the following static methods:**

**i) int mToCm(int m ), which converts metre(m) to centimetre and return it.**

**ii) int multiply(int x, int y ), which returns the product of x and y.**

**Define another class named Rectangle which contains the following data members:**

- length of int type which holds the length of a rectangle in metre.
- breadth of int type which holds the breadth of a rectangle in centimetre.

**Create member functions:**

**i) Constructor to initialize the data members.**

**ii) Convert the length to centimetre by calling the relevant method of the above class.**

**iii) Compute the area by calling the relevant method of the above class and display the result.**

**Ans.** class Conversion //Question 6

```

{
    static int mToCm(int m)

```

```

    {
        return m*100;
    }

    static int multiply(int x,int y)
    {
        return x*y;
    }
}

public class Rectangle
{
    int length,breadth;
    Rectangle(int l,int b)
    {
        length=l;
        breadth=b;
    }

    void convert()
    {
        length=Conversion.mTocm(length);
    }

    void compute()
    {
        int area;
        convert();
        area=Conversion.multiply(length,breadth);
        System.out.println("Area="+area);
    }
}

```

**7. Design a class named Numbers, which will contain the following members:**

**Data Members: a , b and c of int data type.**

**Member Functions:**

**i) Parameterized constructor to initialize a and b.**

**ii) void show( ) to display the contents of a, b and c.**

**iii) void compute( ) to add a and b and store it in c.**

**In the main( ) create an object and initialize a and b with any value and add a and b by invoking the compute( ) function and display the contents of a, b and c using show( ) function.**

**Ans.** class Numbers //Question 7

```

{
    int a,b,c;
    Numbers(int x,int y)

```



```

{
    a=x;
    b=y;
}

void show()
{
    System.out.println(a+" "+b+" "+c);
}

void compute()
{
    c=a+b;
}

public static void main(String args[])
{
    Numbers ob=new Numbers(5,6);
    ob.compute();
    ob.show();
}
}

```

**8. Design a class named SumIntegers, which will contain the following members:**

**Data members: a and b of int data type.**

**Member Function:**

**i) Parameterized constructor to initialize a and b.**

**ii) Default constructor to initialize a and b with 0.**

**iii) Accepts an object as parameter and return that object among the current and passed object whose a + b (sum of the data members) is largest.**

**Also create the main ( ) to show its implementation.**

**Ans.** class SumIntegers //Question 8

```

{
    int a,b;
    SumIntegers(int x,int y)
    {
        a=x;
        b=y;
    }

    SumIntegers()
    {
        a=b=0;
    }
}

```

```

SumIntegers sum(SumIntegers p)
{
    if(a+b>p.a+p.b)
        return this;
    else
        return p;
}

public static void main(String args[])
{
    SumIntegers ob1=new SumIntegers(5,6);
    SumIntegers ob2=new SumIntegers(7,8);
    SumIntegers ob3=new SumIntegers();
    ob3=ob1.sum(ob2);
    System.out.println(ob3.a+" "+ob3.b);
}
}

```

## Lab Exercises

**1. Create a class named 'SumDemo' which contains three members: 'n' of int type and static members 'sum' of int type, 'count' of int type and 'average' of float type. Create a static block to initialize the 'sum' and 'count' with 0. Use a parameterized constructor to initialize the value of 'n'. Create a member function addIt( ) to add the value of 'n' of the current object to 'sum' and increment the value of 'count'**

**by 1. Create a static function calculate( ) to calculate the average=sum/count. Create another static function showAverage( ) which displays the value of 'average'. Also create the main( ) to create three objects and initialize them and call the addIt( ) function to add the value of all the three 'n's' of the three objects to 'sum' and 'count' the number of numbers added and display their average.**

**Ans.** class SumDemo //Question 1

```

{
    int n;
    static int sum;
    static int count;
    static float average;

    static
    {
        sum=count=0;
    }

    SumDemo(int p)
    {
        n=p;
    }
}

```

```

}

void addIt()
{
    sum+=n;
    count++;
}

static void calculate()
{
    average=(float)sum/count;
}

static void showAverage()
{
    System.out.println(average);
}

public static void main(String args[])
{
    SumDemo ob1=new SumDemo(5);
    SumDemo ob2=new SumDemo(6);
    SumDemo ob3=new SumDemo(7);
    ob1.addIt();
    ob2.addIt();
    ob3.addIt();
    calculate();
    showAverage();
}
}

```

**2. Create a class having default access specifier named 'Average' which contains a static method to accept two integers as parameters and return their average. Create another class named having public access specifier named 'FindAverage', which will contain the following members: a, b and c of int type as data members. Parameterized constructor to initialize a and b only. Another member function to find the average of a and b of the above class and store it in c, and display it.**

**Ans.** class Average //Question 2

```

{
    static int calc(int a,int b)
    {
        int c;
        c=(a+b)/2;
        return c;
    }
}

```

```

}

public class FindAverage
{
    int a,b,c;
    FindAverage(int x,int y)
    {
        a=x;
        b=y;
    }

    void calculate()
    {
        c=Average.calc(a,b);
        System.out.println(c);
    }
}

```

**3. Create a class named Number with the following members:**

**Data members: a of int data type.**

**Member Functions:**

**i) Parameterized constructor to initialize the data member.**

**ii) Default constructor to initialize the data member with 0.**

**iii) To display 'a' only.**

**iv) Which accepts an instance of Number class as parameter and creates another object, whose a will contain the sum the a's of the current and the passed object. This function should return the object newly instantiated.**

**Also create a main( ) to show its implementation.**

**Ans.** class Number //Question 3

```

{
    int a;
    Number(int p)
    {
        a=p;
    }

    Number()
    {
        a=0;
    }

    void display()
    {
        System.out.println(a);
    }
}

```

```

Number sum(Number ob)
{
    Number s=new Number();
    s.a=a+ob.a;
    return s;
}

public static void main(String args[])
{
    Number ob1=new Number(5);
    Number ob2=new Number(6);
    Number ob3=new Number();
    ob3=ob1.sum(ob2);
    ob3.display();
}
}

```

**4. Create a class named Number having 'a' as instance variable and 'b' as class variable. Use a static block to initialize 'b' with 0. Also create a parameterized constructor to initialize 'a'. Now create a member function which adds the 'a' of the current object to the class variable 'b'. Also create a class function to display 'b'. In the main( ) create 3 objects, initialize the a's with any values, and store their sum inside the 'b' and display it.**

**Ans.** class Number //Question 4

```

{
    int a;
    static int b;
    static
    {
        b=0;
    }

    Number(int x)
    {
        a=x;
    }

    void add()
    {
        b+=a;
    }

    static void display()
    {
        System.out.println(b);
    }
}

```

```

public static void main(String args[])
{
    Number ob1=new Number(5);
    Number ob2=new Number(6);
    Number ob3=new Number(7);
    ob1.add();
    ob2.add();
    ob3.add();
    display();
}
}

```

**5. Create a class named Number having 'a' as data member. Use a parametrized constructor to initialize 'a'. Create a class function which accepts two objects as parameter and interchange their content. Also create a member function to display the value of 'a'. In the main( ) create two objects, initialize them and interchange their contents. Show the contents of the object before and after the interchange.**

**Ans.** class Number //Question 5

```

{
    int a;
    Number(int x)
    {
        a=x;
    }

    static void swap(Number ob1, Number ob2)
    {
        int t;
        t=ob1.a;
        ob1.a=ob2.a;
        ob2.a=t;
    }

    void display()
    {
        System.out.println(a);
    }

    public static void main(String args[])
    {
        Number ob1=new Number(5);
        Number ob2=new Number(6);
        ob1.display();
        ob2.display();
    }
}

```

```

        swap(ob1,ob2);
        ob1.display();
        ob2.display();
    }
}

```

**6. One of the most easiest ways of rounding off a real number to its near integer is to take only the integer part after adding 0.5 to the real number. For example 12.7 when 0.5 is added and only the integer part is taken yields 13, 14.4 when 0.5 is added and only the integer part is taken yields 14. Design a class name RoundOff which will contain 'a' of integer type. Use a parameterized constructor to initialize it with a certain real number. Create a member function to round off 'a' to nearest integer and return it. In the main( ) create an object, initialize it with a real number and round it off to nearest integer and display it.**

**Ans.** import java.util.\*;  
class RoundOff //Question 6  
{  
 float a;  
 RoundOff(float b)  
 {  
 a=b;  
 }  
  
 void roundIt()  
 {  
 int b;  
 b=(int)(a+0.5f);  
 System.out.println("Rounded Off="+b);  
 }  
  
 public static void main(String args[])  
 {  
 Scanner sc=new Scanner(System.in);  
 float t;  
 System.out.println("Enter a real number:");  
 t=sc.nextFloat();  
 RoundOff ob=new RoundOff(t);  
 ob.roundIt();  
 }  
}

**7. You are to print the telephone bill of a subscriber. Create a class having the following data members: Phone Number of long data type (for storing the phone number). Name of String type (for storing the name of the subscriber).**

**Hire Charge** a symbolic constant of int type (to store monthly hire charge say Rs.200).

**Units Consumed** of int type (to store the monthly units consumed by the subscriber).

**Bill Amount** of float type (to store the bill amount that is payable).

Create member functions for the following:

i) Constructor to initialize all data members except Hire Charge and Bill Amount.

ii) Calculate the bill amount payable which is Hire Charge+(Rs.1 per unit for the first 100 units, Rs.1.50 per unit for the next 100 units and Rs.2.00 per unit thereafter).

iii) To display the Bill for the subscriber.

**Ans.** class Telephone //Question 7

```
{
    long phone;
    String name;
    final int hire=200;
    int units;
    float bill;
    Telephone(long p, String n, int u)
    {
        phone=p;
        name=n;
        units=u;
    }

    void calculate()
    {
        if(units<=100)
            bill=units*1;
        else if(units<=200)
            bill=100*1+(units-100)*1.50f;
        else
            bill=100*1+100*1.50f+(units-200)*2.00f;

        bill=hire+bill;
    }

    void display()
    {
        System.out.println("Phone Number="+phone);
        System.out.println("Name="+name);
        System.out.println("Hire Charge="+hire);
        System.out.println("Units Consumed="+units);
        System.out.println("Bill Amount="+bill);
    }
}
```



```
}  
}
```

**8. Define a class Taximeter having the following description:**

**Data members/instance variables:**

**int taxino** : to store taxi number

**String name** : to store passenger's name

**int km** : to store number of kilometres travelled

**Member functions:**

**Taximeter( )** : constructor to initialize taxino to 0, name to "" and km to 0.

**input( )** : to store taxino,name,km

**calculate( )** : to calculate bill for a customer according to given conditions:

| Kilometres travelled(km) | Rate/km |
|--------------------------|---------|
| 1 km                     | ₹ 25    |
| $1 < km \leq 6$          | ₹ 10    |
| $6 < km \leq 12$         | ₹ 15    |
| $12 < km \leq 18$        | ₹ 20    |
| $> 18$ km                | ₹ 25    |

**display( )** : To display the details in the following format

**Taxino**                      **Name**                      **Kilometres travelled**                      **Bill amount**

\_\_\_\_\_

**Ans.** import java.util.\*;  
class Taximeter //Question 8

```
{  
    int taxino;  
    String name;  
    int km;  
    Taximeter()  
    {  
        taxino=0;  
        name="";  
        km=0;  
    }  
  
    void input()  
    {  
        Scanner sc=new Scanner(System.in);  
        System.out.println("Enter the taxi number:");  
        taxino=sc.nextInt();  
        sc.nextLine();//dummy input  
        System.out.println("Enter the name:");  
        name=sc.nextLine();  
        System.out.println("Enter the kilometers travelled:");
```

```

        km=sc.nextInt();
    }

    int calculate()
    {
        int bill,rate;
        if(km==1)
            rate=25;
        else if(km<=6)
            rate=10;
        else if(km<=12)
            rate=15;
        else if(km<=18)
            rate=20;
        else
            rate=25;
        bill=km*rate;
        return bill;
    }

    void display()
    {
        System.out.println("Taxi No.\tName\tKilometres travelled\tBill amount");
        System.out.println(taxino+"\t\t"+name+"\t"+km+"\t\t"+calculate());
    }
}

```

**9. Create a class which will contain the following components:**

**Data Members: a and b of int type.**

**Member Functions:**

**i) Constructor to initialize a and b.**

**ii) Return the sum of a and b.**

**iii) To display the value of a and b.**

**Also create a static function which accepts two objects as parameters and print a and b of the object whose sum is the maximum.**

**In the main( ) create two objects, initialize them and display the data members of that object whose sum of the data-members is the maximum.**

**Ans.** class Question9

```

{
    int a,b;
    Question9(int x,int y)
    {
        a=x;
        b=y;
    }
}

```

```

int sum()
{
    return a+b;
}

void display()
{
    System.out.println(a+" "+b);
}

static void compare(Question9 ob1,Question9 ob2)
{
    if(ob1.a+ob1.b>ob2.a+ob2.b)
        ob1.display();
    else
        ob2.display();
}

public static void main(String args[])
{
    Question9 ob1=new Question9(5,6);
    Question9 ob2=new Question9(7,8);
    compare(ob1,ob2);
}
}

```

**10. Define a class called FruitJuice with the following description:**

**Instance Variables/Data Members:**

**int product\_code** : stores the product code number  
**String flavor** : stores the flavour of the juice(orange, apple, etc.)  
**String pack\_type** : stores the type of packaging (tetra-pack, bottle, etc.)  
**int pack\_size** : stores package size (200ml, 400ml, etc.)  
**int product\_price** : stores the price of the product  
**Member Methods:**  
**FruitJuice( )** : default constructor to initialize integer data members to zero and string data members to "".  
**void input( )** : to input and store the product code, flavour, pack type, pack size and product price.  
**void discount( )** : to reduce the product price by 10.  
**void display( )** : to display the product code, flavour, pack type, pack size and product price.

**Create an object in the main method and call all the above methods in it.**

**Ans.** import java.util.\*;  
class FruitJuice //Question 10  
{  
 int product\_code;  
 String flavour, pack\_type;

```

int pack_size,product_price;
FruitJuice()
{
    product_code=0;
    flavour=pack_type="";
    pack_size=product_price=0;
}

void input()
{
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter the product code:");
    product_code=sc.nextInt();
    sc.nextLine();//dummy input
    System.out.println("Enter the flavour:");
    flavour=sc.nextLine();
    System.out.println("Enter the pack type:");
    pack_type=sc.nextLine();
    System.out.println("Enter the pack size:");
    pack_size=sc.nextInt();
    System.out.println("Enter the product price:");
    product_price=sc.nextInt();
}

void discount()
{
    product_price-=10;
}

void display()
{
    System.out.println("Product code:"+product_code);
    System.out.println("Flavour:"+flavour);
    System.out.println("Pack type:"+pack_type);
    System.out.println("Pack size:"+pack_size);
    System.out.println("Product price:"+product_price);
}

public static void main(String args[])
{
    FruitJuice ob=new FruitJuice();
    ob.input();
    ob.discount();
    ob.display();
}

```

```
}  
}
```

**11. Create a class SalaryCalculation that is described as below:**

**Class Name** : SalaryCalculation  
**Data members** : name (String type data)  
basicPay, specialAlw, conveyanceAlw, gross, pf,  
netSalary, AnnualSal (All double type data)

**Member methods:**

- i) SalaryCalculation( ) : A constructor to assign name of employee (name), basic salary(basicPay) of your choice and conveyance allowance (conveyanceAlw) as Rs. 1000.00
- ii) void SalaryCal( ) : to calculate other allowances and salaries as given: specialAlw = 25% of basic salary. pf = 11% of basic salary. gross = basicPay + specialAlw + conveyanceAlw. netSalary = gross – pf. AnnualSal = 12 months netSalary.
- iii) void display( ) : to print the name and other calculations with suitable headings.

**Write a program in Java to calculate all the details mentioned above and print them all.**

```
Ans. import java.util.*;  
class SalaryCalculation //Question 11  
{  
    String name;  
    double basicPay, specialAlw,conveyanceAlw,gross,pf,netSalary,AnnualSal;  
    SalaryCalculation()  
    {  
        Scanner sc=new Scanner(System.in);  
        System.out.println("Enter the name:");  
        name=sc.nextLine();  
        System.out.println("Enter the Basic Pay:");  
        basicPay=sc.nextDouble();  
        conveyanceAlw=1000.00;  
    }  
  
    void salaryCal()  
    {  
        specialAlw=25/100.0*basicPay;  
        pf=11/100.0*basicPay;  
        gross=basicPay+specialAlw+conveyanceAlw;  
        netSalary=gross-pf;  
        AnnualSal=12*netSalary;  
    }  
  
    void display()
```

```

{
    System.out.println("Name="+name);
    System.out.println("Basic Pay="+basicPay);
    System.out.println("Special Allowance="+specialAlw);
    System.out.println("Conveyance Allowance="+conveyanceAlw);
    System.out.println("Gross="+gross);
    System.out.println("Provident Fund="+pf);
    System.out.println("Net Salary="+netSalary);
    System.out.println("Annual Salary="+AnnualSal);
}

public static void main(String args[])
{
    SalaryCalculation ob=new SalaryCalculation();
    ob.salaryCal();
    ob.display();
}
}

```

**12. A class Compound is created to calculate the compound interest using:**

$$CI = P \left( 1 + \frac{r}{100} \right)^t - P$$

**Where P — is the Principal amount, r – rate of interest and t–time period in years.**

**Data members of class: pamt, rate (double data type to store principal amount and rate of interest), time (integer to store time period)**

**Functions of the class:**

**i) Compound( )— constructor to assign default values to all the data members.**

**ii) void input( )— to input the principal, rate and time from the user.**

**iii) double findInterest( )-to find and return compound interest using the given formula.**

**iv) void printData( )— to print the principal, rate and time.**

**Write a main( ) function to input required data and by invoking suitable functions print the entered data and compound interest.**

**Ans.** import java.util.\*;

class Compound //Question 12

```

{
    double pamt,rate;
    int time;
    Compound()
    {
        pamt=rate=time=0;
    }
}

```

```

void input()
{
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter the Principal:");
    pamt=sc.nextDouble();
    System.out.println("Enter the Rate:");
    rate=sc.nextDouble();
    System.out.println("Enter the Time:");
    time=sc.nextInt();
}

double findInterest()
{
    double ci;
    ci=pamt*Math.pow(1+rate/100,time)-pamt;
    return ci;
}

void printData()
{
    System.out.println("Principal:"+pamt);
    System.out.println("Rate:"+rate);
    System.out.println("Time:"+time);
}

public static void main(String args[])
{
    Compound ob=new Compound();
    ob.input();
    ob.printData();
    System.out.println("Compound Interest="+ob.findInterest());
}
}

```

**13. Define a class PhoneBill with the following descriptions:**

**Data members:**

|                     |                                |
|---------------------|--------------------------------|
| <b>customerName</b> | <b>of type character array</b> |
| <b>phoneNumber</b>  | <b>of type long</b>            |
| <b>no_of_units</b>  | <b>of type int</b>             |
| <b>rent</b>         | <b>of type int</b>             |
| <b>amount</b>       | <b>of type float.</b>          |

**Member Functions:**

**i) calculate( ) This member function should calculate the value of amount as rent+ cost for the units.**

**Where cost for the units can be calculated according to the following conditions:**

|                       |             |
|-----------------------|-------------|
| <b>No_of_units</b>    | <b>Cost</b> |
| <b>First 50 calls</b> | <b>Free</b> |

Next 100 calls            0.80 @ unit  
Next 200 calls            1.00 @ unit  
Remaining calls           1.20 @ unit

ii) A constructor to assign initial values of customerName as "Raju", phoneNumber as 259461, no\_of\_units as 50, rent as 100, amount as 100.

iii) A function accept() which allows user to enter customerName, phoneNumber, no\_of\_units and rent and should call function calculate().

iv) A function display() to display the values of all the data members on the screen.

**Ans.** package LabExercises;

//Question 13

import java.util.\*;

class PhoneBill

```
{
    String customerName;
    long phoneNumber;
    int no_of_units;
    int rent;
    float amount;

    void calculate()
    {
        float cost;
        if(no_of_units<=50)
            cost=0;
        else if(no_of_units<=150)
            cost=50*0+(no_of_units)*0.80f;
        else if(no_of_units<=200)
            cost=50*0+100*0.80f+(no_of_units-150)*1.00f;
        else
            cost=50*0+100*0.80f+200*1.00f+(no_of_units-350)*1.20f;
        amount=rent + cost;
    }

    PhoneBill()
    {
        customerName="Raju";
        phoneNumber=259461;
        no_of_units=50;
        rent=100;
        amount=100;
    }

    void accept()
    {
        Scanner sc=new Scanner(System.in);
```



```

        System.out.println("Enter the Customer name:");
        customerName=sc.nextLine();
        System.out.println("Enter the Phone Number:");
        phoneNumber=sc.nextLong();
        System.out.println("Enter number of units consumed:");
        no_of_units=sc.nextInt();
        System.out.println("Enter rent:");
        rent=sc.nextInt();
        calculate();
    }

    void display()
    {
        System.out.println("Customer Name:"+customerName);
        System.out.println("Phone Number:"+phoneNumber);
        System.out.println("Number of units consumed:"+no_of_units);
        System.out.println("Rent:"+rent);
        System.out.println("Amount:"+amount);
    }
}

```

**14. Define a class Sports with following descriptions:**

**Data members:**

**s\_code** of type long

**s\_name** of type (String)

**fees** of type integer

**duration** of type integer

**Member Functions:**

i) Constructor to assign initial values of s\_code as 1001, s\_name as "Cricket", fees as 500, duration as 70.

ii) A function newSports( ) which allows user to enter s\_code, s\_name and duration.

Also assign the values to fees as per the following conditions:

| s_name       | Fees |
|--------------|------|
| Table Tennis | 2000 |
| Swimming     | 4000 |
| Football     | 3000 |

iii) A function displaySports( ) to display all the details.

**Ans.** //Question 14

```

import java.util.*;
class Sports
{
    long s_code;
    String s_name;
    int fees;
    int duration;
}

```

```

Sports()
{
    s_code=1001;
    s_name="Cricket";
    fees=500;
    duration=70;
}

void newSports()
{
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter the Sports code:");
    s_code=sc.nextLong();
    sc.nextLine();//dummy input
    System.out.println("Enter the Sports name:");
    s_name=sc.nextLine();
    System.out.println("Enter the duration:");
    duration=sc.nextInt();
    if(s_name.equals("Table Tennis"))
        fees=2000;
    else if(s_name.equals("Swimming"))
        fees=4000;
    else if(s_name.equals("Football"))
        fees=3000;
}

void displaySports()
{
    System.out.println("Sports Code:"+s_code);
    System.out.println("Sports Name:"+s_name);
    System.out.println("Duration:"+duration);
    System.out.println("Fees:"+fees);
}
}

```

**15. Create a class called GeneratePrime which will be used to generate 'n' number of prime numbers.**

**The class should have the following members:**

**Data Members: n of int data type.**

**Member Functions:**

- i) Parameterized constructor to initialize the value of n.**
- ii) Method called isPrime() which accepts an integer as a parameter and return true if it is a prime number otherwise return false.**
- iii) Method called display() which displays the first n prime numbers by calling the above function.**

**Ans. //Question 15**

```

import java.util.*;
class GeneratePrime
{
    int n;
    GeneratePrime(int x)
    {
        n=x;
    }

    boolean isPrime(int p)
    {
        int i,c=0;
        for(i=1;i<=p;i++)
        {
            if(p%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    void display()
    {
        int i=0,p=2;
        while(i<n)
        {
            if(isPrime(p))
            {
                System.out.print(p+" ");
                i++;
            }
            p++;
        }
    }
}

```

**16.** Create a class called PrimeDigits to find the sum of the prime digits in an integer. The class should have the following members.

**Data Members:** n of int data type.

**Member Functions:**

**i) Parameterized constructor to initialize the value of n.**

**ii) Method called isPrime() which accepts an integer as a parameter and return true if it is a prime number otherwise return false.**

**Static Method called sumOfPrimeDigits() which accepts an integer as a parameter and find the sum of prime digits only.**

**Ans.** //Question 16

```
import java.util.*;
class PrimeDigits
{
    int n;
    PrimeDigits(int x)
    {
        n=x;
    }

    boolean isPrime(int p)
    {
        int i,c=0;
        for(i=1;i<=p;i++)
        {
            if(p%i==0)
                c++;
        }
        if(c==2)
            return true;
        else
            return false;
    }

    static void sumOfPrimeDigits(int m)
    {
        PrimeDigits ob=new PrimeDigits(m);
        int t=ob.n,d,s=0;
        while(t!=0)
        {
            d=t%10;
            if(ob.isPrime(d))
                s=s+d;
            t=t/10;
        }
        System.out.println("Sum of Prime Digits="+s);
    }
}
```

**17. Create a class called Series which will contain the following members:**

**Data Members:**        x of double type

                          n of int type

**Member Functions:**

**i) Parameterized constructor to initialize the data members.**

ii) Which accepts an integer as parameter and return its factorial.

iii) To calculate and print the sum of the following series:

$$x + x/2! + x/3! + x/4! + \dots + x/n!$$

iv) To calculate and print the sum of the following series:

$$x/2! + x^2/3! + x^3/4! + x^4/5! + \dots + x^n/(n+1)!$$

v) To calculate and print the sum of the following series:

$$x/2! - x^2/3! + x^3/4! - x^4/5! + \dots \pm x^n/(n+1)!$$

where the symbol ! stands for factorial, e.g.  $5! = 5*4*3*2*1$ ,  $3! = 3*2*1$

**Ans.** //Question 17

```
import java.util.*;
```

```
class Series
```

```
{
```

```
    double x;
```

```
    int n;
```

```
    Series(double xx,int nn)
```

```
    {
```

```
        x=xx;
```

```
        n=nn;
```

```
    }
```

```
    long factorial(int p)
```

```
    {
```

```
        int i;
```

```
        long f=1;
```

```
        for(i=1;i<=p;i++)
```

```
        {
```

```
            f=f*i;
```

```
        }
```

```
        return f;
```

```
    }
```

```
    void sumOfseries1()
```

```
    {
```

```
        int i;
```

```
        long f;
```

```
        double s=0;
```

```
        for(i=1;i<=n;i++)
```

```
        {
```

```
            f=factorial(i);
```

```
            s=s+x/f;
```

```
        }
```

```
        System.out.println("Sum of the Series="+s);
```

```
    }
```

```
    void sumOfseries2()
```

```

{
    int i;
    long f;
    double s=0;
    for(i=1;i<=n;i++)
    {
        f=factorial(i+1);
        s=s+x/f;
    }
    System.out.println("Sum of the Series="+s);
}
void sumOfseries3()
{
    int i;
    long f;
    double s=0;
    for(i=1;i<=n;i++)
    {
        f=factorial(i+1);
        if(i%2==0)
            s=s-x/f;
        else
            s=s+x/f;
    }
    System.out.println("Sum of the Series="+s);
}
}

```

**18.** Create a class named **Rounding** which contains the following members:

**Data members:** n and r of double type

**Member Functions:**

i) Constructor to read a real number from the keyboard into n.

ii) to round off n to two places of decimal and store it in r.

iii) to display the value of the data members.

**Ans.** //Question 18

```

import java.util.*;
class Rounding
{
    double n,r;
    Rounding()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter a real number:");
        n=sc.nextDouble();
    }
}

```

```
void roundOff2()
{
    double t=n*100;
    r=Math.round(t)/100.0;
}

void display()
{
    System.out.println("Original Number="+n);
    System.out.println("Rounded Off to 2 places of decimal="+r);
}
}
```

# Chapter 12: String Manipulation

## A. Answer the following questions:

### 1. What is a String? Explain its use.

**Ans.** The **String** data type is a class defined in the **java.lang** package and every variable that you create of **String** data type is actually an object or object reference of the **String** class.

Using **String** variable in Java it is possible to manipulate strings i.e. a group of characters, by extracting characters and manipulate it.

### 2. Why is String considered to be immutable?

**Ans.** String is considered to be immutable as it is not possible to insert or remove characters from it. Moreover every time a new value is assigned to a String variable it occupies a different memory location as compared to the previous allocation.

### 3. What are the common methods of initializing String objects?

**Ans.** A string may be initialised either using a String reference. For example,  
`String s="COMPUTER";`

Or initialised using a constructor while creating a String object. For example,  
`String s=new String("COMPUTER");`

### 4. What is concatenation? Give an example to show how '+' operator is used to perform concatenation.

**Ans.** Joining of two or more strings to form a single string is termed as concatenation. For example,

```
String a="AB", b="CD",c;  
c=a+b;
```

Thus the value of the variable c is "ABCD".

### 5. Explain, with the help of an example, the functionality of toString().

**Ans.** The `toString()` function in Java is used to convert a value of any data type to String. For example, the following statements convert a value (say 45) stored in an integer variable to a String.

```
int a=45;//45 is stored in int type variable  
String s=Integer.toString(a); // Content of variable a is converted to String.
```

### 6. Give the code to extract the first and the last character of a string in Java.

**Ans.** In case the String is stored in a variable named 's'.

To extract the first character use: `s.charAt(0)`

To extract the last character use: `s.charAt(s.length()-1)`

### 7. State the use of the following String member functions, along with examples:

#### i) substring()

#### ii) replace()

#### iii) trim()

**Ans. i) substring()** : It is used a part of a string from a given string . It has two syntaxes:

- `substring(s,e)`: Extracts characters from s to e-1(s and e are indexes and should be of int type). For example, "INDIA".`substring(1,4)` will give "NDI".



b) `substring(s)`: Extracts characters from `s` till the end of the string (`s` is an index and should be of int type). For example, `"Laserjet".substring(5)` will yield "jet".

ii) `replace()` : It is used to replace a character/string from a string with another character/string. Its general syntax is:  
`replace(c,d)`: It is used to replace the character `c` with character `d`. For example, `"BEETEL".replace('E','O')` will yield "BOOTOL"  
 iii) `trim()` : It is used to remove leading and trailing spaces in a string. For example, `"COMPUTER ".trim()` will yield "COMPUTER"

**8. Give the difference between the following functions:**

- i) `==` and `equals()` [ICSE 2008]
- ii) `toLowerCase()` and `toUpperCase()` [ICSE 2005]
- iii) `startsWith()` and `endsWith()`
- iv) `indexOf()` and `lastIndexOf()` [ICSE 2010]

Ans. i)

| <code>==</code>                                                                                        | <code>equals()</code>                                 |
|--------------------------------------------------------------------------------------------------------|-------------------------------------------------------|
| Checks whether two strings belong to the same memory location or not.                                  | Checks whether two string quantities are same or not. |
| Checks whether two quantities are equal or not. The quantities may be any data type other than string. | Checks only two string quantities are same or not.    |

ii)

| <code>toLowerCase()</code>                            | <code>toUpperCase()</code>                            |
|-------------------------------------------------------|-------------------------------------------------------|
| Converts a given character or a string to lower case. | Converts a given character or a string to upper case. |

iii)

| <code>startsWith()</code>                                                                                         | <code>endsWith()</code>                                                                                         |
|-------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| Checks whether a string begins with a certain character or a string or not and returns true or false accordingly. | Checks whether a string ends with a certain character or a string or not and returns true or false accordingly. |

iv)

| <code>indexOf()</code>                                                                                                                                                              | <code>lastIndexOf()</code>                                                                                                                                                          |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Searches for a certain character or string from left to right in a given string and return the index for the first occurrence of the found character or string otherwise return -1. | Searches for a certain character or string from right to left in a given string and return the index for the first occurrence of the found character or string otherwise return -1. |

**9. Explain `compareTo()` function, with respect to the values that it returns.**

[ICSE 2010]

Ans. The `compareTo()` function is used to lexicographically check two strings where the Unicode differs and accordingly return their difference.

The value returned by the `compareTo()` function is the difference between the corresponding Unicodes of the characters, where it differs. If at no point the character differs in the strings a 0 is returned.

For example,

```
System.out.println("abc".compareTo("abde"));
```

Will display `-1`, as the strings differ from the character at position 2 (i.e. 'c' and 'd'). The Unicode for 'c' is 99 and 'd' is 100 and therefore their difference is `-1`.

```
System.out.println("COMPUTRONICS".compareTo("COMPUTER"));
```

Will display `13`, as the pair of strings differ from the character at position 6 (i.e. 'R' and 'E'). The Unicode for 'R' is 82 and 'E' is 69 and therefore their difference is `13`.

If there is no index position at which they differ, then the shorter string lexicographically precedes the longer string. In this case, it returns the difference of the lengths of the strings.

For example,

```
System.out.println("AB".compareTo("ABCD"));
```

will display `-2` as `"AB".length()-"ABCD".length()=2-4=-2`

**10. Differentiate between `compareTo()` and `equals()` methods. [ICSE 2006]**

**Ans.**

| <code>compareTo()</code>                                                             | <code>equals()</code>             |
|--------------------------------------------------------------------------------------|-----------------------------------|
| Checks two strings lexicographically.                                                | Checks two strings for equality.  |
| Returns the first difference between the Unicode of the characters where it differs. | Returns either a true or a false. |
| Return type is int.                                                                  | Return type is Boolean.           |

**11. What is Unicode? How can you get the unicode value of a Java character?**

**Ans.** Unicode is a universal international standard character encoding that is capable of representing most of the world's written languages. Using this system of encoding it is possible to represent 65,536 characters.

To get the Unicode value of a Java character simply type-cast it with int. For example,

```
char x='B';
```

```
int p=(int)x; //It will assign the Unicode of the character x into p.
```

**12. Explain what arithmetic operation can be performed with Java characters.**

**Ans.** You can use increment or decrement operator with character variables to get the previous or next character. For example, if

```
char x='B';
```

```
char y='d';
```

```
x++;
```

```
y--;
```

`x++` will make the character x to 'C' and will make the character y to 'c'.

**B. Answer as directed:**

**1. Give the output of the following program fragment:**

```
String s=new String("He went to the market");
```

```
String r;
```

```
r=s.replace("went","is going");
```

```
System.out.println(r);
```

**Ans.** He is going to the market

2. Give the output of the following program fragment:

```
String s="String";
int a=12,b=45;
System.out.println(s+a+b);
System.out.println(a+s+b);
System.out.println(a+b+s);
```

**Ans.** String1245  
12String45  
57String

3. Give the output of the following program fragment:

```
String s="india",s1="IndIA",s2=s;
System.out.println(s.equals(s1));
System.out.println(s.equalsIgnoreCase(s1));
System.out.println(s2==s);
System.out.println(s.toUpperCase( )==s1.toUpperCase( ));
System.out.println(s.startsWith("IN".toLowerCase( )));
System.out.println(s1.endsWith("iA".toUpperCase( )));
```

**Ans.** false  
true  
true  
false  
true  
true

4. What do the following functions return for:

[ICSE 2005]

```
String x ="hello";
String y = "world"
System.out.println(x + y);
System.out.println(x.length( ));
System.out.println(x.charAt(3));
System.out.println(x.equals(y));
```

**Ans.** helloworld  
5  
l  
false

5. What is the output of the following:

[ICSE 2007]

```
i) System.out.println ("four :" + 4 + 2);
System.out.println (" four : "+(2+2));
ii) String S1 = "Hi";
String S2 = "Hi";
String S3 = "there";
String S4 = "HI";
System.out.println(S1 + "equals" + S2 + "→" + S1.equals(S2));
System.out.println(S1 + "equals" + S3 + "→" + S1.equals(S3));
System.out.println(S1 + "equals" + S4 + "→" + S1.equals(S4));
System.out.println(S1 + "equalsIgnoreCase" +S4 + "→" +
S1.equalsIgnoreCase(S4));
```

**Ans.** i) four :42

four : 4

- ii) HiequalsHi→true  
Hiequalsthere→false  
HiequalsHI→false  
HiequalsIgnoreCaseHI→true

**6. If, String x = "Computer"; [ICSE 2008]  
String y = "Applications";**

**What do the following functions return for?**

- i) `System.out.println (x.substring(1,5));`
- ii) `System.out.println (x.indexOf(x.charAt(4)));`
- iii) `System.out.println (y+x.substring(5));`
- iv) `System.out.println (x.equals(y));`

**Ans.** ompu  
4  
Applicationster  
false

**7. What will be the output for the following program segment? [ICSE 2006]**

`String s = new String("abc");  
System.out.println(s.toUpperCase( ) );`

**Ans.** ABC

**8. What will be the output of the following code? [ICSE 2010]**

`char x = 'A' ; int m;  
m=(x= 'a') ? 'A' : 'a';  
System.out.println("m="+m);`

**Ans.** m=97

**9. Write a program to count the number of non-blank characters in a given sentence.**

**Ans.** `import java.util.*;`

```
class Question9
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s;
        char x;
        int i,c=0;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                c++;
        }
        System.out.print("No. of non-blank characters:"+c);
    }
}
```

**10. Write a method to accept a word and print it in the following way:**

**Parameter: TRIAL**

**Output:**  
L  
AL  
IAL  
RIAL  
TRIAL

**Ans.** class Question10

```
{
    static void pattern(String s)
    {
        int i,j;
        char x;
        for(i=s.length()-1;i>=0;i--)
        {
            for(j=i;j<=s.length()-1;j++)
            {
                x=s.charAt(j);
                System.out.print(x);
            }
            System.out.println();
        }
    }
}
```

**11. Write a program to count the number of words in a given sentence. Assume that there may be any number of blank spaces between the words.**

**Ans.** import java.util.\*;

class Question11

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s;
        int i,c=1;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s+" ";
        for(i=0;i<s.length()-1;i++)
        {
            if(s.charAt(i)==' ' && s.charAt(i+1)!=' ')
                c++;
        }
        System.out.print("No. of words:"+c);
    }
}
```

**12. Write a method to accept a name as parameter and print its initials.**

**Example,**

**Parameter-> AJAY PRATAP SINGH**

**Output-> A.P.S.**

**Ans.** import java.util.\*;

class Question12

```
{
    static void main(String s)
    {
        s=s.trim();
        s=" "+s;
        String w="";
        int i;
        for(i=0;i<s.length()-1;i++)
        {
            if(s.charAt(i)==' ' && s.charAt(i+1)!=' ')
                w=w+s.charAt(i+1)+".";
        }
        System.out.print(w);
    }
}
```

**13. Write a program to print the first and last word in a given sentence.**

**Ans.** import java.util.\*;

class Question13

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,first,last;
        int i;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        i=s.indexOf(" ");
        first=s.substring(0,i);
        i=s.lastIndexOf(" ");
        last=s.substring(i+1);
        System.out.println("First word:"+first);
        System.out.println("Last word:"+last);
    }
}
```

**14. Write a method to accept a name as parameter and print the initial first and then the title.**

**Example,**

**Parameter: AJAY PRATAP SINGH RATHORE**

**Output: A.P.S. RATHORE**

```
Ans. import java.util.*;
class Question14
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,w="",last;
        int i,l;
        char x,y;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s.toUpperCase();
        s=" "+s;
        l=s.lastIndexOf(' ');
        last=s.substring(l+1);
        for(i=0;i<l;i++)
        {
            if(s.charAt(i)==' ' && s.charAt(i+1)!=' ')
                w=w+s.charAt(i+1)+". ";
        }
        w=w+last;
        System.out.println(w);
    }
}
```

**15. Write a program to input a string and print out the text with the uppercase and lowercase letters reversed, but all other characters should remain the same as before. [ICSE 2008]**

**Example :**

**INPUT : WelCome TO School**

**OUTPUT : wELcOMe to sCHOOL**

```
Ans. import java.util.*;
```

```
class Question15
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,rs="";
        int i;
        char x;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
```

```

        if(x>='A' && x<='Z')
            x=(char)(x+32);
        else if(x>='a' && x<='z')
            x=(char)(x-32);
        rs=rs+x;
    }
    System.out.println("Reversed Sentence:"+rs);
}
}

```

**16. Write a program to input a sentence and change it to its Title case. Title case means to change the first alphabet of every word to capital and rest to small letters.**

**Ans.** import java.util.\*;  
class Question16  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 String s,t="",w="";  
 int i;  
 char x,y;  
 System.out.print("Enter a sentence:");  
 s=sc.nextLine();  
 s=s.trim();  
 s=s.toLowerCase();  
 s=s+" ";  
 for(i=0;i<s.length();i++)  
 {  
 x=s.charAt(i);  
 if(x!=' ')  
 w=w+x;  
 else  
 {  
 y=w.charAt(0);  
 y=Character.toUpperCase(y);  
 w=y+w.substring(1);  
 t=t+w+" ";  
 w="";  
 }  
 }  
 System.out.println(t);  
 }  
}

**17. Write a program to input a sentence and display only those words which begin with alphabet 'A' or 'a'.**

**Ans.** import java.util.\*;  
class Question17



```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,t="",w="";
        int i;
        char x,y;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s.toLowerCase();
        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                y=w.charAt(0);
                if(y=='a' || y=='A')
                    t=t+w+" ";
                w="";
            }
        }
        System.out.println(t);
    }
}

```

**18. Write a program to input a sentence and display only those words which end with alphabet 'A' or 'a'.**

**Ans.** import java.util.\*;  
class Question18

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,t="",w="";
        int i;
        char x,y;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s.toLowerCase();
        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);

```

```

        if(x!=' ')
            w=w+x;
        else
        {
            y=w.charAt(w.length()-1);
            if(y=='a' || y=='A')
                t=t+w+" ";
            w="";
        }
    }
    System.out.println(t);
}
}

```

**19. Write a program to input a sentence and display only those words which begin with a vowel.**

**Ans.** import java.util.\*;

class Question19

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,t="",w="";
        int i;
        char x,y;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s.toLowerCase();
        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                y=w.charAt(0);
                if(y=='a' || y=='A' || y=='e' || y=='E' || y=='i' ||
                    y=='I' || y=='o' || y=='O' || y=='u' || y=='U')
                    t=t+w+" ";
                w="";
            }
        }
        System.out.println(t);
    }
}
}

```

**20. Write a program to input a sentence and display only those words which begin and ends with the same alphabet.**

```
Ans. import java.util.*;
class Question20
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,t="",w="";
        int i;
        char x,y,z;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s.toLowerCase();
        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                y=w.charAt(0);
                z=w.charAt(w.length()-1);
                if(y==z)
                    t=t+w+" ";
                w="";
            }
        }
        System.out.println(t);
    }
}
```

**21. Write a program to input a sentence and print the longest word, assume that there is only one longest word in the given sentence.**

```
Ans. import java.util.*;
class Question21
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,lng="",w="";
        int i,f=0;
        char x;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
```

```

s=s+" ";
for(i=0;i<s.length();i++)
{
    x=s.charAt(i);
    if(x!=' ')
        w=w+x;
    else
    {
        if(f==0)
        {
            lng=w;
            f=1;
        }
        if(w.length()>lng.length())
            lng=w;
        w="";
    }
}
System.out.println("Longest Word="+lng);
}
}

```

**22. Write a program to print the smallest word in a given sentence, assume that there is only one smallest word in the given sentence.**

**Ans.** import java.util.\*;  
class Question22  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 String s,sma="",w="";  
 int i,f=0;  
 char x;  
 System.out.print("Enter a sentence:");  
 s=sc.nextLine();  
 s=s.trim();  
 s=s+" ";  
 for(i=0;i<s.length();i++)  
 {  
 x=s.charAt(i);  
 if(x!=' ')  
 w=w+x;  
 else  
 {  
 if(f==0)  
 {  
 sma=w;  
 f=1;  
 }  
 }  
 }  
 }  
}

```

    }
    if(w.length()<sma.length())
        sma=w;
        w="";
    }
}
System.out.println("Smallest Word="+sma);
}
}

```

**23. Write a program to input a sentence and display only those words which have even number of vowels.**

**Ans.** import java.util.\*;

class Question23

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,t="",w="";
        int i,c=0;
        char x;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
            {
                w=w+x;
                if(x=='a' || x=='A' || x=='e' || x=='E' || x=='i' ||
                    x=='I' || x=='o' || x=='O' || x=='u' || x=='U')
                    c++;
            }
            else
            {
                if(c%2==0)
                    t=t+w+" ";
                w="";
                c=0;
            }
        }
        System.out.println(t);
    }
}

```

**24. Write a program to print the word with maximum number of vowels in a given sentence. Assume that there is only one word having maximum number of vowels.**

**Ans.** import java.util.\*;

class Question24

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,max="",w="";
        int i,c=0,f=0,l=0;
        char x;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
            {
                w=w+x;
                if(x=='a' || x=='A' || x=='e' || x=='E' || x=='i' ||
                    x=='I' || x=='o' || x=='O' || x=='u' || x=='U')
                    c++;
            }
            else
            {
                if(f==0)
                {
                    max=w;
                    l=c;
                    f=1;
                }
                else
                {
                    if(c>l)
                    {
                        l=c;
                        max=w;
                    }
                }
                w="";
                c=0;
            }
        }
        System.out.println("Word having maximum number of vowels:"+max);
    }
}
```

```
}  
}
```

**25. Write a program to input a sentence and print the word having least number of vowels. Assume that there is only one word having least number of vowels.**

**Ans.** import java.util.\*;

```
class Question25
```

```
{  
    static void main()  
    {  
        Scanner sc=new Scanner(System.in);  
        String s,min="",w="";  
        int i,c=0,f=0,l=0;  
        char x;  
        System.out.print("Enter a sentence:");  
        s=sc.nextLine();  
        s=s.trim();  
  
        s=s+" ";  
        for(i=0;i<s.length();i++)  
        {  
            x=s.charAt(i);  
            if(x!=' ' )  
            {  
                w=w+x;  
                if(x=='a' || x=='A' || x=='e' || x=='E' || x=='i' ||  
                    x=='I' || x=='o' || x=='O' || x=='u' || x=='U')  
                    c++;  
            }  
            else  
            {  
                if(f==0)  
                {  
                    min=w;  
                    l=c;  
                    f=1;  
                }  
                else  
                {  
                    if(c<l)  
                    {  
                        l=c;  
                        min=w;  
                    }  
                }  
                w="";  
                c=0;  
            }  
        }  
    }  
}
```

```

    }
    System.out.println("Word having minimum number of vowels:"+min);
}
}

```

**26. Write a method to accept a sentence as parameter and print the longest word with least number of vowels in it. Assume that there is only one such word in the given sentence.**

**Ans.** import java.util.\*;

class Question26

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,lng="",w="",min="";
        int i,f=0,l=0,c=0;
        char x;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();

        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                if(f==0)
                {
                    lng=w;
                    f=1;
                }
                if(w.length()>lng.length())
                    lng=w;
                w="";
            }
        }
        f=0;
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
            {
                w=w+x;
                if(x=='a' || x=='A' || x=='e' || x=='E' || x=='i' ||
                    x=='I' || x=='o' || x=='O' || x=='u' || x=='U')

```



```

        c++;
    }
    else
    {
        if(w.length()==lng.length())
        {
            if(f==0)
            {
                min=w;
                l=c;
                f=1;
            }
            else
            {
                if(c<l)
                {
                    l=c;
                    min=w;
                }
            }
        }
        w="";
        c=0;
    }
}
System.out.println("Longest Word having least number of vowels:"+min);
}
}

```

**27. Write a program to input a sentence and print only those words which have all the five vowels present in it (i.e. A, E, I, O, U), in a given sentence.**

**For example,**

**INPUT**

**Enter a sentence: EDUCATION IS A MUST FOR THE DEVELOPMENT  
OF THE COUNTRY**

**OUTPUT**

**EDUCATION**

**Ans.** import java.util.\*;

class Question27

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,t="",w="";
        int i,fa=0,fe=0,fi=0,fo=0,fu=0;
        char x;
        System.out.print("Enter a sentence:");
    }
}

```

```

s=sc.nextLine();
s=s.trim();

s=s+" ";
for(i=0;i<s.length();i++)
{
    x=s.charAt(i);
    if(x!=' ')
    {
        w=w+x;
        if(x=='a' || x=='A') fa=1;
        if(x=='e' || x=='E') fe=1;
        if(x=='i' || x=='I') fi=1;
        if(x=='o' || x=='O') fo=1;
        if(x=='u' || x=='U') fu=1;
    }
    else
    {
        if(fa==1 && fe==1 && fi==1 && fo==1 && fu==1)
            t=t+w+" ";
        w="";
        fa=fe=fi=fo=fu=0;
    }
}
System.out.println("Word having all the five vowels:"+t);
}
}

```

**28. Write a method to accept a word as parameter and print the alphabets in alphabetical order.**

**e.g. Input: world**

**Output: DLORW**

**e.g. Input: HUMAN**

**Output: AHMNU**

**Ans.** class Question28

```

{
    static void alphabetical(String w)
    {
        w=w.toUpperCase();
        int i;
        char x;
        String t="";
        for(x='A';x<='Z';x++)
        {
            for(i=0;i<w.length();i++)
            {
                if(x==w.charAt(i))

```

```

        t=t+x;
    }
}
System.out.println(t);
}
}

```

**29. Write a method to accept a word as parameter and return true if all its alphabets are unique otherwise return false. Now use it in another function to print those words whose all alphabets are unique in a given sentence.**

**Ans.** import java.util.\*;

class Question29

```

{
    static boolean isUnique(String w)
    {
        int i,j;
        boolean f=true;
        for(i=0;i<w.length();i++)
        {
            for(j=i+1;j<w.length();j++)
            {
                if (w.charAt(i)==w.charAt(j))
                    f=false;
            }
        }
        return f;
    }

    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int i;
        char x;
        String s,w="",t="";
        System.out.println("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                if(isUnique(w))
                    t=t+w+" ";
                w="";
            }
        }
    }
}

```

```

    }
    }
    System.out.println(t);
}
}

```

**30. Write a method to accept a sentence as a parameter and print the word which occurs the most. Assume that there is only one word occurring maximum number of times.**

**Ans.** import java.util.\*;

class Question30

```

{
    static void maxOccuring(String s)
    {
        s=s.trim();
        s=s+" ";
        int i,j,c=0,l=0;
        char x,y;
        String w="",m="",max="";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                c=0;
                for(j=0;j<s.length();j++)
                {
                    y=s.charAt(j);
                    if(y!=' ')
                        m=m+y;
                    else
                    {
                        if(m.equalsIgnoreCase(w))
                            c++;
                        m="";
                    }
                }
                if(c>l)
                {
                    l=c;
                    max=w;
                }
                w="";
            }
        }
        System.out.println("Word occuring maximum number of time:"+max);
    }
}

```

```
}  
}
```

**31. Write a method to accept a sentence as parameter and print the longest word at prime locations in the sentence. That is the 2nd word, 3rd word, 5th word, 7th word, etc.**

**Ans.** import java.util.\*;

```
class Question31
```

```
{  
    static boolean isPrime(int n)  
    {  
        int i,c=0;  
        for(i=1;i<=n;i++)  
        {  
            if (n%i==0)  
                c++;  
        }  
        if(c==2)  
            return true;  
        else  
            return false;  
    }  
    static void main()  
    {  
        Scanner sc=new Scanner(System.in);  
        String s,lng="",w="";  
        int i,f=0,c=0;  
        char x;  
        System.out.print("Enter a sentence:");  
        s=sc.nextLine();  
        s=s.trim();  
        s=s+" ";  
        for(i=0;i<s.length();i++)  
        {  
            x=s.charAt(i);  
            if(x!=' ')  
                w=w+x;  
            else  
            {  
                c++;  
                if(isPrime(c))  
                {  
                    if(f==0)  
                    {  
                        lng=w;  
                        f=1;  
                    }  
                    if(w.length()>lng.length())
```

```

        lng=w;
    }
    w="";
}
}
System.out.println("Longest Word="+lng);
}
}

```

**32. Write a program using a method Palin( ), to check whether a string is a palindrome or not. A palindrome is a string that reads the same from left to right and vice versa.**

[ICSE 2007]

**E.g. MADAM, ARORA, ABBA, etc.**

**Ans.** class Question32

```

{
    static void Palin(String w)
    {
        int i;
        String r="";
        for(i=0;i<w.length();i++)
        {
            r=w.charAt(i)+r;
        }
        if(w.equalsIgnoreCase(r))
            System.out.println("Palindrome");
        else
            System.out.println("Not Palindrome");
    }
}

```

**33. Write a program to input a sentence and print only the polindrome words.**

**Ans.** import java.util.\*;

class Question33

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,t="",w="",r="";
        int i;
        char x;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
            {

```

```

        w=w+x;
        r=x+r;
    }
    else
    {
        if(w.equalsIgnoreCase(r))
            t=t+w+" ";
        r=w="";
    }
}
System.out.println(t);
}
}

```

**34. Write a program to print the longest palindrome word in a given sentence, if any.**

**Ans.** import java.util.\*;

class Question34

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,lng="",w="",r="";
        int i,f=0;
        char x;
        System.out.print("Enter a sentence:");
        s=sc.nextLine();
        s=s.trim();
        s=s+" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
            {
                w=w+x;
                r=x+r;
            }
            else
            {
                if(w.equalsIgnoreCase(r))
                {
                    if(f==0)
                    {
                        lng=w;
                        f=1;
                    }
                    if(w.length()>lng.length())
                        lng=w;
                }
            }
        }
    }
}

```

```

        r=w="";
    }
}
if(f==1)
    System.out.println("Longest Palindrome Word="+lng);
else
    System.out.println("No Palindrome Words found");
}
}

```

**35. Write a program to input any given string to calculate the total number of characters and vowels present in the string and also reverse the string:**

**Example: INPUT**

Enter String: SNOWY

**OUTPUT**

Total number of characters: 05

Number of Vowels : 01

Reverse string : YWONS

[ICSE 2005]

**Ans.** import java.util.\*;

class Question35

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,r="";
        int i,c=0;
        char x;
        System.out.print("Enter String:");
        s=sc.nextLine();
        s=s.trim();
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x=='a' || x=='A' || x=='e' || x=='E' || x=='i' ||
                x=='I' || x=='o' || x=='O' || x=='u' || x=='U')
                c++;
            r=x+r;
        }
        System.out.println("Total number of characters:"+s.length());
        System.out.println("Number of Vowels:"+c);
        System.out.println("Reverse string:"+r);
    }
}

```

**36. Consider the following statement: "January 26 is celebrated as the Republic Day of India". Write a program to change "26" to "15", "January" to "August", "Republic" to "Independence" and finally print "August 15 is celebrated as the Independence Day of India".**

[ICSE 2006]

**Ans.** class Question36



```

{
static void main()
{
String s="January 26 is celebrated as the Republic Day of India";
String w="",t="";
int i,c=0;
char x;
s=s.trim();
s=s+" ";
for(i=0;i<s.length();i++)
{
x=s.charAt(i);
if(x!=' ')
w=w+x;
else
{
if(w.equalsIgnoreCase("January"))
w="August";
else if(w.equalsIgnoreCase("26"))
w="15";
else if(w.equalsIgnoreCase("Republic"))
w="Independence";
t=t+w+" ";
w="";
}
}
}
System.out.println(t);
}
}

```

**37. Write a program to enter a sentence from the keyboard and count the number of times a particular word occurs in it. Display the frequency of the search word.**

**Example:**

**INPUT:**

Enter a sentence : the quick brown fox jumps over the lazy dog.

Enter a word to be searched : the

**OUTPUT:**

Searched word occurs : 2 times.

[ICSE 2007]

**Ans.** import java.util.\*;

class Question37

```

{
static void main()
{
Scanner sc=new Scanner(System.in);
String s,ws="",w="";
int i,c=0;
char x;
System.out.print("Enter a sentence:");

```

```

s=sc.nextLine();
System.out.print("Enter a word to be searched:");
ws=sc.nextLine();
s=s.trim();
s=s+" ";
for(i=0;i<s.length();i++)
{
    x=s.charAt(i);
    if(x!=' ')
        w=w+x;
    else
    {
        if(w.equals(ws))
            c++;
        w="";
    }
}
System.out.println("Searched word occurs :"+c+" times");
}
}

```

**38. Write a program to input a sentence and print the number of characters found in the longest word of the given sentence. For example if S= "India is my country" then the output should be 7. [ICSE 2009]**

**Ans.** import java.util.\*;  
class Question38  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 String s,w="";  
 int i,l=0;  
 char x;  
 System.out.print("Enter a sentence:");  
 s=sc.nextLine();  
 s=s.trim();  
 s=s+" ";  
 for(i=0;i<s.length();i++)  
 {  
 x=s.charAt(i);  
 if(x!=' ')  
 w=w+x;  
 else  
 {  
 if(w.length(>l)  
 l=w.length();  
 w="";  
 }  
 }  
 }  
}

```

    }
    System.out.println("Length of longest word="+l);
}
}

```

**39. Design a class to overload a function num\_calc( ) as follows:**

**i) void num\_calc(int num, char ch) with one integer argument and one character argument, computes the square of integer argument if choice ch is 's' otherwise finds its cube.**

**ii) void num\_calc(int a, int b, char ch) with two integer arguments and one character argument. It computes the product of integer arguments if ch is 'p' else adds the integer.**

**iii) void num\_calc(String s1, String s2) with two string arguments, which prints whether the strings are equal or not.**

**Ans.** class Question39

```

{
    static void num_calc(int num, char ch)
    {
        if(ch=='s')
            System.out.println(num*num);
        else
            System.out.println(num*num*num);
    }

    static void num_calc(int a,int b, char ch)
    {
        if(ch=='p')
            System.out.println(a*b);
        else
            System.out.println(a+b);
    }

    static void num_calc(String s1,String s2)
    {
        if(s1.equalsIgnoreCase(s2))
            System.out.println("Equal");
        else
            System.out.println("Not Equal");
    }
}

```

**40. In a set of 50 names, it is intended to find the total number of names which contain at least one pair of consecutive letters, e.g. SURABHI. In this 'A' and 'B' are consecutive letters and 'H' and 'I' are consecutive letters. Write a program for the above situation.**

**Ans.** import java.util.\*;

class Question40

```

{
    static void main()

```

```

{
Scanner sc=new Scanner(System.in);
String s,w="";
int i,j,f,c=0;
char x,y;
System.out.println("Enter 50 names:");
for(i=0;i<50;i++)
{
s=sc.nextLine();
s=s.toLowerCase();
f=0;
for(j=0;j<s.length()-1;j++)
{
x=s.charAt(j);
y=s.charAt(j+1);
if((char)(x+1)==y)
f=1;
}
if(f==1)
c++;
}
System.out.println("Number of consecutive letter words="+c);
}
}

```

**41. Write a program to input a line of text consisting of sentences terminated by either "." or "!" or "?".**

**The words in the sentences may be separated by multiple spaces. The program should output:**

- i) The total number of words and**
- ii) The number of alphabets in the given text.**

**Ans.** import java.util.\*;

class Question41

```

{
static void main()
{
Scanner sc=new Scanner(System.in);
String s,w="";
int i,c=0,p=0;
char x,y;
System.out.println("Enter a string:");
s=sc.nextLine();
for(i=0;i<s.length();i++)
{
x=s.charAt(i);
if(x!=' ' && x!='.' && x!='!' && x!='?' )
{
w=w+x;
}
}
}
}

```

```

        if(x>='A' && x<='Z' || x>='a' && x<='z')
            p++;
    }
    else
    {
        if(w!="")
            c++;
        w="";
    }
}
System.out.println("Total number of words="+c);
System.out.println("Total number of alphabets="+p);
}
}

```

**42. Write a program to input a simple mathematical expression consisting of a binary operator and two integer operands and evaluate it. For example, if**

**INPUT: 55+11**

**OUTPUT: 66**

**INPUT: 5\*12**

**OUTPUT: 60**

**Ans.** import java.util.\*;

class Question42

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,w="";
        int i,f,l;
        float eval=0;
        char x=' ';
        System.out.println("Enter an expression:");
        s=sc.nextLine();
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x=='+' || x=='-' || x=='/' || x=='%' || x=='*')
                break;
        }
        f=Integer.parseInt(s.substring(0,i).trim());
        l=Integer.parseInt(s.substring(i+1).trim());
        switch(x)
        {
            case '*':
                eval=f*l;
                break;
            case '%':
                eval=f%l;

```

```

        break;
    case '/':
        eval=(float)f/l;
        break;
    case '+':
        eval=f+l;
        break;
    case '-':
        eval=f-l;
    }
    System.out.println(eval);
}
}

```

**43. Write a program to accept any string: Count and print the number of pairs of consecutive letters present in words in the forward direction only. Assume that all the letters in the string are in the same case, consecutive letters in two different words are not counted and 'za' or 'ZA' in any word is not a consecutive pair.**

**For example:**

**INPUT:**

**Enter String: ABSTRACT STRING IS BEING COUNTED EDUCATIONALLY.**

**OUTPUT:**

**Pairs of consecutive letters: 3**

**Ans.** import java.util.\*;

class Question43

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s,w="";
        int i,j,c=0;
        float eval=0;
        char x;
        System.out.print("Enter String:");
        s=sc.nextLine();
        s=s.trim();
        s=s+" ";
        s=s.toUpperCase();
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                for(j=0;j<w.length()-1;j++)
                {
                    if((char)(w.charAt(j)+1)==w.charAt(j+1))

```

```

        c++;
    }
    w="";
}
}
System.out.println("Pair of consecutive letters:"+c);
}
}

```

**44. Pig Latin is a language game of alterations played in English. To form the Pig Latin form of an English word the initial consonant sound is transposed to the end of the word and an "ay" is affixed (for example, trash yields ashtray and plunder yields underplay). Write a program to input a word and change it to Pig Latin.**

**Ans.** import java.util.\*;  
class Question44  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 String w,pigLatin;  
 int i;  
 char x;  
 System.out.print("Enter a word:");  
 w=sc.nextLine();  
 w=w.trim();  
 w=w.toUpperCase();  
 for(i=0;i<w.length();i++)  
 {  
 x=w.charAt(i);  
 if(x=='A' || x=='E' || x=='I' || x=='O' || x=='U')  
 break;  
 }  
 pigLatin=w.substring(i)+w.substring(0,i)+"AY";  
 System.out.println("Pig Latin:"+pigLatin);  
 }  
}

**45. An anagram is a word or a phrase made by transposing the letters of another word or phrase; for example, "parliament" is an anagram of "partial men," and "software" is an anagram of "swear oft". Write a program that figures out whether one string is an anagram of another string. The program should ignore white space and punctuation.**

**Ans.** import java.util.\*;  
class Question45  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 String s1,s2;

```

int i,p,j,c,f=0;
char x,y;
System.out.print("Enter a string:");
s1=sc.nextLine();
System.out.print("Enter another string:");
s2=sc.nextLine();
s1=s1.trim();
s2=s2.trim();
s1=s1.toUpperCase();
s2=s2.toUpperCase();
for(i=0;i<s1.length();i++)
{
    x=s1.charAt(i);
    if(x>='A' && x<='Z')
    {
        c=0;
        //Check the number of times it is present in the same string
        for(j=0;j<s1.length();j++)
        {
            y=s1.charAt(j);
            if(x==y)
                c++;
        }
        p=0;
        //Check the number of times it is present in another string
        for(j=0;j<s2.length();j++)
        {
            y=s2.charAt(j);
            if(x==y)
                p++;
        }
        if(c!=p)
            f=1;
    }
}
if(f==0)
    System.out.println("ANAGRAM");
else
    System.out.println("NOT ANAGRAM");
}
}

```

**46. Write a program to input a string in upper case and print the frequency of each character.** **[ICSE 2010]**

**INPUT: COMPUTER HARDWARE**

**OUTPUT:**

**CHARACTERS**

**FREQUENCY**

**A**

**2**



|   |   |
|---|---|
| C | 1 |
| D | 1 |
| E | 2 |
| H | 1 |
| M | 1 |
| P | 1 |
| R | 3 |
| T | 1 |
| U | 1 |
| W | 1 |

```

Ans. import java.util.*;
class Question46
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String s;
        int i,c;
        char x,y;
        System.out.print("Enter a string:");
        s=sc.nextLine();
        s=s.toUpperCase();
        System.out.println("CHARACTERS\t\tHARDWARE");
        for(x='A';x<='Z';x++)
        {
            c=0;
            for(i=0;i<s.length();i++)
            {
                y=s.charAt(i);
                if(x==y)
                    c++;
            }
            if(c>0)
                System.out.println(x+"\t\t"+c);
        }
    }
}

```

**47. Write a program to input two words and find out the set that will be formed by the intersection of the characters found in both the strings.**

**INPUT:**

Enter the first word: TOPPLE

Enter the second word: CRIPPLE

**OUTPUT:**

**Intersecting Set: PLE**

```

Ans. import java.util.*;
class Question47
{
    static void main()

```

```

{
Scanner sc=new Scanner(System.in);
String w1,w2,t="";
int i,j,f;
char x,y;
System.out.print("Enter the first word:");
w1=sc.nextLine();
w1=w1.toUpperCase();
System.out.print("Enter the second word:");
w2=sc.nextLine();
w2=w2.toUpperCase();
for(i=0;i<w1.length();i++)
{
f=0;
for(j=0;j<i;j++)
{
if(w1.charAt(i)==w1.charAt(j))
f=1;
}
if(f==0)
{
for(j=0;j<w2.length();j++)
{
if(w1.charAt(i)==w2.charAt(j))
f=1;
}
if(f==1)
t=t+w1.charAt(i);
}
}
System.out.println("Intersecting Set:"+t);
}
}

```

**48. Write a program to input two words and find out the set that will be formed by the union of the characters found in both the strings.**

**INPUT:**

Enter the first word: TOPPLE

Enter the second word: CRIPPLE

**OUTPUT:**

Union Set: TOPLECRI

**Ans.** import java.util.\*;

class Question48

```

{
static void main()
{
Scanner sc=new Scanner(System.in);
String w1,w2,t="",con="";

```

```

int i,j,f;
char x,y;
System.out.print("Enter the first word:");
w1=sc.nextLine();
w1=w1.toUpperCase();
System.out.print("Enter the second word:");
w2=sc.nextLine();
w2=w2.toUpperCase();
con=w1+w2;
for(i=0;i<con.length();i++)
{
    f=0;
    for(j=0;j<i;j++)
    {
        if(con.charAt(i)==con.charAt(j))
            f=1;
    }
    if(f==0)
        t=t+con.charAt(i);
}
System.out.println("Union Set:"+t);
}
}

```

**49. Write a program to input an integer of the Decimal Number System and convert it to corresponding Hexadecimal.**

**INPUT:**

Enter an integer of the Decimal Number System: 2578

**OUTPUT:**

Equivalent Hexadecimal Number: A12

**Ans.** import java.util.\*;

class Question49

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String hex="";
        int n,d;
        System.out.print("Enter an integer of the Decimal Number System:");
        n=sc.nextInt();
        while(n!=0)
        {
            d=n%16;
            if(d<=9)
                hex=d+hex;
            else
                hex=(char)(d+55)+hex;
            n=n/16;
        }
    }
}

```

```

    }
    System.out.println("Union Set:"+hex);
}
}

```

**50. Write a program to input an integer of the Hexadecimal Number System and convert it to a number of Decimal Number System.**

**INPUT:**

Enter a hexadecimal number: 12BC

**OUTPUT:**

Decimal Number: 4796

**Ans.** import java.util.\*;

class Question50

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String hex="";
        int i,d=0,dec=0,c=0;
        char x;
        System.out.print("Enter a Hexadecimal Number:");
        hex=sc.nextLine();
        for(i=hex.length()-1;i>=0;i--)
        {
            x=hex.charAt(i);
            if(x>='0' && x<='9')
                d=x-48;
            else if(x>='A' && x<='Z')
                d=x-55;
            dec=dec+d*(int)Math.pow(16,c++);
        }
        System.out.println("Decimal Number:"+dec);
    }
}
}

```

# Chapter 13:

# Arrays

## A. Answer the following questions:

### 1. What are arrays?

**Ans.** An array is a homogeneous collection of data, referred by a name and occupying contiguous memory space.

### 2. How do you declare single or 2-dimensional arrays?

**Ans.** To declare a single dimensional array the following syntax is used:

The general syntax is,

```
<data type> <array-name>[ ];
```

or,

```
<data type>[ ] <array-name>;
```

For example, to declare an integer array use,

```
int a[ ];
```

```
int[ ] a;
```

### 3. How do you create memory spaces for arrays?

**Ans.** To allocate memory space for a declared array the following general syntax is used.

```
<array-name>=new<data-type>[<size>;
```

For example, to allocate 10 memory spaces for the integer array use,

```
a=new int[10];
```

### 4. How do you initialize single dimensional arrays. Explain with examples.

**Ans.** To initialise a single dimensional array the following general syntax is used:

```
<data-type> < array-name>[ ]={ <list of values >};
```

For example, to initialise an integer array you may use,

```
int a[ ]={1,2,3,4,5};
```

### 5. With the help of an example, explain how arrays of objects are used in a Java program.

**Ans.** The declaration for an array of objects uses the following general syntax,

```
<class-name> <array-name>[ ]=new <class-name>[<size>;
```

Or,

```
<class-name>[ ] <array-name>=new <class-name>[<size>;
```

Or,

```
<class-name> <array-name>[ ];
```

```
<array-name>=new <class-name>[<size>;
```

For example, to declare an object of size 10 for a class named Cuboid, you may use,

```
Cuboid obj[ ]=new Cuboid[10];
```

or,

```
Cuboid[ ] obj=new Cuboid[10];
```

or,

```
Cuboid obj[ ];
```

```
obj=new Cuboid[10];
```

### 6. With the help of an example, show how arrays in Java are dynamic in nature.

**Ans.** Arrays in Java are always dynamic in nature in the sense that the size of an array is specified at run-time. Here is an example, that takes an integer(say n) as input using Scanner and then declares an integer array of size n.

```
Scanner sc=new Scanner(System.in);
int n;
System.out.println("Enter the value of n:");
n=sc.nextInt( );
int a[ ]=new int [n];
```

Note that the array 'a' of size n is declared at runtime.

**7. "Array variables are references to a block of elements". What do you understand by this statement?**

**Ans.** An array variable is always a reference to a block of memory and therefore can be used to refer to a memory location which is referred by another array variable.

For example,

```
int[ ] a = { 100, 99, 98}; // "a" references the array object.
```

```
int[ ] b; // "b" doesn't refer to anything.
```

```
b = a; // Now "b" refers to the SAME array as "a"
```

```
b[1] = 0; // Also changes a[1] to 0 because a and b refer to the same array.
```

**8. How do you determine the length of an array?**

**Ans.** Length of an array refers to the number of elements in the array. To determine the length of an array the following general syntax is used:

```
<array-name>.length( );
```

**9. With the help of an example show how arrays are passed to a function as Call by Reference.**

**Ans.** An array when passed to a function, it is always passed as a reference, therefore any changes made to the formal parameter is reflected by to the actual parameter.

The following example shows how this is performed:

```
class ArrayDemo
{
    /*Method to double the value of each element in the
    array.*/
    static void doubleIt(int b[ ])
    {
        for(int i=0;i<b;i++)
        {
            b[i]=2*b[i];
        }
    }
    public static void main(String args[ ])
    {
        int a[ ]={4,5,12,7,8,3};
        System.out.println("Original Array...");
        for(int i=0;i<6;i++)
            System.out.print(a[i]+"\\t");
        doubleIt(a); //this is how you pass an array as
        parameter.
        System.out.println("Updated Array...");
        for(int i=0;i<6;i++)
```

```

        System.out.print(a[i]+"\\t");
    }
}

```

Output when main( ) is executed will be,

Original Array...

4 5 12 7 8 3

Updated Array...

8 10 24 14 16 6

**10. Explain:**

**i) Linear Search ii) Binary Search**

**Ans. i) Linear Search:** Linear Search is a search process technique that involves checking each element sequentially with the given element to be searched for.

ii) **Binary Search:** Binary search is a search process technique where an element to be searched is checked with the central element of the array and depending upon its value, it is checked in the sub-array before the central position or in the sub-array after the central position. The same search process continues even with the sub-array.

**11. Compare Linear Sorting, Selection Sorting and Bubble Sorting.**

**Ans. Linear Sorting:** The linear sorting process checks each element of the array with the remaining elements of the array after it. In case the array element is greater than either of the remaining elements, the elements are interchanged. The process continues until all elements are exhausted.

**Bubble Sorting:** The bubble sorting process checks adjacent elements of the array, i.e. first element with the second element, the second element with the third element and the process continues until all elements are exhausted. In case the array element is greater than the adjacent element, the elements are interchanged. The process continues until all elements are exhausted.

**Selection Sort:** During the selection sort process, an element is taken from the first and from the remaining elements the smallest or largest number (for ascending or descending order respectively) is found and interchanged.

**12. Differentiate between Binary Search and Linear Search. [ICSE 2007]**

**Ans.**

| Linear search                                                           | Binary Search                                                                                                                                                                                                                      |
|-------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Works with both sorted and unsorted arrays.                             | Works only with sorted arrays.                                                                                                                                                                                                     |
| Can have repetition of elements if index is required                    | All elements should be unique if index is required.                                                                                                                                                                                |
| Search the entire array in sequential manner, i.e. one after the other. | Search the array by seeking the middle element and depending upon it, it will be checked in the sub-arrays before or after it. The same process continues with the sub-arrays until a match is found or the element isn't present. |

**13. Which element is num[9] of the array num? [ICSE 2005]**

**Ans.** It is the 10<sup>th</sup> element in the array num.

**14. Name the keyword that is used for allocating memory to an array. [ICSE 2009]**

**Ans.** The 'new' command is used for allocating memory to an array.

**15. State two advantages and disadvantages of using arrays.**

**Ans. Advantages:**

- i) Gives a concise code as declaration, allocation of space and initialization can be done in a single line.
- ii) Java arrays are dynamic, unlike other languages like C, C++ where it is static usually.

**Disadvantages:**

- i) Arrays allow you to store elements of only one data type, i.e. only homogeneous data can be stored, thus data of heterogeneous data type cannot be stored, which is often required by a programmer.
- ii) Arrays always require contiguous free memory location to be allocated for storing elements. But there may be situations when the available memory is discrete in nature and therefore array cannot be used to efficiently use the memory.

**B. Answer as directed:**

**1. What is the difference between these two statements:**

**i) int sum[ ]=new int[10]            ii) sum[1]=10;**

**Ans.** i) int sum[ ]=new int [10] is used to declare an integer array named 'sum' of size 10.  
ii) sum[1]=10 on the other hand is used to assign the value 10 to the index position 1 in the array 'sum'.

**2. Write the correct form of the following 'float' arrays:**

**i) A1(2)                            ii) Name[2/5]                            iii) Roll[2;]                            iv) Matrix(5)**

**Ans.** i) float A1[ ]=new float[2];  
ii) float Name=new float[25];  
iii) float Roll[]=new float[2];  
iv) float Matrix[]=new float[5];

**3. Write the value of individual array element.**

**int c[ ]={78,23,45,12,16};**

**Ans.** c[0]=78, c[1]=23, c[2]=45, c[3]=12 and c[4]=16.

**4. Give the proper array declaration for the following:**

- i) Declare an integer array, which can hold 25 values.**
- ii) Declare an integer array initialised with 12, 36, 49, 50, 67.**
- iii) Declare an array of float data type initialised to 13.7, 14.6, 19.3, 12.**

**Ans.** i) int a[]=new int[25];  
ii) int a[]={12,36,49,50,67};  
iii) float a[]={13.7f, 14.6f,19.3f,12};

**5. What will be the output of the following program:**

```
class Output
{
    public static void main(String args[])
    {
        int a[ ]={6,5,4,3,2,1};
        int x;
        for(x=5;x>=0;x--)
```



```

        {
            System.out.println(a[x]);
        }
    }
}

```

**Ans.** 1  
2  
3  
4  
5  
6

**6. What will be the output of the following program?**

```

class First
{
    public static void main(String args[ ])
    {
        int a[ ]={5,1,15,20,25};
        int i,j;
        int m;
        i=++a[1];
        j=a[2]++;
        m=a[i++];
        System.out.print(i+" "+j+" "+m);
    }
}

```

**Ans.** 3 15 16

**7. What will be the output of the following program, when method invoke( ) is called?**

```

public class StringArray
{
    void change(String arr[ ])
    {
        for(int i=0;i<arr.length;i++)
        {
            arr[i]=arr[i].substring(0,1).toUpperCase( )+arr[i].substring(1);
        }
    }
}
void invoke( )
{
    String ar[ ]={"kolkata","gangtok","bangalore"};
    for(int i=0;i<ar.length;i++)
    System.out.println(ar[i]);
    change(ar);
    for(int i=0;i<ar.length;i++)

```

```

        System.out.println(ar[i]);
    }
}

```

**Ans.** kolkata  
 gangtok  
 bangalore  
 Kolkata  
 Gangtok  
 Bangalore

**8. Find and correct the errors in the following program segment:**

```

int n[ ] = (2,4,6,8,10);
for (int i = 0; i < = 5; i ++ )
System.out.println("n[" + i + "] = " + n[i ]);

```

**[ICSE 2006]**

**Ans.** int n[ ] = {2,4,6,8,10};  
 for (int i = 0; i < 5; i ++ )  
 System.out.println("n[" + i + "] = " + n[i ]);

**9. Identify error(s), if any, in the following program:**

```

class First
{
    public static void main(String args[ ])
    {
        int i;
        int a[6]={0,1,8,7,6,4};
        for(i=0;i<=a.length( );i++)
        System.out.println(a[i]);
    }
}

```

**Ans.** class First  
 {  
 public static void main(String args[ ])  
 {  
 int i;  
 int a[6]={0,1,8,7,6,4};  
 for(i=0;i<a.length;i++)  
 System.out.println(a[i]);  
 }  
 }

**10. If, array[ ] = {1,9,8,5,2};** **[ICSE 2008]**

**i) What is array.length?**

**ii) What is array[2]?**

**Ans.** i) 5  
 ii) 8

**11. Write a program to input 10 numbers into an integer array and find the sum of all the numbers in it.**

```

Ans. import java.util.*;
class Question11
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];
        int i,s=0;
        System.out.println("Enter 10 integers:");
        for(i=0;i<10;i++)
        {
            a[i]=sc.nextInt();
            s+=a[i];
        }
        System.out.println("Sum of the numbers:"+s);
    }
}

```

**12. Write a program to input 10 numbers into a floating point array and find the average of all the numbers in it.**

```

Ans. import java.util.*;
class Question12
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        float a[]=new float[10],s=0,av;
        int i;
        System.out.println("Enter 10 floating point numbers:");
        for(i=0;i<10;i++)
        {
            a[i]=sc.nextInt();
            s+=a[i];
        }
        av=s/10;
        System.out.println("Average of the numbers:"+av);
    }
}

```

**13. Write a program to input 10 numbers into an integer array and find the sum of even numbers only.**

```

Ans. import java.util.*;
class Question13
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];

```

```

int i,s=0;
System.out.println("Enter 10 integers:");
for(i=0;i<10;i++)
{
    a[i]=sc.nextInt();
    if(a[i]%2==0)
        s+=a[i];
}
System.out.println("Sum of even numbers:"+s);
}
}

```

**14. Write a program to input 10 numbers into an integer array and find the sum of even as well as odd numbers separately.**

**Ans.** import java.util.\*;  
class Question14

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];
        int i,se=0,so=0;
        System.out.println("Enter 10 integers:");
        for(i=0;i<10;i++)
        {
            a[i]=sc.nextInt();
            if(a[i]%2==0)
                se+=a[i];
            else
                so+=a[i];
        }
        System.out.println("Sum of even numbers:"+se);
        System.out.println("Sum of odd numbers:"+so);
    }
}
}

```

**15. Write a program to input 10 numbers into an integer array and count the number of 5-digit numbers that are present in the array.**

**Ans.** import java.util.\*;  
class Question15

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];
        int i,c=0;
        System.out.println("Enter 10 integers:");
        for(i=0;i<10;i++)

```

```

    {
        a[i]=sc.nextInt();
        if(a[i]>=10000 && a[i]<=99999 || a[i]>=-99999 && a[i]<=-10000)
            c++;
    }
    System.out.println("Number of 5 digit numbers:"+c);
}
}

```

**16. Create a method to accept a integer array as parameter and count the number of positive, negative and zeroes in it.**

**Ans.** class Question16

```

{
    static void main(int a[])
    {
        int i,sp=0,sn=0,so=0;
        for(i=0;i<a.length;i++)
        {
            if(a[i]>0)
                sp++;
            else if(a[i]<0)
                sn++;
            else
                so++;
        }
        System.out.println("Sum of Positive numbers:"+sp);
        System.out.println("Sum of Negative numbers:"+sn);
        System.out.println("Sum of Zeroes:"+so);
    }
}

```

**17. Write a program to input 10 numbers into an integer array and find the average of two digit numbers only.**

**Ans.** import java.util.\*;

class Question17

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];
        int i,s=0,c=0;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
            if(a[i]>=10 && a[i]<=99)
            {
                s=s+a[i];
            }
        }
    }
}

```

```

        c++;
    }
}
if(c>0)
    System.out.println("Average of 2-digit numbers:"+(float)s/c);
else
    System.out.println("No 2-digit numbers present");
}
}

```

**18. Write a program to input 10 numbers into an integer array and find the average of even as well as odd numbers separately.**

**Ans.** import java.util.\*;

class Question18

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];
        int i,se=0,ce=0,so=0,co=0;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
            if(a[i]%2==0)
            {
                se=se+a[i];
                ce++;
            }
            else
            {
                so=so+a[i];
                co++;
            }
        }
        if(ce>0)
            System.out.println("Average of even numbers:"+(float)se/ce);
        else
            System.out.println("No even numbers present");
        if(co>0)
            System.out.println("Average of odd numbers:"+(float)so/co);
        else
            System.out.println("No odd numbers present");
    }
}
}

```

**19. Create a method to accept a float type array as parameter and print the average of the largest and smallest number.**

**Ans.** class Question19

```
{
    static void main(float a[])
    {
        int i;
        float lar=0,sma=0;
        for(i=0;i<a.length;i++)
        {
            if(i==0)
                lar=sma=a[i];
            if(a[i]>lar)
                lar=a[i];
            if(a[i]<sma)
                sma=a[i];
        }
        System.out.println("Average of largest and smallest number:"+(float)(lar+sma)/2);
    }
}
```

**20. Write a program to input 10 numbers into an integer array and print the position of the largest number.**

**Ans.** import java.util.\*;

class Question20

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];
        int i,l=0;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
            if(i==0)
                l=a[i];
            if(a[i]>l)
                l=a[i];
        }
        System.out.println("Positions where largest number is present are:");
        for(i=0;i<a.length;i++)
        {
            if(a[i]==l)
                System.out.print(i+" ");
        }
    }
}
```

```
}  
}
```

**21. Create a method to accept an integer array as parameter and print the position of the largest and smallest.**

**Ans.** class Question21

```
{  
    static void main(int a[])  
    {  
        int i,l=0,s=0;  
        for(i=0;i<a.length;i++)  
        {  
            if(i==0)  
                s=l=a[i];  
            if(a[i]>l)  
                l=a[i];  
            if(a[i]<s)  
                s=a[i];  
        }  
        System.out.println("Positions where largest number is present are:");  
        for(i=0;i<a.length;i++)  
        {  
            if(a[i]==l)  
                System.out.print(i+" ");  
        }  
        System.out.println("\nPositions where smallest number is present are:");  
        for(i=0;i<a.length;i++)  
        {  
            if(a[i]==s)  
                System.out.print(i+" ");  
        }  
    }  
}
```

**22. Write a program to input 10 numbers into an integer array and print the smallest prime number (if any) in the array. (Note that there may not be any prime number in the array and therefore getting the smallest prime number is beyond question and therefore you need to print a relevant message to show its absence.)**

**Ans.** import java.util.\*;

class Question22

```
{  
    static void main()  
    {  
        Scanner sc=new Scanner(System.in);  
        int a[]=new int[10];  
        int i,s=0,j,c,f=0;  
        System.out.println("Enter 10 numbers:");
```



```

for(i=0;i<a.length;i++)
{
    a[i]=sc.nextInt();
    c=0;
    for(j=1;j<=a[i];j++)
    {
        if(a[i]%j==0)
            c++;
    }
    if(c==2)
    {
        if(f==0)
        {
            s=a[i];
            f=1;
        }
        if(a[i]<s)
            s=a[i];
    }
}
if(f==1)
    System.out.println("Smallest Prime Number:"+s);
else
    System.out.println("No Prime No. present");
}
}

```

**23. Create a method to accept a String type array as parameter and print the smallest word in it.**

**Ans.** class Question23

```

{
    static void smallest(String a[])
    {
        String s="";
        int i;
        for(i=0;i<a.length;i++)
        {
            if(i==0)
                s=a[i];
            if(a[i].length()<s.length())
                s=a[i];
        }
        System.out.println("Smallest Word="+s);
    }
}
}

```

**24. Create a method to accept an array as parameter and print the second largest number in it.**

**Ans.** class Question24

```
{
    static void secondLargest(int a[])
    {
        int i,l=0,sl=0,f=0;
        for(i=0;i<a.length;i++)
        {
            if(i==0)
                l=a[i];
            if(a[i]>l)
                l=a[i];
        }
        for(i=0;i<a.length;i++)
        {
            if(a[i]!=l)
            {
                if(f==0)
                {
                    sl=a[i];
                    f=1;
                }
                if(a[i]>sl)
                    sl=a[i];
            }
        }
        System.out.println("Second Largest Number="+sl);
    }
}
```

**25. Create a method to accept an array as parameter and print the position of the second largest number in it.**

**Ans.** class Question25

```
{
    static void secondLargest(int a[])
    {
        int i,l=0,sl=0,f=0,p=0;
        for(i=0;i<a.length;i++)
        {
            if(i==0)
                l=a[i];
            if(a[i]>l)
                l=a[i];
        }
        for(i=0;i<a.length;i++)
        {
            if(a[i]!=l)
            {
```

```

        if(f==0)
        {
            sl=a[i];
            p=i;
            f=1;
        }
        if(a[i]>sl)
        {
            sl=a[i];
            p=i;
        }
    }
}
System.out.println("Position of Second Largest Number="+p);
}
}

```

**26. Write a program to input 10 numbers into an integer array and replace all prime numbers in it (if any) by 0 and print the modified array. For example, if the array contains the following elements:**

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 3    | 7    | 67   | 34   | 15   | 16   | 89   | 13   |

after replacing the prime numbers with 0 the modified array should have the elements as:

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 0    | 0    | 67   | 34   | 15   | 16   | 89   | 0    |

**Ans.** class Question26

```

{
    static void main(int a[])
    {
        int i,j,c;
        for(i=0;i<a.length;i++)
        {
            c=0;
            for(j=1;j<=a[i];j++)
            {
                if(a[i]%j==0)
                    c++;
            }
            if(c==2)
                a[i]=0;
        }
        System.out.println("Modified Array:");
        for(i=0;i<a.length;i++)
        {
            System.out.print(a[i]+" ");
        }
    }
}

```

```
    }  
  }  
}
```

**27. Write a program to input an integer array and replace all Armstrong numbers in it with 0 and display the modified array.**

**Ans.** class Question27

```
{  
    static void main(int a[])  
    {  
        int i,j,s,t,d;  
        for(i=0;i<a.length;i++)  
        {  
            s=0;t=a[i];  
            while(t!=0)  
            {  
                d=t%10;  
                s+=d*d*d;  
                t=t/10;  
            }  
            if(s==a[i])  
                a[i]=0;  
        }  
        System.out.println("Modified Array:");  
        for(i=0;i<a.length;i++)  
        {  
            System.out.print(a[i]+" ");  
        }  
    }  
}
```

**28. Write a program to input 10 numbers into an integer array and interchange the largest number with the smallest number within the array and print the modified array. Assume that there is only one largest and smallest number.**

**For example, if array contains**

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 3    | 7    | 67   | 34   | 15   | 16   | 89   | 5    |

↑  
Smallest

↑  
Largest

after interchange it should have the elements arranged as:

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 89   | 7    | 67   | 34   | 15   | 16   | 3    | 5    |

↑  
Largest

↑  
Smallest

```

Ans. import java.util.*;
class Question28
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];
        int i,l=0,lp=0,s=0,sp=0;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
            if(i==0)
            {
                s=l=a[i];
                sp=lp=i;
            }
            if(a[i]>l)
            {
                l=a[i];
                lp=i;
            }
            if(a[i]<s)
            {
                s=a[i];
                sp=i;
            }
        }
        a[lp]=s;
        a[sp]=l;
        System.out.println("Modified Array:");
        for(i=0;i<a.length;i++)
        {
            System.out.print(a[i]+" ");
        }
    }
}

```

```
    }  
  }  
}
```

**29. Write a program to bubble-sort the following set of values in ascending order: 5,3,8,4,9,2,1,12,98,16** [ICSE 2005]  
**OUTPUT: 1 2 3 4 5 8 9 12 16 98**

**Ans.** class Question29  
{  
 static void main()  
 {  
 int a[]={5,3,8,4,9,2,1,12,98,16};  
 int i,j,t;  
 for(i=a.length-1;i>0;i--)  
 {  
 for(j=0;j<i;j++)  
 {  
 if(a[j]>a[j+1])  
 {  
 t=a[j];  
 a[j]=a[j+1];  
 a[j+1]=t;  
 }  
 }  
 }  
 System.out.println("Sorted Array:");  
 for(i=0;i<a.length;i++)  
 System.out.print(a[i]+" ");  
 }  
}

**30. Write a program to input 10 numbers into a float type array and arrange the numbers in descending order using Selection Sorting technique.**

**Ans.** import java.util.\*;  
class Question30  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 float a[]=new float[10],l;  
 int i,j,p;  
 System.out.println("Enter 10 numbers:");  
 for(i=0;i<a.length;i++)  
 {  
 a[i]=sc.nextFloat();  
 }  
 for(i=0;i<a.length-1;i++)

```

    {
        l=a[i];p=i;
        for(j=i+1;j<a.length;j++)
        {
            if(a[j]>l)
            {
                l=a[j];
                p=j;
            }
        }
        a[p]=a[i];
        a[i]=l;
    }

    System.out.println("Sorted Array:");
    for(i=0;i<a.length;i++)
    {
        System.out.print(a[i]+" ");
    }
}
}

```

**31. Write a program to accept 15 integers from the keyboard, assuming that no integer entered is a zero. Perform selection sort on the integers and then print them in ascending order. [ICSE 2006]**

**Ans.** import java.util.\*;  
class Question31  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int a[]=new int[15],s,i,j,p;  
 System.out.println("Enter 15 numbers:");  
 for(i=0;i<a.length;i++)  
 {  
 a[i]=sc.nextInt();  
 }  
 for(i=0;i<a.length-1;i++)  
 {  
 s=a[i];p=i;  
 for(j=i+1;j<a.length;j++)  
 {  
 if(a[j]<s)  
 {  
 s=a[j];  
 p=j;  
 }  
 }  
 }  
 }  
}

```

    }
    a[p]=a[i];
    a[i]=s;
}

System.out.println("Sorted Array:");
for(i=0;i<a.length;i++)
{
    System.out.print(a[i]+" ");
}
}
}

```

**32. Write a program to input 10 numbers into an integer array and arrange only the even numbers in ascending order using Bubble Sorting technique.**

**Ans.** import java.util.\*;  
class Question32  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int a[]=new int[10],t,i,j;  
 System.out.println("Enter 10 numbers:");  
 for(i=0;i<a.length;i++)  
 {  
 a[i]=sc.nextInt();  
 }  
 for(i=a.length-1;i>0;i--)  
 {  
 for(j=0;j<i;j++)  
 {  
 if(a[j]>a[j+1] && a[j]%2==0 && a[j+1]%2==0)  
 {  
 t=a[j];  
 a[j]=a[j+1];  
 a[j+1]=t;  
 }  
 }  
 }  
 }  
  
 System.out.println("Sorted Array:");  
 for(i=0;i<a.length;i++)  
 {  
 System.out.print(a[i]+" ");  
 }  
}



```
}
```

**33. Write a program to input 10 numbers into an integer array and arrange only the numbers at even positions in descending order using Bubble Sorting technique.**

```
Ans. import java.util.*;
class Question33
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10],t,i,j;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
        }
        for(i=8;i>0;i-=2)
        {
            for(j=0;j<i;j+=2)
            {
                if(a[j]<a[j+2])
                {
                    t=a[j];
                    a[j]=a[j+2];
                    a[j+2]=t;
                }
            }
        }

        System.out.println("Sorted Array:");
        for(i=0;i<a.length;i++)
        {
            System.out.print(a[i]+" ");
        }
    }
}
```

**34. Write a program to input 10 numbers into an integer array and reverse the array and print the modified array.**

**For example, if array contains**

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 3    | 7    | 89   | 34   | 15   | 16   | 67   | 25   |

after reversal it should have the elements arranged as:

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 25   | 67   | 16   | 15   | 34   | 89   | 7    | 3    | 12   | 9    |

```

Ans. import java.util.*;
class Question34
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10],t,i,j;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
        }
        for(i=0;i<a.length/2;i++)
        {
            t=a[i];
            a[i]=a[a.length-1-i];
            a[a.length-1-i]=t;
        }

        System.out.println("Modified Array:");
        for(i=0;i<a.length;i++)
        {
            System.out.print(a[i]+" ");
        }
    }
}

```

**35. Write a program to input 10 numbers into an integer array and interchange the consecutive elements in it. That is, interchange a[0] with a[1], a[2] with a[3], a[4] with a[5].**

**For example, if array contains**

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 3    | 7    | 89   | 34   | 15   | 16   | 67   | 25   |

after interchange it should have the elements arranged as:

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 12   | 9    | 7    | 3    | 34   | 89   | 16   | 15   | 25   | 67   |

```

Ans. import java.util.*;
class Question35
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10],t,i,j;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {

```

```

        a[i]=sc.nextInt();
    }
    for(i=0;i<a.length-1;i+=2)
    {
        t=a[i];
        a[i]=a[i+1];
        a[i+1]=t;
    }

    System.out.println("Modified Array:");
    for(i=0;i<a.length;i++)
    {
        System.out.print(a[i]+" ");
    }
}
}

```

**36. Write a program to input 10 numbers into an integer array and interchange the elements of the first half with the corresponding elements of the second half. That is, interchange a[0] with a[5], a[1] with a[6], a[2] with a[7], a[3] with a[8] and a[4] with a[9].**

**For example, if array contains**

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 3    | 7    | 89   | 34   | 15   | 16   | 67   | 25   |

after interchange it should have the elements arranged as:

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 34   | 15   | 16   | 67   | 25   | 9    | 12   | 3    | 7    | 89   |

**Ans.** import java.util.\*;  
class Question36  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int a[]=new int[10],t,i,j;  
 System.out.println("Enter 10 numbers:");  
 for(i=0;i<a.length;i++)  
 {  
 a[i]=sc.nextInt();  
 }  
 for(i=0;i<a.length/2;i++)  
 {  
 t=a[i];  
 a[i]=a[i+5];  
 a[i+5]=t;  
 }  
 }  
}

```

System.out.println("Modified Array:");
for(i=0;i<a.length;i++)
{
    System.out.print(a[i]+" ");
}
}
}

```

**37. Write a program to input 10 numbers into an integer array and find the frequency of the largest number.**

**For example, if the array contains**

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 89   | 3    | 7    | 89   | 34   | 15   | 16   | 67   | 89   |

**Output should be:**

**Frequency of largest number = 3**

**Ans.** import java.util.\*;

class Question37

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10],l=0,c=0,i;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
            if(i==0)
                l=a[i];
            if(a[i]>l)
                l=a[i];
        }
        for(i=0;i<a.length;i++)
        {
            if(l==a[i])
                c++;
        }

        System.out.println("Frequency of the largest number="+c);
    }
}

```

**38. Create a method to accept two arrays of integer type as parameter (say A and B), which are arranged in ascending order and merge them into a third array in such a manner that the elements are in ascending order. Do not use any known sorting technique to sort.**

**Ans.** class Question38

```

{
static void merge(int a[],int b[])
{
int ia=0,ib=0,ic=0;
int c[]=new int[a.length+b.length];
while(ia<a.length && ib<b.length)
{
if(a[ia]<b[ib])
c[ic]=a[ia++];
else
c[ic]=b[ib++];
ic++;
}
while(ia<a.length)
c[ic++]=a[ia++];

while(ib<b.length)
c[ic++]=b[ib++];

System.out.println("Merged Array:");
for(ic=0;ic<c.length;ic++)
{
System.out.print(c[ic]+" ");
}
}
}

```

**39. Write a program to input 10 numbers into an integer array and input a position. Now delete the element at that position by shifting the rest of the numbers to the left and insert a 0 at the end.**

**For example, if array contains**

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 3    | 7    | 89   | 34   | 15   | 16   | 67   | 25   |

and element at position 4 is to be deleted, the resultant array should be:

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 3    | 7    | 34   | 15   | 16   | 67   | 25   | 0    |

**Ans.** import java.util.\*;

class Question39

```

{
static void main()
{
Scanner sc=new Scanner(System.in);
int a[]=new int[10],p,i;
System.out.println("Enter 10 numbers:");
for(i=0;i<a.length;i++)
{

```

```

        a[i]=sc.nextInt();
    }
    System.out.println("Enter the position to delete:");
    p=sc.nextInt();
    for(i=p;i<a.length-1;i++)
    {
        a[i]=a[i+1];
    }
    a[9]=0;
    System.out.println("Modified Array:");
    for(i=0;i<a.length;i++)
    {
        System.out.print(a[i]+" ");
    }
}
}

```

**40. Write a program to input 10 numbers into an integer array and input a number and a position. Now insert the number at that position by shifting the rest of the numbers to the right. The last element is therefore removed from the array.**

**For example, if array contains**

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 3    | 7    | 89   | 34   | 15   | 16   | 67   | 25   |

and if 36 is to be inserted at position 4, the resultant array should be:

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 9    | 12   | 3    | 7    | 36   | 89   | 34   | 15   | 16   | 67   |

**Notice that the last element i.e. 25 got removed after the shifting process.**

**Ans.** import java.util.\*;

class Question40

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10],p,i,n;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
        }
        System.out.println("Enter the number to insert:");
        n=sc.nextInt();
        System.out.println("Enter the position where to insert:");
        p=sc.nextInt();
        for(i=a.length-1;i>p;i--)
        {

```

```

        a[i]=a[i-1];
    }
    a[p]=n;
    System.out.println("Modified Array:");
    for(i=0;i<a.length;i++)
    {
        System.out.print(a[i]+" ");
    }
}
}

```

**41. Write a program to input 10 positive or negative numbers (no zero) into an integer array and shift all positive numbers to the beginning of the array and negative numbers to the end of the array, without changing the order of the numbers.**

**For example, if the array contains the following elements:**

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| -9   | -12  | -3   | -7   | 89   | -34  | 15   | 16   | -67  | 25   |

after shifting the array should contain the elements arranged in the following manner:

| a[0] | a[1] | a[2] | a[3] | a[4] | a[5] | a[6] | a[7] | a[8] | a[9] |
|------|------|------|------|------|------|------|------|------|------|
| 12   | 89   | 15   | 16   | 25   | -9   | -3   | -7   | -34  | -67  |

⏟
⏟

Positive numbers
Negative numbers

**Ans.** import java.util.\*;  
class Question41  
{  
static void main()  
{  
Scanner sc=new Scanner(System.in);  
int a[]=new int[10],t,i,j;  
System.out.println("Enter 10 numbers:");  
for(i=0;i<a.length;i++)  
{  
a[i]=sc.nextInt();  
}  
for(i=a.length-1;i>0;i--)  
{  
for(j=0;j<i;j++)  
{  
if(a[j]<0 && a[j+1]>0)  
{  
t=a[j];  
a[j]=a[j+1];  
a[j+1]=t;  
}  
}  
}

```

    }
}

System.out.println("Modified Array:");
for(i=0;i<a.length;i++)
{
    System.out.print(a[i]+" ");
}
}
}

```

**42. Write a program to input 10 numbers into an integer array and shift all even numbers to the beginning of the array and odd numbers to the end of the array, without changing the order of the numbers.**

**Ans.** import java.util.\*;  
class Question42  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int a[]=new int[10],t,i,j;  
 System.out.println("Enter 10 numbers:");  
 for(i=0;i<a.length;i++)  
 {  
 a[i]=sc.nextInt();  
 }  
 for(i=a.length-1;i>0;i--)  
 {  
 for(j=0;j<i;j++)  
 {  
 if(a[j]%2!=0 && a[j+1]%2==0)  
 {  
 t=a[j];  
 a[j]=a[j+1];  
 a[j+1]=t;  
 }  
 }  
 }  
 }  
}

System.out.println("Modified Array:");
for(i=0;i<a.length;i++)
{
 System.out.print(a[i]+" ");
}
}
}



**43. Write a program to input 10 numbers into an integer array and print those single-digit positive numbers which are not present in the array.**

**Ans.** import java.util.\*;

class Question43

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10],x,i,f;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
        }
        System.out.println("Single Digit Positive Numbers that are not present in the array
are:");
        for(x=1;x<=9;x++)
        {
            f=0;
            for(i=0;i<a.length;i++)
            {
                if(a[i]==x)
                    f=1;
            }
            if(f==0)
                System.out.print(x+" ");
        }
    }
}
```

**44. Write a program to input 10 numbers into an integer array and find the frequency of each two-digit positive numbers present in it.**

**Ans.** import java.util.\*;

class Question44

```
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10],x,i,c;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
        }
        for(x=10;x<=99;x++)
        {
            c=0;
```

```

        for(i=0;i<a.length;i++)
        {
            if(a[i]==x)
                c++;
        }
        if(c>0)
            System.out.println("Frequency of "+x+"is="+c);
    }
}
}

```

**45. Create a method to accept an integer array as parameter and print the frequency of each element in it.**

**Ans.** class Question45

```

{
    static void main(int a[])
    {
        int i,j,f,c;
        for(i=0;i<a.length;i++)
        {
            f=0;
            for(j=0;j<i;j++)
            {
                if(a[i]==a[j])
                    f=1;
            }
            if(f==0)
            {
                c=0;
                for(j=i;j<a.length;j++)
                {
                    if(a[i]==a[j])
                        c++;
                }
                System.out.println("Frequency of "+a[i]+"is="+c);
            }
        }
    }
}
}

```

**46. Write a program to input 10 numbers into an integer array and print the number having maximum frequency.**

**Ans.** import java.util.\*;

class Question46

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);

```

```

int a[]=new int[10],i,j,c,l=0,n=0;
System.out.println("Enter 10 numbers:");
for(i=0;i<a.length;i++)
{
    a[i]=sc.nextInt();
}
for(i=0;i<a.length;i++)
{
    c=0;
    for(j=0;j<a.length;j++)
    {
        if(a[i]==a[j])
            c++;
    }
    if(i==0)
    {
        l=c;
        n=a[i];
    }
    if(c>l)
    {
        l=c;
        n=a[i];
    }
}
System.out.println("Number having maximum frequency="+n);
}
}

```

**47. Write a program to input 10 numbers into an integer array and print the number having least frequency.**

**Ans.** import java.util.\*;  
class Question47  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int a[]=new int[10],i,j,c,s=0,n=0;  
 System.out.println("Enter 10 numbers:");  
 for(i=0;i<a.length;i++)  
 {  
 a[i]=sc.nextInt();  
 }  
 for(i=0;i<a.length;i++)  
 {  
 c=0;  
 for(j=0;j<a.length;j++)

```

        {
            if(a[i]==a[j])
                c++;
        }
        if(i==0)
        {
            s=c;
            n=a[i];
        }
        if(c<s)
        {
            s=c;
            n=a[i];
        }
    }
    System.out.println("Number having least frequency="+n);
}
}

```

**48. Write a program to input 10 words into an integer array and print only those words that begin with an 'A' or 'a'.**

**Ans.** import java.util.\*;  
class Question48

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String a[]=new String[10];
        int i;
        System.out.println("Enter 10 words:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextLine();
        }
        System.out.println("Words that begins with character A or a");
        for(i=0;i<a.length;i++)
        {
            if(a[i].charAt(0)=='A' || a[i].charAt(0)=='a')
                System.out.println(a[i]);
        }
    }
}
}

```

**49. Write a program to input 10 numbers into an integer array and store only the even numbers into another array and display it.**

**Ans.** import java.util.\*;  
class Question49  
{



```

    }
    if(c==2)
        b[p++]=a[i];
    }
    System.out.println("Numbers that are stored in array b:");
    for(i=0;i<p;i++)
    {
        System.out.print(b[i]+" ");
    }
}
}

```

**51. Write a program to input 10 numbers into an integer array and store only the unique numbers into another array and display it.**

**Ans.** import java.util.\*;  
class Question51  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int a[]=new int[10];  
 int b[]=new int[10];  
 int i,c=0,j,p=0;  
 System.out.println("Enter 10 numbers:");  
 for(i=0;i<a.length;i++)  
 {  
 a[i]=sc.nextInt();  
 }  
 for(i=0;i<a.length;i++)  
 {  
 c=0;  
 for(j=0;j<a.length;j++)  
 {  
 if(a[i]==a[j])  
 c++;  
 }  
 if(c==1)  
 b[p++]=a[i];  
 }  
 System.out.println("Numbers that are stored in array b:");  
 for(i=0;i<c;i++)  
 {  
 System.out.print(b[i]+" ");  
 }  
 }  
}

**52. Write a program to input 10 numbers into an integer array and store each number only once into another array irrespective of the number of times it is present in the array.**

```
Ans. import java.util.*;
class Question52
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];
        int b[]=new int[10];
        int i,c=0,j,p=0;
        System.out.println("Enter 10 numbers:");
        for(i=0;i<a.length;i++)
        {
            a[i]=sc.nextInt();
        }
        for(i=0;i<a.length;i++)
        {
            c=0;
            for(j=0;j<i;j++)
            {
                if(a[i]==a[j])
                    c=1;
            }
            if(c==0)
                b[p++]=a[i];
        }
        System.out.println("Numbers that are stored in array b:");
        for(i=0;i<p;i++)
        {
            System.out.print(b[i]+" ");
        }
    }
}
```

**53. Write a program to input 10 numbers into an integer array and store all even numbers into one array and all odd numbers into another array. Display all the three arrays.**

```
Ans. import java.util.*;
class Question53
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a[]=new int[10];
        int b[]=new int[10];
```

```

int c[]=new int[10];
int i,p=0,q=0;
System.out.println("Enter 10 numbers:");
for(i=0;i<a.length;i++)
{
    a[i]=sc.nextInt();
}
for(i=0;i<a.length;i++)
{
    if(a[i]%2==0)
        b[p++]=a[i];
    else
        c[q++]=a[i];
}
System.out.println("Original Array:");
for(i=0;i<10;i++)
{
    System.out.print(a[i]+" ");
}
System.out.println("\nArray where even numbers are stored:");
for(i=0;i<p;i++)
{
    System.out.print(b[i]+" ");
}
System.out.println("\nArray where odd numbers are stored:");
for(i=0;i<q;i++)
{
    System.out.print(c[i]+" ");
}
}
}

```

**54. Write a program to input 10 numbers and store each one of them into an integer array in such a way that the array is always maintained in ascending order.**

**Ans.** import java.util.\*;  
class Question54  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int a[]=new int[10];  
 int i,n,j,f=0,k;  
 System.out.println("Enter 10 numbers:");  
 for(i=0;i<a.length;i++)



```

{
    n=sc.nextInt();
    f=0;
    for(j=0;j<i;j++)
    {
        if(a[j]>n)
        {
            f=1;
            break;
        }
    }
    if(f==1)
    {
        for(k=i;k>j;k--)
        {
            a[k]=a[k-1];
        }
        a[j]=n;
    }
    else
        a[i]=n;
}
for(i=0;i<a.length;i++)
{
    System.out.print(a[i]+" ");
}
}
}

```

**55. Write a program to store the first 10 prime numbers into an integer array and display it.**

**Ans.** class Question55

```

{
    static void main()
    {
        int a[]=new int[10];
        int i=0,n=2,j,c;
        while(i<10)
        {
            c=0;
            for(j=1;j<=n;j++)
            {
                if(n%j==0)
                    c++;
            }
            if(c==2)

```

```

        a[i++] = n;
        n++;
    }
    System.out.println("First 10 prime numbers:");
    for(i=0; i<10; i++)
    {
        System.out.print(a[i] + " ");
    }
}
}

```

**56. Write a program to generate the first 20 numbers of the Fibonacci series using an array.**

**Ans.** class Question56

```

{
    static void main()
    {
        int a[] = new int[20];
        int i;
        a[0] = a[1] = 1;
        for(i=2; i<20; i++)
        {
            a[i] = a[i-1] + a[i-2];
        }
        System.out.println("First 20 numbers of the fibonacci series:");
        for(i=0; i<20; i++)
        {
            System.out.print(a[i] + " ");
        }
    }
}

```

**57. Create a method to accept an integer array as parameter and find the mean, median and mode of it.**

**Ans.** class Question57

```

{
    static void main(int a[])
    {
        int i, j, t, s=0, c, l=0, n=0;
        for(i=0; i<a.length; i++)
        {
            s += a[i];
        }
        System.out.println("Mean=" + (float)s/10);
        for(i=0; i<a.length-1; i++)
        {
            for(j=i+1; j<a.length; j++)
            {

```

```

        if(a[i]>a[j])
        {
            t=a[i];
            a[i]=a[j];
            a[j]=t;
        }
    }
}
if(a.length%2!=0)
    System.out.println("Median:"+a[a.length/2]);
else
    System.out.println("Median:"+(float)(a[a.length/2]+a[a.length/2-1])/2);

for(i=0;i<a.length;i++)
{
    c=0;
    for(j=0;j<a.length;j++)
    {
        if(a[i]==a[j])
            c++;
    }
    if(c>1)
    {
        l=c;
        n=a[i];
    }
}
System.out.println("Number having maximum frequency:"+n);
}
}

```

**58. Create a method to accept an integer array as parameter and check whether all numbers in it are two-digit numbers or not.**

**Ans.** class Question58

```

{
    static void is2digit(int a[])
    {
        int i;
        boolean f=true;
        for(i=0;i<a.length;i++)
        {
            if(!(a[i]>=10 && a[i]<=99 || a[i]>=-99 && a[i]<=-10))
                f=false;
        }
        if(f)
            System.out.println("All are 2-digit numbers");
        else
    }
}

```

```
        System.out.println("All are not 2-digit numbers");
    }
}
```

**59. Create a method to accept an integer array as parameter and check whether all numbers in it are even numbers or not.**

**Ans.** class Question59

```
{
    static void isEven(int a[])
    {
        int i;
        boolean f=true;
        for(i=0;i<a.length;i++)
        {
            if(a[i]%2!=0)
                f=false;
        }
        if(f)
            System.out.println("All are even numbers");
        else
            System.out.println("All are not even numbers");
    }
}
```

**60. Create a method to accept an integer array as parameter and check whether all numbers in it are prime numbers or not.**

**Ans.** class Question60

```
{
    static void isPrime(int a[])
    {
        int i,j,c;
        boolean f=true;
        for(i=0;i<a.length;i++)
        {
            c=0;
            for(j=1;j<=a[i];j++)
            {
                if(a[i]%j==0)
                    c++;
            }
            if(c!=2)
                f=false;
        }
        if(f)
            System.out.println("All are prime numbers");
        else
            System.out.println("All are not prime numbers");
    }
}
```

```
}
```

**61. Create a method to accept an integer array as parameter and check whether all numbers are sorted in ascending order or not.**

**Ans.** class Question61

```
{
    static void isAscending(int a[])
    {
        int i;
        boolean f=true;
        for(i=0;i<a.length-1;i++)
        {
            if(a[i]>a[i+1])
                f=false;
        }
        if(f)
            System.out.println("All are in ascending order");
        else
            System.out.println("All are not in ascending order");
    }
}
```

**62. Create a method to accept an integer array as parameter and print, the largest even number in it (if any), otherwise print 0.**

**Ans.** class Question62

```
{
    static void largestEven(int a[])
    {
        int i,l=0,f=0;
        for(i=0;i<a.length-1;i++)
        {
            if(a[i]%2==0)
            {
                if(f==0)
                {
                    l=a[i];
                    f=1;
                }
                if(a[i]>l)
                    l=a[i];
            }
        }
        if(f==1)
            System.out.println("Largest Even Number="+l);
        else
            System.out.println("No even number found="+l);
    }
}
```

```
}  
}
```

**63. Create a method to accept a char array as parameter and check whether all elements in it are unique or not.**

**Ans.** class Question63

```
{  
    static void isUnique(char a[])  
    {  
        int i,j;  
        boolean f=true;  
        for(i=0;i<a.length-1;i++)  
        {  
            for(j=i+1;j<a.length;j++)  
            {  
                if(a[i]==a[j])  
                    f=false;  
            }  
        }  
    }  
    if(f)  
        System.out.println("All characters in it are unique");  
    else  
        System.out.println("All characters are not unique");  
}  
}
```

**64. Create a method to accept an integer array as parameter and check whether all elements in it come under the Fibonacci series or not. Assume that the numbers may be in any order.**

**Ans.** class Question64

```
{  
    static void isFibo(int a[])  
    {  
        int i,x,y,z;  
        boolean f=true;  
        for(i=0;i<a.length;i++)  
        {  
            x=1;y=0;z=0;  
            while(z<a[i])  
            {  
                z=x+y;  
                x=y;  
                y=z;  
            }  
            if(z!=a[i])  
                f=false;  
        }  
    }  
    if(f)
```



```

    }
    if(i==0)
    {
        l=c;
        n=a[i];
    }
    if(c<l)
    {
        l=c;
        n=a[i];
    }
}
System.out.println("Number having least frequency="+n);
}

```

```

static void leastFrequency(char a[])
{
    int i,j,l=0,c=0;
    char ch=' ';
    for(i=0;i<a.length;i++)
    {
        c=0;
        for(j=0;j<a.length;j++)
        {
            if(a[i]==a[j])
                c++;
        }
        if(i==0)
        {
            l=c;
            ch=a[i];
        }
        if(c<l)
        {
            l=c;
            ch=a[i];
        }
    }
    System.out.println("Character having least frequency="+ch);
}
}

```

**67. Create a method which accepts two integers as parameters and return their HCF. Now use it in another method to find the HCF of the elements in an integer array.**

**Ans.** class Question67

```

{

```



```

static int hcf(int n,int d)
{
    int r;
    do
    {
        r=n%d;
        n=d;
        d=r;
    }while(r!=0);
    return n;
}

```

```

static void findHCF(int a[])
{
    int i,h=0;
    for(i=0;i<a.length;i++)
    {
        if(i==0)
            h=a[i];
        else
            h=hcf(h,a[i]);
    }
    System.out.println("HCF="+h);
}
}

```

**68. Create a class containing the following members:**

**Data Members:**

**farr[ ]** a float type array reference.

**iarr[ ]** a int type array reference.

**Member functions:**

**i) Parameterized constructor to initialize farr[ ].**

**ii) To display both the arrays.**

**iii) To round off each elements of farr[ ] and store it in iarr[ ].**

**Also create the main to show its implementation.**

**Ans.** class Array //Question 68

```

{
    float farr[];
    int iarr[];
    Array(float f[])
    {
        farr=f;
        iarr=new int[farr.length];
    }

    void display()
    {

```

```

    int i;
    for(i=0;i<farr.length;i++)
    {
        System.out.println(farr[i)+"\t"+iarr[i]);
    }
}

void roundOff()
{
    int i;
    for(i=0;i<farr.length;i++)
    {
        iarr[i]=Math.round(farr[i]);
    }
}

public static void main(String args[])
{
    float f[]={ 12.3f,14.9f,15.5f,16,7f,23,4f};
    Array ob=new Array(f);
    ob.roundOff();
    ob.display();
}
}

```

**69. Create a class named Student containing the following instance variables:**

**Instance Variables:**

**roll[ ] of int type array reference.**

**name[ ] of String type array reference.**

**Member functions:**

**i) Parameterized constructor to initialize the data members.**

**ii) To accept a roll as parameter and search for it in roll[ ]. If found, the corresponding name should be displayed otherwise a relevant message should be displayed.**

**Ans.** class Student //Question 69

```

{
    int roll[];
    String name[];

    Student(int r[],String n[])
    {
        roll=r;
        name=n;
    }

    void search(int r)

```

```

{
    int i;
    boolean f=false;
    for(i=0;i<roll.length;i++)
    {
        if(roll[i]==r)
        {
            System.out.println(name[i]);
            f=true;
            break;
        }
    }
    if(!f)
        System.out.println("Not Found!");
}
}

```

**70. Create a method which accepts a sentence as a parameter and print the words according to ascending order of length.**

**Ans.** class Question70

```

{
    static void ascending(String s)
    {
        s=s.trim();
        s=s+" ";
        String w="",t="";
        int i,j,c=0;
        char x;
        String a[]=new String[s.length()];
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                if(w!="")
                    a[c++]=w;
                w="";
            }
        }
        for(i=0;i<c-1;i++)
        {
            for(j=i+1;j<c;j++)
            {
                if(a[i].length(>a[j].length())
                {

```

```

        t=a[i];
        a[i]=a[j];
        a[j]=t;
    }
}
}
for(i=0;i<c;i++)
{
    System.out.print(a[i]+" ");
}
}
}

```

**71. Create a method which accepts a sentence as a parameter and print the words in alphabetical order.**

**Ans.** class Question71

```

{
    static void ascending(String s)
    {
        s=s.trim();
        s=s+" ";
        String w="",t="";
        int i,j,c=0;
        char x;
        String a[]=new String[s.length()];
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                if(w!="")
                    a[c++]=w;
                w="";
            }
        }
    }
    for(i=0;i<c-1;i++)
    {
        for(j=i+1;j<c;j++)
        {
            if(a[i].compareTo(a[j])>0)
            {
                t=a[i];
                a[i]=a[j];
                a[j]=t;
            }
        }
    }
}

```

```

    }
}
for(i=0;i<c;i++)
{
    System.out.print(a[i]+" ");
}
}
}

```

**72. Write a program to initialize an array of 5 names and initialize another array with their respective telephone numbers. Search for a name input by the User, in the list. If found, display "Search Successful" and print the name along with the telephone number, otherwise display "Search unsuccessful. Name not enlisted". [ICSE 2005]**

```

Ans. import java.util.*;
class Question72
{
    static void search()
    {
        Scanner sc=new Scanner(System.in);
        String name[]{"Amit","Deb","Somesh","Trina","Ashish"};
        long
phone[]={9832164084L,9476278923L,8936745126L,9866733497L,9832349667L};
        String n;
        boolean f=false;
        int i;
        System.out.println("Enter the name to search:");
        n=sc.nextLine();
        for(i=0;i<name.length;i++)
        {
            if(n.equalsIgnoreCase(name[i]))
            {
                System.out.println("Search Successful");
                System.out.println(name[i]+"\\t"+phone[i]);
                f=true;
                break;
            }
        }
        if(!f)
            System.out.println("Search unsuccessful. Name not enlisted");
    }
}
}

```

**73. The marks obtained by 50 students in a subject are tabulated as follows: -**

| Name | Marks |
|------|-------|
|      |       |
|      |       |
|      |       |

**Write a program to input the names and the marks of the students in the subject. Calculate and display:**

**i) The subject average marks (subject average marks = subject total / 50)**

**ii) The highest mark in the subject and the name of the student. (The maximum marks in the subject are 100). [ICSE 2006]**

```
Ans. import java.util.*;
class Question73
{
    static void main()
    {
        String name[]=new String[50],n="";
        int marks[]=new int[50];
        int i,tot=0,max=0;
        float avg;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the names and marks for 50 students:");
        for(i=0;i<name.length;i++)
        {
            name[i]=sc.nextLine();
            marks[i]=sc.nextInt();
            sc.nextLine();//dummy input
        }
        for(i=0;i<name.length;i++)
        {
            tot=tot+ marks[i];
        }
        avg=(float)tot/50;
        System.out.println("Subject Average Marks:"+avg);
        for(i=0;i<name.length;i++)
        {
            if(i==0)
            {
                max=marks[i];
                n=name[i];
            }
            if(marks[i]>max)
            {
                max=marks[i];
                n=name[i];
            }
        }
    }
}
```

```

    }
    System.out.println("Highest mark:"+max+" Name:"+n);
}
}

```

**74. Define a class and store the given city names in a single dimensional array. Sort these names in alphabetical order using the Bubble Sort technique only.**

**INPUT: Delhi, Bangalore, Agra, Mumbai, Calcutta**

**OUTPUT: Agra, Bangalore, Calcutta, Delhi, Mumbai**

**[ICSE 2008]**

**Ans.** class Question74

```

{
    static void main()
    {
        String a[]={"Delhi","Bangalore","Agra","Mumbai","Calcutta"};
        int i,j;
        String t="";
        for(i=a.length-1;i>=0;i--)
        {
            for(j=0;j<i;j++)
            {
                if(a[j].compareTo(a[j+1])>0)
                {
                    t=a[j];
                    a[j]=a[j+1];
                    a[j+1]=t;
                }
            }
        }

        System.out.println("Sorted Array:");
        for(i=0;i<a.length;i++)
        {
            System.out.println(a[i]);
        }
    }
}

```

**75. Write a program to perform binary search on a list of integers given below, to search for an element input by the user. If it is found display the element along with its position, otherwise display the message "Search element not found". 5,7,9,11,15,20,30,45,89,97 [ICSE 2010]**

**Ans.** import java.util.\*;

class Question75

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);

```

```

int a[]={5,7,9,11,15,20,30,45,89,97};
int n,l=0,u=a.length-1,m;
boolean f=false;
System.out.println("Enter the number to search:");
n=sc.nextInt();
while(l<=u)
{
    m=(l+u)/2;
    if(a[m]==n)
    {
        System.out.println("Element:"+a[m]);
        System.out.println("Position:"+m);
        f=true;
        break;
    }
    else if(a[m]>n)
        u=m-1;
    else
        l=m+1;
}
if(!f)
    System.out.println("Search element not found");
}
}

```

**76. The annual examination results of 50 students in a class are tabulated as follows:**

| Roll no. | Subject A | Subject B | Subject C |
|----------|-----------|-----------|-----------|
| -----    | -----     | -----     | -----     |

**Write a program to read the data, calculate and display the following: (a) Average mark obtained by each student, (b) Print the roll number and average marks of the students whose average mark is above 80, (c) Print the roll number and average marks of the students whose average mark is below 40. [ICSE 2009]**

**Ans.** import java.util.\*;

class Question76

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int rollno[]=new int[50];
        int subA[]=new int[50];
        int subB[]=new int[50];
        int subC[]=new int[50];
        float avg[]=new float[50];
        int i;
        System.out.println("Enter the roll no. and marks in 3 subjects for 50 students:");
    }
}

```



```

for(i=0;i<rollno.length;i++)
{
    rollno[i]=sc.nextInt();
    subA[i]=sc.nextInt();
    subB[i]=sc.nextInt();
    subC[i]=sc.nextInt();
    sc.nextLine();//dummy input
}
System.out.println("Average of each student:");
for(i=0;i<rollno.length;i++)
{
    avg[i]=(float)(subA[i]+subB[i]+subC[i])/3;
    System.out.println(rollno[i]+" \t\t"+avg[i]);
}
System.out.println("Students whose average is more than 80:");
for(i=0;i<rollno.length;i++)
{
    if(avg[i]>80)
        System.out.println(rollno[i]+" \t\t"+avg[i]);
}
System.out.println("Students whose average is less than 40:");
for(i=0;i<rollno.length;i++)
{
    if(avg[i]<40)
        System.out.println(rollno[i]+" \t\t"+avg[i]);
}
}
}

```

**77. Create a method which accepts an integer as parameter and return true if it comes under the Fibonacci series, otherwise return false. Now use it another method which accepts a two-dimensional integer array as parameter and check whether all elements in it comes under the Fibonacci series or not.**

**Ans.** class Question77

```

{
    static boolean isFibo(int n)
    {
        int a=1,b=0,c=0;
        while(c<n)
        {
            c=a+b;
            a=b;
            b=c;
        }
        if(c==n)
            return true;
    }
}

```

```

    else
        return false;
}

static void main(int a[])
{
    int i,f=0;
    for(i=0;i<a.length;i++)
    {
        if(isFibo(a[i])==false)
            f=1;
    }
    if(f==0)
        System.out.println("All are fibonacci numbers");
    else
        System.out.println("All are not fibonacci numbers");
}
}

```

**78. Write a program to store 6 elements in an array P, and 4 elements in an array Q and produce a third array R, containing all elements of array P and Q. Display the resultant array. [ICSE 2010]**

**EXAMPLE:**

| INPUT | OUTPUT |       |
|-------|--------|-------|
| P [ ] | Q [ ]  | R [ ] |
| 4     | 19     | 4     |
| 6     | 23     | 6     |
| 1     | 7      | 1     |
| 2     | 8      | 2     |
| 3     |        | 3     |
| 10    |        | 10    |
|       |        | 19    |
|       |        | 23    |
|       |        | 7     |
|       |        | 8     |

**Ans.** class Question78

```

{
    static void main()
    {
        int P[]={4,6,1,2,3,10};
        int Q[]={19,23,7,8};
        int R[]=new int[P.length+Q.length];
        int i,c=0;
        for(i=0;i<P.length;i++)
        {
            R[c++]=P[i];
        }
    }
}

```

```

    for(i=0;i<Q.length;i++)
    {
        R[c++]=Q[i];
    }

    for(i=0;i<R.length;i++)
    {
        System.out.println(R[i]);
    }
}
}

```

**79. Write a program to enter 10 words into a String type array and print only those words which end with 'A' or 'a'.**

**Ans.** import java.util.\*;  
class Question79  
{  
 static void main()  
 {  
 String s[]=new String[10];  
 int i;  
 Scanner sc=new Scanner(System.in);  
 System.out.println("Enter 10 words:");  
 for(i=0;i<10;i++)  
 {  
 s[i]=sc.nextLine();  
 }  
 System.out.println("Words that end with A or a:");  
 for(i=0;i<10;i++)  
 {  
 char x=s[i].charAt(s[i].length()-1);  
 if(x=='a' || x=='A')  
 System.out.println(s[i]);  
 }  
 }  
}

**80. Write a program to input 10 words into a String type array and print only those words which begin and end with the same alphabet.**

**Ans.** import java.util.\*;  
class Question80  
{  
 static void main()  
 {  
 String s[]=new String[10];  
 int i;  
 Scanner sc=new Scanner(System.in);

```

System.out.println("Enter 10 words:");
for(i=0;i<10;i++)
{
    s[i]=sc.nextLine();
}
System.out.println("Words that begins and ends with same alphabet:");
for(i=0;i<10;i++)
{
    char x=s[i].charAt(s[i].length()-1);
    char y=s[i].charAt(0);
    if(x==y)
        System.out.println(s[i]);
}
}
}

```

**81. Write a program to input 10 words into a String type array and print the longest word.**

**Ans.** import java.util.\*;  
class Question81  
{  
 static void main()  
 {  
 String s[]=new String[10],l="";  
 int i;  
 Scanner sc=new Scanner(System.in);  
 System.out.println("Enter 10 words:");  
 for(i=0;i<10;i++)  
 {  
 s[i]=sc.nextLine();  
 }  
 for(i=0;i<10;i++)  
 {  
 if(i==0)  
 l=s[i];  
 if(s[i].length()>l.length())  
 l=s[i];  
 }  
 System.out.println("Longest Word="+l);  
 }  
}

# Chapter 14: Console Input / Output

## Practice Questions

### A. Answer the following questions:

**1. What is the function of the System class? What are the stream variables associated with the Stream?**

**Ans.** class `System` is a built-in class present in `java.lang` package, having a **final** modifier, which does not allow it to be inherited and contains pre-defined methods and fields, which provide facilities like standard input, output, etc.

The Stream variables associated with the Stream class are `System.in` and `System.out`.

**2. Explain the concept of Streams in Java.**

**Ans.** A **stream** in Java is an ordered sequence of bytes of undetermined length. Streams are ordered and in sequence so that the Java virtual machine can understand and work upon the stream. There are two categories of streams in Java, namely **Byte** stream and **Character** stream.

**3. Briefly explain the two types of streams available in Java.**

**Ans.** *Byte streams* carry integers with values that range from 0 to 255. A diverse assortment of data can be expressed in byte format, including numerical data, executable programs, Internet communications and bytecode—the class files run by a Java virtual machine.

*Character streams* are a specialized type of byte stream that handles only textual data. They are distinguished from byte streams because Java's character set supports Unicode, a standard that includes many more characters than could be expressed easily using bytes.

**4. What is the function of the BufferedReader class? In this respect explain the function of the InputStreamReader class. Also state the package under which the above classes are associated.**

**Ans.** The `BufferedReader` class read text from a character input stream, buffering characters so as to provide for the efficient reading of characters, arrays, and lines. The buffer size may be specified, or the default size may be used.

The `InputStreamReader` class is a bridge from byte streams to character streams: It reads bytes and decodes them into characters using a specified char set. The char set that it uses may be specified by name or may be given explicitly, or the platform's default character set may be accepted.

The package under which these classes are available is `java.io`.

**5. Show, with the help of an example, how character and String can be entered through console using BufferedReader class.**

**Ans.** Program to input a character from the console:

```
import java.io.*;
class readCharDemo
{
    static void test() throws IOException
    {
        char x;
        BufferedReader br=new BufferedReader(new
        InputStreamReader(System.in));
        System.out.println("Enter a character:");
```

```

        x=(char)br.read();
        /*typecasting is necessary as the method read()
        returns the unicode of the corresponding
        character entered by the user.*/
        System.out.println("You have input:"+x);
    }
}

```

Program to input a String from the console:

```

import java.io.*;
class readCharDemo
{
    static void test() throws IOException
    {
        String s;
        BufferedReader br=new BufferedReader(new
        InputStreamReader(System.in));
        System.out.println("Enter a string:");
        s=br.readLine();
        System.out.println("You have input:"+s);
    }
}

```

**6. State the difference between print( ) and println( ) functions of the System class.**

**Ans.** `System.out.print( )` is used to display a String on the console and keeps the cursor in the same line.

`System.out.println( )` is used to display a String on the console and brings the cursor to the next line.

**7. Generally all numeric data you print using System.out.print( ) and System.out.println( ) is left-aligned if you want a formatted output, where a numeric data will be right-aligned, what function would you use?**

**Ans.** `System.out.printf( )` should be used.

**8. What is the function of the Scanner class? Under which package is it available?**

**Ans.** The `Scanner` class is a class in `java.util`, which allows the user to read values of various types, from either keyboard or a file.

**9. What are tokens as identified by the Scanner class?**

**Ans.** A token as identified by the `Scanner` class is a series of characters that ends with what Java calls whitespace. A whitespace character can be a blank, a tab character, a carriage return, or the end of the file.

**10. What is a delimiter? How would you make Scanner class read data from the keyboard separated by , or ; or space?**

**Ans.** A delimiter is a character or pattern that marks the end of a token. The delimiter to make a `Scanner` class read data from the keyboard separated by , or ; or space is

```
sc.useDelimiter("[,; ]");
```

where `sc` is the name of the `Scanner` object.

**11. Give an example where you will read an integer and an entire line of text (say a sentence) using the Scanner class.**

**Ans.** `import java.util.Scanner;`

```

class Example11
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n;
        String s;
        System.out.println("Enter an integer:");
        n=sc.nextInt();
        sc.nextLine();//dummy input
        System.out.println("Enter a string:");
        s=sc.nextLine();
        System.out.println("Integer:"+n);
        System.out.println("String:"+s);
    }
}

```

**12. What is the function of hasNext( ) function of the Scanner class?**

**Ans.** The **hasNext...()** member function of the **Scanner** class checks whether the next token is the specified token or not.

**B. Write programs for the following:**

**1. Write a program to read three real numbers (double data type) from the keyboard using Buffered Reader class and find their average.**

**Ans.** import java.io.\*;

```

class Question1
{
    static void main() throws IOException
    {
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        double a,b,c,avg;
        System.out.println("Enter three real numbers:");
        a=Double.parseDouble(br.readLine());
        b=Double.parseDouble(br.readLine());
        c=Double.parseDouble(br.readLine());
        avg=(a+b+c)/3;
        System.out.println("Average="+avg);
    }
}

```

**2. Create a class named *Number* which will have the following members:**

**Data Members:** *a* and *b* of int type.

**Member Functions:**

**i) Constructor to read two integers from keyboard using Scanner class into *a* and *b*.**

**ii) Member function to display the sum of *a* and *b*.**

**Ans.** //Question 2

import java.util.\*;

class Number

```

{

```

```

int a,b;
Number()
{
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter 2 numbers:");
    a=sc.nextInt();
    b=sc.nextInt();
}

void sum()
{
    int c;
    c=a+b;
    System.out.println("Sum="+c);
}
}

```

**3. Write a program to input two integers and interchange their values:**

**i) Using a third variable.**

**ii) Without using a third variable.**

**Ans.** import java.util.\*;

class Question3

```

{
    static void swap_i()
    {
        Scanner sc=new Scanner(System.in);
        int a,b,c;
        System.out.println("Enter 2 integers:");
        a=sc.nextInt();
        b=sc.nextInt();
        System.out.println("Before swapping a="+a+" b="+b);
        c=a;
        a=b;
        b=c;
        System.out.println("After swapping a="+a+" b="+b);
    }

    static void swap_ii()
    {
        Scanner sc=new Scanner(System.in);
        int a,b;
        System.out.println("Enter 2 integers:");
        a=sc.nextInt();
        b=sc.nextInt();
        System.out.println("Before swapping a="+a+" b="+b);
        a=a+b;
        b=a-b;
        a=a-b;
    }
}

```



```

        System.out.println("After swapping a="+a+" b="+b);
    }
}

```

**4. Write a program to input an integer and a digit and count the number of times the digit is present in the integer.**

**Ans.** import java.util.\*;  
class Question4  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int n,digit,c=0,d;  
 System.out.println("Enter an integer:");  
 n=sc.nextInt();  
 System.out.println("Enter a Digit:");  
 digit=sc.nextInt();  
 while(n!=0)  
 {  
 d=n%10;  
 if(d==digit)  
 c++;  
 n=n/10;  
 }  
 System.out.println("No. of times the digit is present is:"+c);  
 }  
}

**5. Write a program to input 10 integers in an array with scanner using ',' (comma) as delimiter and find the largest integer.**

**Ans.** import java.util.\*;  
class Question5  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 sc.useDelimiter(",");  
 int a[]=new int[10];  
 int i,l=0;  
 System.out.println("Enter 10 integers:");  
 for(i=0;i<10;i++)  
 {  
 a[i]=sc.nextInt();  
 if(i==0)  
 l=a[i];  
 if(a[i]>l)  
 l=a[i];  
 }  
 System.out.println("Largest Number:"+l);  
 }  
}

```
}  
}
```

**6. Write a program to input the roll numbers and names of 10 students and print it according to descending order of roll number.**

```
Ans. import java.util.*;  
class Question6  
{  
    static void main()  
    {  
        Scanner sc=new Scanner(System.in);  
        int roll[]=new int[10];  
        String name[]=new String[10],tn="";  
        int i,j,tr;  
        System.out.println("Enter roll and name of 10 students:");  
        for(i=0;i<10;i++)  
        {  
            roll[i]=sc.nextInt();  
            sc.nextLine();//dummy input  
            name[i]=sc.nextLine();  
        }  
        for(i=0;i<9;i++)  
        {  
            for(j=i+1;j<10;j++)  
            {  
                if(roll[i]>roll[j])  
                {  
                    tr=roll[i];  
                    roll[i]=roll[j];  
                    roll[j]=tr;  
                    tn=name[i];  
                    name[i]=name[j];  
                    name[j]=tn;  
                }  
            }  
        }  
        System.out.println("Ascending order of roll no.:");  
        for(i=0;i<10;i++)  
        {  
            System.out.println(roll[i]+"\\t\\t"+name[i]);  
        }  
    }  
}
```

**7. Write a program to input a character and 10 words into a String array and count the number of words that begin with the given character.**

```
Ans. import java.util.*;  
class Question7  
{
```

```

static void main()
{
    Scanner sc=new Scanner(System.in);
    String words[]=new String[10];
    int i;
    char x;
    System.out.println("Enter a character:");
    x=sc.next().charAt(0);
    sc.nextLine();//dummy input
    System.out.println("Enter 10 words:");
    for(i=0;i<10;i++)
    {
        words[i]=sc.nextLine();
    }
    System.out.println("Words that begin with "+x+" are:");
    for(i=0;i<10;i++)
    {
        if(words[i].charAt(0)==x)
            System.out.println(words[i]);
    }
}
}

```

**8. Using Scanner, write a program to input an integer and find the sum of its digits. The program should ensure that the user enters only an integer. In case, data of any other type is entered by the user, the user is again given a chance to input an integer, the process continues until the user inputs an integer, whose sum of the digits is found.**

**e.g. INPUT**

**Enter an integer;**

**46.3**

**Enter an integer:**

**ABC**

**Enter an integer:**

**569**

**OUTPUT**

**Sum of the digits of 569 is = 20**

**Ans.** import java.util.\*;

class Question8

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n=0,d,s=0;
        boolean f;
        do
        {
            f=false;

```

```

System.out.println("Enter an integer:");
if(sc.hasNextInt())//check whether the token is an integer
{
    n=sc.nextInt();
    f=true;
}
else
    sc.next();//skip the token
}while(f==false);
while(n!=0)
{
    d=n%10;
    s+=d;
    n=n/10;
}
System.out.println("Sum of the digits="+s);
}
}

```

**9. Write a program to input an integer (of Binary Number System) and check whether it is a valid binary number or not. A number is said to be a Binary Number because all digits in it contains only 0 and 1, any other digit other than 0 and 1, it is considered to be a valid Binary Number.**

**Ans.** import java.util.\*;

```

class Question9
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n,d;
        boolean f=true;
        System.out.println("Enter a binary number:");
        n=sc.nextInt();
        while(n!=0)
        {
            d=n%10;
            if(d!=1 && d!=0)
                f=false;
            n=n/10;
        }
        if(f)
            System.out.println("Valid binary number");
        else
            System.out.println("Not a Valid binary number");
    }
}
}

```

# Chapter 15: Encapsulation

## Practice Questions

### ----->Objective-Type Questions<-----

#### A. State whether true or false:

1. A class is used to implement encapsulation. [True]
2. Instance variables are used to represent behaviour of an object. [False]
3. You need to prefix the keyword default before data members which should have default/friendly access specifier. [False]
4. The access specifier private allows accessibility of a member only within the class where the member is declared/defined, but not outside the class. [True]
5. Java source code should always have the .jav extension. [False]

#### B. Fill in the blanks:

1. The access specifier protected allows accessibility within the same package by both classes and sub-classes but only by sub-classes in a different package.
2. In case there are multiple classes in a source code the class from where the program should start execution should have the access specifier as public.
3. The private access specifier is not allowed for a **class**.
4. The extends keyword is used to perform inheritance in Java.
5. The keyword used to include a class in a package is package.

#### C. Choose the correct answer:

1. Which among the following allows global accessibility?  
a) public                      b) private                      c) protected                      d) All of these

Ans. a) public

2. Which among the following allows accessibility only within the class where the member is defined or declared?  
a) public                      b) private                      c) protected                      d) All of these

Ans. b) private

3. In case, a source code has multiple classes; the **class** from which the program begins should have the \_\_\_\_\_ access specifier.  
a) public                      b) private                      c) protected                      d) All of these

Ans. a) public

4. Which among the following is true, if a constructor is declared as **private** for a **class**?  
a) Another **class** may declare the object of that **class**.  
b) An object may be declared only within that **class**.  
c) An object may be declared in a different package.  
d) None of these.

Ans. b) An object may be declared only within that class.

5. Which among the following is not true for inheritance?  
a) The class which is being inherited is called the super-class and the class that inherits it is called the sub-class.  
b) The **extends** keyword is used to perform inheritance.  
c) private members of a super-class is also inherited in the sub-class.

d) The member functions of a super-class can manipulate the values of the data-members of a sub-class.

**Ans. d) The member-functions of a super-class can manipulate the values of the data-members of a sub-class.**

**6.** Which among the following statements is true with respect to packages?

a) The command **package <package-name>**; should be the given as the first line inside the class.

b) The command **package <package-name>**; should be the given just after the import command.

c) The command **package <package-name>**; should be the given as the first line even before the import statement.

d) The command **package <package-name>**; may be provided anywhere in the source code.

**Ans. c) The command package <package-name>; should be the given as the first line even before the import statement.**

**7.** Which operator is used to access individual subpackages and classes in a package?

a) + operator                      b) new operator                      c) this operator                      d) dot operator

**Ans. d) dot operator**

**8.** Which operator is used to import all classes within a package?

a) \*                                      b) all                                      c) +                                      d) None of these

**Ans. a) \***

**9.** Which menu among the following is used to create a package in BlueJ?

a) Project                              b) Edit                                      c) Tools                                      d) Help

**Ans. b) Edit**

**10.** Which among the following access specifiers allows accessibility outside that package?

a) public and private                                      b) public and protected  
c) public and default                                      d) private and protected.

**Ans. b) public and protected**

## ----->Subjective-Type Questions<-----

**Answer the following questions:**

**1. What is a wrapper class? Give few examples of wrapper classes.**

**Ans.** The wrapper class encapsulate or wrap the primitive data-types within a class to form an object representation of it.

Examples of wrapper classes

| Wrapper Class | Primitive Data Type |
|---------------|---------------------|
| Boolean       | boolean             |
| Byte          | byte                |
| Short         | short               |
| Long          | long                |
| Integer       | int                 |
| Float         | float               |
| Double        | double              |
| Character     | char                |

## 2. What are access specifiers?

**Ans.** Access specifiers are keywords which is used to declare which entity cannot be accessed from where. Its effect has different consequences when used on a class, class member (variable or method), constructor. java offers four access specifiers, listed below in decreasing order of accessibility:

- public
- protected
- default/friendly
- private

## 3. Explain the statement:

**Character c='A';**

**Ans.** An object named 'c' of the wrapper class Character is being initialized with the character constant 'A'.

## 4. Given a statement:

**String s= "123";**

**Write a statement so that the numeric value of the variable s may be stored in an integer.**

**int a=?**

**Ans.** int a=Integer.parseInt(s);

## 5. Given a statement:

**double d=12.36;**

**Write a statement so that the numeric value of the variable d may be stored in a String variable.**

**String a=?**

**Ans.** String a=Double.toString(d);

**6. Name the wrapper class function that converts an integer to a:**

**i) Binary Number**

**ii) Octal Number**

**iii) Hexadecimal Number**

**Ans.** i) Integer.toString( );

ii) Integer.toOctalString( );

iii) Integer.toHexString( );

**7. State the difference between parse...( ) and valueOf() method.**

**Ans.** The **parse...( )** method of the wrapper class converts a String to its corresponding primitive data type and returns it whereas **valueOf()** method converts a String to the corresponding wrapper object and returns it.

**8. State the similarity difference between private and default access specifier.**

**Ans.** Similarity between private and default access specifier is that both allows accessibility within the class.

Difference between private and default is that private access specifier allows accessibility only within the class and default access specifier allows accessibility to all classes within a package.

**9. What is a package? State its significance.**

**Ans.** Package is a folder that contains compiled classes having both a name and a visibility control mechanism for using its function in another class.

A package gives you an organized way of managing classes in Java. You can group related classes into a package thus making it more organized. Similarly it is possible to define classes inside a package that are not accessible by code outside the package. There may also be classes that are only exposed to other members of the same package. This allows the classes to have intimate knowledge of each other, but do not expose it to the entire world.

**10. Why can't a class be declared as private?**

**Ans.** A class when defined as private, it will be impossible to access its members and thus making it unusable.

**11. Write the statement to import all classes from the Simple package.**

**Ans.** import Simple.\*;

## Lab Exercises

**Write programs for the following:**

**1. Create a class name Sentence having the following:**

**private data members: s of String type.**

**public member functions:**

**i) Parameterized constructor to initialize it with a sentence.**

**ii) To accepts a word as parameter and return true if it is palindrome else return false.**

**iii) To print the longest palindrome word in s.**

**Create another public class called Main and input a sentence and print the longest word in the sentence using the above class.**

**Ans.** //Question 1

```
import java.util.*;
class Sentence
{
    private String s;
    public Sentence(String t)
    {
```



```

        s=t;
    }

    public boolean isPalindrome(String w)
    {
        int i;
        char x;
        String r="";
        for(i=0;i<w.length();i++)
        {
            x=w.charAt(i);
            r=x+r;
        }
        if(r.equalsIgnoreCase(w))
            return true;
        else
            return false;
    }

    public void longestPalindrome()
    {
        int i;
        char x;
        String w="",l="";
        s=s.trim();
        s+=" ";
        for(i=0;i<s.length();i++)
        {
            x=s.charAt(i);
            if(x!=' ')
                w=w+x;
            else
            {
                if(isPalindrome(w))
                {
                    if(w.length()>l.length())
                        l=w;
                }
                w="";
            }
        }
        System.out.println("Longest Palindrome Word="+l);
    }

    public static void main(String args[])
    {

```

```

String m;
Scanner sc=new Scanner(System.in);
System.out.println("Enter a sentence:");
m=sc.nextLine();
Sentence ob=new Sentence(m);
ob.longestPalindrome();
}
}

```

**2. Define a class Employee having the following description: [ICSE 2008]**

**Data Members:**

|                                |                                         |
|--------------------------------|-----------------------------------------|
| <b>int <i>pan</i></b>          | <b>to store personal account number</b> |
| <b>String <i>name</i></b>      | <b>to store name</b>                    |
| <b>double <i>taxincome</i></b> | <b>to store annual taxable income.</b>  |
| <b>double <i>tax</i></b>       | <b>to store tax that is calculated</b>  |

**Member functions:**

|                  |                                                                     |
|------------------|---------------------------------------------------------------------|
| <b>input()</b>   | <b>: Store the <i>pan</i> number, <i>name</i>, <i>taxincome</i></b> |
| <b>calc()</b>    | <b>: Calculate <i>tax</i> for an employee</b>                       |
| <b>display()</b> | <b>: Output details of an employee</b>                              |

**Write a program to compute the tax according to the given conditions and display the output as per given format.**

**Total Annual Taxable Income Tax Rate**

**Upto Rs. 1,00,000 No tax**

**From 1,00,001 to 1,50,000 10% of the income exceeding Rs.1,00,000**

**From 1,50,001 to 2,50,000 Rs.5000+20% of the income exceeding Rs.1,50,000**

**Above Rs.2,50,000 Rs. 25,000+30% of the income exceeding Rs.2,50,000**

|               |            |               |             |                   |            |
|---------------|------------|---------------|-------------|-------------------|------------|
| <b>Ouput:</b> | <b>Pan</b> | <b>Number</b> | <b>Name</b> | <b>Tax-income</b> | <b>Tax</b> |
|               | ----       | ----          | ----        | -----             | ----       |
|               | ----       | ----          | ----        | -----             | ----       |

**Ans.** //Question 2

```

import java.util.*;
class Employee
{
    int pan;
    String name;
    double taxincome,tax;

    void input()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the pan no.:");
        pan=sc.nextInt();
        sc.nextLine();//dummy input
        System.out.println("Enter your name:");
        name=sc.nextLine();
        System.out.println("Enter the annual taxable income:");
        taxincome=sc.nextDouble();
    }
}

```

```

void calc()
{
    if(taxincome<=100000)
        tax=0;
    else if(taxincome<=150000)
        tax=10/100.0*(taxincome-100000);
    else if(taxincome<=250000)
        tax=5000+20/100.0*(taxincome-150000);
    else
        tax=25000+30/100.0*(taxincome-250000);
}

void display()
{
    System.out.println("Pan Number\t\tName\t\tTax-income\t\tTax");
    System.out.println(pan+"\t\t"+name+"\t\t"+taxincome+"\t\t"+tax);
}
}

```

**3. Define a class Salary described as below:-**

**private data Members: Name, Address, Phone, Subject Specialization, Monthly Salary, Income Tax.**

**public Member methods:**

**i) To accept the details of a teacher including the monthly salary.**

**ii) To display the details of the teacher.**

**iii) To compute the annual Income Tax as 5% of the annual salary above Rs.1,75,000/-.**

**Write a main method to create object of the class and call the above member method.**

**Ans.** //Question 3

```

import java.util.*;
class Salary
{
    private long phone;
    private String name,address,subjectSpecialization;
    private double monthlySalary,incomeTax;

    public void input()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the phone no.:");
        phone=sc.nextLong();
        sc.nextLine();//dummy input
        System.out.println("Enter the name:");
        name=sc.nextLine();
        System.out.println("Enter the address:");
        address=sc.nextLine();
        System.out.println("Enter the Subject Specialization:");
    }
}

```

```

        subjectSpecialization=sc.nextLine();
        System.out.println("Enter the monthly salary:");
        monthlySalary=sc.nextDouble();
    }

    public void display()
    {
        System.out.println("Phone no.:"+phone);
        System.out.println("Name:"+name);
        System.out.println("Address:"+address);
        System.out.println("Subject Specialization:"+subjectSpecialization);
        System.out.println("Monthly salary:"+monthlySalary);
        System.out.println("Income Tx:"+incomeTax);
    }

    public void calc()
    {
        double annualSalary;
        annualSalary=12*monthlySalary;
        if(annualSalary>175000)
            incomeTax=5/100.0*(annualSalary-175000);
        else
            incomeTax=0;
    }

    public static void main(String args[])
    {
        Salary ob=new Salary();
        ob.input();
        ob.calc();
        ob.display();
    }
}

```

**4. A special number is a number in which the sum of the factorial of each digit is equal to the number itself. For example,  $145=1!+4!+5! =1+24+120$**

**Design a class Special to check if a given number is a special number using the given members:**

|                             |                                                                   |
|-----------------------------|-------------------------------------------------------------------|
| <b>Class name</b>           | <b>: Special</b>                                                  |
|                             | <b>Data members/instance variables</b>                            |
| <b><i>n</i></b>             | <b>: Integer</b>                                                  |
| <b>Member functions:</b>    |                                                                   |
| <b>Special()</b>            | <b>: constructor to assign 0 to <i>n</i></b>                      |
| <b>Special(int)</b>         | <b>: parameterized constructor to assign a value to 'n'.</b>      |
| <b>int factorial(int p)</b> | <b>: calculate and return the factorial of <i>p</i>.</b>          |
| <b>void is Special()</b>    | <b>: check and display if the number 'n' is a special number.</b> |

**Also create another class named Main to input a number and check whether the number is a Special Number or not.**

**Ans.** //Question 4

```
import java.util.*;
class Special
{
    int n;

    Special()
    {
        n=0;
    }

    Special(int x)
    {
        n=x;
    }

    int factorial(int p)
    {
        int i,f=1;
        for(i=1;i<=p;i++)
            f=f*i;

        return f;
    }

    void isSpecial()
    {
        int t=n,d,s=0;
        while(t!=0)
        {
            d=t%10;
            s=s+factorial(d);
            t=t/10;
        }
        if(s==n)
            System.out.println("Special Number");
        else
            System.out.println("Not a Special Number");
    }
}

public class Main
{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
```

```

        int n;
        System.out.println("Enter a number:");
        n=sc.nextInt();
        Special ob=new Special(n);
        ob.isSpecial();
    }
}

```

**5. A class Modify has been defined with the following details:**

**Class name : Modify**

**Data members:**

**St : stores a string**

**Len : to store the length of the string**

**Member Functions:**

**void read() : to accept the string in Uppercase alphabets**

**void putin(int,char) : to insert a character at the specified position in the string and display the changed string.**

**void takeout(int) : to remove character from the specified position in the string and display the changed string**

**void change() : to replace each character in the original string by the character which is at a distance of 2 moves ahead.**

**For example,**

**“ABCD” becomes “CDEF”**

**“XYZ” becomes “ZAP”**

**Specify the class Modify giving details of the functions void read(), void putin(int, char), void takeout(int) and void change(). The main function need not be written.**

**Ans.** //Question 5

```

import java.util.*;
class Modify
{
    String st;
    int len;

    void read()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter a string:");
        st=sc.nextLine().toUpperCase();
        len=st.length();
    }

    void putin(int p,char x)
    {
        st=st.substring(0,p)+x+st.substring(p);
    }

    void takeout(int p)
    {
        st=st.substring(0,p)+st.substring(p+1);
    }
}

```

```

    }

    void change()
    {
        int i;
        char x;
        String tst="";
        for(i=0;i<len;i++)
        {
            x=st.charAt(i);
            if(x=='Y')
                x='A';
            else if(x=='Z')
                x='C';
            else
                x=(char)(x+2);
            tst+=x;
        }
        st=tst;
    }
}

```

**6. Design a class named *Myclass* inside a package named *myPackage*, which will contain two protected integer data members and a member function to add these two data members and display them.**

**Now inherit this class into another class named *SubClass* outside *myPackage*. Create member functions to initialize the two inherited data members and display their sum. Also include a `main()` to show its implementation.**

**Ans.** //Question 6

```

package myPackage;

public class Myclass
{
    protected int a,b;
    protected void sum()
    {
        int c;
        c=a+b;
        System.out.println(c);
    }
}

```

//Question 6

```

import myPackage.Myclass;
class SubClass extends Myclass
{
    void initialize(int x,int y)
    {

```

```

        a=x;
        b=y;

    }

    public static void main(String args[])
    {
        SubClass ob=new SubClass();
        ob.initialize(5,6);
        ob.sum();
    }
}

```

**7. Create a class with the following members:**

**class name: Numbers**

**data member: int ar[ ]**

**Member Functions:**

**i) Constructor to allocate 20 spaces for ar.**

**ii) Generate twenty unique random numbers between 1 to 20 and store it in ar[ ].**

**Display the contents of ar[ ].**

**Also create another class named Main to implement the above class.**

**Ans.** //Question 7

```

class Numbers
{
    int ar[];
    Numbers()
    {
        ar=new int[20];
    }

    void generate()
    {
        int i,j,f,r;
        for(i=0;i<20;i++)
        {
            f=0;
            r=1+(int)(Math.random()*20);
            for(j=0;j<i;j++)
            {
                if(ar[j]==r)
                    f=1;
            }
            if(f==0)
                ar[i]=r;
            else
                i--;
        }
    }
}

```



```

void display()
{
    int i;
    for(i=0;i<20;i++)
    {
        System.out.print(ar[i]+" ");
    }
}

public static void main(String args[])
{
    Numbers ob=new Numbers();
    ob.generate();
    ob.display();
}
}

```

**8. Design a class with the following properties**

**Class Name: Student**

**Data Members: *roll*: of int type**

***name*: of String type**

**Member Functions:**

**i) Parameterized constructor to initialize *roll* and *name*.**

**ii) to display only the *name*.**

**In the main () create an array of 10 objects initialize it by taking data through user input. Now input a roll number and check which roll within the array of objects matches. Upon matching print, the corresponding name, otherwise print a relevant message.**

**Ans. //Question 8**

```

import java.util.*;
class Student
{
    int roll;
    String name;

    Student(int r,String n)
    {
        roll=r;
        name=n;
    }

    void display()
    {
        System.out.println(name);
    }

    public static void main(String args[])

```

```

{
    Student ob[]=new Student[10];
    int i,r,f=0;
    String n;
    Scanner sc=new Scanner(System.in);
    for(i=0;i<10;i++)
    {
        System.out.println("Enter Roll No.:");
        r=sc.nextInt();
        sc.nextLine();//dummy input
        System.out.println("Enter Name:");
        n=sc.nextLine();
        ob[i]=new Student(r,n);
    }

    System.out.println("Enter a roll no. to search:");
    r=sc.nextInt();
    for(i=0;i<10;i++)
    {
        if(ob[i].roll==r)
        {
            ob[i].display();
            f=1;
        }
    }
    if(f==0)
        System.out.println("Search Unsuccessful");
}
}

```

**9. Design a class with the following properties**

**Class Name: Number System**

**Data Members: *a* and *b* of int data type.**

**Member Functions:**

**i) Parameterized constructor to initialize *a* and *b* with two binary numbers.**

**ii) boolean isBinary(int *n*) to check whether *n* is a binary number or not.**

**iii) Add the two binary numbers and display their result only if both the numbers are binary.**

**In the main () create an object initialize it with two binary numbers and find their sum after checking whether both the numbers are binary numbers or not.**

```

Ans. import java.util.*;
//Question 9
class NumberSystem
{
    int a,b;
    NumberSystem(int x,int y)
    {
        a=x;

```

```

        b=y;
    }

    boolean isBinary(int n)
    {
        int d;
        boolean f=true;
        while(n!=0)
        {
            d=n%10;
            if(d!=1 && d!=0)
                f=false;
            n=n/10;
        }
        return f;
    }

    void add()
    {
        int d1,d2,s=0,c=0,carry=0,x;
        if(isBinary(a) && isBinary(b))
        {
            while(a!=0 || b!=0)
            {
                d1=a%10;
                d2=b%10;
                x=d1+d2+carry;
                if(x>=2)
                {
                    x=x-2;
                    carry=1;
                }
                s=s+x*(int)Math.pow(10,c++);
                a=a/10;
                b=b/10;
            }
            s=s+carry*(int)Math.pow(10,c);
            System.out.println("Sum="+s);
        }
        else
            System.out.println("Not a valid binary number");
    }

    public static void main(String args[])
    {
        int x,y;

```

```
Scanner sc=new Scanner(System.in);
System.out.println("Enter 2 valid binary numbers:");
x=sc.nextInt();
y=sc.nextInt();
NumberSystem ob=new NumberSystem(x,y);
ob.add();
    }
}
```

# Chapter 16:

# Errors

## Practice Questions

### A. Answer the following questions:

#### 1. What is a bug?

**Ans.** Bug is any error that may have crept in a program because of typographical error or logical errors or some run-time errors.

#### 2. State the different types of errors you might encounter during coding.

**Ans.** a) Syntax/Semantic error  
b) Logical error  
c) Run-time error

#### 3. Give three examples when you will get compile-time error.

**Ans.** Compile time errors may be generated in the following situations:

- Missing semicolon after a statement.
- A keyword is used as an identifier.
- A keyword is mis-spelt.

#### 4. What are run-time errors?

**Ans.** Run-time errors occur during execution of a program when an exceptional situation may occur, which disrupts the proper execution of the program.

#### 5. What are Exceptions? What do you understand by the statement "throwing an exception"?

**Ans.** An exception is an event, which occurs during the execution of a program, that disrupts the normal flow of the program's instructions. Whenever an error occurs during the execution of a method, the method creates an object and hands it off to the run-time system. The object, called an exception object, contains information about the error, including its type and the state of the program when the error occurred.

During execution of a program, if a run-time error occurs, the method in which it occurs creates an *Exception* object and passes it on to the Java run-time system. This is called **throwing** an object.

#### 6. What do you understand by the statement "A Java code must honour the Catch/Specify Requirement"?

**Ans.** A Java program must honour the Catch or Specify Requirement. This means that code that might throw certain exceptions must be enclosed by either of the following:

- A try statement that catches the exception.
- A method that specifies that it can throw the exception. The method must provide a throws clause that lists the exception.

#### 7. What is an Exception Handler?

**Ans.** The run-time system searches the call stack for a method that contains a block of code that can handle the exception. This block of code is called an exception handler.

#### 8. What are the different types of exceptions?

**Ans.** There are two types of exceptions:

- Checked Exceptions
- Unchecked Exceptions

**9. Give three common Exception class names.**

- Ans.** i) ArithmeticException  
ii) ArrayIndexOutOfBoundsException  
iii) InputMismatchException

**10. State the difference between throw and throws.**

**Ans.**

| <b>throws</b>                                                                                                                  | <b>throw</b>                                                                                  |
|--------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|
| If a method is capable of causing an exception that it doesn't handle, it must specify this behaviour using the throws clause. | The <b>throw</b> clause is used by the programmer to deliberately invoke an exception object. |

**11. What is Encapsulation?**

**Ans.** Wrapping up of characteristics (represented by data members) and behaviour (represented by member functions) into a single unit (using a class) is called Encapsulation.

**12. What are the different access specifiers available in Java?**

- Ans.** i) default  
ii) private  
iii) private  
iv) protected

**Ans. 13. State the difference between:**

- i) **public and protected access specifier.**  
ii) **private and default access specifier.**

i)

| <b>public</b>                                                                             | <b>protected</b>                                                                                   |
|-------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| It allows accessibility by all classes within the package as well as outside the package. | It allows accessibility by classes within the package but only by sub-classes outside the package. |

ii)

| <b>private</b>                                                         | <b>default</b>                                             |
|------------------------------------------------------------------------|------------------------------------------------------------|
| It allows accessibility only within the class of which it is a member. | It allows accessibility by all classes within the package. |

**14. What happens when a constructor is given a private access?**

**Ans.** When a constructor is given a private access, its object can only be created within the class of which it is a member.

**15. Give an example to show how overloaded constructors can have different access specifiers.**

**Ans.** public class Example  
{  
    int a,b;  
    Example()  
    {  
        a=b=0;  
    }  
    private Example(int x)  
    {

```

        a=b=x;
    }

    public Example(int x,int y)
    {
        a=x;
        b=y;
    }

    protected Example(int x,int y,int z)
    {
        a=x;
        b=y+z;
    }

    void show()
    {
        System.out.println(a+"\t"+b);
    }
}

```

**16. What is inheritance? Which command is used for inheritance in Java?**

**Ans.** Inheritance is the property by which all members of one class becomes members of another class. The class which is being inherited is called the base or super class and the class which inherits it is called the derived or sub-class.

The command used for inheritance is 'extends'.

**17. What are packages? Which package is imported by default in Java?**

**Ans.** Package is a folder that contains compiled classes having both a name and a visibility control mechanism for using its function in another class.

The package which is imported by default is java.lang.

**18. State the conditions necessary to make a class part of a package.**

**Ans.** To make a class part of a package, the first line of the source code should include the code:

```
package <package-name>;
```

**19. What is the function of import command in Java? Show with the help of an example how a single class within a package is imported. Also state how all classes may be imported in a program.**

**Ans.** The import command in Java is used to include a class or group of classes that are within the package for its functions and variables to be reused within a class.

To import a single class of a package, use the following syntax:

```
import <package-name>.<class-name>;
```

To import an all classes of a package, use the following syntax:

```
import <package-name>.*;
```

**20. What is a Wrapper class?**

**Ans.** For every primitive data-type there is corresponding class that wraps around the primitive data-type to form an object representation of it. This is called Wrapper class.

**B. Write programs for the following:**

**1. Write a program to input integers and find their sum until the user inputs a character. In that case an exception should be raised and ultimately the sum of the integers should be printed.**

```
Ans. import java.util.*;
class Question1
{
    static void main()
    {
        int n,s=0;
        Scanner sc=new Scanner(System.in);
        try
        {
            while(true)
            {
                System.out.println("Enter an integer:");
                n=sc.nextInt();
                s=s+n;
            }
        }
        catch(InputMismatchException e)
        {
            System.out.println("Enter only integers!");
        }
        finally
        {
            System.out.println("Sum="+s);
        }
    }
}
```

**2. Write a program to input two integers and find the quotient and the remainder upon dividing the Larger number by the Smaller number.**

```
Ans. import java.util.*;
class Question2
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int a,b;
        float c;
        try
        {
            System.out.println("Enter two integers:");
            a=sc.nextInt();
            b=sc.nextInt();
```



```

        c=(float)a/b;
        System.out.println("Quotient="+c);
    }
    catch(InputMismatchException e)
    {
        System.out.println("Enter only Integers!");
    }
    catch(ArithmeticException e)
    {
        System.out.println("Cannot divide by 0!");
    }
}
}

```

**3. Write a program to input an integer and check whether it is 5-digit number or not, in case it is not raise a number format exception and the program should end, otherwise print it in a five-digit number.**

**Ans.** import java.util.\*;  
class Question3  
{  
 static void main()  
 {  
 Scanner sc=new Scanner(System.in);  
 int n;  
 try  
 {  
 System.out.println("Enter an integer:");  
 n=sc.nextInt();  
 if(n>=1000 && n<=99999 || n>=-99999 && n<=-1000)  
 System.out.println("It is a 5 digit number!");  
 else  
 throw new NumberFormatException();  
 }  
 catch (Exception exp)  
 {  
 System.out.println("Error: "+exp.getMessage());  
 }  
 }  
}

**4. Write a program to input a list of integers in a single line separated by comma (,) and store it in an integer array of size 10. In case the number of elements is more than the size of the array an array index out of bound exception is to be raised.**

**Ans.** import java.util.\*;  
class Question4  
{  
 static void main()  
 {

```

Scanner sc=new Scanner(System.in);
sc.useDelimiter(",");
int a[]=new int[10],i=0;
boolean f;
System.out.println("Enter 10 integers separated by comma:");
try
{
    while(true)
    {
        a[i]=sc.nextInt();
        i++;
    }
}
catch(ArrayIndexOutOfBoundsException e)
{
    System.out.println("Only 10 integers!");
}
}
}

```

**5. Write a program to continue entering integers and find the sum of even and odd integers separately unless a zero is entered. In that case throw any type of exception and the program should end.**

**Ans.** import java.util.\*;

```

class Question5
{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        int n,se=0,so=0;
        System.out.println("Enter the sum of even and odd integers:");
        try
        {
            while(true)
            {
                n=sc.nextInt();
                if (n==0)
                    throw new Exception("Only non-zero integers!");
                if(n%2==0)
                    se+=n;
                else
                    so+=n;
            }
        }
        catch(Exception e)
        {
            System.out.println("Error: "+e.getMessage());
        }
    }
}

```

```

    }
    finally
    {
        System.out.println("Sum of even numbers:"+se);
        System.out.println("Sum of odd numbers:"+so);
    }
}
}
}

```

**6. Write a program to input a line of text and search for a given word, use all relevant exceptions while using try-catch-finally block.**

**Ans.** import java.util.\*;

class Question6

```

{
    static void main()
    {
        Scanner sc=new Scanner(System.in);
        String txt,w;
        int p=0,i=0,f=0;
        try m,n
        {
            System.out.println("Enter the line of text:");
            txt=sc.nextLine();
            System.out.println("Enter the word:");
            w=sc.next();
            txt=txt.trim();
            txt=" "+txt+" ";
            w=w.trim();
            w=" "+w+" ";
            do
            {
                p=txt.indexOf(w,i);
                if(p!=-1)
                {
                    System.out.println("Found!");
                    f=1;
                    break;
                }
                i++;
            }while(i<txt.length());
            if(f==0)
                System.out.println("Not Found!");
        }
        catch(StringIndexOutOfBoundsException e)
        {
            System.out.println(e);
        }
        catch(Exception e)
    }
}

```

```

        {
            System.out.println(e);
        }
    }
}

```

**7. Create a class name Sentence having the following:**

**Data members: s of String type.**

**Member functions:**

**i) Parameterized constructor to initialize it with a sentence.**

**ii) To accept a word as parameter and return true if it is palindrome, else return false.**

**iii) To print the longest palindrome word in s.**

**Ans.** //Question 7

```
import java.util.*;
```

```
class Sentence
```

```
{
    private String s;
    public Sentence(String t)
    {
        s=t;
    }

```

```
public boolean isPalindrome(String w)
```

```
{
    int i;
    char x;
    String r="";
    for(i=0;i<w.length();i++)
    {
        x=w.charAt(i);
        r=x+r;
    }
    if(r.equalsIgnoreCase(w))
        return true;
    else
        return false;
}

```

```
public void longestPalindrome()throws StringIndexOutOfBoundsException
```

```
{
    int i;
    char x;
    String w="",l="";
    s=s.trim();
    s+=" ";
    for(i=0;i<s.length();i++)
    {

```

```

        x=s.charAt(i);
        if(x!=' ')
            w=w+x;
        else
        {
            if(isPalindrome(w))
            {
                if(w.length()>l.length())
                    l=w;
            }
            w="";
        }
    }
    System.out.println("Longest Palindrome Word="+l);
}

```

```

public static void main(String args[])
{
    String m;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter a sentence:");
    m=sc.nextLine();
    Sentence ob=new Sentence(m);
    ob.longestPalindrome();
}
}

```

**8. Define a class Employee having the following description: [ICSE 2008]**

**Data Members:**

|                                |                                  |
|--------------------------------|----------------------------------|
| <b>int <i>pan</i></b>          | to store personal account number |
| <b>String <i>name</i></b>      | to store name                    |
| <b>double <i>taxincome</i></b> | to store annual taxable income   |
| <b>double <i>tax</i></b>       | to store tax that is calculated  |

**Member functions:**

**input( )** : Store the *pan* number, *name*, *taxableincome*

**calc( )** : Calculate *tax* for an employee

**display( )** : Output details of an employee

Write a program to compute the tax according to the given conditions and display the output as per given format.

**Total Annual Taxable Income Tax Rate**

**Upto Rs.1,00,000 No tax**

**From 1,00,001 to 1,50,000 10% of the income exceeding Rs.1,00,000**

**From 1,50,001 to 2,50,000 Rs.5000+20% of the income exceeding Rs.1,50,000**

**Above Rs.2,50,000 Rs. 25,000+30% of the income exceeding Rs.2,50,000**

|                |            |               |             |                   |            |
|----------------|------------|---------------|-------------|-------------------|------------|
| <b>Output:</b> | <b>Pan</b> | <b>Number</b> | <b>Name</b> | <b>Tax-income</b> | <b>Tax</b> |
|                | ---        | ----          | ----        | -----             | ----       |
|                | ---        | ----          | ----        | -----             | ----       |

**Ans.** //Question 8

```
import java.util.*;
```

```
class Employee
```

```

{
    int pan;
    String name;
    double taxincome,tax;

    void input()throws InputMismatchException
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the pan no.:");
        pan=sc.nextInt();
        sc.nextLine();//dummy input
        System.out.println("Enter your name:");
        name=sc.nextLine();
        System.out.println("Enter the annual taxable income:");
        taxincome=sc.nextDouble();
    }

    void calc()
    {
        if(taxincome<=100000)
            tax=0;
        else if(taxincome<=150000)
            tax=10/100.0*(taxincome-100000);
        else if(taxincome<=250000)
            tax=5000+20/100.0*(taxincome-150000);
        else
            tax=25000+30/100.0*(taxincome-250000);
    }

    void display()
    {
        System.out.println("Pan Number\t\tName\t\tTax-income\t\tTax");
        System.out.println(pan+"\t\t"+name+"\t\t"+taxincome+"\t\t"+tax);
    }
}

```

**9. Define a class salary described as below:**

**Data Members: Name, Address, Phone, Subject Specialization, Monthly Salary, Income Tax.**

**Member methods:**

**i) To accept the details of a teacher including the monthly salary.**

**ii) To display the details of the teacher.**

**iii) To compute the annual Income Tax as 5% of the annual salary above Rs.1,75,000/-.**

**Write a main method to create object of the class and call the above member method.**

**Ans.** //Question 9

```
import java.util.*;
```

```

class Salary
{
    private long phone;
    private String name,address,subjectSpecialization;
    private double monthlySalary,incomeTax;

    public void input()throws InputMismatchException
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the phone no.:");
        phone=sc.nextLong();
        sc.nextLine();//dummy input
        System.out.println("Enter the name:");
        name=sc.nextLine();
        System.out.println("Enter the address:");
        address=sc.nextLine();
        System.out.println("Enter the Subject Specialization:");
        subjectSpecialization=sc.nextLine();
        System.out.println("Enter the monthly salary:");
        monthlySalary=sc.nextDouble();
    }

    public void display()
    {
        System.out.println("Phone no.:"+phone);
        System.out.println("Name:"+name);
        System.out.println("Address:"+address);
        System.out.println("Subject Specialization:"+subjectSpecialization);
        System.out.println("Monthly salary:"+monthlySalary);
        System.out.println("Income Tx:"+incomeTax);
    }

    public void calc()
    {
        double annualSalary;
        annualSalary=12*monthlySalary;
        if(annualSalary>175000)
            incomeTax=5/100.0*(annualSalary-175000);
        else
            incomeTax=0;
    }

    public static void main(String args[])
    {
        Salary ob=new Salary();
        ob.input();
    }
}

```

```

    ob.calc();
    ob.display();
}
}

```

**10. A special number is a number in which the sum of the factorial of each digit is equal to the number**

**itself. For example,  $145=1!+4!+5! =1+24+120$**

**Design a class Special to check if a given number is a special number using the given members:**

**Class name : Special**

**Data members/instance variables**

***n* : Integer**

**Member functions:**

**Special( )** : constructor to assign 0 to *n*

**Special(int)** : parameterized constructor to assign a value to 'n'.

**int factorial(int p)** : calculate and return the factorial of *p*.

**void isSpecial( )** : check and display if the number 'n' is a special number.

**Ans.** //Question 10

```
import java.util.*;
```

```
class Special
```

```
{
    int n;
```

```
    Special()
```

```
{
        n=0;
    }
```

```
    Special(int x)
```

```
{
        n=x;
    }
```

```
    int factorial(int p)throws ArithmeticException
```

```
{
        int i,f=1;
        for(i=1;i<=p;i++)
            f=f*i;
```

```
        return f;
    }
```

```
    void isSpecial()throws ArithmeticException
```

```
{
        int t=n,d,s=0;
        while(t!=0)
        {
```



```

        d=t%10;
        s=s+factorial(d);
        t=t/10;
    }
    if(s==n)
        System.out.println("Special Number");
    else
        System.out.println("Not a Special Number");
    }
}

```

**11. A class Modify has been defined with the following details:**

**Class name: Modify**

**Data members**

**St:** stores a string

**Len:** to store the length of the string

**Member Functions:**

**void read ()** : to accept the string in Uppercase alphabets

**void putin(int,char)** : to insert a character at the specified position in the string and display the changed string.

**void takeout(int)** : to remove character from the specified position in the string and display the changed string

**void change ()** : to replace each character in the original string by the character which is at a distance of 2 moves ahead. For, example

"ABCD" becomes "CDEF"

"XYZ" becomes "ZAP"

**Specify the class Modify giving details of the functions void read (), void putin(int, char), void takeout(int) and void change (). The main function need not be written.**

**Ans.** //Question 11

```
import java.util.*;
```

```
class Modify
```

```
{
```

```
    String st;
```

```
    int len;
```

```
    void read()
```

```
{
```

```
    Scanner sc=new Scanner(System.in);
```

```
    System.out.println("Enter a string:");
```

```
    st=sc.nextLine().toUpperCase();
```

```
    len=st.length();
```

```
}
```

```
    void putin(int p,char x)throws StringIndexOutOfBoundsException
```

```
{
```

```
    st=st.substring(0,p)+x+st.substring(p);
```

```
}
```

```

void takeout(int p)throws StringIndexOutOfBoundsException
{
    st=st.substring(0,p)+st.substring(p+1);
}

```

```

void change()throws StringIndexOutOfBoundsException
{
    int i;
    char x;
    String tst="";
    for(i=0;i<len;i++)
    {
        x=st.charAt(i);
        if(x=='Y')
            x='A';
        else if(x=='Z')
            x='C';
        else
            x=(char)(x+2);
        tst+=x;
    }
    st=tst;
}
}

```

**12. Design a class named *Myclass* inside a package named *myPackage*, which will contain two protected integer data members and a member function to add these two data members and display them.**

**Now inherit this class into another class named *SubClass* outside *myPackage* and initialize create member functions to initialize the two inherited data members and display their sum. Also include a main () to show its implementation.**

**Ans.** //Question 12

```
package myPackage;
```

```

public class Myclass
{
    protected int a,b;
    protected void sum()
    {
        int c;
        c=a+b;
        System.out.println(c);
    }
}

```

**Ans.** //Question 12

```

import myPackage.Myclass;
class SubClass extends Myclass

```

```

{
    void initialize(int x,int y)
    {
        a=x;
        b=y;
    }

    public static void main(String args[])
    {
        SubClass ob=new SubClass();
        ob.initialize(5,6);
        ob.sum();
    }
}

```

**13. Create a class with the following members:**

**class name** : Numbers

**data member** : int ar[ ]

**Member Functions:**

**i) Constructor to allocate 20 spaces for ar.**

**ii) Generate twenty unique random numbers between 1 to 20 and store it in ar[ ].**

**iii) Display the contents of ar[ ].**

**Ans.** //Question 13

```
class Numbers
```

```

{
    int ar[];
    Numbers()
    {
        ar=new int[20];
    }

    void generate()throws ArrayIndexOutOfBoundsException
    {
        int i,j,f,r;
        for(i=0;i<20;i++)
        {
            f=0;
            r=1+(int)(Math.random()*20);
            for(j=0;j<i;j++)
            {
                if(ar[j]==r)
                    f=1;
            }
            if(f==0)
                ar[i]=r;
            else
                i--;
        }
    }
}

```

```

    }

    void display()throws ArrayIndexOutOfBoundsException
    {
        int i;
        for(i=0;i<20;i++)
        {
            System.out.print(ar[i]+" ");
        }
    }

    public static void main(String args[])
    {
        Numbers ob=new Numbers();
        ob.generate();
        ob.display();
    }
}

```

**14. Design a class with the following properties**

**Class Name** : Student

**Data Members** : *roll*: of int type

*name*: of String type

**Member Functions:**

**i) Parameterized constructor to initialize *roll* and *name*.**

**ii) to display only the *name*.**

**In the main () create an array of 10 objects initialize it by taking data through user input. Now input a roll number and check which roll within the array of objects matches. Upon matching print, the corresponding name, otherwise print a relevant message.**

**Ans.** //Question 14

```

import java.util.*;
class Student
{
    int roll;
    String name;

    Student(int r,String n)
    {
        roll=r;
        name=n;
    }

    void display()
    {
        System.out.println(name);
    }
}

```

```

public static void main(String args[])throws
ArrayIndexOutOfBoundsException,NumberFormatException
{
    Student ob[]=new Student[10];
    int i,r,f=0;
    String n;
    Scanner sc=new Scanner(System.in);
    for(i=0;i<10;i++)
    {
        System.out.println("Enter Roll No.:");
        r=sc.nextInt();
        sc.nextLine();//dummy input
        System.out.println("Enter Name:");
        n=sc.nextLine();
        ob[i]=new Student(r,n);
    }

    System.out.println("Enter a roll no. to search:");
    r=sc.nextInt();
    for(i=0;i<10;i++)
    {
        if(ob[i].roll==r)
        {
            ob[i].display();
            f=1;
        }
    }
    if(f==0)
        System.out.println("Search Unsuccessful");
}
}

```

**15. Design a class with the following properties**

**Class Name: Number System**

**Data Members: *a* and *b* of int data type.**

**Member Functions:**

**i) Parameterized constructor to initialize *a* and *b* with two binary numbers.**

**ii) boolean isBinary(int *n*) to check whether *n* is a binary number or not.**

**iii) Add the two binary numbers and display their result only if both the numbers are binary.**

**In the main () create an object initialize it with two binary numbers and find their sum after checking whether both the numbers are binary numbers or not.**

**Ans.** import java.util.\*;

//Question 15

class NumberSystem

```

{
    int a,b;
    NumberSystem(int x,int y)

```

```
{
    a=x;
    b=y;
}
```

```
boolean isBinary(int n)throws ArithmeticException
```

```
{
    int d;
    boolean f=true;
    while(n!=0)
    {
        d=n%10;
        if(d!=1 && d!=0)
            f=false;
        n=n/10;
    }
    return f;
}
```

```
void add()throws ArithmeticException
```

```
{
    int d1,d2,s=0,c=0,carry=0,x;
    if(isBinary(a) && isBinary(b))
    {
        while(a!=0 || b!=0)
        {
            d1=a%10;
            d2=b%10;
            x=d1+d2+carry;
            if(x>=2)
            {
                x=x-2;
                carry=1;
            }
            s=s+x*(int)Math.pow(10,c++);
            a=a/10;
            b=b/10;
        }
        s=s+carry*(int)Math.pow(10,c);
        System.out.println("Sum="+s);
    }
    else
        System.out.println("Not a valid binary number");
}
```

```
public static void main(String args[])throws InputMismatchException
```

```
{
    int x,y;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter 2 valid binary numbers:");
    x=sc.nextInt();
    y=sc.nextInt();
    NumberSystem ob=new NumberSystem(x,y);
    ob.add();
}
}
```

