



3.

(a) What is static variable and static method? Explain why main method in java is always static.

(2 Marks)

Ans:

Static → It can be used with a variable, a method or block of code. A static methods or variable cannot instance specific. There is no need to create an object of the class to access the static features of the class.

Class suman

{

Static int i=10;

}

Class aman

{

public static void main (string args[])

{

System.out.println("my number is="+suman.i)

}

}

(b) What is inheritance? Explain the advantage of inheritance with an example program. What are different types of inheritance supported by java?

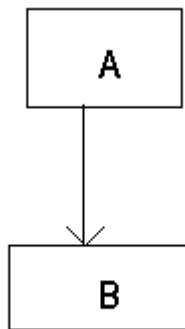
(5 Marks)

Ans:

Inheritance: it is a process to create a new class from existing class or classes. Class 'B' is a sub class that is derived from class 'A'. Thus class 'A' is a parent class and class 'B' is a child class. It provides reusability of code.

Advantage:

- Code reusability
- Function overriding
- Save compilation and programmer time
- Increase modularity



Example:-

```
class suman
{
int c;
public void sum (int a, int b)
{
c=a+b;
system.out.println(c);
}
}
class pix extends suman
{
public static void main()
{
suman s=new suman();
s.sum(4,5);
}
}
```

(c) Explain the steps involved in creating a distributed application using Remote Method Invocation (RMI). (3 Marks)

Ans:-

Steps to write the RMI program.

- Create the remote interface
- Provide the implementation of the remote interface
- Compile the implementation class and create the stub and skeleton objects using the rmic tool
- Start the registry service by rmiregistry tool
- Create and start the remote application
- Create and start the client application



```
class TCPServer
{
    public static void main(String argv[]) throws Exception
    {
        String clientSentence;
        String capitalizedSentence;
        ServerSocket welcomeSocket = new ServerSocket(6789);

        while(true)
        {
            Socket connectionSocket = welcomeSocket.accept();
            BufferedReader inFromClient =
                new BufferedReader(new
InputStreamReader(connectionSocket.getInputStream()));
            DataOutputStream outToClient = new
DataOutputStream(connectionSocket.getOutputStream());
            clientSentence = inFromClient.readLine();
            System.out.println("Received: " + clientSentence);
            capitalizedSentence = clientSentence.toUpperCase() + '\n';
            outToClient.writeBytes(capitalizedSentence);
        }
    }
}
```

(c) Explain the need of JDBC? Explain steps involved in connecting a databases using JDBC.

(3 Marks)

Ans:

Steps:

- **Register the driver class**

The forName() method of Class class is used to register the driver class. This method is used to dynamically load the driver class.

Class.forName("oracle.jdbc.driver.OracleDriver");

- **Create the connection object**

The getConnection() method of DriverManager class is used to establish connection with the database.

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");