

Two Marks Questions with Answers**Q.1 Mention some of the separators used in Java programming .****AU : IT : Dec.-12**

Ans. : Some of the separators that are used in Java are as given below –

Name	Description
.	The period or dot is used to select a field or method from an object. It is also used to separate out the package name from sub-packages or class names.
:	The colon is used in loops after the loop label.
,	The comma is used to separate out variables or to separate out the parameters.
;	The semicolon is used to terminate the statement in Java.
()	The round opening and closing parenthesis are used to enclose the parameters of the method definition or call. It adjusts the precedence in arithmetic expression. For type casting purpose also these parenthesis are used.
{}	For enclosing the function or the class block or for array initialization these parentheses are used.
[]	These parenthesis are used in array declaration.

Q.2 How dynamic initialization of variables is achieved in java ?**AU : IT : Dec.-12**

Ans. : The variables in Java are allowed to get initialized at run time. Such type of initialization is called dynamic initialization.

For example

```
double a=5.0;
```

```
double b=20.0
```

```
double c=Math.sqrt((a+b));
```

Here the expression is evaluated at run time and its square root value is calculated and then the variable c is initialized dynamically.

Q.3 What is the output of the main method in the given code ?**AU : CSE : Dec.-13**

```
public static void main(String[] args)
{
    screen.write(sum());
}
static int sum()
{
    int A=12;
    int B=13;
    return =A+B;
}
```

Ans. : It will generate error at the statement return = A+B as “Illegal start of expression”.

Q.4 What is Java Virtual Machine ?**AU : IT : May-13, CSE : Dec.-13**

Ans. : Java Virtual Machine is a set of software and program components which takes the **byte code** as an input and generates output corresponding to underlying operating system. Due to Virtual Machine Java can achieve platform independence feature.

Q.5 What are the features of Java ?**AU : Dec.-13****Ans. :**

- (1) Java is simple to implement.
- (2) Java is platform independent.
- (3) It is an object oriented programming language.
- (4) It is a robust programming language.
- (5) Java is designed for distributed system.

Q.6 Define objects and classes in Java.**AU : CSE : Dec.-10, 18**

Ans. : An object is an instance of a class. The objects represent the real world entity. The objects are used to provide a practical basis for the real world. Each class is a collection of data and the functions that manipulate the data. The data components of class are called data fields and the function components of the class are called member functions or methods.

Q.7 Why are classes important in OO technology ?

Ans. : Classes bind together the relative data and methods in a cohesive unit. Due to this arrangement, only certain methods are allowed to access the corresponding data. Secondly, if any modification is needed, then the class can be viewed as one module and the changes made in one class does not spoil rest of the code. Moreover, finding error from such a source code becomes simple.

Hence use of class is very important thing in OO technology.

Q.8 What is the difference between object and class ?**Ans. :**

Following are some differences between the class and the object –

Sr. No.	Class	Object
1.	For a single class there can be any number of objects. For example - If we define the class as River then Ganga, Yamuna, Narmada can be the objects of the class River.	There are many objects that can be created from one class. These objects make use of the methods and attributes defined by the belonging class.

2.	The scope of the class is persistent throughout the program.	The objects can be created and destroyed as per the requirements.
3.	The class cannot be initialized with some property values.	We can assign some property values to the objects.
4.	A class has unique name.	Various objects having different names can be created for the same class.

Q.9 What is the difference between structure and class ?

Ans. :

Sr. No.	Structure	Class
1.	By default the members of structure are public.	By default the members of class are private.
2.	The structure can not be inherited.	The class can be inherited.
3.	The structures do not require constructors.	The classes require constructors for initializing the objects.
4.	A structure contains only data members.	A class contains the data as well as the function members.

Q.10 What is the difference between static and non static variables ?

AU : IT : Dec.-10

Ans. : A static variable is shared among all instances of class, whereas a non static variable (also called as instance variable) is specific to a single instance of that class.

Q.11 Define encapsulation.

AU : IT : Dec.-11

Ans. : Encapsulation means binding of data and method together in a single entity called class.

Q.12 What is an abstract class ?

AU : IT : Dec.-11,12, May-12

Ans. : When apply inheritance, the class becomes more and more specific as we move down. Sometimes a situation may occur that the superclass becomes more general and less specific. Such a class lists out only common features of other classes. Such a super class is called as **abstract class**.

Q.13 Define class. Give example.

AU : Comp : Dec.-11, IT : May-12

Ans. : Each class is a collection of data and the functions that manipulate the data. The data components of class are called data fields and the function components of the class are called member functions or methods.

For example

```
class Customer
{
    int ID;
    String Name; //Data Field
    Customer() //Constructor
    { }
    double withdraw_money() //method
    { ...}
}
```

Q.14 What is the default access to a member in a class ?

Ans. : If public or private is not used then by default the classes, methods and data fields are accessible by any class in the same package. This is called **package-private or package-access**.

Q.15 What are the benefits of encapsulation ? Should abstractions be user centric or developer-centric ?

Ans. : Due to encapsulation the corresponding data and the methods get bound together by means of class. The data inside the class is accessible by the function in the same class. It is normally not accessible from outside component. Thus the unwanted access to the data can be protected.

The abstraction should be user centric. While developing the system using the OO principles it is important to focus the user and not the developer.

Q.16 How would you declare an object of type animal named lion that takes a weight of 500 and length of 45 as parameters ?

Ans. :

```
public class Animal
{
    int weight,length;
    Animal(int wt,int len)
    {
        weight=wt;
        length=len;
    }
    void Show()
    {
        System.out.println("\n The weight of animal is: "+weight);
        System.out.println("\n The length of animal is: "+length);
    }
}
class AnimalMain
{
```

```
public static void main(String args[])
{
    Animal Lion=new Animal(500,45);
    Lion.Show();
}
}
```

Output

The weight of animal is: 500

The length of animal is: 45

Q.17 What is meant by private access specifier ?

Ans. : The private access specifier allows classes, methods and data fields accessible only from within the own class. The functions outside the class cannot access the data members or the member functions.

Q.18 What do you mean by instance variable ?

AU : CSE : May-12

Ans. : An Object is an instance of a class. The variables that the object contains are called instance variables. For example consider a class Student

```
class Student
{
    int RollNo;
    char name[10];
}
Student S; //Creation of object of type Student class.
S= new Student();
```

If we create an object of the class Student say S, then using this object the variables RollNo and name can be accessed. These variables that belong to object S are then called as instance variables. Thus the Instance variables are any variables, without "static" field modifier, that are defined within the class body.

Q.19 Define constructor.

AU : CSE : Dec.-12

Ans. : Constructor is a specialized method used for initializing the objects. The name of this method and the name of the class must be the same. The constructor is invoked whenever an object of its associated class is created.

Q.20 What is Static in Java ?

AU : IT : May-13

Ans. : In java, static is a member of a class that is not associated with an instance of a class. If there is a need for a variable to be common to all the objects of a single java class, then the static keyword should be used in the variable declaration.

Q.21 What is the difference between a constructor and a method ?

AU : IT : May-13

Ans. :

Constructor	Method
Name of the constructor must be the same as the name of the class.	There is no such restriction on the name of the method.
Constructor does not have any return type.	Method has return type. If the method does not return anything then it must have void data type.
Constructors are chained, in the sense that they are called in some specific order.	Methods are not chained. That means - invoking of methods depend upon the logic of the program.

Q.22 What are key characteristics of objects ?

AU : CSE : May-13

Ans. :

1. Object is an instance of a class
2. Objects are runtime entities.
3. Using object of a class the member variable and member functions of a class can be accesses.

Q.23 What is meant by parameter passing constructors? Give example.

AU : CSE : Dec.-13

Ans. : The constructor methods to which the parameter are passed are called parameter passing constructors

Example -

```
Rectangle(int h,int w)
{
    height=h;
    weight=w;
}
```

Q.24 What is package ?

Ans. : Package represent a collection of classes,methods and interfaces. The name of the package must be written as the first statement in the Java source program. The syntax of specifying the package in the Java program is

`package name_of_package`

Due to package the systematic arrangement of classes is possible. All standard classes in Java are stored in named packages.

Q.25 Mention the necessity for import statement.

AU : IT : May-12

Ans. : The import statement is used to refer the classes and methods that are present in particular package. This statement is written at the beginning of any Java program. For example -

```
import java.io.*
```

This statement allows to use the useful functionalities for performing input and output operations.

Q.26 Enumerate two situations in which static methods are used.

AU : May-14

Ans. :

- (1) The situation in which object of belonging class is not created and we want to use the method of that class, then the method must be static.
- (2) For calling another static method, one such static method is used.
- (3) For accessing static data the static method must be used.

Q.27 How is constant defined in Java ?

AU : May-15

Ans. : Refer section 1.7.2.

Q.28 List any four JavaDoc comments.

AU : CSE : Dec.-11

Ans. :

Tags	Description
@author	This can be used in the class level comment. It describes the name of the author who is writing the document/
@param	This tag describes the name of the method used in the class
@return	This tag describes the return tag of the method.
@throws	This tag describes the exertion that can be thrown by the method.
@exception	This tag describes the exception

Q.29 What is the need for javadoc multiline comments ?

Ans. : Javadoc multiline comments can be used to document all java source code. Comments follow a standard format consisting of a description followed by block tags. The first sentence of the description should be clear and concise as it is used in the summary of the API item.

Q.30 List any four packages in java and highlight their features.**Ans. :**

Package	Feature
java.io	This package provide useful functionalities for performing input and output operations through data streams and through file system.
java.net	This package is for providing useful classes and interfaces for networking applications which are used in sockets and URL.
java.lang	This package provides core classes and interfaces used in design of Java language.
java.util	Some commonly used utilities such as random number generation, event model, date, time and some of the system properties are represented using java util package.

Q.31 What is API package ?**Ans. :** • The API packages are application programming interface packages used by java.

- These packages include important classes and interfaces which are used by the programmer while developing the java applications.
- For example java.io, java.util, java.net and so on are the commonly used API.

Q.32 Explain any three uses of package.**Ans. :**

1. The classes defined in the packages of other program can be easily reused.
2. Two classes from two different packages can have the same name. By using the package name the particular class can be referred.
3. Packages provide the complete separation between the two phases- design and coding.

Q.33 Explain the term CLASSPATH.**Ans. :** The packages are nothing but the directories. For locating the specified package the java run time system makes use of current working directory as its starting point. This directory path is called CLASSPATH.**Q.34 What are the ways of importing the java packages ?****OR Write the syntax for importing packages in a Java source file and give an example. AU : May-19****Ans. :** The **import** statement can be written at the beginning of the Java program, using the keyword import. For example -

```
import java.io.File
or
import java.io.*
```

Either a class name can be specified explicitly or a * can be used which indicated that the Java compiler should import the entire package.

Q.35 Define access specifier. AU : Dec.-18, 19**Ans. :** Access Specifier is used to set the accessibility of class members. There are three access specifiers in Java - (1) Public (2) Private (3) Protected.

Q.36 What is Javadoc ?**AU : Dec.-19**

Ans. : The Javadoc is a standard way to comment the java code. It has a special format of commenting the java code.

Q.37 Can Java source file be saved using a name other than the class name. Justify.**AU : May-19**

Ans. : Yes, we can save the java source file using a name other than the class name. But we should compile the program with file name and should run the program with the class name in which the main method exist. And there must not be any public class defined inside this java source file. For example, consider following Java code which is saved using the file name **test.java**

```
class A
{
    public static void main(String args[])
    {
        System.out.println("Class A");
    }
}
class B
{
    public static void main(String args[])
    {
        System.out.println("Class B");
    }
}
```

Output

Step 1 : Compile the above program using following command

```
javac test.java
```

Step 2 : Now execute the above program using

```
java A
The output will be
Class A
```

Step 3 : If you execute the above code as

```
java B
The output will be
ClassB
```

Q.38 What are inline function ? Give examples.**AU : May-19**

Ans. : Inline function is a function that is expanded in line when it is called. When the inline function is called whole code of the inline function gets inserted or substituted at the point of inline function call.

The inline function is one of the important feature of C++.

In java there is no inline function



Sol. :

```

class StringDemo
{
    public static void main(String[] args)
    {
        String temp;
        int n = args.length;
        int i,j,c;
        for (i=0; i<n-1; i++)
        {
            for (j=i+1; j<n; j++)
            {
                c = args[i].compareTo(args[j]);
                if (c >0)
                {
                    temp = args[i];
                    args[i] = args[j];
                    args[j] = temp;
                }
            }
        }
        for (i=0; i<n ;i++)
        {
            System.out.println(args[i]);
        }
    }
}

```

Output

```

D:\>javac StringDemo.java
D:\>java StringDemo Swapna Radha Archana Shilpa
Archana
Radha
Shilpa
Swapna

```

Two Marks Questions with Answers**Q.1 In Java what is the use of interface ?****AU : CSE : Dec.-10****OR What is interface mention its use.****AU : IT : Dec.-11,19**

Ans. : In java, the interface is used to specify the behaviour of a group of classes. Using interfaces the concept of multiple inheritance can be achieved.

Q.2 Define Interface and write the syntax of the Interface.**AU : CSE : Dec.-12**

Ans. : The interface can be defined using following syntax

```

access_modifier interface name_of_interface
{
    return_type method_name1(parameter1,parameter2,...parameterN);
    ...
    return_type method_name1(parameter1,parameter2,...parameterN);
}

```

```

type static final variable_name=value;
...
}

```

The interface is a collection of various methods that can be used by the class that is attached to the interface. The interface name must be preceded by the keyword interface.

Q.3 What modifiers may be used with an interface declaration ?

Ans. : An interface may be declared as public or abstract.

Q.4 Differentiate copying and cloning.

Ans. : In object copying the copy must be done by copying each field of object to another. This is poor method of duplicating the object. But the object cloning is a method in which the object is duplicated without explicitly copying each field. It makes use of the cloneable interface to duplicate the object.

Q.5 What is object cloning ?

AU : CSE : Dec.-13,19

Ans. : The object cloning is a method of duplicating the object. It is the simplest technique in which the cloneable interface is used. This interface requires a method clone(). The clone method returns duplicated object.

Q.6 What is an anonymous inner class ?

AU : IT : Dec.- 10

Ans. : Anonymous inner class is a local class without any name. This class is created in following situations -

1. When the class has very short body.
2. Only one instance of the class is needed.
3. Class is used immediately after defining it.

Q.7 How does the declaration of a local inner class differ from the declaration of an anonymous inner class ?

AU : IT : May-11

Ans. : The local inner class is an inner class within the body of a method just like the local variable. For example -

```

Access_modifier class OuterClass
{
    //code
    Access_modifier return_type methodname(arguments)
    {
        class InnerClass
        {
            //code

```

```
    }  
    //code  
}
```

Anonymous inner class is a local class without any name. Because it has no name it can only be used once at place of declaration. For example

```
class MyInnerClass implements Runnable  
{  
    public void run()  
    {  
        System.out.println("Hello");  
    }  
}  
class DemoClass  
{  
    public static void main(String[] arg)  
    {  
        MyInnerClass my=new MyInnerClass();  
        Thread th=new Thread(my);  
        my.start();  
    }  
}
```

Q.8 State the difference between inner class and anonymous class.

AU : IT : Dec.-11

Ans. : The inner class is a kind of class within the body of a method. An inner class can have any accessibility including private. Anonymous inner class is a local class without any name. Because it has no name it can only be used once at place of declaration.

Q.9 Why the variables in interface static and final ?

Ans. : The members of interface are static and final because -

- 1) The reason for being static - The members of interface belong to interface only and not object.
- 2) The reason for being final - Any implementation can change value of fields if they are not defined as final. Then these members would become part of the implementation. An interface is pure specification without any implementation.

Q.10 What is the purpose of nested interface ?

Ans. : The nested interfaces are used to group related interfaces so that they can be easy to maintain. The nested interface must be referred by the outer interface or class. It can't be accessed directly.

Q.11 What are the properties of nested interface ?

Ans. :

- 1) Nested interfaces are static by default. You don't have to mark them static explicitly.
- 2) Nested interfaces declared inside class can take any access modifier.
- 3) Nested interface declared inside interface is public implicitly.

Q.12 If a class B is derived from class A and extends the interface myinterface, then how will you write this statement for class B definition ?

Ans. :

class B extends class A implements myinterface

```
{  
    ....//body of class B  
}
```



int getLineNumber()	Represents the name of the line containing execution point represented by StackTraceElement .
int hashCode()	It returns the hash code element for the stack trace element.
boolean isNativeMethod()	It represents whether the method containing is native or not, using StackTraceElement.
String toString()	It returns a string represented by the StackTraceElement .

Review Question

1. What is meant by exceptions ? Why it is needed ? Describe the exception hierarchy. Write note on Stack Trace Elements. Give example.

AU : CSE : Dec.-13, Marks 16

Two Marks Questions with Answers

Q.1 What is an exception ? Give example.

AU : IT, CSE : May-12

Ans : Exception is a mechanism which is used for handling unusual situation that may occur in the program. For example -

ArithmeticException : This exception is used to handle arithmetic exceptions such as divide by zero.

IOException : This exception occurs when an illegal input/output operation is performed.

Q.2 What will happen if an exception is not caught ?

AU : CSE : Nov.-10

Ans. : An uncaught exception results in invoking of the uncaughtException() method. As a result eventually the program will terminate in which it is thrown.

Q.3 Give any methods available in Stack Trace Element.

AU : CSE : Dec.-10

Ans. : int getLineNumber() - This method represents the name of the line containing execution point represented by StackTraceElement.

Q.4 What are stack trace elements.

AU : CSE : Dec.-12

Ans. : Stack trace element is created to represent the specific execution point. The syntax of stack trace element is `StackTraceElement(String ClassName, String MethodName, String filename,int lineNumber);`

The class name represents the name of the class for which the execution point is specified.

Q.5 What is the benefit of exception handling ?

Ans. : When calling method encounters some error then the exception can be thrown. This avoids crashing of the entire application abruptly.

Q.6 What is the difference between error and exception in java?

Ans. : Errors are mainly caused by the environment in which an application is running. For example, `OutOfMemoryError` happens there is shortage of memory. Where as exceptions are mainly caused by the application itself. For example, `NullPointerException` occurs when an application tries to access null object.

Q.7 What is compile time and run time error ?

Ans. :

1. The errors that are detected by the Java compiler during the compile time are called compiler time errors.
2. The runtime errors are basically the logically errors that get caused due to wrong logic.

Q.8 What is the use of try, catch keywords ?

Ans. : try - A block of source code that is to be monitored for the exception.

catch - The catch block handles the specific type of exception along with the try block.

Q.9 What is the difference between throw and throws ?

Ans. :

1. The throw keyword is used to explicitly throw an exception.
2. The throws keyword is used to declare an exception.

Q.10 What is `ArrayIndexOutOfBoundsException` ?

Ans. : When we use an array index which is beyond the range of index then `ArrayIndexOutOfBoundsException` occurs.

Q.11 What is the need of multiple catch ?

Ans. :

- There may be the situations in which different exceptions may get raised by a single try block statements and depending upon the type of exception thrown it must be caught.
- To handle such situation multiple catch blocks may exist for the single try block statements.

Q.12 There are three statements in a try block – statement1, statement2 and statement3. After that there is a catch block to catch the exceptions occurred in the try block. Assume that exception has occurred in statement2. Does statement3 get executed or not?

Ans. : No. Once a try block throws an exception, remaining statements will not be executed. Control comes directly to catch block.

Q.13 Explain the types of exceptions.

Ans. : There are two types of exceptions :

1. **Checked Exception :** These types of exceptions need to be handled explicitly by the code itself either by using the try-catch block or by using throws. These exceptions are extended from the java.lang.Exception class.

For example : IOException which should be handled using the try-catch block.

2. **Unchecked Exception :** These type of exceptions need not be handled explicitly. The Java Virtual Machine handles these type of exceptions. These exceptions are extended from java.lang.RuntimeException class.

For example : ArrayIndexOutOfBoundsException, NullPointerException, RuntimeException.

Q.14 Does finally block get executed if either try or catch blocks are returning the control ?

Ans. : Yes, finally block will be always executed no matter whether try or catch blocks are returning the control or not.

Q.15 Can we have empty catch block ?

Ans. : Yes we can have empty catch block, but it is an indication of poor programming practice. We should never have empty catch block because if the exception is caught by that block, we will have no information about the exception and it will create problem while debugging the code.

Q.16 Which class is the super class for all types of errors and exceptions in java ?

Ans. : The java.lang.Throwable is the super class for all types of errors and exceptions in java.

Q.17 What is runtime exceptions ?

AU : Dec.-18

Ans. : RuntimeException is the superclass of those exceptions that can be thrown during the normal operation of the Java Virtual Machine. RuntimeException is the parent class in all exceptions of the Java.

Q.18 What is assert keyword ?

AU : Dec.-18, 19

Ans. : The assert keyword is used for testing. It can be used for assumption about the program. While executing the assertion it is assumed to be true.

Q.19 What is exception handling ?**AU : Dec.-19**

Ans. : Exception handling is a mechanism in which the statements that are likely to cause an exception are enclosed within try block. As soon as exception occurs it is handled using catch block.

Thus exception handling is a mechanism that prevents the program from crashing when some unusual situation occurs.

Q.20 What happens when the statement `int value = 25/0` is executed ?**AU : May-19**

Ans. : The exception Arithmetic Exception : Divide by zero will occur.



```
    }
    catch (Exception ex)
    {
        ex.printStackTrace();
    }
//Read object from OUT.txt
try
{
    FileInputStream fileIn = new FileInputStream("D:\\OUT.txt");
    ObjectInputStream objectIn = new ObjectInputStream(fileIn);
    Object obj;
    System.out.println(" FirstName LastName Age");
    for(int i=0;i<100;i++)
    {
        obj = objectIn.readObject();
        System.out.println((Student)obj);
    }
    objectIn.close();
}
catch (Exception ex)
{
    ex.printStackTrace();
}
}
```

Review Question

1. Explain in detail about the following with sample program :
i) Reading from a file ii) Writing in a file.

AU : Dec.-19, Marks 13**Two Marks Questions with Answers****Q.1 What is stream ?**

Ans. : Stream is basically a channel on which the data flow from sender to receiver.

Q.2 What is input stream and output stream ?

Ans. : An input object that reads the stream of data from a file is called input stream and the output object that writes the stream of data to a file is called output stream.

Q.3 What is byte stream ? Enlist its super classes.

Ans. : The byte stream is used for inputting or outputting the bytes. The InputStream and OutputStream are the superclass of byte stream.

Q.4 What is character stream ? Enlist its super classes.

Ans. : The character stream is used for inputting or outputting the characters. The Reader and Writer are the superclass of character stream.

Q.5 List the byte stream classes.

Ans. : The Byte Stream classes are –

1. FileInputStream
2. PipedInputStream
3. FilterInputStream
4. ByteArrayInputStream
5. FileOutputStream
6. PipedOutputStream
7. FilterOutputStream
8. ByteArrayOutputStream

Q.6 What is the purpose of BufferedInputStream and BufferedOutputStream classes ?

Ans. :

- These are efficient classes used for speedy read and write operations.
- We can specify Buffer size while using these classes.

Q.7 Give an example on stream.

Ans. :

- There are two types of streams - Byte stream and Character stream.
- Stream classes for byte stream : FileInputStream, PipedInputStream, BufferedInputStream, FileOutputStream, PipedOutputStream and so on.
- Stream classes for character stream : FileReader, PipeReader, BufferedReader, FileWriter, PipeWriter, BufferedWriter and so on.

Q.8 What is absolute file name ?

Ans. : The filename that can be specified with the complete path name and drive letter is called absolute file name.

Q.9 What is relative file name ?

Ans. : The file name which is specified as a path relative to the current directory is called relative file name. For example `new File("myprogram.html");`

Q.10 What is the use of seek method ?

Ans. : The `seek()` allows you to specify the index from where the read and write operations will start. The syntax is

```
void seek(long position);
```

Q.11 Write a Java code to check if the command line argument is file or not.

Ans. :

```
class Test
{
    public static void main(String [] args)
    {
        File obj=new File(args[0]);
```

```
        If(obj.isFile())
        {
            System.out.println("This is a file name"+obj.getPath());
        }
    }
}
```

Q.12 Enlist the two common constructors of FileInputStream.

Ans. : The common constructors of FileInputStream are

```
FileInputStream(String filename);
FileInputStream(File fileobject);
```

Q.13 What is the use of Input Stream Reader and Output Stream Writer ?

Ans. : The InputStreamReader converts byte into character and OutputStreamWriter converts the characters written into byte.

Q.14 Give an example for reading data from files using FileInputStream

AU : May 19

Ans. :

```
import java.io.FileInputStream;
public class test
{
    public static void main(String args[])
    {
        try
        {
            FileInputStream fin=new FileInputStream("D:\\input.txt");
            int i=0;
            while((i=fin.read())!=-1)
            {
                System.out.print((char)i);
            }
            fin.close();
        }catch(Exception e){System.out.println(e);}
    }
}
```



```
    }  
    public void paint(Graphics g)  
    {  
        g.drawImage(Img, 50,50, this);  
    }  
}
```

Review Question

1. Explain how to display an image in applet ?

Two Marks Questions with Answers

Q.1 What are the steps needed to show a Frame ?

AU : CSE : Dec.-10

OR How are frames created in Java ?

AU : CSE : Dec.-11

Ans. : Frame can be created by following two ways -

- 1) The frame can be created by extending the frame class. For example
class FrameDemo extends Frame

```
{  
    public void static main(String[] arg)  
    {  
        FrameDemo fr=new FrameDemo();  
        fr.setSize(300,300);  
        fr.setVisible(true);  
    }  
}
```

- 2) The frame can be created by using the instance of Frame class -
class FrameDemo1

```
{  
    public static void main(String[] arg)  
    {  
        Frame fr=new Frame();  
        fr.setSize(300,300);  
        fr.setVisible(true);  
    }  
}
```

Q.2 What is meant by frame window ?

AU : IT : May-12

Ans. : The frame window is a standard graphical window. It can be displayed by the class named **Frame**. This frame window has standard minimize, maximize and close buttons.

Q.3 What is AWT ?**AU : IT : May-12, Dec.-12**

Ans. : The AWT stands for Abstract Window Toolkit. It is a class library that contains various classes and interfaces that are required for graphical programming. The AWT provides the support for drawing graphical shapes, windows, buttons, text boxes, menus and so on.

Q.4 How do you manage the color and font of graphics in applet ?**AU : CSE : May-12**

Ans. : The color can be specified as
`color(int R,int G,int B);`

The background color can be set by
`setBackground(Color colname)`

The color of the text can be set by
`setForeground(Color colname)`

The font can be set by **setFont()** method. This function requires a reference parameter which can be created by the method **Font()**. The **Font()** method has first parameter as name of the font, second parameter denotes the style of the font which can be BOLD, ITALIC or PLAIN. The third parameter denotes the size of the font.

Q.5 Code a Graphics method in Java to draw the String "Hello World" from the coordinates(100,200)**AU : IT : Dec.-13****Ans. :**

```
Import java.applet;
/*
<applet code="MsgDemo" width=300 height=300>
</applet>
*/
public class MsgDemo extends Applet
{
    public void paint(Graphics g)
    {
        g.drawString("Hello World",100,200);
    }
}
```

Q.6 How does radio button in Java differ from check box ?**AU : IT : Dec.-13**

Ans. : The radio buttons are mutually exclusive and user must select exactly one choice from the given list. Whereas the check box is for selecting any number options from the given choices.

The applet has to register itself as a listener to various events (*here the same applet acts as a event source as well as event listener*). Hence inside **init()** method applet registers itself as a listener by following statements

```
addMouseListener(this);
addMouseMotionListener(this);
```

And then simple methods of mouse events are defined. The **getX()** and **getY()** methods return the current x and y positional values. To each of these methods an object of **MouseEvent** is passed which is shown by a variable 'm'.

Review Question

1. Discuss mouse listener and mouse motion listener. Give an example program.

AU : Dec.-19, Marks 13

9.6 AWT Event Hierarchy

AU : IT : Dec.-12, 13, Marks 16

In Java event handling is done using object oriented methodology. **EventObject** class is defined in java.util package and all the events are descendant of this class. The **AWTEvent** is a class derived from EventObject class.

The AWT Event Hierarchy is shown by following Fig. 9.6.1

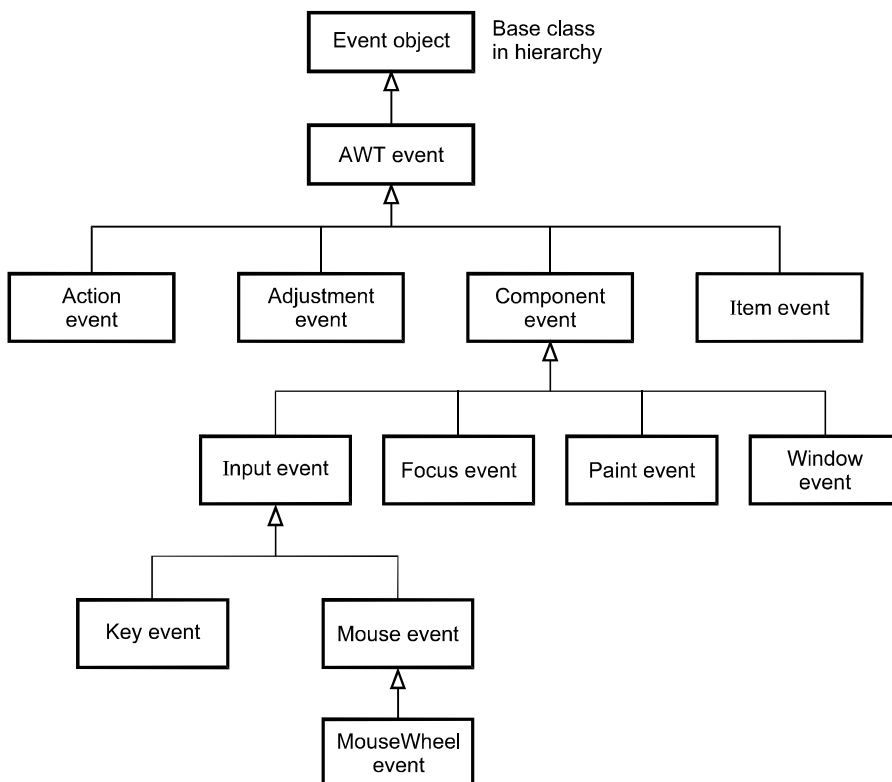


Fig. 9.6.1 AWT Event Hierarchy

Review Question

1. Explain in detail about AWT event hierarchy.

AU : IT : Dec.-12, Marks 16**Two Marks Questions with Answers****Q.1 What are the advantages of event delegation model ?****AU : IT : Dec.-10**

Ans. : Following are the advantages of event delegation model -

1. In this model, the events are handled using objects. This allows a clear separation between usage of the components and the design.
2. It accelerates the performance of the application in which multiple events are used.

Q.2 When do the following events occur ? i) AdjustmentEvent ii) ItemEvent**AU : IT : May-11**

Ans. :

(i) AdjustmentEvent : This event is generated when the scrollbars are used.

(ii) ItemEvent : This event is generated when an item from a list is selected. For example if choice is made to if a checkbox is selected.

Q.3 Which Java.util classes and interfaces support event handling ?**AU : IT : Dec.-11**

Ans. : EventObject is a class which is useful in event handling and this class belongs to java.util package.

Similarly EventListener interface of java.util package support event handling.

Q.4 Which class is the top of the AWT event hierarchy ?**AU : IT : Dec.-11**

Ans. : The **Component** class is at the top of AWT event hierarchy.

Q.5 What method is used to distinguish between single, double and triple mouse clicks ?

Ans. : public int **getClickCount()** : This methods returns the number of mouse clicks. Hence we can check -

```
if (mouseEvent.getClickCount() == 1)
{
    System.out.println("Single Click");
}
else if (mouseEvent.getClickCount() == 2)
{
    System.out.println("Double Click");
}
```


void interrupt()	This method is used to interrupt all the threads in the group.
void list	This method is used to display the information about the thread group.

Example Program

public class ThreadGroupProg implements Runnable

```
{
    public void run()
    {
        System.out.println(Thread.currentThread().getName());
    }
    public static void main(String[] args)
    {
        ThreadGroupProg runnable = new ThreadGroupProg();
        ThreadGroup tg1 = new ThreadGroup("Colorful ThreadGroup");
        System.out.println("Thread Group Name: "+tg1.getName());
        Thread t1 = new Thread(tg1, runnable,"Red");
        t1.start();
        Thread t2 = new Thread(tg1, runnable,"Green");
        t2.start();
        Thread t3 = new Thread(tg1, runnable,"Blue");
        t3.start();
        Thread t4 = new Thread(tg1, runnable,"Yellow");
        t4.start();
        System.out.println("The active threads are: "+tg1.activeCount());
    }
}
```

Output

```
Thread Group Name: Colorful ThreadGroup
Red
Blue
Green
Yellow
The active threads are: 4
```

Two Marks Questions with Answers

Q.1 What do you mean by threads in Java ?

AU : CSE : Dec.-11; IT : May-12, Dec.-13

Ans. : Thread is tiny program running continuously. It is a light weight process in Java.

Q.2 Give the difference between process and thread.

AU : IT : Dec.-11

Ans. : Refer section 6.1.

Q.3 What are different stages in thread ?

AU : CSE : Dec.-10, 19

Ans. : Various stages in thread are -

1. New state
2. Runnable state
3. Waiting state
4. Time waiting state
5. Blocked state
6. Terminated state

Q.4 What do you mean by synchronization ?

AU : CSE : Dec.-10, May-12

Ans. : When two or more threads need to access shared memory, then there is some way to ensure that the access to the resource will be by only one thread at a time. The process of ensuring one access at a time by one thread is called synchronization.

Q.5 What are the three ways by which the thread can enter in waiting stage ?

AU : IT : Dec.-10

Ans. :

- i) **Waiting state :** Sometimes one thread has to undergo in waiting state because another thread starts executing. This can be achieved by using wait() state.
- ii) **Timed waiting state :** There is a provision to keep a particular threading waiting for some time interval. This allows to execute high prioritized threads first. After the timing gets over, the thread in waiting state enters in runnable state. This can be achieved by using the sleep() command.
- iii) **Blocked state :** When a particular thread issues an input/output request then operating system sends it in blocked state until the I/O operation gets completed. After the I/O completion the thread is sent back to the runnable state. This can be achieved by using suspend() method.

Q.6 What is multithreading ?

AU : CSE : Dec.-11, May-12; IT : Dec.-12, 18

Ans. : Multithreading is an environment in which multiple threads are created and they can execute simultaneously. The multiple threads can be created either by extending the thread class or by implementing the runnable interface.

Q.7 Mention two mechanisms for protecting a code block from concurrent access.

Ans. : There are two mechanisms for protecting a code block from concurrent access -

1. Use of reentrant code
2. Use synchronized block

Q.8 What is meant by notify methods in multithreading ?

Ans. : The notify() method is used for inter thread communication. If a particular thread is in the sleep mode then that thread can be resumed by using the notify call.

Q.9 What are the two ways of creating a thread ?**AU : IT : Dec.-12**

Ans. : Thread can be created using

1. Thread class
2. runnable interface.

Q.10 Why do we need run() and start() method both ? Can we achieve it with only run() method ?**AU : IT : May-13**

Ans. : We use start() method to start a thread instead of calling run() method because the run() method is not a regular class method. It should only be called by the JVM. If we call the run() method directly then it will simply invoke the caller's thread instead of its own thread. Hence the start() method is called which schedules the thread with the JVM.

Q.11 What is the need for thread ?**AU : CSE : May-13**

Ans. : In Java threads are used to handle multiple tasks together. This kind of programming is called concurrent programming.

Q.12 Name any four thread constructor.**AU : CSE : May-13**

Ans. :

1. Thread()
2. Thread(String name)
3. Thread(Runnable target)
4. Thread(Runnable target, String name)

Q.13 When thread is initiated and created, what is its initial stage ?**AU : May-15**

Ans. : A thread is in "Ready" state after it has been created and started. This state signifies that the thread is ready for execution. From here, it can be in the running state.

Q.14 "Thread is a lightweight process" – Comment**AU : Dec.-19**

Ans. : Threads do not require separate address for its execution. It runs in the address space of the process to which it belongs to. Hence thread is a lightweight process.

Q.15 Sketch the life cycle of thread**AU : May-19**

Ans. : Refer Fig. 6.3.1



```

    Test<Number> obj1 = new Test<Number>(123);
    System.out.println("The integer is: "+obj1.getT());
    //While Creating object by passing String as a type parameter, it gives compile time
    //error
    Test<String> obj2 = new Test<String>("I am string"); //Compile time error
    System.out.println("The string is: "+obj2.getT());
}
}

```

Review Question

1. Explain the use of bounded types in Generics with illustrative program.

7.5 Restrictions and Limitations

1. In Java, generic types are compile time entities. The runtime execution is possible only if it is used along with raw type.
2. Primitive type parameters is not allowed for generic programming.
For example: Stack<int> is not allowed.
3. For the instances of generic class throw and catch instances are not allowed.

For example :

```

    public class Test<T> extends Exception
    {
        //code    // Error:can't extend the Exception class
    }

```

4. Instantiation of generic parameter T is not allowed.

For example :

```

    new T() //Error
    new T[10]

```

5. Arrays of parameterized types are not allowed.

For example :

```

    new Stack<String>[10];//Error

```

6. Static fields and static methods with type parameters are not allowed.

Two Marks Questions with Answers

Q.1 Why generic programming is required ?

OR

What is the need for generic programming ?

AU : CSE : May-13, Dec.-18

Ans. : Following some reasons for the need of generic programming -

1. It saves the programmers burden of creating separate methods for handling data belonging to different data types.

2. It allows the code reusability.
3. Compact code can be created.

Q.2 List out motivation needed in generic programming.

Ans. :

1. Suppose we want to create a stack and if we create a stack of integers then it will store only the integer elements, if we try to push any string or any double type element then a compile time error will occur. If we want to push any string then we need to create a separate stack, separate class and separate methods for handling the String elements. Same is true for the elements of any other data type. This results in complex code building. To avoid such complexity, a concept of generic programming is introduced.
2. It saves the programmers burden of creating separate methods for handling data belonging to different data types.
3. It allows the code reusability.
4. Compact code can be created.

Q.3 With an example define a generic class. Or Describe generic classes.

AU : CSE : Dec.-12, 13, May-19

Ans. : A generic class contains one or more variables of generic data type. Following is a simple example which shows how to define the generic class .

```
public class Test<T>
{
    public Test(){val=null;}
    public Test(T val)
    {
        this.val=val;
    }
    public getVal()
    {    return val;
    }
    public setVal()
    {
        val=newValue;
    }
    private T val; //variable defined as of generic type
}
```

Q.4 Can generic be used with inheritance in several ways ? What are they ?

Ans. : Following are the ways by which generic can be used with inheritance -

- Consider a class and a subclass such as **Employee** and **Trainee** . There are two pair classes **Pair<Employee>** and **Pair<Trainee>** which do not possess an inheritance relation. That is,

Program Explanation

For the confirm dialog box, the option type could be `OK_CANCEL`, `YES_NO_OPTION`, `OK_OPTION`, `NO_OPTION` which are associated with some constant values.

In above program, when the user clicks the button for OK and Cancel message then appropriate message will be displayed. If user clicks the OK button then the message "You have clicked OK button will be displayed" otherwise the message "You have clicked Cancel button" will be displayed.

Two Marks Questions with Answers

Q.1 List any four features of swing not present in java .

Ans. :

1. Swing has a rich set of GUI components such as buttons and check boxes, trees, tabbed panes and so on.
2. It has got pluggable look and feel support.
3. Support for data transfer is built into Swing and works between Swing components within an application, between Java applications, and between Java and native applications.
4. If you want your program to run within a browser window, you can create it as an applet and run it.

Q.2 What is the function of a) SetLayout b) FlowLayout ?

AU : CSE : Dec.-10

Ans. : The `setLayout` is used for setting the layout of the device. The parameter to this function is an object of the layout manager.

The `FlowLayout` manager is the simplest layout manager. Using this layout manager the components are arranged from top left corner lying down from left to right and from top to bottom.

Q.3 Write down the statements used to create a JFrame and b JButton and then add the JButton to the JFrame.

AU : IT : May-11

Ans. : The code segment is -

```
JFrame f=new JFrame("Frame Demo");
f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
f.setVisible(true);
f.setSize(300,300);
Container content=f.getContentPane();
```

- Inside the **finalize() method** you will specify those actions that must be performed before an object is destroyed.
- The garbage collector runs periodically checking for objects that are no longer referenced by any running state or indirectly through other referenced objects.
- To add finalizer to a class simply define the **finalize** method. The syntax to finalize() the code is -

```
void finalize()
{
    finalization code
}
```

Note that finalize() method is called just before the garbage collection. It is not called when an object goes out-of-scope.

- The finalize method of an object is called when garbage collector is about to clean up the object.
- The purpose of finalization is to perform some action before the objects get cleaned up.
- The clean up code can be kept in the finalize method block.

Review Question

1. State the purpose of finalize () method in java. With an example explain how finalize () method can be used in java program.

AU : Dec.-18, Marks 7

Two Marks Questions with Answers

Q.1 Define the term inheritance.

Ans. : Inheritance is a mechanism in which the derived class borrows the properties of the base class.

Q.2 Explain the use of extend keyword with suitable example.

Ans. :

- The extend keyword is used in inheritance.
- When the child class is derived from its parent class then the keyword extends is used.

Q.3 What is the difference between superclass and subclass ?

Ans. :

- The superclass is a parent class or base class from which another class can be derived.
- The subclass is always a derived class which inherits the properties of the base class or superclass.
- The superclass is normally a generalized class whereas the subclass is normally for specific purpose.

Q.4 Enlist various forms of inheritance.

Ans. : Various forms of inheritance are –

1. Single level inheritance
2. Multi-level inheritance
3. Multiple inheritance
4. Hierarchical inheritance
5. Hybrid inheritance

Q.5 What is the use of super keyword ?

Ans. : The super class is used to access immediate parent class from the subclass. It is used to

1. access parent's variable,
2. access parent's method
3. access parent's constructor invocation

Q.6 Differentiate between inheritance and polymorphism.

Ans. :

Sr. No.	Inheritance	Polymorphism
1.	Inheritance is a property in which some of the properties and methods of base class can be derived by the derived class.	Polymorphism is ability for an object to used different forms. The name of the function remains the same but it can perform different tasks.
2.	Various types of inheritance can be single inheritance, multiple inheritance, multilevel inheritance and hybrid inheritance.	Various types of polymorphism are compile time polymorphism and run time polymorphism. In compile time polymorphism there are two types of overloading possible. - Functional overloading and operator overloading. In run time polymorphism there is a use of virtual function.

Q.7 Distinguish between static and dynamic binding.

AU : IT : May-11

Ans. : Static binding is a type of binding in which the function call is resolved at compile time.

The dynamic binding is a type of binding in which the function call is resolved at run time.

The static binding is called the early binding and the dynamic binding is called late binding.

Q.8 What is the purpose of 'final' keyword

AU : IT : Dec.-11, CSE : Dec.-12

Ans. : The keyword 'final' is used to avoid further modification of a variable, method or a class. For instance if a variable is declared as final then it can not be modified further, similarly, is a method is declared as final then the method overriding is avoided.

Q.9 Define inheritance hierarchy.

AU : CSE : Dec.-11, 18

Ans. : The inheritance hierarchy represents the collection of classes the inherit from their base classes and thereby make use of the code present in the base class. The data reusability can be achieved by inheritance hierarchy. For example

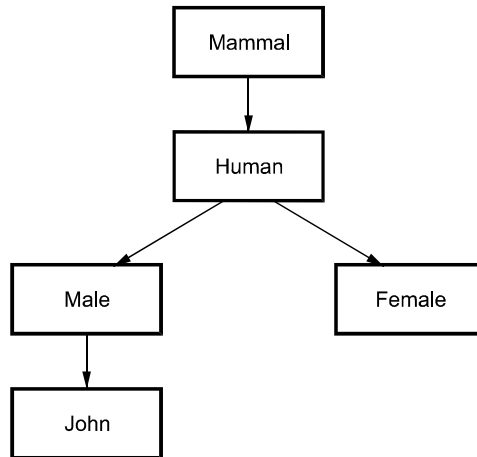


Fig. 2.1

Q.10 How to prevent inheritance ?

Ans. : If we declare particular class as **final** then no class can be derived from it. For example

```

final class Test
{
...
...
...
}
class Test1 extends Test
{
...
...
...
}
  
```

The above code will produce an error "cannot inherit from final Test".

Thus the use of keyword final for the class prevents the inheritance.

**Q.11 Write a class declarations for the following relationship, assuming that all classes are public:
a Bulldog is a kind of dog, and a Dog is a kind of Animal.**

Ans. :

```

class Animal
{
...
}
  
```

```

class dog extends Animal
{
...
}
class Bulldog extends dog
{
...
}

```

Q.12 What is meant by dynamic method dispatch ?

Ans. : The dynamic method dispatch is run time polymorphism in which a call to overridden function is resolved at runtime instead of at compile time. Hence is the name. For example

```

class Test1
{
    public void method1()
    { }
}
class Test2 extends Test1
{
    public void method1()
    { }
}
void main()
{
    Test1 obj=new Test2();
    obj.method1();
}

```

The **method1** of class **Test2** will be invoked here, thus the code supports the dynamic method dispatch.

Q.13 Can an abstract class be final ? Why ?

AU : IT : Dec.-10

Ans. : The abstract class cannot be final because once we declared the abstract base class with the keyword final then it can not be inherited.

Q.14 Can an abstract class in Java be instantiated ? Give the reason.

AU : IT : Dec.-13

Ans. : The abstract class can not be instantiated(i.e. we can not create the object of this class using new operator) because the abstract class is very much general and less specific. It does nothing and simply lists out only common features of other classes.

Q.15 Why is multiple inheritance using classes a disadvantage in Java ?

AU : IT : May-13

Ans. : In multiple inheritance, the child class is derived from two or more parent classes. It can lead to ambiguity when two base classes implement a method with the same name.

Q.16 State the condition for method overriding in Java

AU : May 19

Ans. : Method overriding occurs only when the name of the two methods(method in super class and method in subclass) are same and their type signature is same.



Notes