1. Write a program to prompt the user for a hostname and then looks up the IP address for the hostname and displays the results.

```
import java.net.*;
  public class IpTest
  {
     public static void main(String arg[])throws UnknownHostException
     {
       InetAddress addr=InetAddress.getLocalHost();
 String Ipad=addr.getHostAddress();
       String HostName=addr.getHostName();
       System.out.println("IP address is"+Ipad);
 System.out.println("HostName is"+HostName);
     }
  }
```

2. Write a program to read the webpage from a website and display the contents of the webpage.

/*Embedding HTML code into JAVA*/

```
import javax.swing.JEditorPane;
import java.net.URL;
import javax.swing.JFrame;

   public class HtmlContent extends JFrame
   {
      public static void main(String args[])
      {
       new HtmlContent().start();
      }
       void start()
       {
          try
           {
              String html;
              html="<html><head><title>SimplePage</title></head>;
              html="<body bgcolor='#777779'><Font size=50>
              <p>This is AJ class</Font></p>
              html+="</body></html>";
              JEditorPane ed1=new JEditorPane("text/html",html);
              add(ed1);
```

```
            setVisible(true);
            setSize(600,600);
            setDefaultCloseOperation(Exit_On_Close);
          }
        }
       catch(Exception e)
          {
           e.printStackTrace();
           System.out.println("some problem has occured"+e.getMessage());
          }
        }
      }
```

/*Using an URL*/

```
  import javax.swing.JEditorPane;
  import java.net.URL;
  import javax.swing.JFrame;

    public class HtmlContent extends JFrame
    {
      public static void main(String args[])
```

```
 {
  new HtmlContent().start();
 }
  void start()
  {
    try
     {
       URL u=new URL("http://www.gmail.com");
       JEditorPane ed2=new JEditorPane(u);
 add(ed2);
       setVisible(true);
       setSize(600,600);
       setDefaultCloseOperation(Exit_On_Close);
     }
  }
 catch(Exception e)
     {
      e.printStackTrace();
      System.out.println("some problem has occured"+e.getMessage());
     }
  }
}
```

3. Write programs for TCP server and Client interaction as per given below. i. A program to create TCP server to send a message to client. ii. A program to create TCP client to receive the message sent by the server.

/*To write a C program on server side*/

```
#include<stdio.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<string.h>

   int main()
   {
     int welcomeSocket,newSocket;
     char buffer[1024];
     struct sockaddr_in serverAddr;
     struct sockaddr_storage serverStorage;
     socklen_t addr_size;
     welcomeSocket=socket(PF_INET,sock_STREAM,0);
```

```
serverAddr.sin_family=AF_INET;
serverAddr.sin_port=htcons(999);
serverAddr.sin_addr.s_addr=inet_addr("192.168.1.147");

/*set all bits of the padding field to 0*/

memset(serverAddr.sin_zero,'\0',sizeof serverAddr.sin_zero);
bind(welcomeSocket,(struct sockaddr*)&serverAddr,sizeof(serverAddr));

/*...listen on the socket,with 5 max connection requests queued...*/

if(Listen(welcomeSocket,5)==0)
 printf("listening\n");
else
 printf("error\n");

/*..accept call creates a new socket for the incoming conneton*/

addr_size=sizeof serverStorage;
newSocket=accept(welcomeSocket,(struct
sockaddr*)&serverStorage,&addr_size);
strcpy(buffer,"hello Aliet\n");
```

```
    send(newSocket,buffer,13,0);
    return 0;
  }
```

/*To write a C program on client side*/

```
#include<stdio.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<string.h>

  int main()
  {
    int clientSocket;
    char buffer[1024];
    struct sockaddr_in serverAddr;
    socklen_t addr_size;
    clientSocket=socket(PF_INET,sock_STREAM,0);
    serverAddr.sin_family=AF_INET;
    serverAddr.sin_port=htcons(999);
    serverAddr.sin_addr.s_addr=inet_addr("192.168.1.147");
```

```
/*set all bits of the padding field to 0*/

 memset(serverAddr.sin_zero,'\0',sizeof serverAddr.sin_zero);
addr_size=sizeof serverAddr;

  connect(clientSocket,(struct sockaddr *)?&serverAddr,sin_zero);
     recv(dataSocket,buffer,1024,0);
     printf("data received: %s",buffer);
     return 0;

   }
```

5. Write a program by using JDBC to execute a SQL query for a database and display the results.

/*To create a table*/

```
import java.sql.*;
public class Create
{
```

```
public static void main(String args[])throws Exception
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
System.out.println("Driver loaded");
Connection
con=DriverManager.getConnection("jdbc:odbc:info","system","tiger");
System.out.println("Connection Established");
Statement st=con.CreateStatement();
int a=st.executeUpdate("Create table student(id integer,name char(10))");
if(a<=0)
System.out.println("table created");
else
System.out.println("table can not be created");
}
}
```

/*To insert values into a table*/

```
import java.sql.*;
public class Insertion
{
public static void main(String args[])throws Exception
```

```
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
System.out.println("Driver loaded");
Connection
con=DriverManager.getConnection("jdbc:odbc:info","system","tiger");
System.out.println("Connection Established");
Statement st=con.CreateStatement();
int a=st.executeUpdate("insert into student values(2,'asha')");
if(a<=0)
System.out.println("Data not inserted");
else
System.out.println("Data inserted");
}
}
```

/*To perform Delete operation*/

```
import java.sql.*;
public class Delete
{
public static void main(String args[])throws Exception
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

```
System.out.println("Driver loaded");

Connection
con=DriverManager.getConnection("jdbc:odbc:info","system","tiger");

System.out.println("Connection Established");

Statement st=con.CreateStatement();

int a=st.executeUpdate("Delete table student");

System.out.println("table deleted");

}

}
```

6. Write a program by using JDBC to execute an update query without using PreparedStatement and display the results.

```
/*To update a table*/


import java.sql.*;

public class Update

{

public static void main(String args[])throws Exception

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

System.out.println("Driver loaded");
```

```
Connection
con=DriverManager.getConnection("jdbc:odbc:info","system","tiger");

System.out.println("Connection Established");

Statement st=con.CreateStatement();

int a=st.executeUpdate("alter table student name='harish' where id=2);

System.out.println("table altered or updated");

} }
```

7. Write a program by using JDBC to execute an update query by using PreparedStatement and display the results.

```
import java.sql.*;

public class Update

{

public static void main(String args[])throws Exception

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

System.out.println("Driver loaded");

Connection
con=DriverManager.getConnection("jdbc:odbc:info","system","tiger");

System.out.println("Connection Established");

PreparedStatement ps=con.PrepareStatement("update student1 set name=?
where id=?");
```

```
ps.setString(1,"bobby");
ps.setInt(2,35);
int a=ps.executeUpdate();


if(a==1)
System.out.println("Record saved");
else
System.out.println("Record not saved");
ps.close();
con.close();
}
}
```

8. Write a program to execute a stored procedure in the database by using CallableStatement and display the results.

```
import java.sql.*;
public class Proc
{
public static void main(String args[])throws Exception
{
```

```
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

System.out.println("Driver loaded");

Connection
con=DriverManager.getConnection("jdbc:odbc:info","system","tiger");

System.out.println("Connection Established");

CallableStatement stmt=con.PrepareCall("{call INSERT(?,?)");

stmt.setString(1,"bobby");

stmt.setInt(2,35);

stmt.execute();

}

}
```

9. Write a program to display a greeting message in the browser by using HttpServlet.

```
import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;
```

```
public class HelloWorld extends HttpServlet {

private String message;

@Override
public void init() throws ServletException {
message"Hello World"
}

@Override
public void doGet(HttpServletRequest request,
HttpServletResponse response) throws ServletException, IOException {

// Set response content type
response.setContentType("text/html");

// Actual logic goes here.
PrintWriter out = response.getWriter();
out.println("<h1>Hurray !!\n Servlet is Working!! </h1>");
}
```

```
@Override
public void doPost(HttpServletRequest request,
HttpServletResponse response) throws ServletException, IOException {


}


@Override
public void destroy() {
// resource release
}
}
```

/* WEB.XML*/

```
<web-app>
<servlet>
<servlet-name>HelloServlet</servlet-name>
<servlet-class>HelloWorld</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>HelloServlet</servlet-name>
```

```
<url-pattern>/HelloServlet</url-pattern>

</servlet-mapping>

</web-app>
```

10.Write a program to receive two numbers from a HTML form and display their sum in the browser by using HttpServlet.

```
import java.io.*;

import javax.servlet.*;

import javax.servlet.http.*;


public class Add extends HttpServlet

{

public void doPost(HttpServletRequest req,HttpServletResponse res)throws
ServletException,IOException

{

res.setContentType("text/html");

PrintWriter pw=res.getWriter();

String n1=req.getParameter("num1");

String n2=req.getParameter("num2");
```

```
int v1=Integer.parseInt(n1);
int v2=Integer.parseInt(n2);
int result=v1+v2;
pw.print("<h1>The sum of two number is"+result);
}
}
```

/*HTML form*/

```
<html>
<head>
<title>hello</title>
<body>
<center>
<form  method="post" action="add">
   <p>Number1:<input type="text" name="num1"/></p>
   <p>Number2:<input type="text" name="num2"/></p>
   <input type="submit" value="ADD"/>
   </form>
</center>
</body>
</html>
```

/*WEB.XML*/

```
<web-app>

   <servlet>
   <servlet-name>cookie1</servlet-name>
   <servlet-class>Add</servlet-class>
   </servlet>

   <servlet-mapping>
   <servlet-name>cookie1</servlet-name>
   <url-pattern>/add</url-pattern>
   </servlet-mapping>
</web-app>
```

11.Write a program to display a list of five websites in a HTML form and visit to the selected website by using Response redirection.

```
import java.io.*;
import java.servlet.*;
import javax.servlet.http.*;
public class MySearcher extends HttpServlet
{
```

```
protected void doGet(HttpServletRequest request,HttpServletResponse
response)throws

servletException,IOException

{

String name=request.getParameter("name");

PrintWriter out=response.getWriter();

if(name.equals("www.google.com"))

else if(name.equals("www.amazon.in"))

response.sendRedirect("http://www.amazon.im");

else if(name.equals("www.facebook.com");

response.sendRedirect("hhtp://wwww.facebook.com");

else if(name.equals("www.yahoo.com"))

response.sendRequest("www.gmail.com"))

else

out.println("web page not found");

}

}
```

/*HTML FORM*/

```
<html>

<head>

<title>
```

Servlets

</title>

</head>

<form action="abcd">

name:input type="text" name="name">

<input type="submit" value="submit">

</form>

</body>

</html>

/*WEB.XML*/

<web-app>

<servlet>

<servlet-name>searcher</servlet-name>

<servlet-class>MySearcher</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>searcher</servlet-name>

<url-pattern>/abcd</url-pattern>

</servlet-mapping>

</web-app>

12.Write a program to store the user information into Cookies. Write another program to display the above stored information by retrieving from Cookies.

```
/* FIRST SERVLET*/
   import java.io.*;
   import javax.servlet.*;
   import javax.servlet.http.*;



   public class FirstServlet1 extends HttpServlet {


     public void doPost(HttpServletRequest request, HttpServletResponse
response){
       try{


       response.setContentType("text/html");
       PrintWriter out = response.getWriter();


       String n=request.getParameter("userName");
       out.print("Welcome "+n);
```

```
        Cookie ck=new Cookie("uname",n);//creating cookie object
        response.addCookie(ck);//adding cookie in the response

        //creating submit button
        out.print("<form method='post' action='servlet2'>");
        out.print("<input type='submit' value='go'>");
        out.print("</form>");

        out.close();

            }catch(Exception e){System.out.println(e);}
      }
    }

/*SECOND SERVLET*/

    import java.io.*;
    import javax.servlet.*;
    import javax.servlet.http.*;

    public class SecondServlet1 extends HttpServlet {
```

```
   public void doPost(HttpServletRequest request, HttpServletResponse response)
    {
      try{

      response.setContentType("text/html");
      PrintWriter out = response.getWriter();

      Cookie ck[]=request.getCookies();
      out.print("Hello "+ck[0].getValue());

      out.close();

          }
catch(Exception e)
{
System.out.println(e);}
      }
  }
```

/*WEB.XML*/

```
<web-app>

<servlet>
<servlet-name>cookie1</servlet-name>
<servlet-class>FirstServlet1</servlet-class>
</servlet>

<servlet-mapping>
<servlet-name>cookie1</servlet-name>
<url-pattern>/servlet1</url-pattern>
</servlet-mapping>

<servlet>
<servlet-name>cookie2</servlet-name>
<servlet-class>SecondServlet1</servlet-class>
</servlet>

<servlet-mapping>
<servlet-name>cookie2</servlet-name>
<url-pattern>/servlet2</url-pattern>
</servlet-mapping>
```

```
   </web-app>
```

/*HTML FORM*/

```
<html>
<head>
<title>hello</title>
<body>
<form  method="post" action="servlet1">
   Name:<input type="text" name="userName"/><br/>
   <input type="submit" value="go"/>
   </form>
</body>
</html>
```

13.Write a program in Java Beans to add a Button to the Bean and display the number of times the button has been clicked.

```
import java.awt.*;

   public class Counter extends Panel
   {
```

```
    private long count=0;
    private Label label;
    private long maxValue=20;

     public void setMaxValue(long max)
     {
        maxValue=max;
     }
public long getMaxValue()
  {
     return maxValue;
  }
public counter()
  {
    setBackground(Color.Blue);
    setForeground(Color.White);
    label=new Label(""+count);
    add(label);
  }
public void increment()
  {
     if(count<maxValue)
```

```
        {
            count++;
            label.setText(count+"");
        }
        else
          label.setText("!!");
     }
   }
```

/*MANIFEST FILE*/

```
  Name:Counter.class
   Java-Bean:True
```

/*CREATION OF JAR FILE*/

```
jar cvfm tissue.jar manifest.mf Counter.class
```

14.Write a program for Java Bean with Simple property by using SimpleBeanInfo class.

```
import java.awt.*;
import java.io.serializable;
public class SimpleBean extends Canvas implements Serializable
```

```
{
  public SimpleBean()
  {
    setSize(60,40);
    setBackground(color,orange);
  }
}

/*MANIFEST FILE*/
Name:SimpleBean.class
 Java-Bean:True

/*JAR FILE CREATION*/
jar cvfm counter.jar MANIFEST.MF SimpleBean.class
```

14.Write a program demonstrate stateless session bean.

Login.html

```
<HTML>
<BODY>
<FONT SIZE= 10 ALIGN=CENTER>  <B> Calculate Interest </B> </FONT>
        <FORM ACTION = "http://127.0.0.1:8080/loanctx/servlet/LoanServlet">
   <TABLE>
        <TR>
        <TD>  RATE </TD> <TD> <INPUT TYPE = TEXT NAME = "rate"> </TD>
        </TR>
        <TR>
        <TD>  TIME </TD> <TD> <INPUT TYPE = TEXT NAME = "time"> </TD>
        </TR>
        <TR>
        <TD>  AMOUNT </TD> <TD>  <INPUT TYPE = TEXT NAME = "amount"> </TD>
```

```
        </TR>
        <TR>
        <TD> <INPUT TYPE = SUBMIT VALUE = "Calculate"> </TD>
        </TR>
   </TABLE>
        </FORM>
</BODY>
</HTML>
```

Loan.java

```
import javax.ejb.EJBObject;
public interface Loan extends EJBObject
{
   public float calculateInterest(float rate, float time, float amount)
throws java.rmi.RemoteException;
}
```

LoanBean.java

```
import java.util.*;
import java.io.*;
import javax.ejb.SessionBean;
import javax.ejb.SessionContext;
public class LoanBean implements SessionBean {
```

```
    private  javax.ejb.SessionContext m_ctx = null;
    public void setSessionContext(SessionContext ctx) {
        m_ctx = ctx;
    }
    public void ejbCreate() throws java.rmi.RemoteException,
javax.ejb.CreateException {
        System.out.println("ejbCreate() on obj " + this);
    }
    public void ejbRemove() {
        System.out.println("ejbRemove() on obj " + this);
    }
    public void ejbActivate() {
        System.out.println("ejbActivate() on obj " + this);
    }
    public void ejbPassivate() {
        System.out.println("ejbPassivate() on obj " + this);
    }
    public float calculateInterest(float rate, float time, float amount)
throws java.rmi.RemoteException {
       float interest = time * amount * (rate / 100);
```

```
      return interest;
   }
}
```

LoanHome.java

```
import javax.ejb.EJBHome;
public interface LoanHome extends EJBHome
{
   public Loan create() throws java.rmi.RemoteException,
javax.ejb.CreateException;
}
```

LoanServlet.java

```
import java.io.*;
import javax.servlet.*;
import javax.naming.*;
import javax.servlet.http.*;
import javax.rmi.PortableRemoteObject;
import javax.ejb.*;
```

```
public class LoanServlet extends HttpServlet
{
  public void doGet (HttpServletRequest request,HttpServletResponse
response) throws ServletException, IOException
  {
        PrintWriter out = response.getWriter();
        response.setContentType("text/html");
        float rate =
Float.valueOf(request.getParameter("rate")).floatValue();
        float time =
Float.valueOf(request.getParameter("time")).floatValue();
        float amount =
Float.valueOf(request.getParameter("amount")).float      Loan
myLoanRemote = null;Value();
    Loan myLoanRemote = null;
    LoanHome myLoanHome = null;
    InitialContext initContext = null;
    try {
      initContext = new InitialContext();
    }catch (Exception e) {
       out.println("First " + e.toString());
```

```
    }
    try {
       String JNDIName = "ejb/SimpleLoan";
       Object obj = initContext.lookup(JNDIName);
       myLoanHome = (LoanHome)PortableRemoteObject.narrow(obj,
LoanHome.class);
    }catch(Exception e) {
       out.println("Second " + e.toString());
    }
   try {
        myLoanRemote = myLoanHome.create();
     }catch(CreateException e) {
       out.println("Third " + e.toString());
     }
     float interest = myLoanRemote.calculateInterest(rate, time,
amount);
     out.println("<B>Interest : " + interest + " </B>");
}
}
```