EX.NO.01 CREATING A WEBPAGE USING IMAGE MAP

Aim

To Create a web page with the following using HTML

i) To embed an image map in a web page

ii) To fix the hot spots

iii) Show all the related information when the hot spots are clicked.

Algorithm

Step1 :Open a notepad.

Step 2:Write the code for imagemap.html.

Step 3: Enter a program that includes tags for <MAP> and other tags.

Step 4: Insert Hyperlink using <A href>.

Step5: Save the file with .html extension.

Step6: Run the program in a web browser

Step: Display Results.

HOME.HTML

```
<html>
   <head>
      <title>Home - States of India!!!</title>
   </head>
   <body bgcolor="gold">
      <h1><u><center>Republic of India</center></u></h1>
      <p>
 India is the Seventh Lagest country in the world by geographical area, the second
most populous country with over 1.2 billion people, and the populus democracy in the
world. India is a federal constitutional republic with a parliamentry democracy consisting
of 28 states and 7 union Terriories.
      </p>
   <center>
      <img align="center" size="100" src="IndiaMap.jpg" usemap="#india"/>
      <map name="india">
         <area shape="circle" coords="160,340,20" href="ANDHRAPRADESH.html">
         <area shape="circle" coords="120,440,10" href="KERALA.html">
         <area shape="circle" coords="110,370,20" href=" KARNATAKA.html">
         <area shape="circle" coords="140,420,20" href="TAMILNADU.html">
      </map>
   </center>
   <h2> Features</h2>
   <ul>
      <li><b>Population</b> - 1,028,783,343(2001 census).
      <li><b>Capital</b> - New Delhi
      <li><b>Largest City</b> - Mumbai
      <li><b>Currency</b> - Indian Rupee
      <li><b>Time Format</b> - IST (UTC + 5:30)
```

```
        <li><b>NAtion Sport</b> - Hockey
        <li><b>Current PM</b> - Manmohan Singh
        <li><b>Current President</b> - Prathiba Patil
        </li>
    </ul>
    <h2>
        <b>To view details of states please click on the specified area in the
              map !!!</b>
</h2>
</body>
</html>
```

ANDHRAPRADESH.HTML

```
<html>
    <head><title>Andhra Pradesh - India</title></head>
    <body bgcolor="tan">
        <h1><center>Andhra Pradesh</center></h1>
        <h3>A.P., is a state situated on the southeastern coast of India. It is
            India's fourth  largest state by area and fifth largest by population.</h3>
        <h3>
            <ul>
                <li>Districts<i> - 23</i>
                <li>Capital City<i> - Hyderabad</i>
                <li>Largest City<i> - Hyderabad</i>
                <li>Governor<i> - E. S. L. Narasimhan</i>
                <li>Chief Minister<i> - N. Kiran Kumar Reddy</i>
                <li>Population<i> - 78,323,330</i>
                <li>Tourist spots<i> - Tirumala Tirupati, Charminar, Golconda Fort,
                        Chandragiri, Chowmahalla Place, Falaknuma Palace etc.,</i>
            </ul>
            <a href="Home.html">back</a>
    </body>
</html>
```

KARNATAKA.HTML

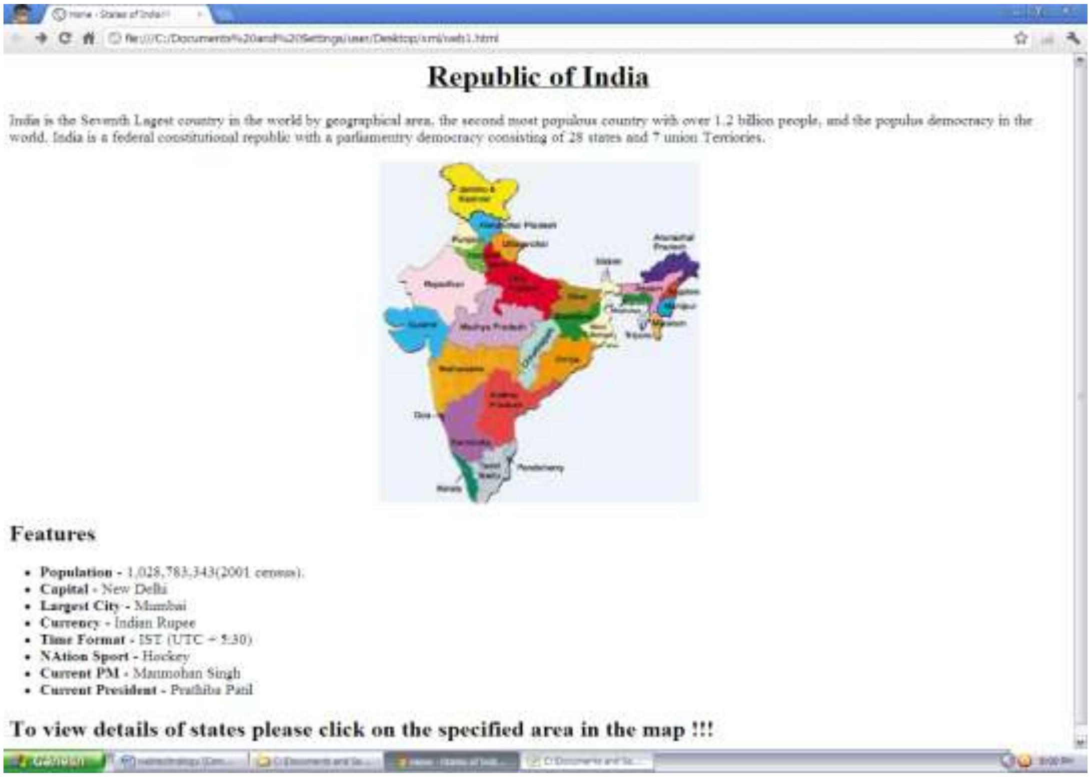
```
<html>
    <head><title>Karnataka - India</title></head>
    <body bgcolor="wheat">
        <h1><center>Karnataka</center></h1>
        <h3>
            <ul>
                <li>Districts<i> - 30</i>
                <li>Capital City<i> - Bangalore</i>
                <li>Largest City<i> - Bangalore</i>
                <li>Governor<i>- Hansraj Bhardwaj</i>
                <li>Chief Minister<i> - D. V. Sadananda Gowda</i>
                <li>Population<i> - 61,130,704</i>
                <li>Tourist spots<i> - Gol Gumbaz, Mysore Palace, Keshava Temple etc.,</i>
            </ul>
        </h3>
        <a href="Home.html">back</a>
    </body>
</html>
```

KERALA.HTML

```
<html>
    <head><title>Kerala - India</title></head>
    <body bgcolor="indianred">
        <h1><center>Kerala</center></h1>
        <h3>
            <ul>
                <li>Districts<i> - 14</i>
                <li>Capital City<i> - Thiruvanandapuram</i>
                <li>Largest City<i> - Thiruvanandapuram</i>
                <li>Governor<i> - Hansraj Bhardwaj</i>
                <li>Chief Minister<i> -Oommen Chandy </i>
                <li>Population<i> - 33,387,677</i>
                <li>Tourist spots<i> - Edakkal Caves, Palayur, Kovalam Beach,Munnar, Kochi, Alapuzha etc.,</i>
            </ul>
        </h3>
        <a href="Home.html">Back</a>
    </body>
</html>
```

TAMILNADU.HTML

```
<html>
    <head><title>Tanil Nadu - India</title></head>
    <body bgcolor="palegreen">
        <h1><center>Tamil Nadu</center></h1>
        <h3>is one of the 28 states of India. Its capital and largest city is Chennai.
            Tamil Nadu lies in the southernmost part of the Indian Peninsula and
            is bordered by the States of puducherry, Kerala,  Karnataka, Andha Pradesh.
        </h3>
        <h3>
            <ul>
                <li>Districts<i> - 32</i>
                <li>Capital City<i> - Chennai</i>
                <li>Largest City<i> - Chennai</i>
                <li>Governor<i> - Konijeti Rosaiah</i>
            <li>Chief Minister<i> - Jayalalithaa</i>
            <li>Population<i> - 72,138,958</i>
            <li>Tourist spots<i> - Mamallapuran, Ooty, Kodaikanal,  Marina,
                  Mudurai Meenakshi Amman Temple, Thanjavur etc.,</i>
         </ul>
         <a href="Home.html">back</a>
   </body>
</html>
```



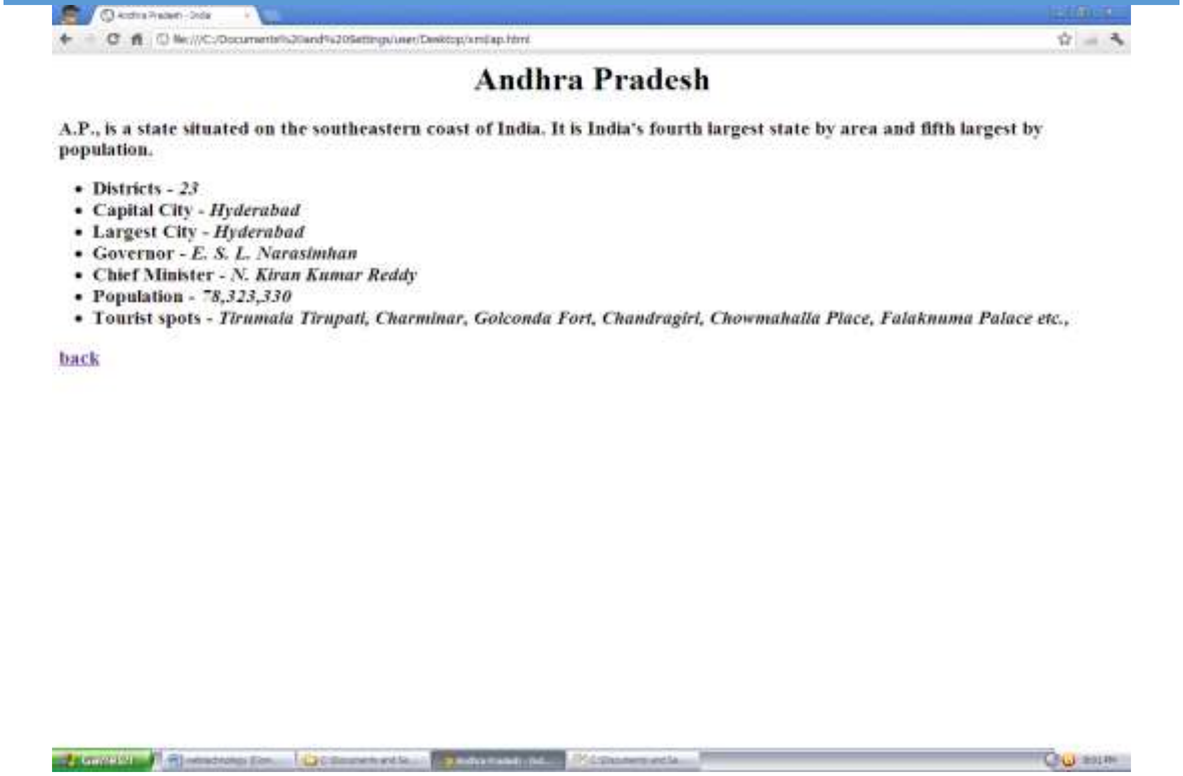
Republic of India

India is the Seventh Lagest country in the world by geographical area, the second most populous country with over 1.2 billion people, and the populus democracy in the world. India is a federal constitutional republic with a parliamentry democracy consisting of 28 states and 7 union Terriories.

Features

- **Population -** 1,028,783,343(2001 census).
- **Capital** - New Delhi
- **Largest City -** Mumbai
- **Currency** - Indian Rupee
- **Time Format** - IST (UTC + 5:30)
- **NAtion Sport -** Hockey
- **Current PM** - Manmohan Singh
- **Current President** - Prathiba Patil

To view details of states please click on the specified area in the map !!!

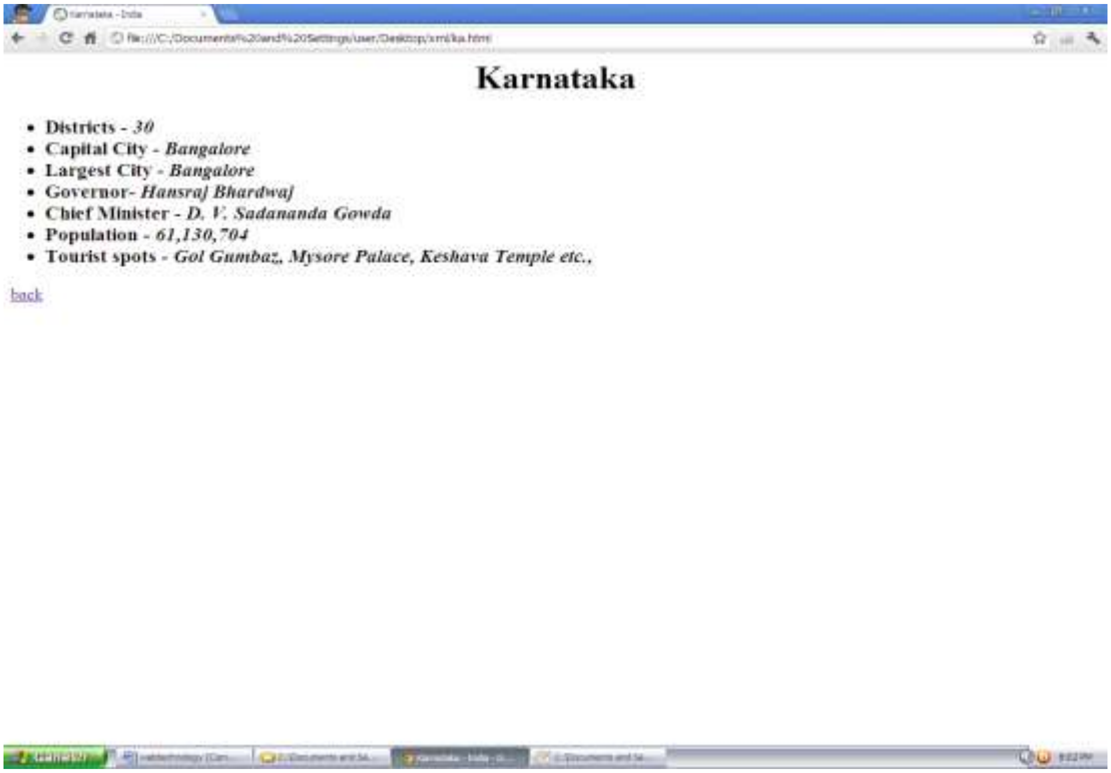


Andhra Pradesh

A.P., is a state situated on the southeastern coast of India. It is India's fourth largest state by area and fifth largest by population.

- **Districts -** ***23***
- **Capital City -** ***Hyderabad***
- **Largest City -** ***Hyderabad***
- **Governor -** ***E. S. L. Narasimhan***
- **Chief Minister -** ***N. Kiran Kumar Reddy***
- **Population -** ***78,323,330***
- **Tourist spots -** ***Tirumala Tirupati, Charminar, Golconda Fort, Chandragiri, Chowmahalla Place, Falaknuma Palace etc.,***

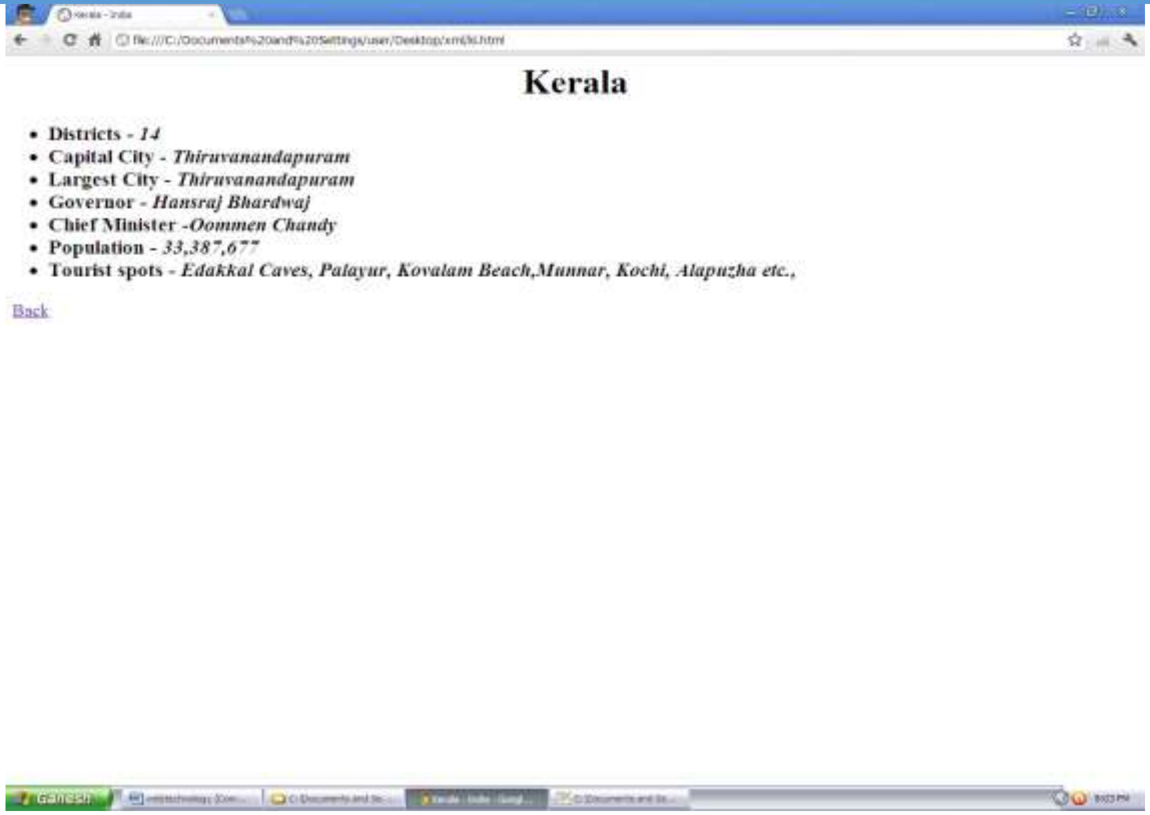
back



Karnataka

- **Districts -** ***30***
- **Capital City -** ***Bangalore***
- **Largest City -** ***Bangalore***
- **Governor-** ***Hansraj Bhardwaj***
- **Chief Minister -** ***D. V. Sadananda Gowda***
- **Population -** ***61,130,704***
- **Tourist spots -** ***Gol Gumbaz, Mysore Palace, Keshava Temple etc.,***

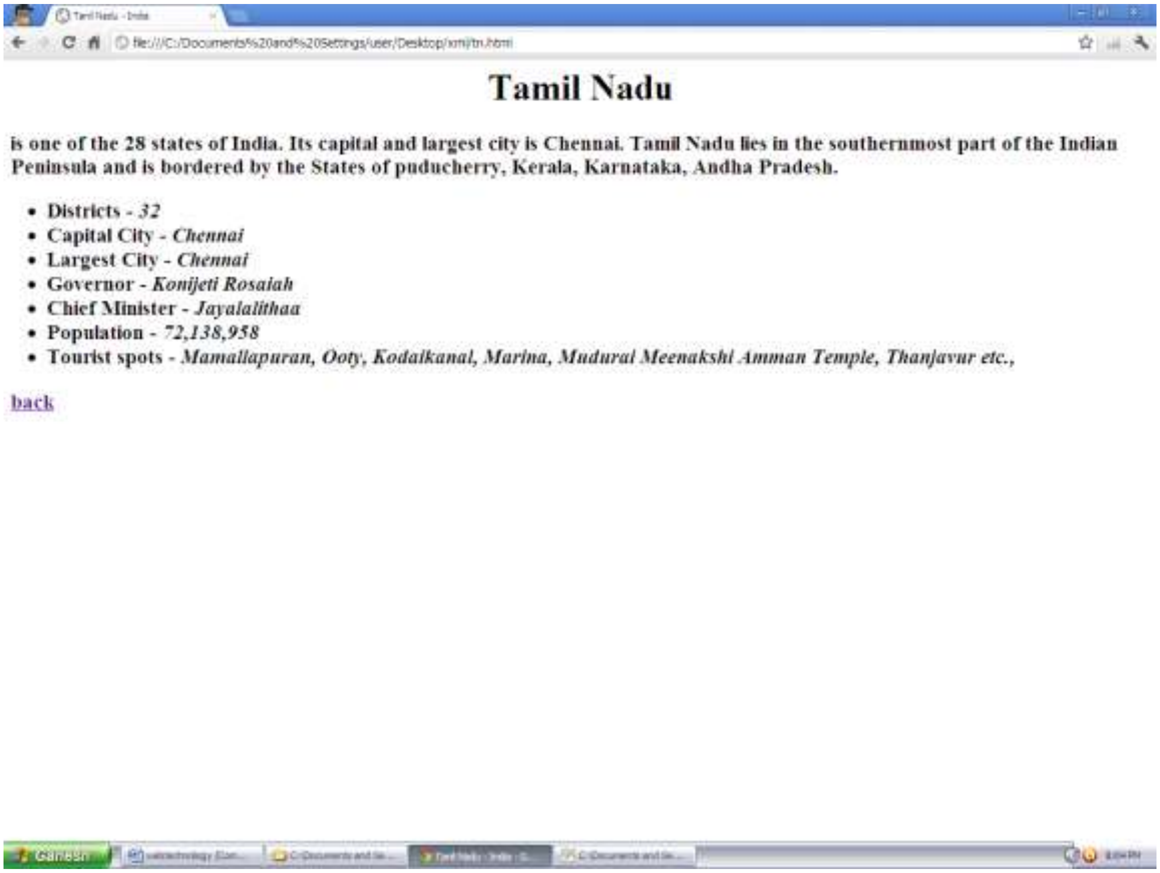
back


Kerala - India
file:///C:/Documents%20and%20Settings/user/Desktop/xml/kl.html
Ganesh

Kerala

- **Districts -** ***14***
- **Capital City -** ***Thiruvanandapuram***
- **Largest City -** ***Thiruvanandapuram***
- **Governor -** ***Hansraj Bhardwaj***
- **Chief Minister -*****Oommen Chandy***
- **Population -** ***33,387,677***
- **Tourist spots -** ***Edakkal Caves, Palayur, Kovalam Beach,Munnar, Kochi, Alapuzha etc.,***

Back


Tamil Nadu - India
file:///C:/Documents%20and%20Settings/user/Desktop/xml/tn.html
Ganesh

Tamil Nadu

is one of the 28 states of India. Its capital and largest city is Chennai. Tamil Nadu lies in the southernmost part of the Indian Peninsula and is bordered by the States of puducherry, Kerala, Karnataka, Andha Pradesh.

- **Districts -** ***32***
- **Capital City -** ***Chennai***
- **Largest City -** ***Chennai***
- **Governor -** ***Konijeti Rosaiah***
- **Chief Minister -** ***Jayalalithaa***
- **Population -** ***72,138,958***
- **Tourist spots -** ***Mamallapuran, Ooty, Kodaikanal, Marina, Mudurai Meenakshi Amman Temple, Thanjavur etc.,***

back

EX.NO.02 CREATION OF WEB PAGE USING CASCADING STYLE SHEET

Aim:

To create a HTML page using types of Cascading Style Sheet.

ALGORITHM

Internal CSS:

STEP 1: Create a HTML program with <style> tag.
STEP 2: Inside the <style> tag, specify the format required for that web page.
STEP3 : Run the program with a web browser.

External CSS:

STEP 4: Open a notepad, type the needed CSS in it and save it with .css extension.
STEP5: Refer this .css file in the HTML using the tag <link>.
STEP 6: Run the program with a web browser.

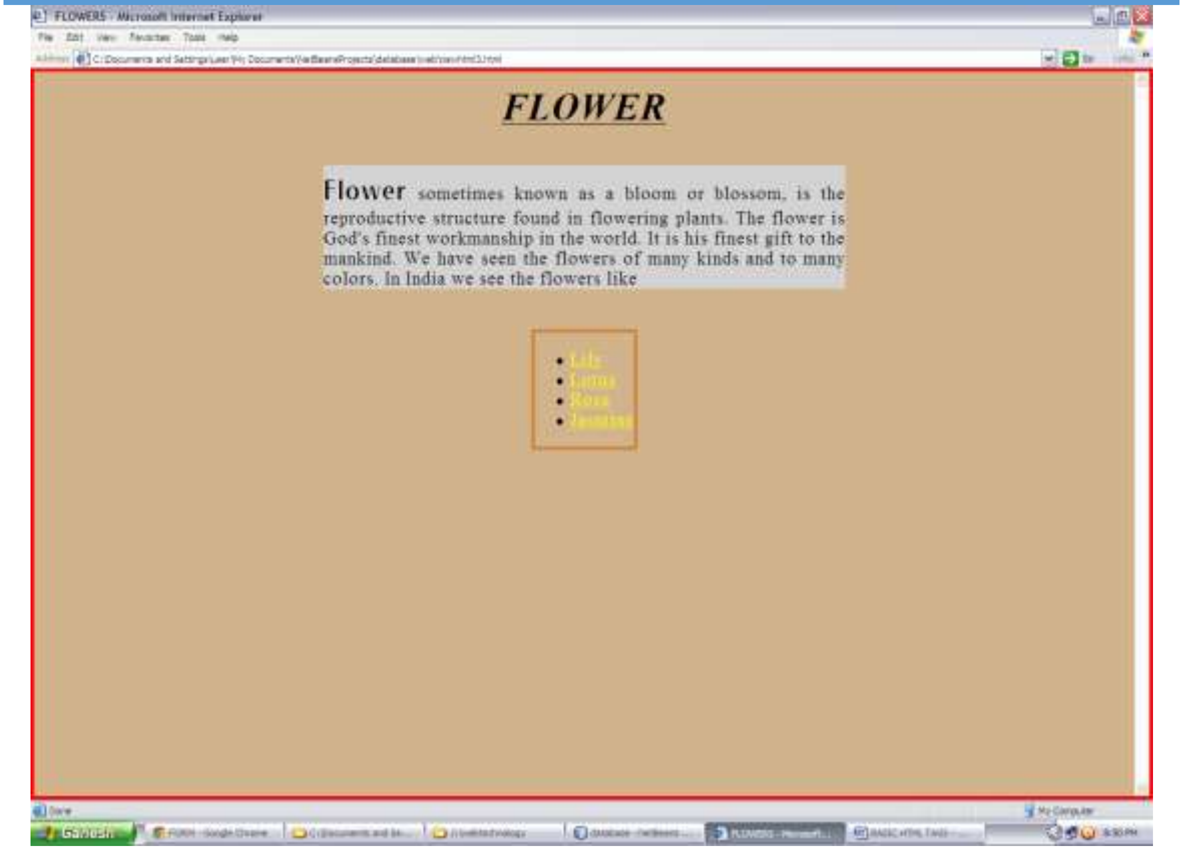
CSSFILE.HTML

```
<html>
<head>
<title>FLOWERS</title>
<!--Extended Style Sheet -->
<link rel="stylesheet" type="text/css" href="newcss.css">
<!--        Embed Style Sheet-->
<style type="text/css">
p{
background-color: lightgrey;
text-align: justify;
margin: 2em 7em;
}

</style>
</head>
<body id="body">
<h1>FLOWER</h1>
<p>
<span style="font: 200 x-large fantasy">Flower</span>
sometimes known as a bloom or blossom,
is the reproductive structure found in flowering plants.
The flower is God's finest  workmanship in the world.
It is his finest gift to the mankind.
We have seen the flowers of many kinds and to many colors.
In India we see the flowers like
</p>
<!--        Inline Sytle Sheet-->
<table style="background-position: center;text-align: center;padding: 3px;">
<tr>
<td align="left">
<div class="div">
<ul>
<li><a href="">Lily</a></li>
```

```
<li><a href="">Lotus</a></li>
<li><a href="">Rose</a></li>
<li><a href="">Jasmine</a></li>
</ul>
</div>
</td>
</tr>
</table>
</body>
</html>
```

CSS.CSS

```
h1,h2{
text-decoration: underline;
font-style: italic;
text-align: center;
}
#body{
background-color: tan;
border: red dotted;
text-align: center;
}
.div{
border: peru solid ;
}
*{
letter-spacing: 1px;
}
a:link{
color: black;
}
a:visited{
color: yellow;
}
a:hover{
color: green;
}
a:active{
color: blue;
}
ul li{
font-size: small;
}
```


FLOWERS - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address C:\Documents and Settings\user\My Documents\NetBeansProjects\database\web\newhtml3.html
Done
My Computer

FLOWER

Flower sometimes known as a bloom or blossom, is the reproductive structure found in flowering plants. The flower is God's finest workmanship in the world. It is his finest gift to the mankind. We have seen the flowers of many kinds and to many colors. In India we see the flowers like

- Lily
- Lotus
- Rose
- Jasmine

EX.NO.03 XML FOR USER DETAILS

Aim

To write a program which takes user id as an input and returns the user details by taking the user information from the XML document

Algorithm

1. Create an xml file containing the user information.
2. The user information are name, collegename, department and CGPA.
3. Write an HTML file in which the user ID is taken as an input,
4. The HTML document will the load the xml file created in step 1.
5. Retrieve the information of the user whose ID is provided.

XML FILE

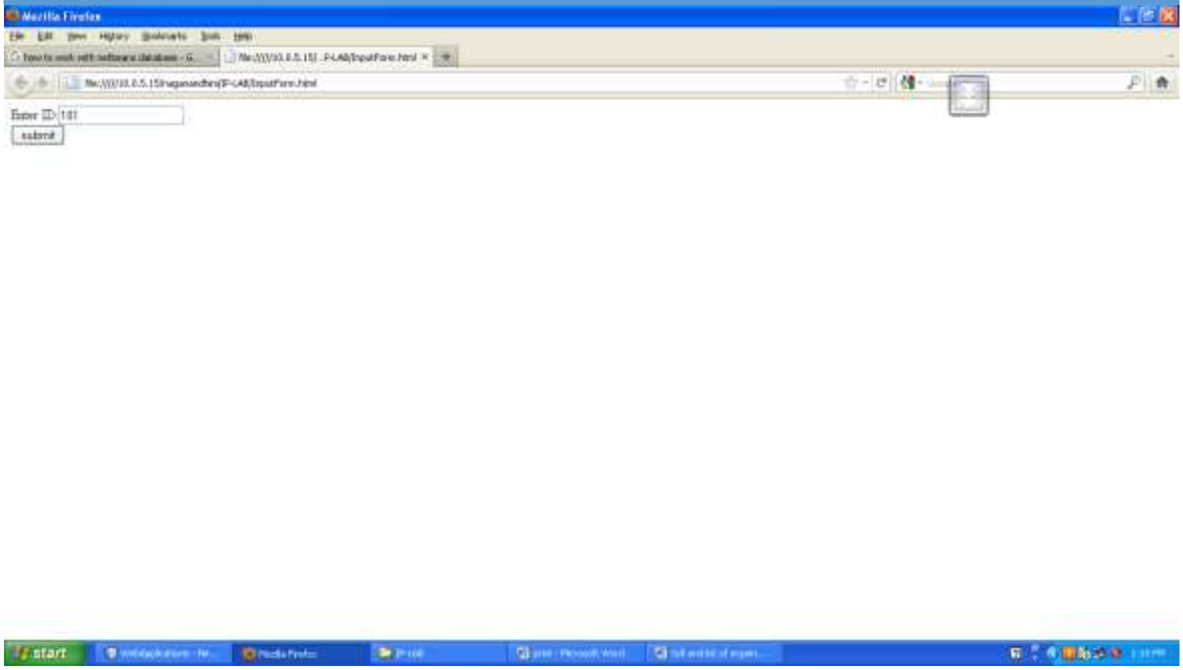
```
<?xml version="1.0" encoding="utf-8"?>
<Student>
<Person-Details>
<id>101</id>
<name>Ramya</name>
<college>JJCET</college>
<dept>CSE</dept>
<cgpa>8</cgpa>
</Person-Details>
<Person-Details>
<id>102</id>
<name>Akshaya</name>
<college>JJCET</college>
<dept>ECE</dept>
<cgpa>8.5</cgpa>
</Person-Details>
<Person-Details>
<id>103</id>
<name>Archana</name>
<college>JJCET</college>
<dept>CSE</dept>
<cgpa>8.7</cgpa>
</Person-Details>
<Person-Details>
<id>104</id>
<name>Priya</name>
<college>JJCET</college>
<dept>Mech</dept>
<cgpa>7.9</cgpa>
</Person-Details>
<Person-Details>
<id>105</id>
<name>tanusri</name>
<college>JJCET</college>
<dept>EEE</dept>
```

```
<cgpa>8.1</cgpa>
</Person-Details>
<Person-Details>
<id>106</id>
<name>Raja</name>
<college>JJCET</college>
<dept>CSE</dept>
<cgpa>8</cgpa>
</Person-Details>
<Person-Details>
<id>107</id>
<name>Kamalini</name>
<college>JJCET</college>
<dept>ECE</dept>
<cgpa>8.5</cgpa>
</Person-Details>
<Person-Details>
<id>108</id>
<name>rani</name>
<college>JJCET</college>
<dept>CSE</dept>
<cgpa>8.7</cgpa>
</Person-Details>
<Person-Details>
<id>109</id>
<name>Meenu</name>
<college>JJCET</college>
<dept>Mech</dept>
<cgpa>7.9</cgpa>
</Person-Details>
<Person-Details>
<id>110</id>
<name>shyam</name>
<college>JJCET</college>
<dept>EEE</dept>
<cgpa>8.1</cgpa>
</Person-Details>
</Student>
```

HTML FILE

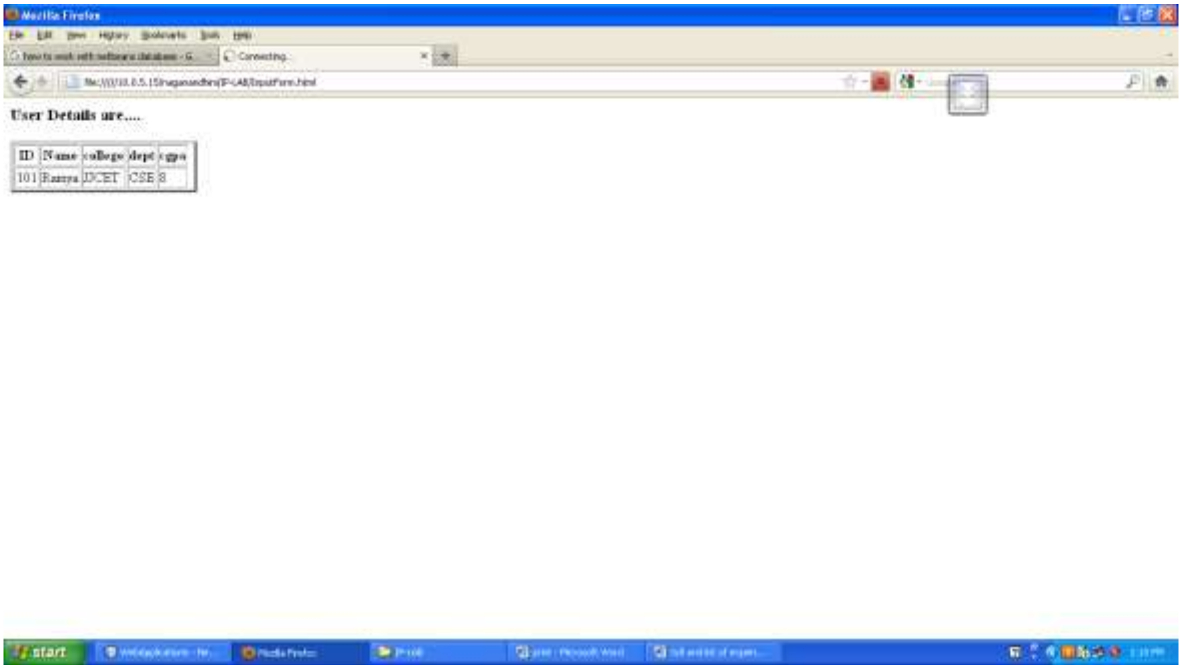
```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<script type="text/javascript">
```

```
function Display()
{
if(window.XMLHttpRequest)
{
//code for IE7+,Firefox,Chrome,Opera,Safari
xmlhttp=new XMLHttpRequest();
}
xmlhttp.open("GET","UserInfo.xml",false);
xmlhttp.send();
xmlDoc=xmlhttp.responseXML;
var x=xmlDoc.getElementsByTagName("Person-Details");
var key_id=document.getElementById('key').value;
for(i=0;i<x.length;i++)
{
if(key_id.match(x[i].getElementsByTagName("id")[0].childNodes[0].nodeValue))
j=i;
}
document.write("<h3> User Details are....</h3>");
document.write("<table
border='3'><tr><th>ID</th><th>Name</th><th>college</th><th>dept</th><th>cgp
a</th></tr>");
document.write("<tr><td>");
document.write(x[j].getElementsByTagName("id")[0].childNodes[0].nodeValue);
document.write("</td><td>");
document.write(x[j].getElementsByTagName("name")[0].childNodes[0].nodeValue);
document.write("</td><td>");
document.write(x[j].getElementsByTagName("college")[0].childNodes[0].nodeValue)
;
document.write("</td><td>");
document.write(x[j].getElementsByTagName("dept")[0].childNodes[0].nodeValue);
document.write("</td><td>");
document.write(x[j].getElementsByTagName("cgpa")[0].childNodes[0].nodeValue);
document.write("</td><td>");
document.write("</table>");
}
</script>
<form name="myform">
Enter ID:<input type="text" id="key"/><br/>
<input type="button" value="submit" onclick="Display()"/>
</form>
</body>
</html>
```



Enter ID: 101

submit



User Details are....

ID	Name	college	dept	cgpa
101	Ramya	JJCET	CSE	8

EX.NO.4A HTTP request

Aim
To write a java socket program to demonstrate HTTP Request
Algorithm

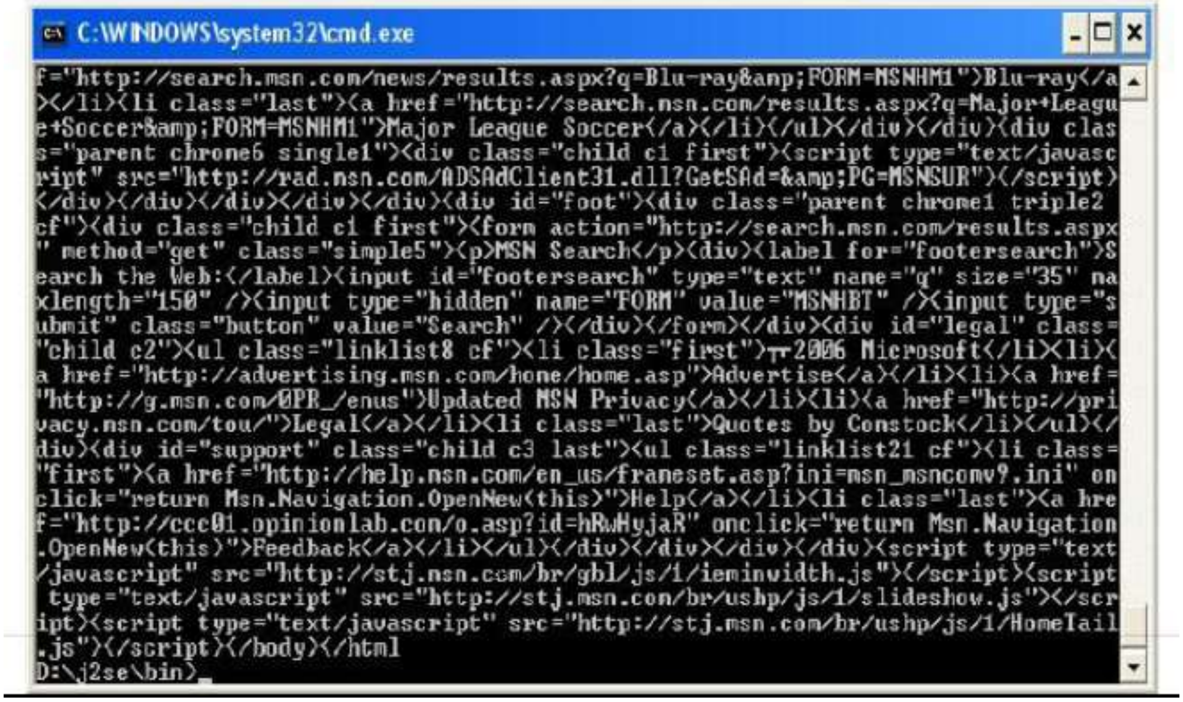
Import all the necessary packages
Create an URL to the server specifying the html page
Get the host and port details from the URL
Request the file from the server using GET method of HTTP Request
Receive the response from the server
Display the response on the console Program

```
import java.io.*;
import java.net.*;
public class HTTP
{
public static void main(String[] args)
{
try
```

Import all the necessary packages
Create an URL to the server specifying the html page
Get the host and port details from the URL
Request the file from the server using GET method of HTTP Request
Receive the response from the server
Display the response on the console

```
import java.io.*;
import java.net.*;
public class HTTP
{
public static void main(String[] args)
{
try
```

```
{
OutputStream to_file = new FileOutputStream("f:\\temp.txt");
URL url = new
URL("http://www.wickedlysmart.com/HeadFirst/HeadFirstJava/HeadFirstJavaInde
x.html");
String protocol = url.getProtocol();
String host = url.getHost();
int port = url.getPort();
if(port == -1) port = 80;
String filename =url.getFile();
System.out.println(filename);
Socket socket = new Socket(host, port);
InputStream from_server = socket.getInputStream();
PrintWriter to_server = new PrintWriter(socket.getOutputStream());
to_server.print("Get" + filename +"\n\n");
to_server.flush();
byte[] buffer = new byte[4096];
int byte_read;
while((byte_read = from_server.read(buffer)) != -1)
{
to_file.write(buffer,0,byte_read);
System.out.print((char)byte_read);
}
socket.close();
to_file.close();
}
catch(Exception e)
{
e.printStackTrace();
}
}}
```



```
f="http://search.msn.com/news/results.aspx?q=Blu-ray&amp;FORM=MSNHM1">Blu-ray</a
></li><li class="last"><a href="http://search.msn.com/results.aspx?q=Major+Leagu
e+Soccer&amp;FORM=MSNHM1">Major League Soccer</a></li></ul></div></div><div clas
s="parent chrome6 single1"><div class="child c1 first"><script type="text/javasc
ript" src="http://rad.msn.com/ADSAdClient31.dll?GetSAd=&amp;PG=MSNSUR"></script>
</div></div></div></div></div><div id="foot"><div class="parent chrome1 triple2
cf"><div class="child c1 first"><form action="http://search.msn.com/results.aspx
" method="get" class="simple5"><p>MSN Search</p><div><label for="footersearch">S
earch the Web:</label><input id="footersearch" type="text" name="q" size="35" na
xlength="150" /><input type="hidden" name="FORM" value="MSNHBT" /><input type="s
ubmit" class="button" value="Search" /></div></form></div><div id="legal" class=
"child c2"><ul class="linklist8 cf"><li class="first">╥2006 Microsoft</li><li><
a href="http://advertising.msn.com/home/home.asp">Advertise</a></li><li><a href=
"http://g.msn.com/0PR_/enus">Updated MSN Privacy</a></li><li><a href="http://pri
vacy.msn.com/tou/">Legal</a></li><li class="last">Quotes by Comstock</li></ul></
div><div id="support" class="child c3 last"><ul class="linklist21 cf"><li class=
"first"><a href="http://help.msn.com/en_us/frameset.asp?ini=msn_msncomv9.ini" on
click="return Msn.Navigation.OpenNew(this)">Help</a></li><li class="last"><a hre
f="http://ccc01.opinionlab.com/o.asp?id=hRwHyjaR" onclick="return Msn.Navigation
.OpenNew(this)">Feedback</a></li></ul></div></div></div></div><script type="text
/javascript" src="http://stj.msn.com/br/gbl/js/1/ieminwidth.js"></script><script
 type="text/javascript" src="http://stj.msn.com/br/ushp/js/1/slideshow.js"></scr
ipt><script type="text/javascript" src="http://stj.msn.com/br/ushp/js/1/HomeTail
.js"></script></body></html
D:\j2se\bin>_
```

EX NO 4B FTP

Aim

To write a java program to demonstrate a simple FTP operation

Algorithm

FTP Client:

Step 1: Establish a connection with the server at a particular port

Step 2: Specify the name of the file to be read

Step 3: Receive the contents of the file from the server

FTP Server:

Accept the connection with the client

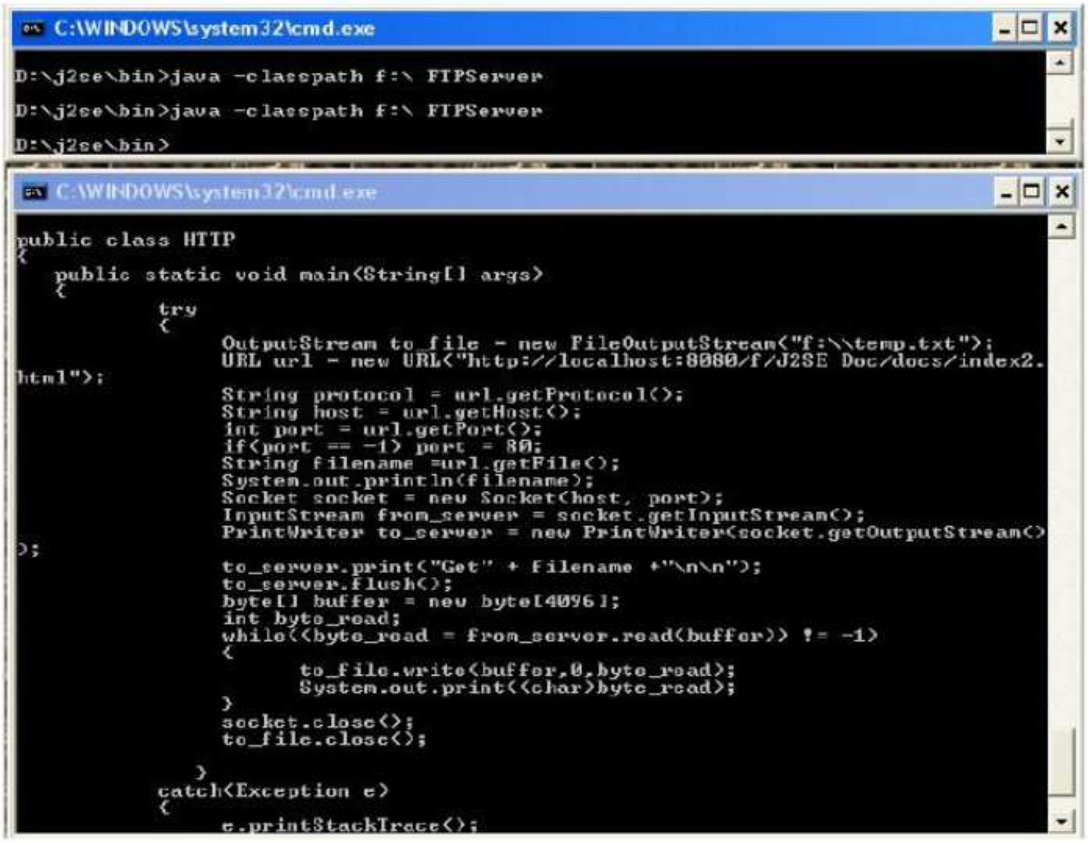
Listen to the port for the name of the file to be sent

Send the file character by character

Terminate the connection

```
import java.io.*;
import java.net.*;
public class FTPClient
{
public static void main(String[] args)
{
try
{
Socket client = new Socket("127.0.0.1",10000);
PrintWriter writer = new PrintWriter(client.getOutputStream());
writer.println("d:\HTTP.java");
writer.flush();
InputStreamReader stream = new InputStreamReader(client.getInputStream());
BufferedReader reader = new BufferedReader(stream);
String str = null;
while((str = reader.readLine()) != null)
{
System.out.println(str);
}
reader.close();
}
catch(Exception e)
{
System.out.println("Connection is terminated by the Server");
} } }
```

```
import java.io.*;
import java.net.*;
public class FTPServer
{
public static void main(String[] arg)
{
try
{
ServerSocket server = new ServerSocket(10000);
Socket client;
client= server.accept();
InputStreamReader stream = new InputStreamReader(client.getInputStream());
BufferedReader reader = new BufferedReader(stream);
String filename = reader.readLine();
PrintWriter writer = new PrintWriter(client.getOutputStream());
FileInputStream fileStream = new FileInputStream(new File(filename));
int ch;
while((ch = fileStream.read()) != -1)
{
writer.write(ch);
writer.flush();
}
writer.close();
}
catch(Exception e)
{
e.printStackTrace();
} } }
```



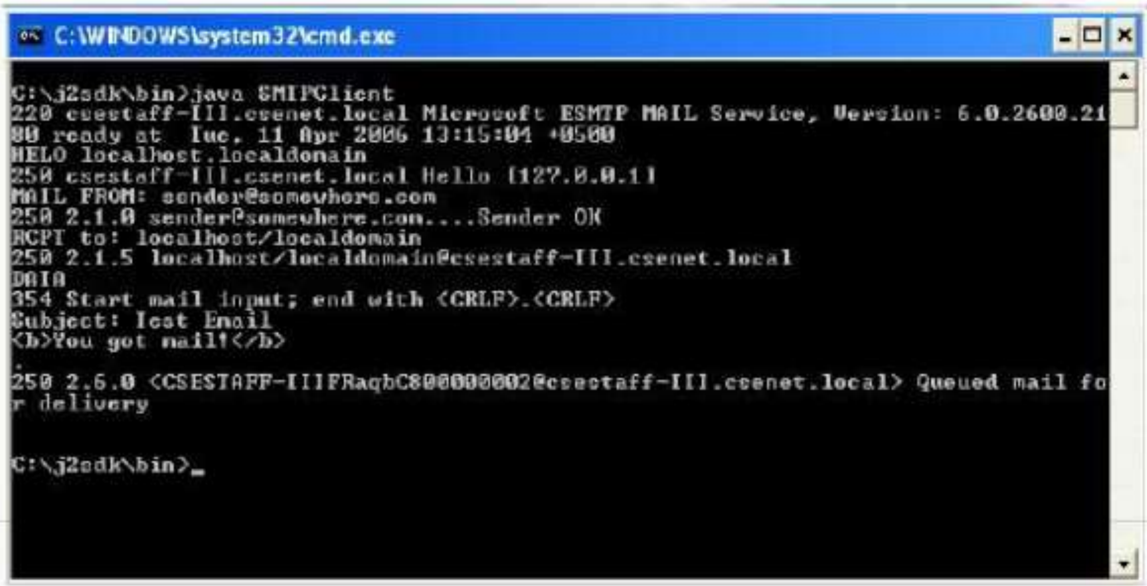
```
D:\j2se\bin>java -classpath f:\ FTPServer

D:\j2se\bin>java -classpath f:\ FTPServer

D:\j2se\bin>
```

```
public class HTTP
{
   public static void main(String[] args)
   {
            try
            {
                 OutputStream to_file - new FileOutputStream("f:\\temp.txt");
                 URL url - new URL("http://localhost:8080/f/J2SE Doc/docs/index2.
html");
                 String protocol = url.getProtocol();
                 String host = url.getHost();
                 int port = url.getPort();
                 if(port == -1) port = 80;
                 String filename =url.getFile();
                 System.out.println(filename);
                 Socket socket = new Socket(host, port);
                 InputStream from_server = socket.getInputStream();
                 PrintWriter to_server = new PrintWriter(socket.getOutputStream()
);
                 to_server.print("Get" + Filename +"\n\n");
                 to_server.flush();
                 byte[] buffer = new byte[4096];
                 int byte_read;
                 while((byte_read = from_server.read(buffer)) != -1)
                 {
                       to_file.write(buffer,0,byte_read);
                       System.out.print((char)byte_read);
                 }
                 socket.close();
                 to_file.close();

              }
            catch(Exception e)
            {
                 e.printStackTrace();
```

EX.NO.4C SMTP

Aim
To write a java socket program to implement a simple SMTP client
Algorithm
Step 1: Import all necessary packages
Step 2: Establish a connection with the server
Step 3: Read the acceptance from the server
Step 4: Say HELO to the server
Step 5: Read the greeting from the server
Step 6: Send sender address to server
Step 7: Read the verification of sender from server
Step 8: Send recipient address to server
Step 9: Read the verification of recipient from server
Step 10: Send DATA command to the server
Step 11: Read the start indication from server
Step 12: Send the message to the server
Step 13: Read the acceptance of message from server
Step 14: Close the connection

```
import java.io.*;
import java.net.Socket;
public class SMTP{
public static void main(String[] args) throws Exception {
String results = send("localhost",25,
"sender@somewhere.com",
"localhost/localdomain",
"Test Email",
"<b>You got mail!</b>");
System.out.println(results);
}
public static String send(String host,int port,String from,String to,String subject,
String message) throws Exception
{
StringBuffer buffer = new StringBuffer();
try {
Socket smtpSocket = new Socket(host, port);
DataOutputStream output = new
DataOutputStream(smtpSocket.getOutputStream());
BufferedReader input =new BufferedReader(new InputStreamReader(
new
DataInputStream(smtpSocket.getInputStream())));
try {
read(input, buffer);
send(output, "HELO localhost.localdomain\r\n", buffer);
read(input, buffer);
send(output, "MAIL FROM: " + from + "\r\n", buffer);
read(input, buffer);
send(output, "RCPT to: " + to + "\r\n", buffer);
read(input, buffer);
send(output, "DATA\r\n", buffer);
read(input, buffer);
send(output, "Subject: " + subject + "\r\n", buffer);
send(output, message, buffer);
send(output, "\r\n.\r\n", buffer);
read(input, buffer);
smtpSocket.close();
}
catch (IOException e) {
System.out.println("Cannot send email as an error occurred.");
}
}
catch (Exception e) {
System.out.println("Host unknown");
}
return buffer.toString();
}
private static void send(DataOutputStream output,String data,StringBuffer buffer)
throws IOException
{
output.writeBytes(data);
buffer.append(data);
}
private static void read(BufferedReader br, StringBuffer buffer) throws IOException
{
int c;
while ((c = br.read()) != -1)
{
buffer.append((char) c);
if (c == '\n')
{
break;
}
}
}}
```



```
C:\j2sdk\bin>java SMTPClient
220 csestaff-III.csenet.local Microsoft ESMTP MAIL Service, Version: 6.0.2600.21
80 ready at  Tue, 11 Apr 2006 13:15:04 +0500
HELO localhost.localdomain
250 csestaff-III.csenet.local Hello [127.0.0.1]
MAIL FROM: sender@somewhere.com
250 2.1.0 sender@somewhere.com....Sender OK
RCPT to: localhost/localdomain
250 2.1.5 localhost/localdomain@csestaff-III.csenet.local
DATA
354 Start mail input; end with <CRLF>.<CRLF>
Subject: Test Email
<b>You got mail!</b>
.
250 2.6.0 <CSESTAFF-IIIFRaqbC800000002@csestaff-III.csenet.local> Queued mail fo
r delivery


C:\j2sdk\bin>_
```

EX.NO.4D POP3

Aim
To write a java socket program to implement a POP3 Client
Algorithm

Get the host name, mailbox user name and password
Establish the connection with the server
Get the number of messages
Retrieve a message whose number is specified by the user
Repeat steps 3 and 4 until the user enters Q to quit Program

```
import java.io.*;
import java.net.*;
import java.util.*;
public class Pop3ClientDemo
{
protected int port = 110;
protected String hostname = "localhost";
protected String username = "";
protected String password = "";
protected Socket socket;
protected BufferedReader br;
protected PrintWriter pw;
// Constructs a new instance of the POP3 client
public Pop3ClientDemo() throws Exception
{
try
{
// Get user input
getInput();
```

Get the host name, mailbox user name and password
Establish the connection with the server
Get the number of messages
Retrieve a message whose number is specified by the user
Repeat steps 3 and 4 until the user enters Q to quit

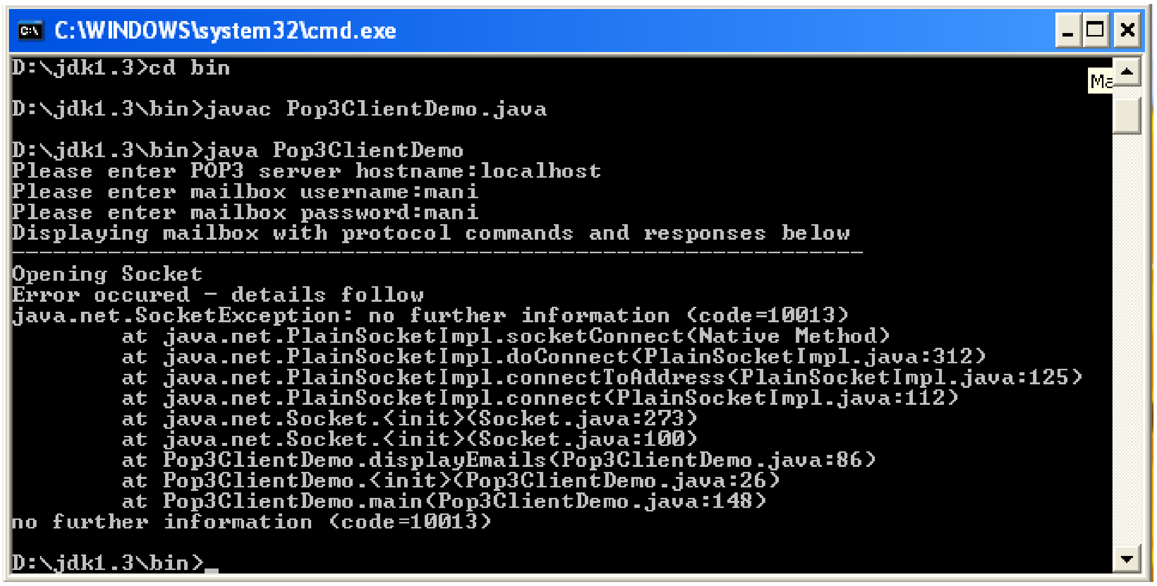
```
import java.io.*;
import java.net.*;
import java.util.*;
public class Pop3ClientDemo
{
protected int port = 110;
protected String hostname = "localhost";
protected String username = "";
protected String password = "";
protected Socket socket;
protected BufferedReader br;
protected PrintWriter pw;
// Constructs a new instance of the POP3 client
```

```
public Pop3ClientDemo() throws Exception
{
try
{
// Get user input
getInput();
//Get the host name, mailbox user name and password
//Establish the connection with the server
//Get the number of messages
//Retrieve a message whose number is specified by the user
//Repeat steps 3 and 4 until the user enters Q to quit
// Get mail messages
displayEmails();
}
catch(Exception e)
{
System.err.println ("Error occured - details follow");
e.printStackTrace();
System.out.println(e.getMessage());
}
}
// Returns TRUE if POP response indicates success, FALSE if failure
protected boolean responseIsOk() throws Exception
{
String line = br.readLine();
System.out.println("< "+line);
return line.toUpperCase().startsWith("+OK");
}
// Reads a line from the POP server, and displays it to screen
protected String readLine(boolean debug) throws Exception
{
String line = br.readLine();
// Append a < character to indicate this is a server protocol response
if (debug)
System.out.println("< "+line);
else
System.out.println(line);
return line;
}
// Writes a line to the POP server, and displays it to the screen
protected void writeMsg(String msg) throws Exception
{
pw.println(msg);
pw.flush();
System.out.println("> "+msg);
}
// Close all writers, streams and sockets
protected void closeConnection() throws Exception
```

```
{
pw.flush();
pw.close();
br.close();
socket.close();
}
// Send the QUIT command, and close connection
protected void sendQuit() throws Exception
{
System.out.println("Sending QUIT");
writeMsg("QUIT");
readLine(true);
System.out.println("Closing Connection");
closeConnection();
}
// Display emails in a message
protected void displayEmails() throws Exception
{
BufferedReader userinput = new BufferedReader( new InputStreamReader
(System.in) );
System.out.println("Displaying mailbox with protocol commands and responses below");
System.out.println("------------------------------------------------------------");
// Open a connection to POP3 server
System.out.println("Opening Socket");
socket = new Socket(this.hostname, this.port);
br = new BufferedReader(new InputStreamReader(socket.getInputStream()));
pw = new PrintWriter(new OutputStreamWriter(socket.getOutputStream()));
// If response from server is not okay
if(! responseIsOk())
{
socket.close();
throw new Exception("Invalid POP3 Server");
}
// Login by sending USER and PASS commands
System.out.println("Sending username");
writeMsg("USER "+this.username);
if(!responseIsOk())
{
sendQuit();
throw new Exception("Invalid username");
}
System.out.println("Sending password");
writeMsg("PASS "+this.password);
if(!responseIsOk())
{
sendQuit();
throw new Exception("Invalid password");
}
```

```
// Get mail count from server ....
System.out.println("Checking mail");
writeMsg("STAT");
// ... and parse for number of messages
String line = readLine(true);
StringTokenizer tokens = new StringTokenizer(line," ");
tokens.nextToken();
int messages = Integer.parseInt(tokens.nextToken());
int maxsize = Integer.parseInt(tokens.nextToken());
if (messages == 0)
{
System.out.println ("There are no messages.");
sendQuit();
return;
}
System.out.println ("There are " + messages + " messages.");
System.out.println("Press enter to continue.");
userinput.readLine();
for(int i = 1; i <= messages ; i++)
{
System.out.println("Retrieving message number "+i);
writeMsg("RETR "+i);
System.out.println("--------------------");
line = readLine(false);
while(line != null && !line.equals("."))
{
line = readLine(false);
}
System.out.println("--------------------");
System.out.println("Press enter to continue. To stop, type Q then enter");
String response = userinput.readLine();
if (response.toUpperCase().startsWith("Q"))
break;
}
sendQuit();
}
public static void main(String[] args) throws Exception
{
Pop3ClientDemo client = new Pop3ClientDemo();
}
// Read user input
protected void getInput() throws Exception
{
String data=null;
BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
System.out.print("Please enter POP3 server hostname:");
data = br.readLine();
if(data == null || data.equals("")) hostname="localhost";
```

```
else
hostname=data;
System.out.print("Please enter mailbox username:");
data = br.readLine();
if(!(data == null || data.equals("")))
username=data;
System.out.print("Please enter mailbox password:");
data = br.readLine();
if(!(data == null || data.equals("")))
password=data;
}
}
```



```
D:\jdk1.3>cd bin

D:\jdk1.3\bin>javac Pop3ClientDemo.java

D:\jdk1.3\bin>java Pop3ClientDemo
Please enter POP3 server hostname:localhost
Please enter mailbox username:mani
Please enter mailbox password:mani
Displaying mailbox with protocol commands and responses below
--------------------------------------------------------------
Opening Socket
Error occured - details follow
java.net.SocketException: no further information (code=10013)
        at java.net.PlainSocketImpl.socketConnect(Native Method)
        at java.net.PlainSocketImpl.doConnect(PlainSocketImpl.java:312)
        at java.net.PlainSocketImpl.connectToAddress(PlainSocketImpl.java:125)
        at java.net.PlainSocketImpl.connect(PlainSocketImpl.java:112)
        at java.net.Socket.<init>(Socket.java:273)
        at java.net.Socket.<init>(Socket.java:100)
        at Pop3ClientDemo.displayEmails(Pop3ClientDemo.java:86)
        at Pop3ClientDemo.<init>(Pop3ClientDemo.java:26)
        at Pop3ClientDemo.main(Pop3ClientDemo.java:148)
no further information (code=10013)

D:\jdk1.3\bin>_
```

EX NO: 5 CHAT APPLICATION USING JAVA

Aim

To write a java program to create a simple chat application with datagram sockets and packets

Algorithm

Server Side

Step 1: Import net and io packages and class
step 2: Create a datagram socket and datagram packet
step 3: While client send datagram packet to server listen to client port
step 4: Get the datagram packet into a string

step 5: Display the string

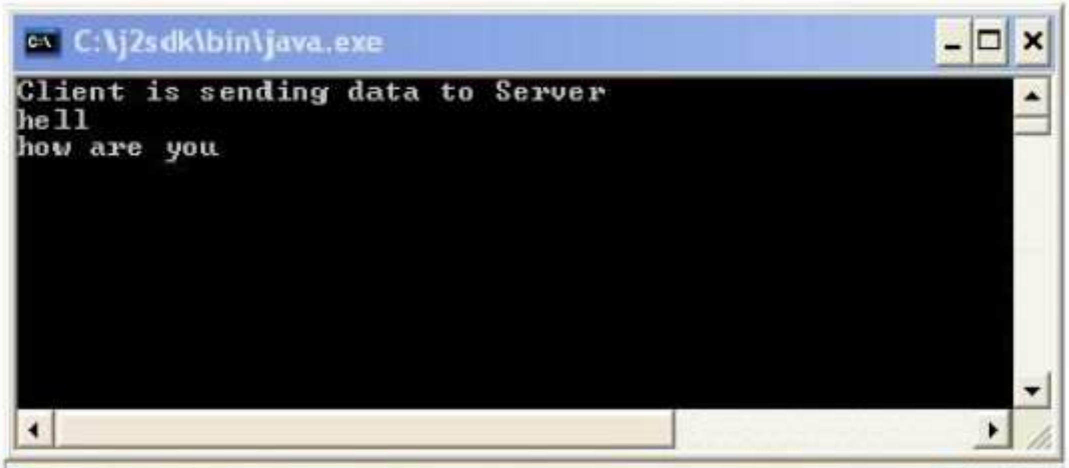
Client Side

step 1: Import net and io packages
step 2: Create a datagram socket and datagram packet
step 3: Get input from the user and convert the string into a datagram packet
step 4: send the datagram packet to the server through serve port

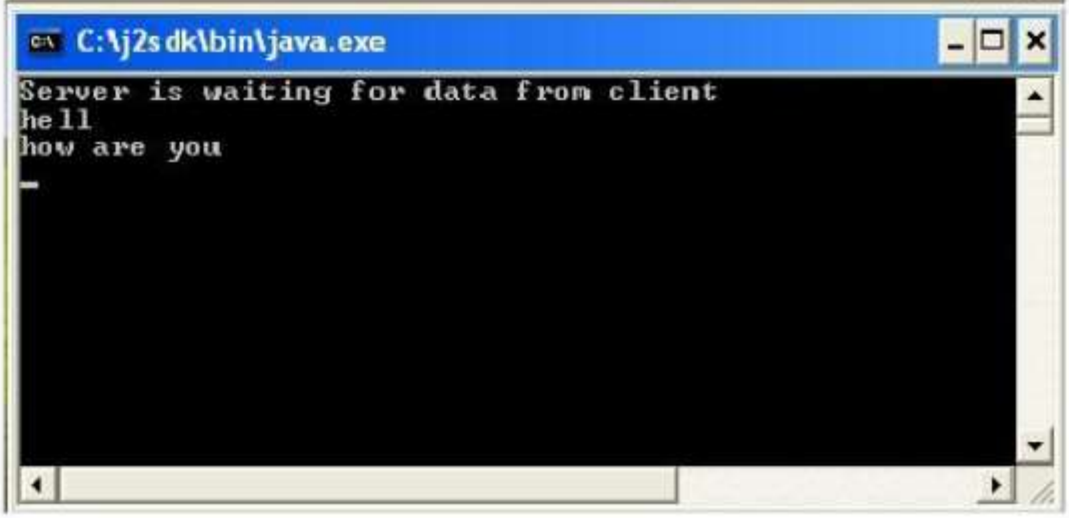
```
import java.io.*;
import java.net.*;
class Client
public static DatagramSocket clientsocket;
public static BufferedReader dis;
public static InetAddress ia;
public static byte buf[]=new byte[1024];
public static int cport=789,sport=790;
public static void main(String a[])throws IOException
{
clientsocket = new DatagramSocket(cport);
dis=new BufferedReader(new InputStreamReader(System.in));
ia=InetAddress.getLocalHost();
System.out.println("Client is sending data to Server ");
while(true)
{
String str=dis.readLine();
buf=str.getBytes();
clientsocket.send(new DatagramPacket(buf,str.length(),ia,sport));
}
}
}
```

```
import java.io.*;
import java.net.*;
class chats
{
```

```
public static DatagramSocket serversocket;
public static DatagramPacket dp;
public static BufferedReader dis;
public static InetAddress ia;
public static byte buf[]=new byte[1024];
public static int cport=789,sport=790;
public static void main(String a[])throws IOException
{
serversocket=new DatagramSocket(sport);
dp=new DatagramPacket(buf,buf.length);
dis=new BufferedReader(new InputStreamReader(System.in));
ia=InetAddress.getLocalHost();
System.out.println("Server is waiting for data from client");
while(true)
{
serversocket.receive(dp);
String s=new String(dp.getData(),0,dp.getLength());
System.out.println(s);
}
}
}
```



```
Client is sending data to Server
hell
how are you
```



```
Server is waiting for data from client
hell
how are you
_
```

EX.NO.06A INVOKING SERVLET FROM HTML FORM

AIM:

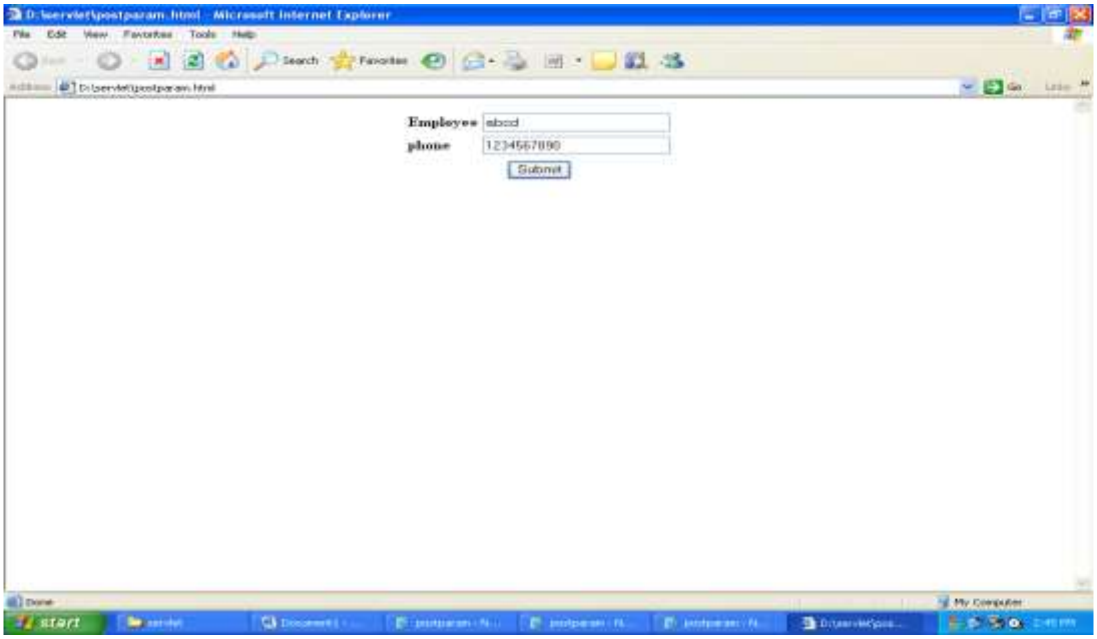
To write a program to invoke Servlet from HTML form.

ALGORITHM

STEP1: Create a HTML page including some form elements.
STEP2: Inside the <form> tag, for the action attribute specify the full path name of the File.
STEP3: Create a .java file that imports javax.http.servlet.*;
STEP4: Set classpath where servlet-api.jar file resides.
STEP5: Compile the servlet program using javac programname.java
STEP6: Place the class file …\Tomcat 5.5\webapps\Examples\WEB-INF\classes\folder.
STEP7: Define the doPost function to process the data obtained from the HTML file.
STEP8: Modify the web.xml file using your servletClassName.
STEP9: Invoke the class file using http://localhost:8080/servetClassName from your browser

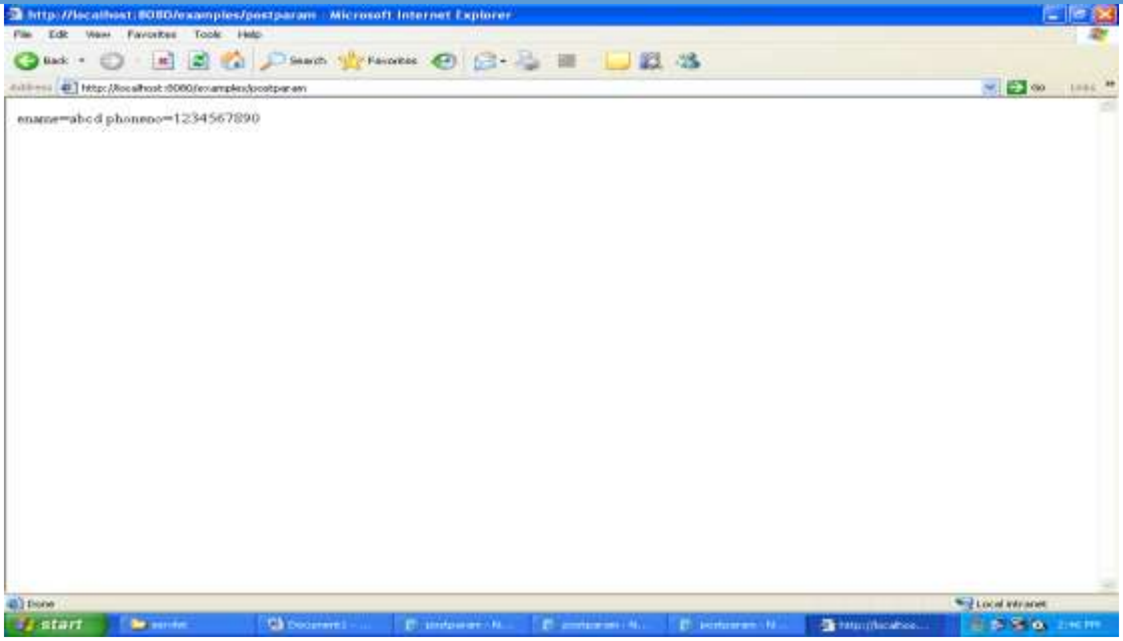
```
//sertun.java
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
public class sertun extends HttpServlet
{
	public void doGet(HttpServletRequest req,HttpServletResponse res) throws
ServletException,IOException
	{
	PrintWriter pw=res.getWriter();
	pw.println("This is the response from the servlet");
	}
}
```

```
//tun.java
import java.awt.*;
import java.net.*;
import java.applet.*;
import java.awt.event.*;
import java.io.*;
//<applet code=tun width=400 height=400> </applet>

public class tun extends Applet implements ActionListener
{
	TextField t1;
	Button b1;
	public void init()
	{
        add(t1=new TextField(20));
        add(b1=new Button("get Message from servlet"));
        b1.addActionListener(this);
        }
        public void actionPerformed(ActionEvent ae)
        {
                try{
                URL u=new URL("http://localhost:8080/servlet/sertun");
                URLConnection ur=u.openConnection();
                DataInputStream din=new DataInputStream(ur.getInputStream());
                t1.setText(din.readLine());
                }catch(Exception e) { }

        }
}
```


D:\servlet\postparam.html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Search Favorites
Address D:\servlet\postparam.html
Go
Submit
Done
My Computer
start

Employee abcd

phone 1234567890


http://localhost:8080/examples/postparam - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address
http://localhost:8080/examples/postparam
Go
Links
Done
Local intranet
start

```
ename=abcd phoneno=1234567890
```

EX.NO.06B INVOKING SERVLET FROM APPLET

Aim

To create an applet servlet communication using tunneling.

Algorithm

1. **Import the servlet package using http servlet class**
2. **Print the response message using the get writer method**
3. **Design the applet window using code base, width and height**
4. **Establish the connection between applet and servlet**
5. **Design the button and text box in the applet window, that gets the message a from servlet.**
6. **Stop the execution.**

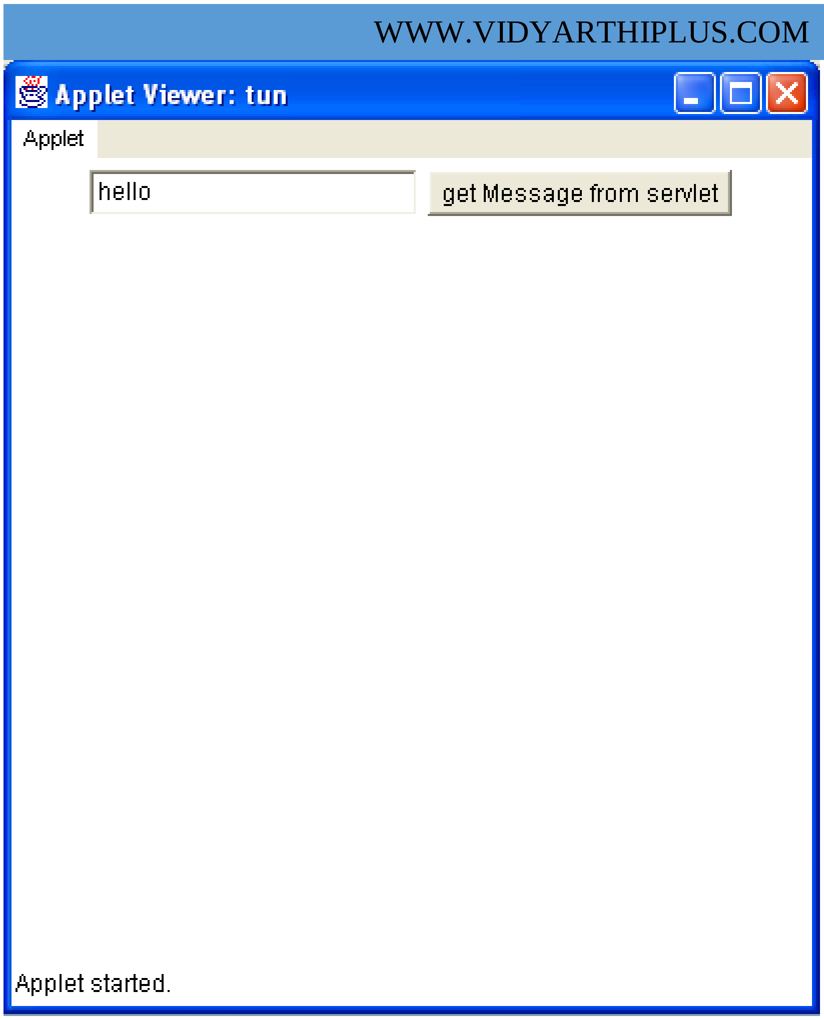
postparam.servlet

```
import java.io.*;
import java.net.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class postparam extends HttpServlet {
public void init(ServletConfig config) throws ServletException {
 super.init(config);
  }
 public void destroy() {}
   protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
   throws ServletException, IOException {
       response.setContentType("text/html");
       PrintWriter pw = response.getWriter();
       Enumeration e=request.getParameterNames();
       while(e.hasMoreElements())
      {
          String pname=(String)e.nextElement();
          pw.print(pname+"=");
          String pvalue=request.getParameter(pname);
          pw.println(pvalue);
       }
        pw.close();
   }
protected void doGet(HttpServletRequest request, HttpServletResponse response)
  throws ServletException, IOException {
   processRequest(request, response);
   }
protected void doPost(HttpServletRequest request, HttpServletResponse response)
   throws ServletException, IOException {
 processRequest(request, response);
   }
 public String getServletInfo() {
  return "Short description";
    }
}
```

postparam.html

```
<html>
  <body>
  <center>
  <form name="Postparam"method=POST
action="http://localhost:8084/WebApplication1/postparam">
  <table>
  <tr>
  <td><B>Employee</B></td>
  <td><input type="textbox"name="ename"size="25"value=" "</td>
  </tr>
  <tr>
  <td><B>phone</B></td>
  <td><input type="textbox"name="phoneno"size="25"value=" "></td>
  </tr>
  </table>
  <input type="submit"value="submit">
  </body>
</html>
```

```
C:\jdk1.3\bin\javac tun.java
C:\jdk1.3\bin\javac sertun.java
C:\jdk1.3\bin\appletviewer tun.java
```



hello

Applet started.

EX.NO.7 ONLINE EXAMINATION USING SERVLET AND DATABASE

AIM:

To write an online Examination using Servlet three tier architecture

ALGORITHM:

STEP 1: Create a user interface form for getting Seat number and Name
STEP2: Create jsp file which calculates the total marks on the server
STEP3: Place the jsp file …\Tomcat 5.5\webapps\Examples\WEB-INF\jsp\ folder.
STEP4: Create a data base to store the existing user details. If a new login has to be created, then display the corresponding page.
STEP5: Create a table in MS ACCESS or ORACLE that contains three fields .
STEP6: Create a DSN to map the Microsoft Access Driver using ODBC in Administrative Tools.
STEP7: Import java.sql.* in the java program to execute the SQL queries.
STEP8: Load the database driver by using Class.forName().
STEP9: Create a Connection and Statement Object.
STEP10: Establish a Database Connection with DSN using DriverManager.getConnection( ).
STEP11: Execute the query using executeQuery() and that gets stored in a Resultset.
STEP12: Until there are records in table specified get the fields in each record one by one and display onto the screen.
STEP13: Close the Statement and Connection Object.
STEP 14: DIsplay the results

```
<!Save as onlineexam.html>
<html>
                <head><title>Online Examination</title>
                    <script language="javascript">
                    function validation(Form_obj)
                    {
                     if(Form_obj.Seat_no.value.length==0)
                     {
                        alert("Please,fill up the Seat Number");
                        Form_obj.Seat_no.focus();
                        return false;
                     }
                               if(Form_obj.Name.value.length==0)
                     {
                        alert("Please,fill up the Name");
                        Form_obj.Name.focus();
                        return false;
                     }
                               return true;
                   }
               </script>
              </head>
              <body bgcolor=lightgreen>
              <center>
```

```
    <h1>OnLine Examination</h1>
</center>
      <form action="StudentServlet3" method="post"
       name="entry" onSubmit="return validation(this)">
      <input type="hidden" value="list" name="action">
       <table>
        <tr>
            <td><h3>Seat Number:</h3></td>
            <td><input type="text" name="Seat_no"></td>
        </tr>
        <tr>
           <td><h3>Name:</h3></td>
           <td><input type="text" name="Name" size="50"></td>
        </tr>
        <hr/>
        <tr>
          <td><b>Total Marks:10(Each question carries equal marks) </b></td>
                <td></td><td></td><td></td><td><b>Time: 15 Min.</b></td>
        </tr>
        </table>
        <hr/>
          <b>1. Apache is an open source web server</b><br/>
          <input type="radio" name="group1" value="True">True
          <input type="radio" name="group1" value="False">False<br>
          <br/>

          <b>2. In Modern PC there is no cache memory.</b><br/>
          <input type="radio" name="group2" value="True">True
          <input type="radio" name="group2" value="False">False<br>
          <br/>

          <b>3. Tim-Berner Lee is the originator of Java.</b><br/>
          <input type="radio" name="group3" value="True">True
          <input type="radio" name="group3" value="False">False<br>
          <br/>

          <b>4.JPG is not a video file extension.</b><br/>
          <input type="radio" name="group4" value="True">True
          <input type="radio" name="group4" value="False">False<br>
          <br/>

          <b>5. HTTP is a statefull protocol</b><br/>
          <input type="radio" name="group5" value="True">True
          <input type="radio" name="group5" value="False">False<br>
          <hr/>

          <center>
            <input type = "submit" value="Submit">
```

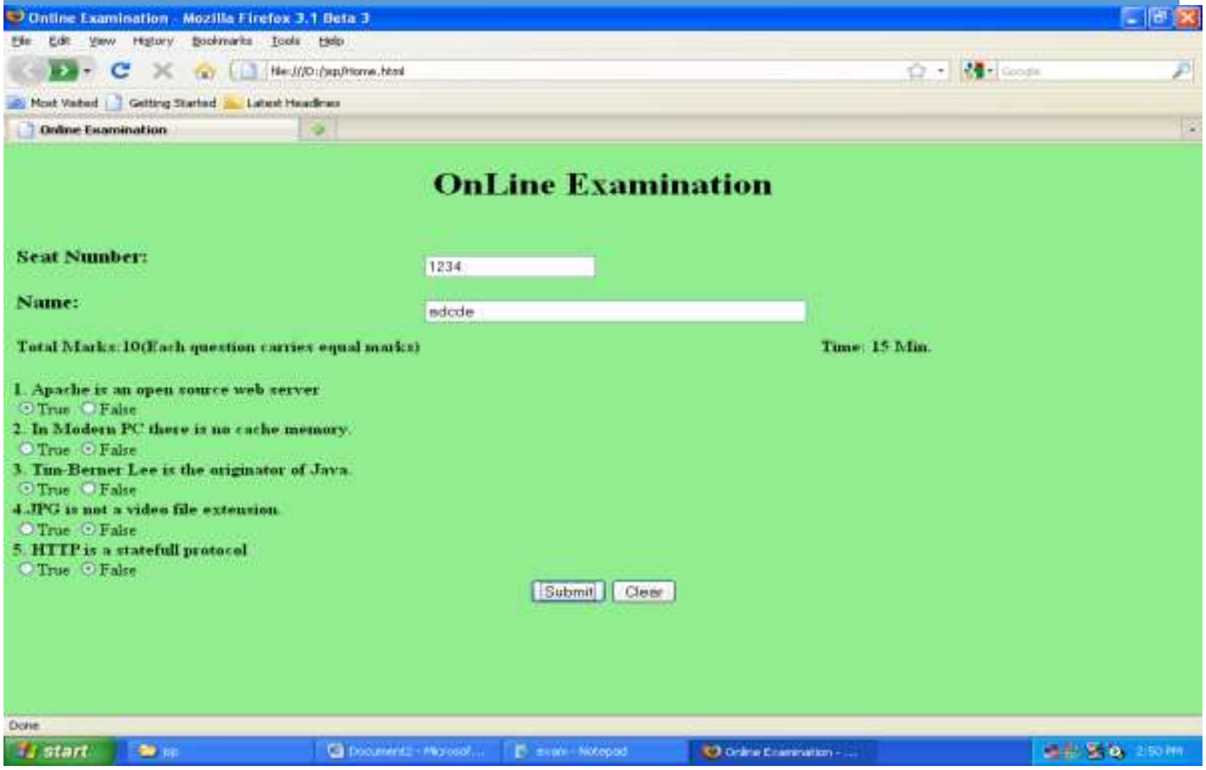
```
                    <input type = "reset" value="Clear"><br><br>
                  </center>
                </form></body></html>
```

StudentServlet.java

```
import  java.sql.*;
import  java.io.*;
 import  java.util.*;
import  javax.servlet.*;
import  javax.servlet.http.*;

public class StudentServlet3 extends HttpServlet
{
        String SeatNum,Name;
        String ans1,ans2,ans3,ans4,ans5;
int a1=0,a2=0,a3=0,a4=0,a5=0;

        Connection connect=null;
        Statement stmt=null;
        ResultSet rs=null;
public void doPost (HttpServletRequest request,HttpServletResponse response) throws
ServletException, IOException
{
//Establishing Connection to the Database
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
      String url = "jdbc:odbc:StudentDB1";
        connect = DriverManager.getConnection(url, " ", " ");
}
catch(ClassNotFoundException e){e.printStackTrace();}
catch(SQLException e){e.printStackTrace();}
catch(Exception e){e.printStackTrace();}

//Gathering parameters from HTML form

         SeatNum = request.getParameter("Seat_no");
         Name = request.getParameter("Name");
         ans1 =request.getParameter("group1");
         if(ans1.equals("True"))
           a1=2;
         else
           a1=0;
          ans2 = request.getParameter("group2");
         if(ans2.equals("True"))
```

```
              a2=0;
            else
              a2=2;
            ans3 = request.getParameter("group3");
            if(ans3.equals("True"))
              a3=0;
            else
              a3=2;
            ans4 = request.getParameter("group4");
            if(ans4.equals("True"))
              a4=2;
            else
              a4=0;
            ans5 = request.getParameter("group5");
            if(ans5.equals("True"))
              a5=0;
            else
              a5=2;
            int Total=a1+a2+a3+a4+a5;

//inserting values in the database
try
{
             stmt = connect.createStatement();
            String query = "INSERT INTO StudentTable (" + "Seat_no,Name,Marks" +  ")
VALUES ('" +SeatNum + "', '" + Name + "', '"+Total+ "')";
            int result = stmt.executeUpdate(query);
            stmt.close();
}catch(SQLException e)    {}

//retrieving the records from database

response.setContentType("text/html");
PrintWriter out=response.getWriter();
out.println("<html>");
out.println("<head>");
out.println("</head>");
out.println("<body bgcolor=pink>");
out.println("<center>");
out.println("<br><br>");
out.println("<h2>Students Database</h2>");
out.println("<table border=5>");
try
{
            stmt = connect.createStatement();
           String query = "SELECT * FROM StudentTable WHERE Name="+"'"+Name+"'";
            rs = stmt.executeQuery(query);
          out.println("<th>"+"Seat_no"+"</th>");
```

```
            out.println("<th>"+"Name"+"</th>");
out.println("<th>"+"Marks"+"</th>");

 while(rs.next())
                {

                          out.println(" <tr>");
                          out.println("  <td>"+rs.getInt(1)+"</td>");
                          out.println(" <td> "+rs.getString(2)+" </td>");
                           out.println(" <td> "+rs.getString(3)+" </td>");
                          out.println(" </tr>");
                   }
out.println("</table>");
}catch(SQLException e){}

finally
{

  try
{
            if(rs!=null)
rs.close();
if(stmt!=null)
                          stmt.close();
if(connect!=null)
                          connect.close();
}catch(SQLException e){}
out.println("<h1>Thanks</h1>\n");
out.println("</table>");
out.println("</center>");
out.println("</body></html>");
}
}}
```


Online Examination - Mozilla Firefox 3.1 Beta 3
File Edit View History Bookmarks Tools Help
file:///D:/jsp/Home.html
Google
Most Visited Getting Started Latest Headlines
Online Examination
Done
start
jsp
Document2 - Microsof...
exam - Notepad
Online Examination - ...
2:50 PM

OnLine Examination

Seat Number: 1234

Name: adcde

Total Marks:10(Each question carries equal marks) **Time: 15 Min.**

1. Apache is an open source web server
● True ○ False

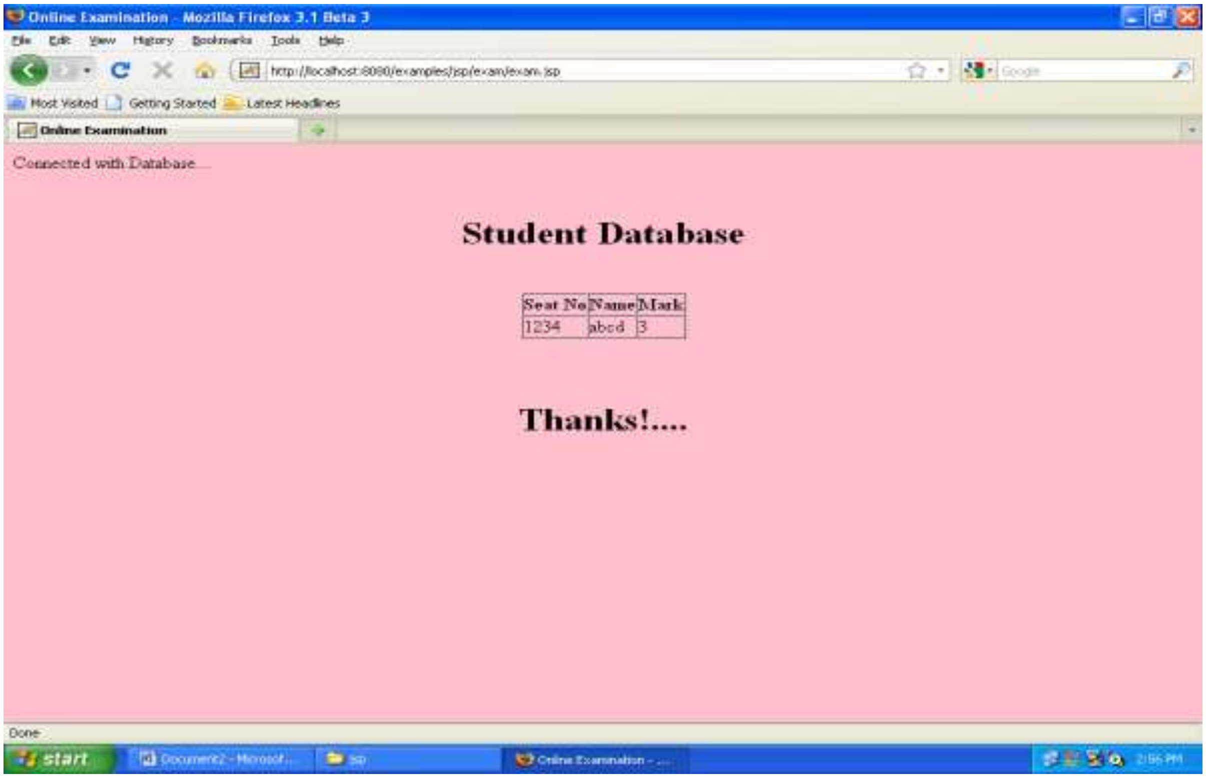
2. In Modern PC there is no cache memory.
○ True ● False

3. Tim-Berner Lee is the originator of Java.
● True ○ False

4.JPG is not a video file extension.
○ True ● False

5. HTTP is a statefull protocol
○ True ● False

Submit Clear


Online Examination - Mozilla Firefox 3.1 Beta 3
File Edit View History Bookmarks Tools Help
http://localhost:8080/examples/jsp/exam/exam.jsp
Google
Most Visited Getting Started Latest Headlines
Online Examination
Done
start
Document2 - Microsof...
jsp
Online Examination - ...
2:56 PM

Connected with Database....

Student Database

Seat No	Name	Mark
1234	abcd	3

Thanks!....

EX.NO.08 LOCK SERVLET

Aim
To create a program to lock servlet itself to a particular port number and ip address using init parameter key.

Algorithm:
Step1: Create an HTML form which contains user interface to allow the user to enter the key.

Step2: Create a servlet program in which the host IP and port number is obtained. It is then converted to numeric form. This numeric value can be set as a key value at which the server IP and port number is locked. On providing this key the authorized access message will be displayed.

Step3:Compile the above code to generate the class file using following command

Step4: Copy the generated class file to the classed folded present inside the C: >tomcat_directory\webapps\exsamples\WEB-INIF folder.Move back to WEB-INF folder in which the web.xml is present. Edit this file by providing the servlet name and init parameter. Following boldfaced lines can be added in web.xml file

Step5: Open the web browser and invoked the HTML document created in Step 1. Type the key and click submit button.

keyform.html

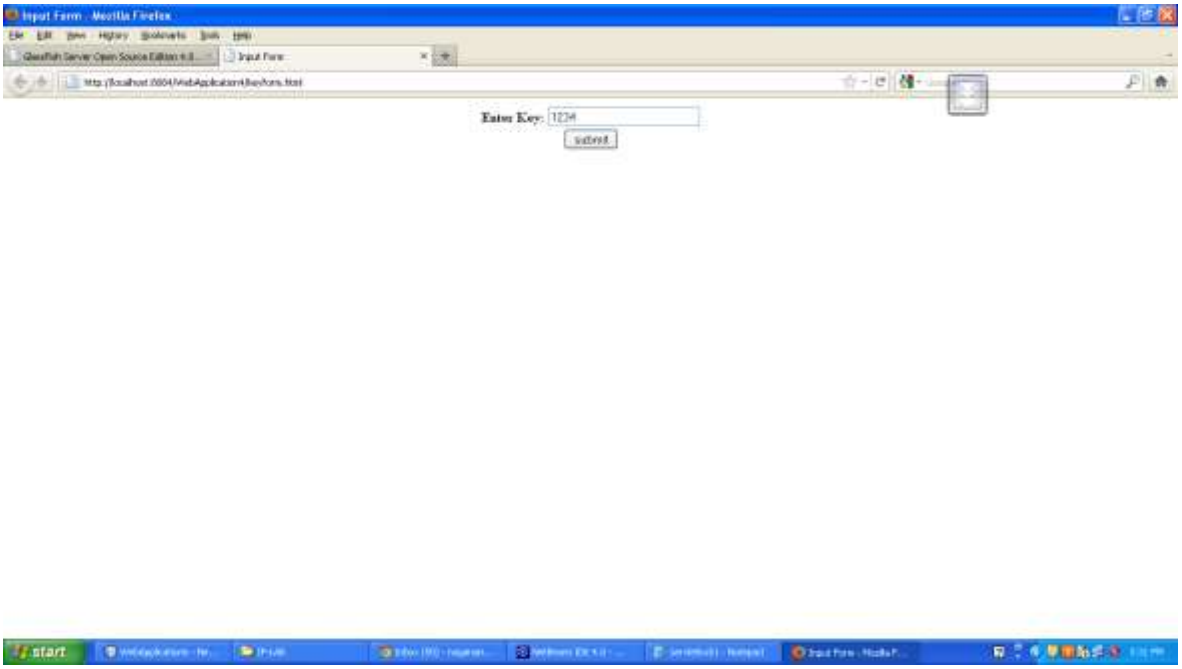
```
<html>
  <head>
    <title>Input Form</title>
      </head>
  <body>
      <center>
          <form name="form1" method="get"
action="http://localhost:8084/Webapplication3/KeyIPLock">
              <table><tr>
              <td><b>Enter Key:</b></td>
              <td><input type="text" name="user_key" size="25" value="">
                  </td>
```

```
            </tr>
            </table>
            <input type="submit" value="submit">
       </form>
    </center>

  </body>
</html>
```

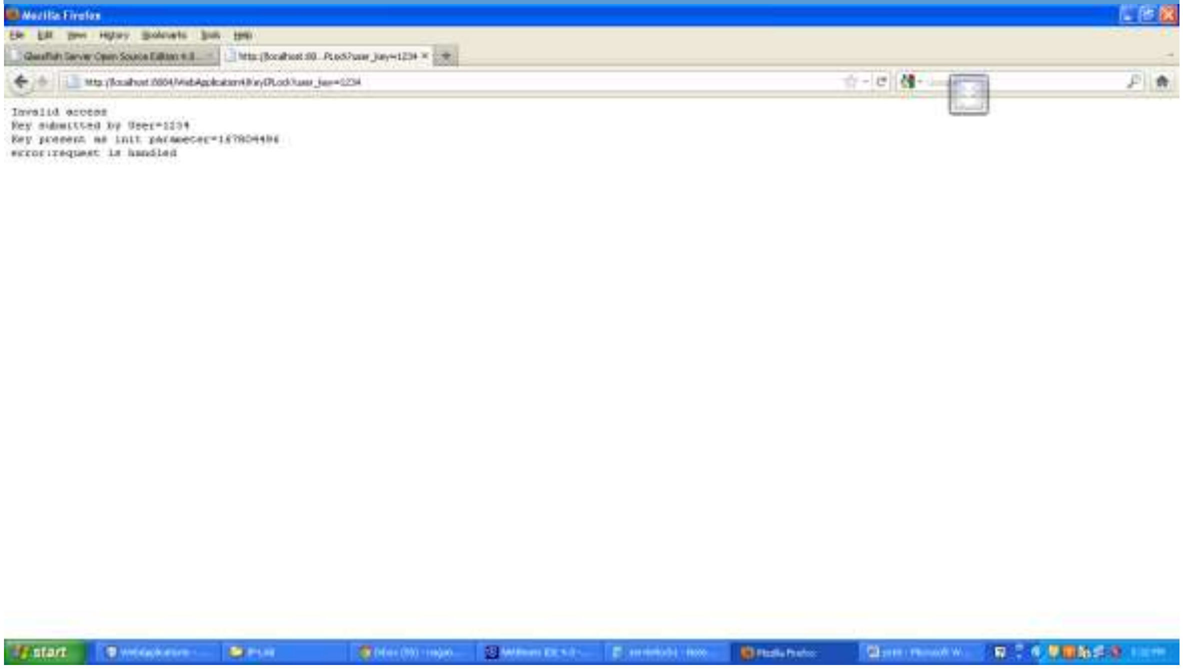
KeyIPLock.java

```
import java.io.*;
import java.net.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class KeyIPLock extends GenericServlet {
  public void service(ServletRequest req,ServletResponse res)throws
ServletException,IOException

  {
     res.setContentType("text/plain");
     PrintWriter out=res.getWriter();
     String userKey=req.getParameter("user_key");
     String key=getInitParameter("key");
     String host=req.getServerName();
     String HostIP="";
     long result=0;
     int port=req.getServerPort();
     if(key==null)

     {
        out.println("Invalid access");
            }
  try
  {
     InetAddress IP=InetAddress.getLocalHost();
     HostIP=IP.getHostAddress();
     String[] ipAddressInArray=HostIP.split("\\.");
     for(int i=3;i>=0;i--)
     {
        long ip=Long.parseLong(ipAddressInArray[3-i]);
        //left shifting 24,16,8,0 and bitwise OR
        //1.192<<24
        //1.168<<16
        //1.1<<8
```

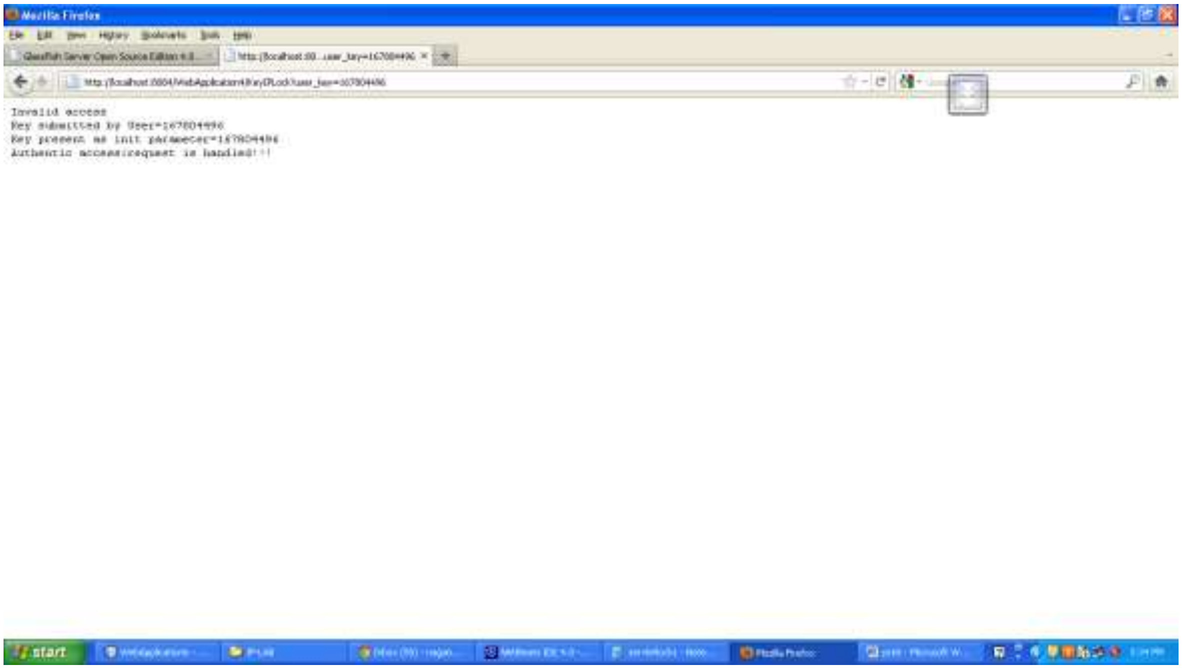
```
            //1.2<<0
            result|=ip<<(i*8);
            //result contains numeric value of IP address
            //now add port number to this numeric value

            result=result+port;
        }
    }
    catch(UnknownHostException e)
    {
        out.println(e);
    }
        long k=Long.parseLong(userKey);
        out.println("Key submitted by User="+k);
         out.println("Key present as init parameter="+result);
         if(k==result)
         {
             out.println("Authentic access:request is handled!!!");
         }
         else
         {
             out.println("error:request is handled");
             }
          }}
```


Input Form - Mozilla Firefox
File Edit View History Bookmarks Tools Help
GlassFish Server Open Source Edition 4.0
Input Form
http://localhost:8084/WebApplication4/keyform.html
submit
start

Enter Key: 1234


Mozilla Firefox
File Edit View History Bookmarks Tools Help
GlassFish Server Open Source Edition 4.0
http://localhost:80...PLock?user_key=1234
http://localhost:8084/WebApplication4/KeyIPLock?user_key=1234
start

```
Invalid access
Key submitted by User=1234
Key present as init parameter=167804496
error:request is handled
```


Mozilla Firefox
File Edit View History Bookmarks Tools Help
GlassFish Server Open Source Edition 4.0
http://localhost:80...user_key=167804496
http://localhost:8084/WebApplication4/KeyIPLock?user_key=167804496
start

```
Invalid access
Key submitted by User=167804496
Key present as init parameter=167804496
Authentic access:request is handled!!!
```

EX.NO.09 SESSION TRACKING USING HIDDEN FORM FIELDS

Aim
To create a servlet program for Session tracking using hidden form fields

Algorithm:

Step1: Create a graphical user interface for allowing the user to enter the data.

Step2: Create the first servlet which will receive these valued and will store them in hidden form files.Then these valued are transferred to the second servlet

Step3: The second servlet will display these values with greeting message

Step4: Now compile both the java sevlets by using the following command

```
Javac Servlet1.java
Javac Servlet2.java
```

Step5:Then edit web.xml file

Step6: Open the web browser and invoke the HTML document created in Step 1.

GUI.html

```
<html>
  <head>
    <title>Session tracking using hidden form field</title>
      </head>
  <body>
      <form method="get" action="http://localhost:8084/WebApplication2/Servlet1">
          Enter name:<input type="text" name="userName"/><br/>
          Enter City:<input type="text" name="userCity"/><br/>
          <input type="submit" value="Submit">
      </form>
  </body>
</html>
```

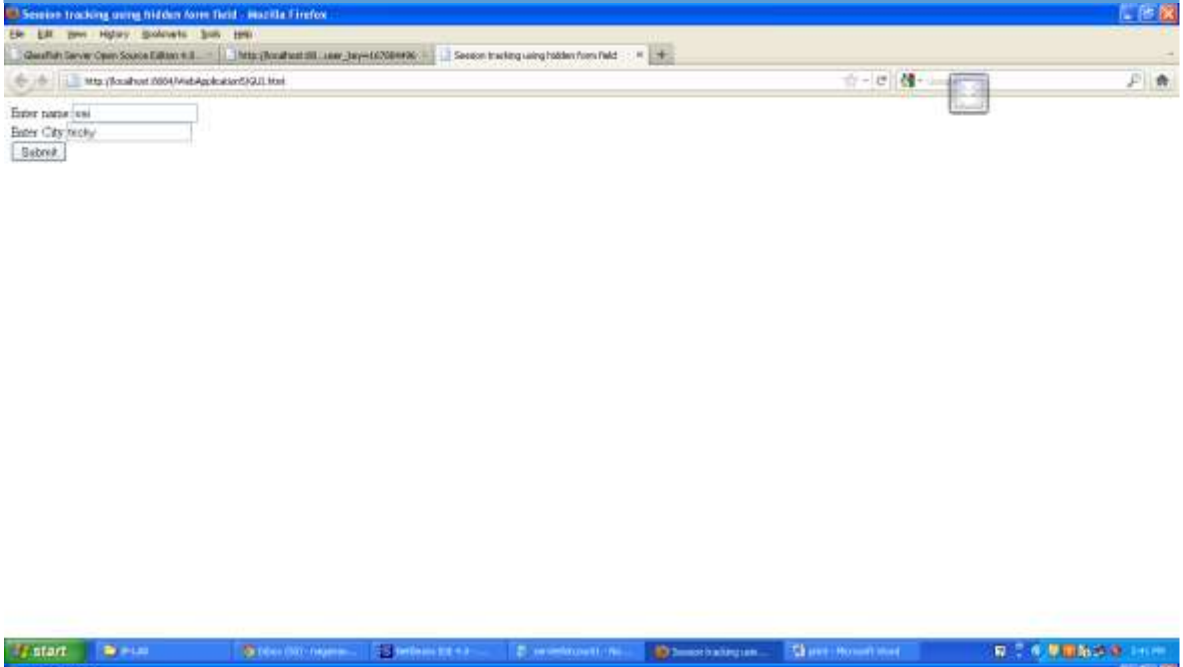
Servlet1.java

```
import java.io.*;
import java.net.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Servlet1 extends HttpServlet {
        public void doGet(HttpServletRequest request,HttpServletResponse response)
        {
            try
            {
```

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

        String n=request.getParameter("userName");
        String c=request.getParameter("userCity");

        out.print("<form action='http://localhost/WebApplication/Servlet2'>");
        out.print("<input type='hidden'name='userName' value='"+n+"'>");
        out.print("<input type='hidden'name='userCity' value='"+c+"'>");
        out.print("<input type='submit' value='just click here'>");
        out.print("</form>");
        out.close();
      }
      catch(Exception e)
      {System.out.println(e);
      }
    }
  }
```

Servlet2.java

```
import java.io.*;
import java.net.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Servlet2 extends HttpServlet {
  public void doGet(HttpServletRequest request,HttpServletResponse response)
  {
        try
    {
      response.setContentType("text/html)");
      PrintWriter out=response.getWriter();
      String n=request.getParameter("userName");
      out.println("Hello"+n);
      String c=request.getParameter("userCity");
      out.println("you are from"+c);
      out.close();
    }
    catch(Exception e)
    {
      System.out.println(e);
    }
  }
}
```

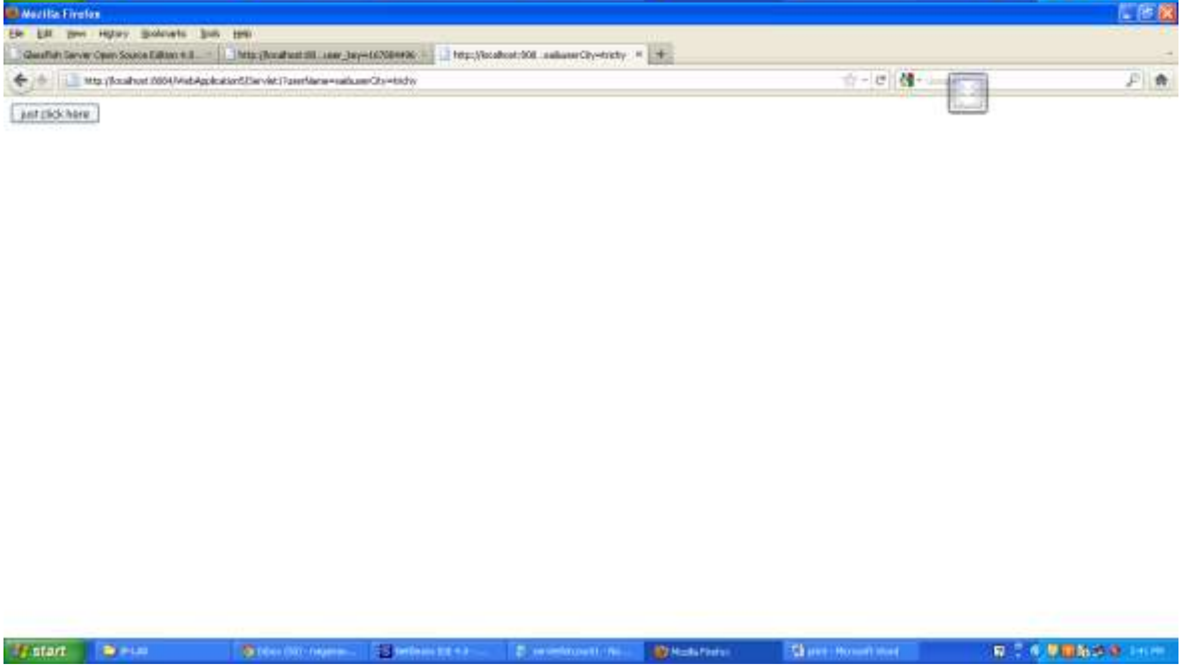

Session tracking using hidden form field - Mozilla Firefox
File Edit View History Bookmarks Tools Help
GlassFish Server Open Source Edition 4.0
Session tracking using hidden form field
start

http://localhost:8084/WebApplication5/GUI.html

Enter name: sai

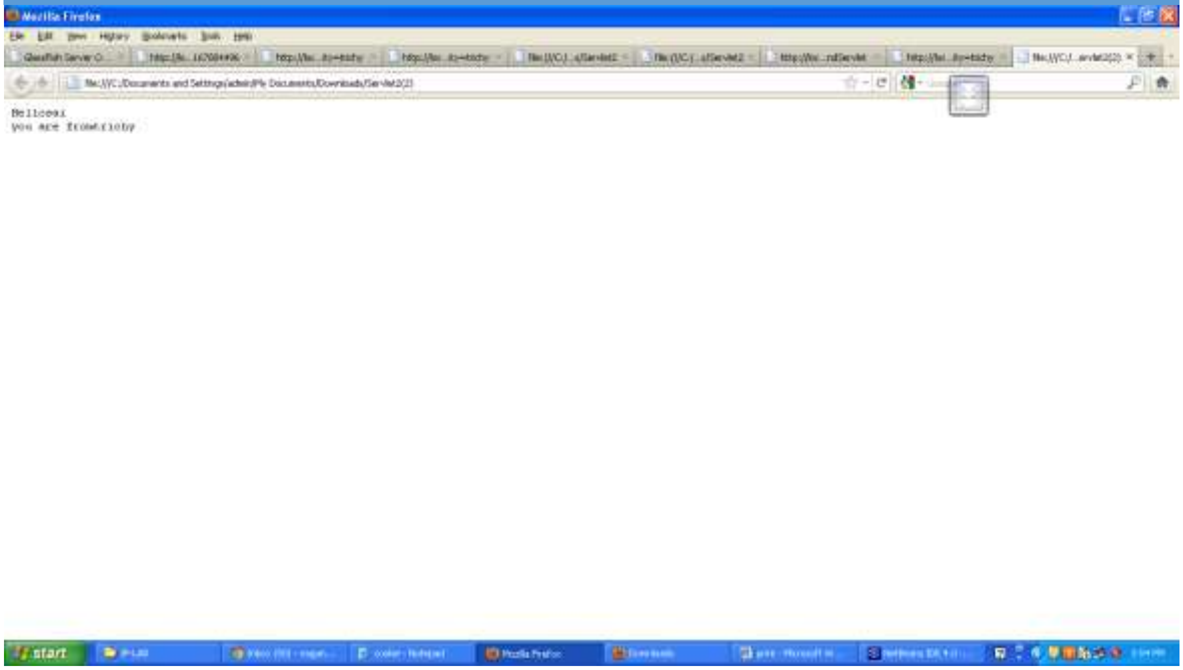
Enter City: trichy

Submit


Mozilla Firefox
File Edit View History Bookmarks Tools Help
GlassFish Server Open Source Edition 4.0
start

http://localhost:8084/WebApplication5/Servlet1?userName=sai&userCity=trichy

just click here


Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///C:/Documents and Settings/admin/My Documents/Downloads/Servlet2(2)
start

```
Hellosai
you are fromtrichy
```

EX.NO.10 COOKIE

Aim

To develop a web application using Servlet and to set the cookie in which the current user name is stored.

Algorithm

Step1:Create a HTML form for entering the user name
Step2: From the above HTML form, the servlet program is invoked in which the validity of the user name checked. If it is a valid user then the welcome message will be displayed otherwise the “invalid user” message will be displayed. In this servlet set the cookies in which the current user name is stored.

Step3: Compile the above servlet LoginServlet.java and copy its class file in tomcats folder at C:\tomcatdirectory\webapps\examples\WEB-INF\classes.Then edit the web.xml in WEB-INF folder.Store the user information such as user name in the web.xml using init-param.

Step4: On successful login, the information from the cookie is checked and page for corresponding user can be displayed.

Step5: Compile the above servlet LoginSuccess.java and copy its class file in tomcat’s folder at C:\tomcatdirectory\webapps\examples\WEB-INF\classes

Step6: Start tomcat web server. Open the web browser and display the login form created in step1.

Index.html

```
<html><body>
<form action="http://localhost:8084/WebApplication6/FirstServlet" >
Name:<input type="text" name="userName"/><br/>
<input type="submit" value="go"/>
</form>
   </body>
   </html>
```

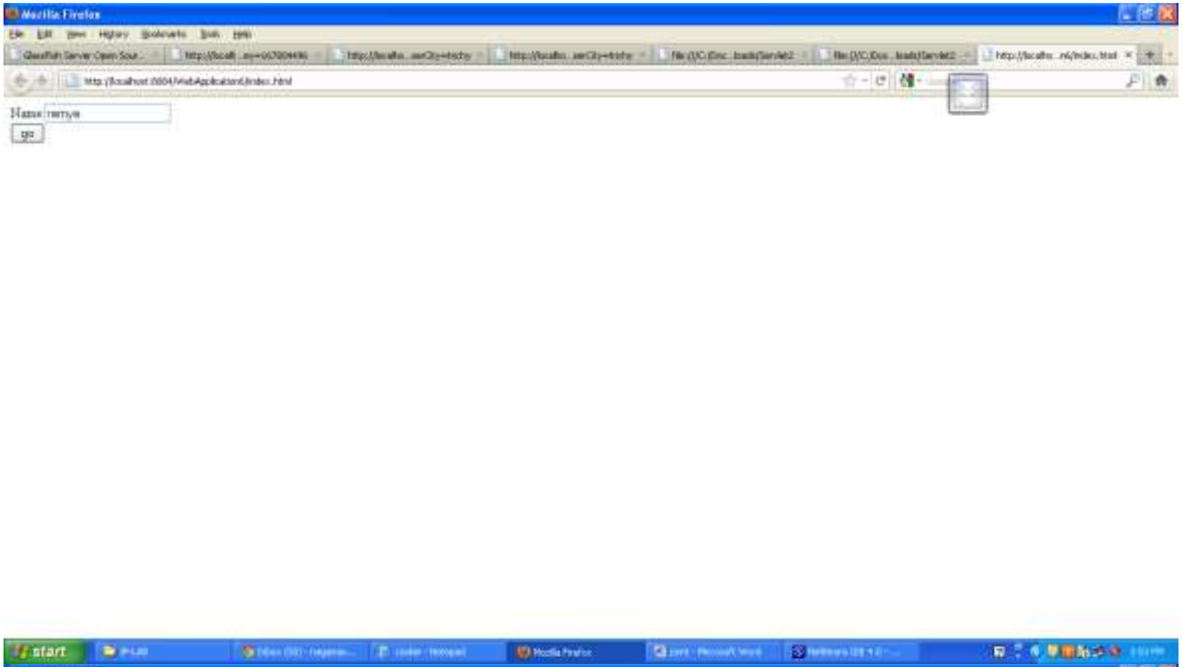
FirstServlet.java

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


   public class FirstServlet extends HttpServlet {
```

```
 public void doGet(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 action='http://localhost:8084/WebApplication6/SecondServlet'>");
  out.print("<input type='submit' value='go'>");
  out.print("</form>");

  out.close();

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

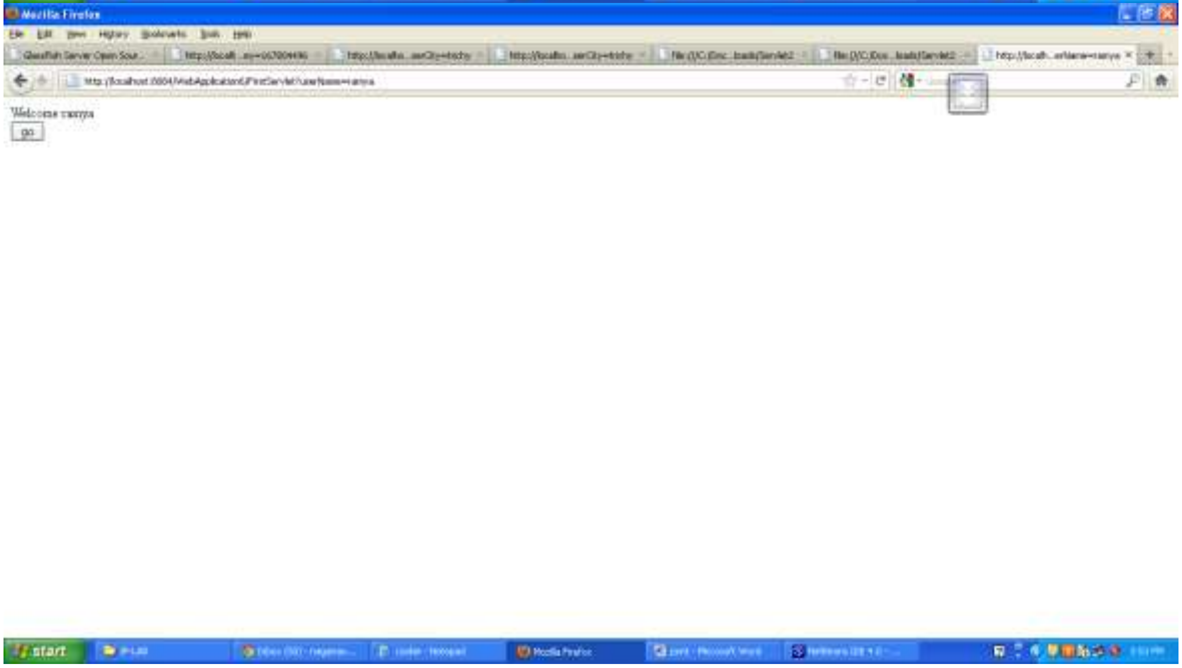
SecondServlet.java

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

public class SecondServlet extends HttpServlet {

public void doGet(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);}
  }
}
```



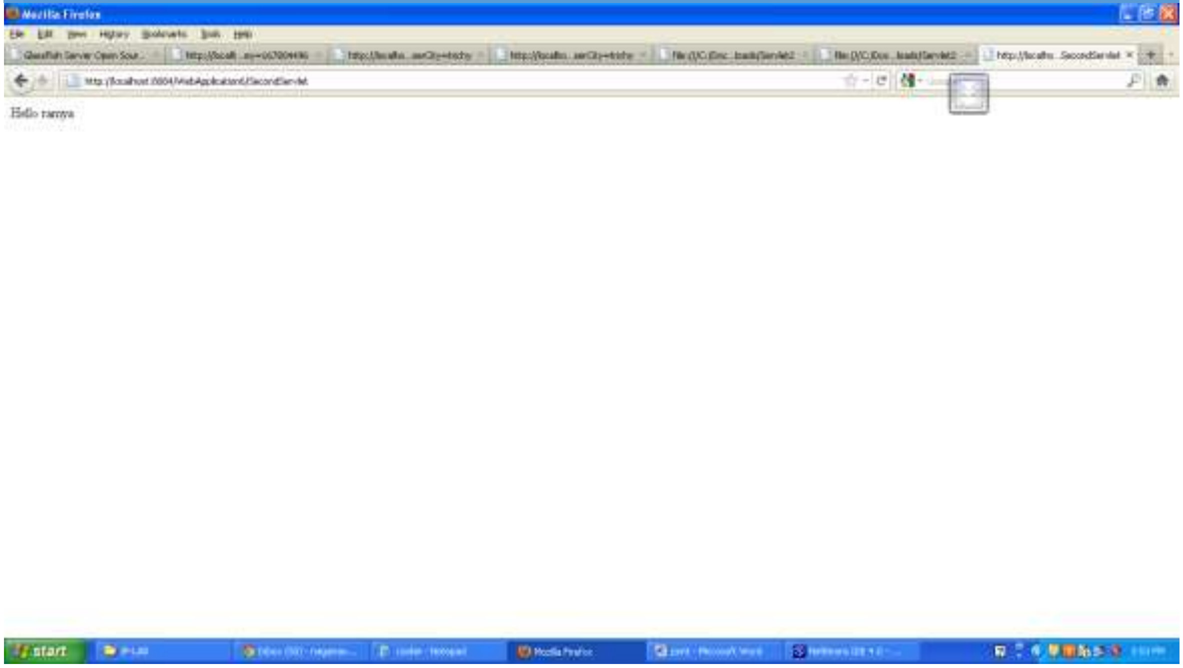
http://localhost:8084/WebApplication6/index.html

Name: ramya



http://localhost:8084/WebApplication6/FirstServlet?userName=ramya

Welcome ramya


Mozilla Firefox
File Edit View History Bookmarks Tools Help
GlassFish Server Open Sour...
http://localhost:8084/WebApplication6/SecondServlet
start
Mozilla Firefox

```
Hello ramya
```

EX.NO.11A SIMPLE JSP STRUTS FRAMEWORK

Aim

To develop a simple program for JSP Struts framework

Algorithm

1. Setting Up a Struts Application
2. Creating JSP Pages
    a. Creating a Login Page
    b. Creating a Success Page
3. Creating an `ActionForm` Bean
4. Creating an `Action` Class
5. Implementing Validation
    a. Accessing Bean Data and Preparing a Forwarding Condition
    b. Setting Up an Error Message
6. Adding `forward` Entries to `struts-config.xml`
7. Configuring and Running the Application
    a. Setting the Welcome Page
    b. Attaching a Stylesheet
    c. Running the Application

```
/*
 LoginForm.java
 */


package com.myapp.struts;

import javax.servlet.http.HttpServletRequest;
import org.apache.struts.action.ActionErrors;
import org.apache.struts.action.ActionMapping;
import org.apache.struts.action.ActionMessage;


public class LoginForm extends org.apache.struts.action.ActionForm {

    private String name;
    private String email;
    private String error;

    public String getError() {
        return error;
    }

    public void setError(String error) {
        this.error = "<span style='color:red'>Please provide valid entries for both
fields</span>";
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }
    private int number;

    /**
     * @return
     */
    public String getName() {
        return name;

    }

    /**
     * @param string
     */
    public void setName(String string) {
        name = string;
    }

    /**
     * @return
     */
    public int getNumber() {
        return number;
    }

    /**
     * @param i
     */
    public void setNumber(int i) {
        number = i;
    }

    /**
     *
     */
    public LoginForm() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * This is the action called from the Struts framework.
     *
     * @param mapping The ActionMapping used to select this instance.
     * @param request The HTTP Request we are processing.
     * @return
     */
    public ActionErrors validate(ActionMapping mapping, HttpServletRequest
request) {
        ActionErrors errors = new ActionErrors();
        if (getName() == null || getName().length() < 1) {
            errors.add("name", new ActionMessage("error.name.required"));
            // TODO: add 'error.name.required' key to your resources
        }
        return errors;
    }
}
```

Loginaction.java

```
package com.myapp.struts;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;


public class LoginAction extends org.apache.struts.action.Action {

    /* forward name="success" path="" */
    private static final String SUCCESS = "success";
    private final static String FAILURE = "failure";

    /**
     * This is the action called from the Struts framework.
     *
     * @param mapping The ActionMapping used to select this instance.
     * @param form The optional ActionForm bean for this request.
     * @param request The HTTP Request we are processing.
     * @param response The HTTP Response we are processing.
     * @throws java.lang.Exception
     * @return
     */
    @Override
    public ActionForward execute(ActionMapping mapping, ActionForm form,
            HttpServletRequest request, HttpServletResponse response)
            throws Exception {
        // extract user data
        LoginForm formBean = (LoginForm) form;
        String name = formBean.getName();
        String email = formBean.getEmail();


        // perform validation
        if ((name == null) || // name parameter does not exist
                email == null || // email parameter does not exist
                name.equals("") || // name parameter is empty
                email.indexOf("@") == -1) {   // email lacks '@'
            formBean.setError("");

            return mapping.findForward(FAILURE);
        }
        return mapping.findForward(SUCCESS);
    }
}
```

Login.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
        <link rel="stylesheet" type="text/css" href="stylesheet.css">
    </head>
    <body>
        <h1>Hello World!
            <html:form action="/login">
                <table border="0">
                    <thead>
                        <tr>
                        <td colspan="2">
                           <bean:write name="LoginForm" property="error"
filter="false"/>
                           &nbsp;</td>
                     </tr>
                  </thead>
                  <tbody>
                     <tr>
                        <td>Enter your name:</td>
                        <td><html:text property="name" /></td>
                     </tr>
                     <tr>
                        <td>Enter your email:</td>
                        <td><html:text property="email" /></td>
                     </tr>
                     <tr>
                        <td></td>
                        <td></td>
                     </tr>
                  </tbody>
               </table>

               <html:submit value="Login" />

            </html:form>
         </h1>
      </body>
</html>
```

stylesheet.css

```
body {
   font-family: Verdana, Arial, sans-serif;
   font-size: smaller;
   padding: 50px;
   color: #555;
   width: 650px;
}

h1 {
   letter-spacing: 6px;
   font-size: 1.6em;
   color: #be7429;
   font-weight: bold;
}

h2 {
   text-align: left;
   letter-spacing: 6px;
   font-size: 1.4em;
   color: #be7429;
   font-weight: normal;
   width: 450px;
}

table {
   width: 550px;
   padding: 10px;
   background-color: #c5e7e0;
  font-family: sans-serif;
}

td {
   padding: 10px;
}

a {
  color: #be7429;
  text-decoration: none;
}

a:hover {
  text-decoration: underline;
}

.popupBox {
  position: absolute;
  top: 170px;
  left: 140px;
}

.popupCell {
   background-color: #fffafa;
}

.popupCell:hover {
  background-color: #f5ebe9;
}

.popupItem {
  color: #333;
  text-decoration: none;
  font-size: 1.2em;
}
```

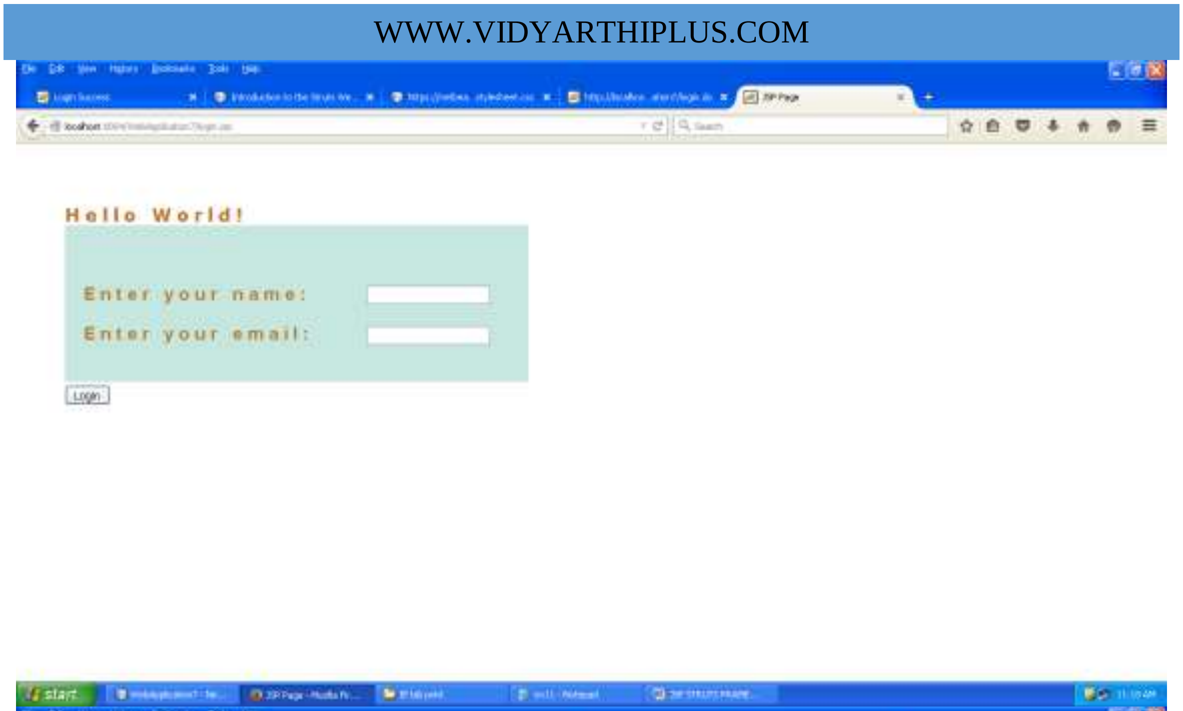
success.jsp

```
<%--
   Document   : success
   Created on : Sep 16, 2015, 2:10:18 PM
   Author     : staff
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>Login Success</title>
   </head>
   <body>
      <h1>Congratulations!</h1>
      <p>You have successfully logged in.</p>

      <p>Your name is: <bean:write name="LoginForm" property="name" /> .</p>

      <p>Your email address is:<bean:write name="LoginForm" property="email" />
.</p>
   </body>
</html>
```

struts-config.xml

```
<?xml version="1.0" encoding="UTF-8" ?>

<!DOCTYPE struts-config PUBLIC
         "-//Apache Software Foundation//DTD Struts Configuration 1.3//EN"
         "http://jakarta.apache.org/struts/dtds/struts-config_1_3.dtd">


<struts-config>
   <form-beans>
      <form-bean name="LoginForm1" type="com.myapp.struts.LoginForm1"/>
      <form-bean name="LoginForm" type="com.myapp.struts.LoginForm"/>

   </form-beans>

   <global-exceptions>

   </global-exceptions>
```

```
    <global-forwards>
        <forward name="welcome"  path="/Welcome.do"/>
    </global-forwards>

    <action-mappings>
        <action name="LoginForm" path="/login" scope="request"
type="com.myapp.struts.LoginAction" validate="false">
            <forward name="success" path="/success.jsp"/>
            <forward name="faliure" path="/login.jsp"/>
        </action>
        <action path="/Welcome" forward="/welcomeStruts.jsp"/>
    </action-mappings>

    <controller processorClass="org.apache.struts.tiles.TilesRequestProcessor"/>

    <message-resources parameter="com/myapp/struts/ApplicationResource"/>

    <!-- ========================= Tiles plugin
===============================-->

    <plug-in className="org.apache.struts.tiles.TilesPlugin" >
        <set-property property="definitions-config" value="/WEB-INF/tiles-defs.xml"
/>
        <set-property property="moduleAware" value="true" />
    </plug-in>

    <!-- ========================= Validator plugin
================================= -->
    <plug-in className="org.apache.struts.validator.ValidatorPlugIn">
        <set-property
            property="pathnames"
            value="/WEB-INF/validator-rules.xml,/WEB-INF/validation.xml"/>
    </plug-in>

</struts-config>
```

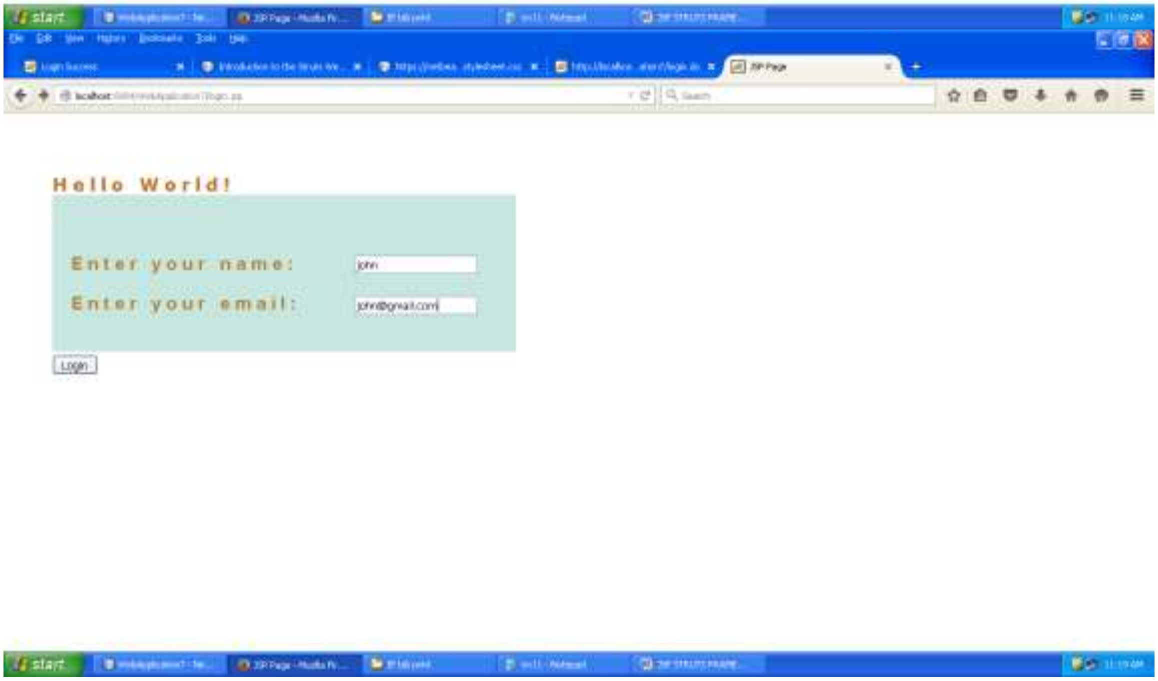


Hello World!

Enter your name:

Enter your email:

Login

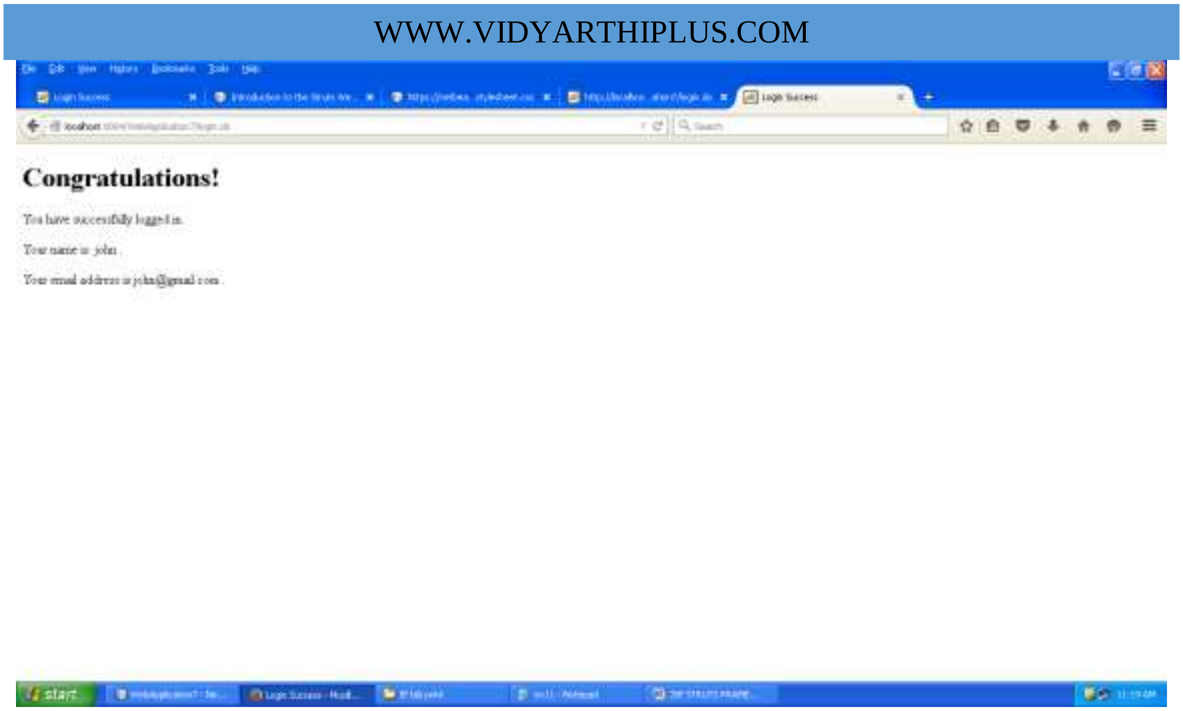


Hello World!

Enter your name: john

Enter your email: john@gmail.com

Login



Congratulations!

You have successfully logged in.

Your name is john .

Your email address is john@gmail.com .

EX.NO.11b SIMPLE SPRING FRAMEWORK

Aim

To develop a simple program using Spring Framework

Algorithm

- Setting up a New Project with Spring Web MVC Support
    - Creating a Spring Web MVC Skeleton Project
    - Running the Skeleton Project
- Overview of the Application
- Implementing a Service
- Implementing the Controller and Model
- Implementing the Views

dispatcher-servlet.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:p="http://www.springframework.org/schema/p"
       xmlns:aop="http://www.springframework.org/schema/aop"
       xmlns:tx="http://www.springframework.org/schema/tx"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-3.1.xsd
       http://www.springframework.org/schema/aop
http://www.springframework.org/schema/aop/spring-aop-3.1.xsd
       http://www.springframework.org/schema/tx
http://www.springframework.org/schema/tx/spring-tx-3.1.xsd">

    <bean
class="org.springframework.web.servlet.mvc.support.ControllerClassNameHandler
Mapping"/>
<bean class="controller.HelloController" p:helloService-ref="helloService"/>
<bean id="urlMapping"
class="org.springframework.web.servlet.handler.SimpleUrlHandlerMapping">
        <property name="mappings">
            <props>
                <prop key="index.htm">indexController</prop>
            </props>
        </property>
    </bean>

    <bean id="viewResolver"
          class="org.springframework.web.servlet.view.InternalResourceViewResolver"
          p:prefix="/WEB-INF/jsp/"
          p:suffix=".jsp" />
<bean name="indexController"
```

```
        class="org.springframework.web.servlet.mvc.ParameterizableViewController"
        p:viewName="index" />
<bean
class="org.springframework.web.servlet.mvc.support.ControllerClassNameHandler
Mapping"/>
</beans>
```

applicationContext.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:p="http://www.springframework.org/schema/p"
      xmlns:aop="http://www.springframework.org/schema/aop"
      xmlns:tx="http://www.springframework.org/schema/tx"
      xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-3.1.xsd
      http://www.springframework.org/schema/aop
http://www.springframework.org/schema/aop/spring-aop-3.1.xsd
      http://www.springframework.org/schema/tx
http://www.springframework.org/schema/tx/spring-tx-3.1.xsd">
<bean name="helloService" class="service.HelloService" />
</beans>
```

HelloController.java

```
package controller;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.springframework.validation.BindException;
import org.springframework.web.servlet.mvc.SimpleFormController;
import org.springframework.web.servlet.ModelAndView;
import service.HelloService;
public class HelloController extends SimpleFormController {
   private HelloService helloService;
   public void setHelloService(HelloService helloService) {
   this.helloService = helloService;
}
   public HelloController() {
        setCommandClass(Name.class);
   setCommandName("name");
   setSuccessView("helloView");
   setFormView("nameView");
   }
@Override
   protected ModelAndView onSubmit(
          HttpServletRequest request,
          HttpServletResponse response,
          Object command,
          BindException errors) throws Exception {

      Name name = (Name) command;
      ModelAndView mv = new ModelAndView(getSuccessView());
      mv.addObject("helloMessage", helloService.sayHello(name.getValue()));
      return mv;
}

}
```

HelloService.java

```
package service;
public class HelloService {
 public static String sayHello(String name) {
        return "Hello " + name + "!";
    }
}
```

Name.java

```
package controller;
public class Name {
    private String value;

    public String getValue() {
        return value;
    }

    public void setValue(String value) {
        this.value = value;
    }
}
```

nameView.jsp

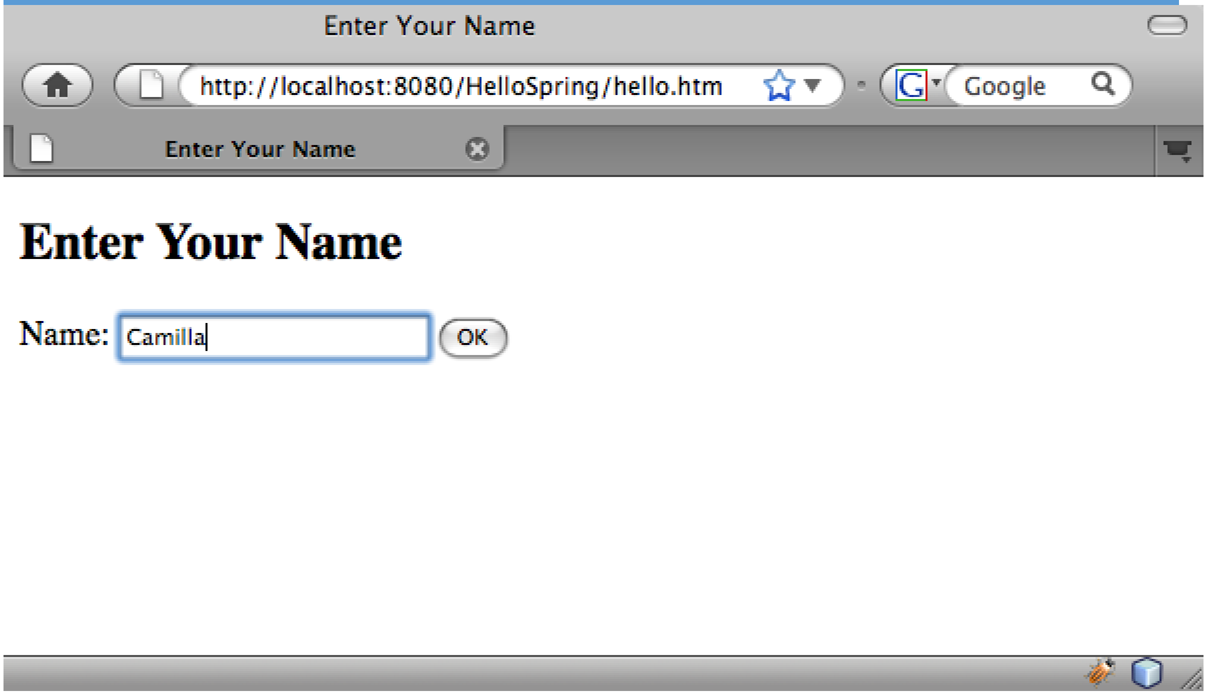
```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib uri="http://www.springframework.org/tags" prefix="spring" %>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Enter your name</title>
    </head>
    <body>
        <spring:nestedPath path="name">
    <form action="" method="post">
        Name:
        <spring:bind path="value">
            <input type="text" name="${status.expression}" value="${status.value}">
        </spring:bind>
        <input type="submit" value="OK">
    </form>
</spring:nestedPath>
    </body>
</html>
```

helloview.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Hello</title>
    </head>
    <body>
         <h1>${helloMessage}</h1>
    </body>
</html>
```

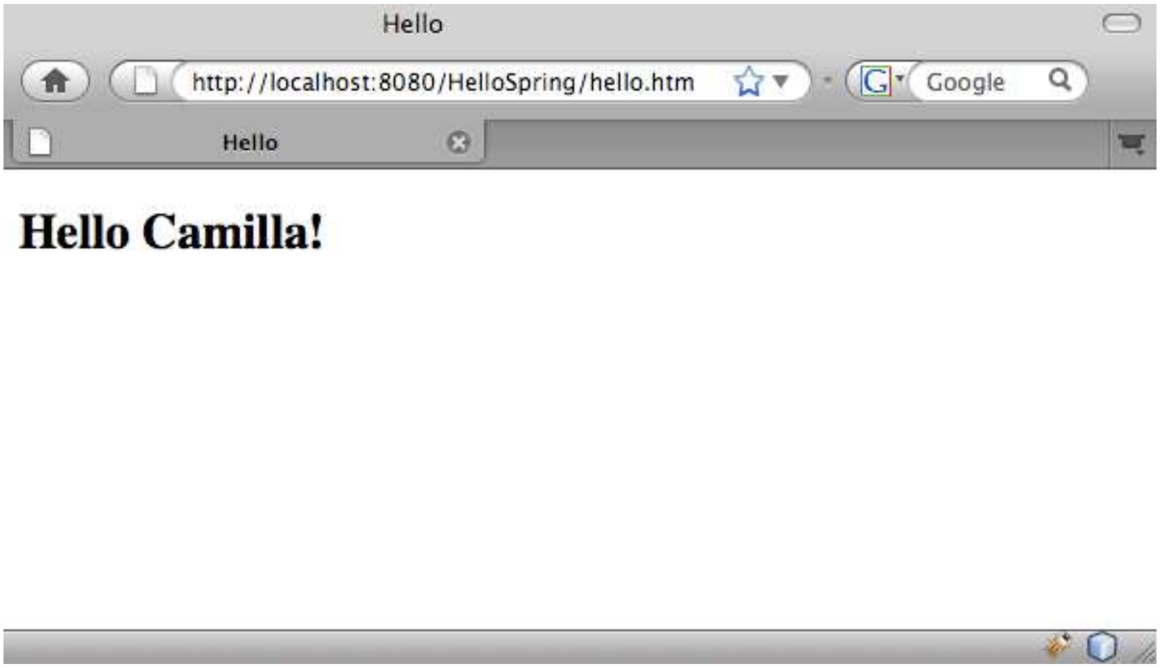
redirect.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<% response.sendRedirect("index.htm"); %>
```



Enter Your Name

Name: Camilla OK



Hello Camilla!

EX.NO.12 Programs using AJAX

Aim

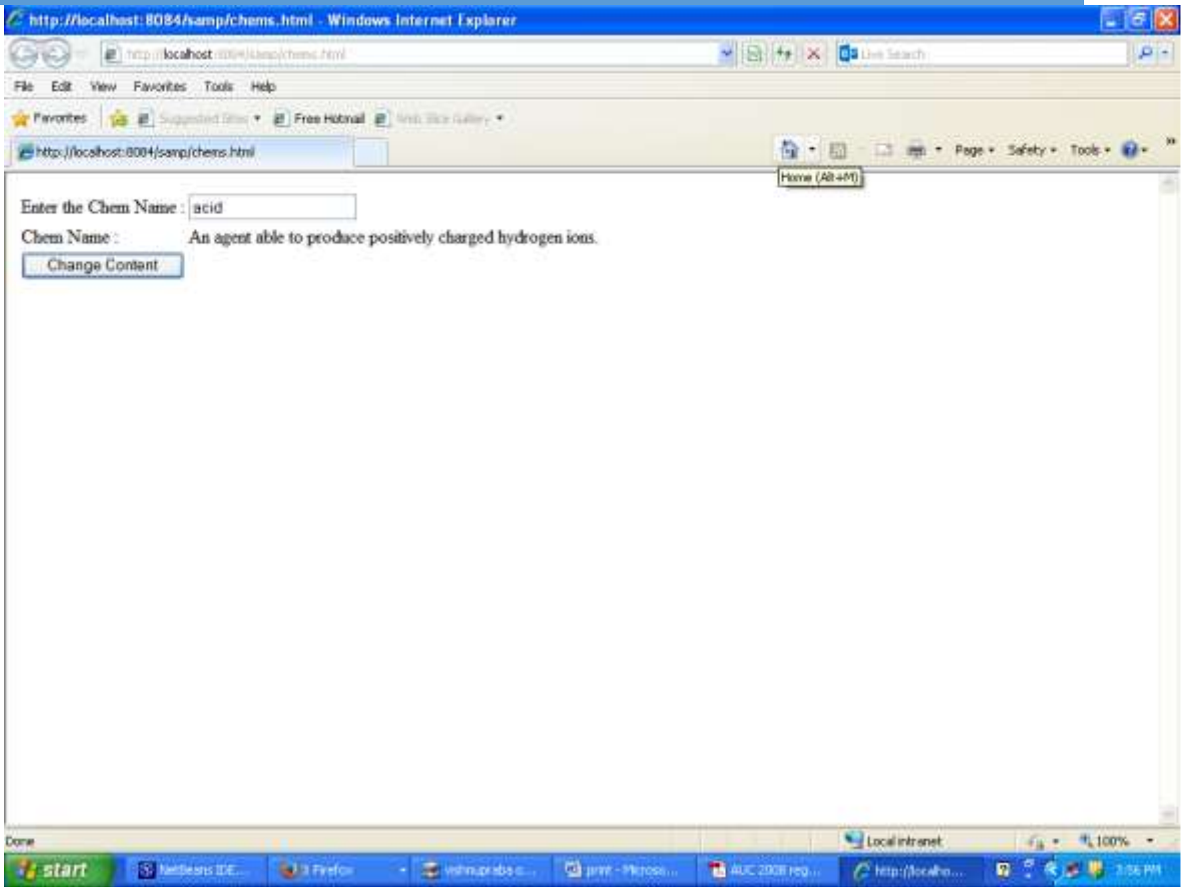
To develop a program using AJAX

Algorithm

1. Start the program
2. Write the JSP program and save as.jsp
3. Write a program html and save as .html
4. In a html program decide a java script to figure the element
5. Compile the html program in internet explorer
6. Display the result
7. Stop the program

```
/* chems.jsp */

<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>
<%String d=request.getParameter("t1");
String[]name;
int i=0,n=0;
name=new String[10];
name[0]="atomic number";
name[1]="catalyst";
name[2]="acid";
name[3]="base";
name[4]="bond energy";
name[5]="chain reactons";
name[6]="covalent bonds";
name[7]="element";
name[8]="enzyme";
name[9]="kinetics";
String[] defn;
defn=new String[10];
defn[0]="It is defined as the number of protons or electrons.";
defn[1]="a catalyst is a substance which fastens a reaction without themselves
undergoing any change.";
defn[2]="An agent able to produce positively charged hydrogen ions.";
defn[3]="A base is a substance that can combine with a proton.";
defn[4]="The energy required to break a particular bond by hompolytic process.";
defn[5]="chain reaction:reactions which proceed by means of a set of repeating cyclic
steps.";
defn[6]="Linkage of two atoms by the sharing  of two electrons.";
defn[7]="a substance which cannot be further subdivided by chemical methods.";
defn[8]="a naturally occuringb substance able to catalyse a chemical reaction.";
defn[9]="The study of rate of reactions.";
for(i=0;i<9;i++)
 { if(d.equals(name[i]))
      n=i; }
```

```
out.println(defn[n]);
 %>


/* chems.html */

<html>
  <head>
    <script type="text/javascript">
function loadXMLDoc()
{ var xmlhttp;
  var data = (t1.value);
 if (window.XMLHttpRequest)
{
  xmlhttp=new XMLHttpRequest();
}
else
{
   xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
 }
    xmlhttp.onreadystatechange=function()
 {
if (xmlhttp.readyState==4 && xmlhttp.status==200)
 {
document.getElementById("myDiv").
            innerHTML=xmlhttp.responseText;
}}

xmlhttp.open("GET","chems.jsp?t1="+data,true);
 xmlhttp.send();
}
</script>
</head>
<body>
<table>
 <tr><td> Enter the Chem Name : </td> <td>
 <Input type=text name=t1 ></td> </tr>
<tr><td> Chem Name : </td><td> <div id="myDiv"></div></td> </tr>
 <tr><td> <button type="button" onClick="loadXMLDoc()">
            Change Content</button></td> </tr>
</table>
</body>
</html>
```



http://localhost:8084/samp/chems.html - Windows Internet Explorer
File Edit View Favorites Tools Help
Favorites
Free Hotmail
Page Safety Tools
Home (Alt+M)
Done
Local intranet
100%
start

Enter the Chem Name : acid

Chem Name : An agent able to produce positively charged hydrogen ions.

Change Content

EX.NO.13 AN AIRLINE SERVICE AND TRAVEL AGENT USING WEBSERVICES AND DATABASE

Aim

To create a web page for a service using the html and agent using web service and database.

Algorithm

1. Create index page for airline service using the html and save the file by index.html format
2. Create a web page about html using tag and give details about the travel agency.
3. If the user click the home page they will get the reservation details from the web page
4. Store the related information in the database by creating the database table.
5. Stop the execution.

index.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">

<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>Travel Agency</title>
   </head>
   <body>
      <center><h1>Travel Easy</h1>
         <h3> - A Smarter way to travel</h3>
         <br><br>
         <b> Enter your Details</b>
         <form name="index" action="AirlineList.jsp" method="post">
            <br>
            <table cellpadding="6" cellspacing="6">
               <tr>
                  <td><b>From</b></td>
                  <td><input type="text" name="txt_from"</td>
               </tr>
               <tr>
                  <td><b>To</b></td>
                  <td><input type="text" name="txt_to"</td>
               </tr>
               <tr>
                  <td><b>Date of Journey</b></td>
                  <td><input type="text" name="txt_depart"</td>
               </tr>
               <tr>
                  <td><b>Number of Passengers</b></td>
            <td><input type="text" name="txt_no"</td>
          </tr>
        <tr>
          <td><b>Type of Flight</b></td>
            <td><input type="Radio" name="group1"
value="domestic">Domestic Flights</td
        </tr>
        <tr>
          <td></td>
          <td><input type="Radio" name="group1"
value="international">International Flights</td>
        </tr>
        <tr>
          <td><b>Select desired Airlines</b></td>
          <td>
            <select name="airline">
              <option>Kingfisher</option>
              <option>SpiceJet</option>
            </select>
          </td>
        </tr>
        </table>
        <br><br>
        <input type="Submit" name="Submit" value="Find Flights">
      </form>

    </center>
  </body>
</html>
```

airlinelist.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Domestic Flights</title>
  </head>
  <body>
    <center><h1>Travel Easy</h1>
      <h3> - A Smarter way to travel</h3>
    </center>
    <br><br>
    <%@ page language="java" %>
    <%@ page import ="java.sql.*" %>
    <%@ page import =
"java.util.Date,java.text.SimpleDateFormat,java.text.ParseException"%>
    <%
      String type = request.getParameter("group1");
      String from= request.getParameter("txt_from");
      String to = request.getParameter("txt_to");
      String depart= request.getParameter("txt_depart");
      String ret = request.getParameter("txt_return");
      String no = request.getParameter("txt_no");
      String airline = request.getParameter("airline");
      Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
      String sTable = type;
      String sSql = "SELECT * FROM "   sTable   " WHERE From='" from "' and
TO='" to "' and DepartDate='" depart "' and SeatCapacity>=" no "";
      String sDBQ = "d:/"  airline ".mdb";

      String database = "jdbc:odbc:Driver={Microsoft Access Driver
(*.mdb)};DBQ="   sDBQ   ";DriverID=22;READONLY=true";
      Connection cn = null;
      Statement st = null;
      ResultSet rs = null;
      try {
        cn = DriverManager.getConnection( database ,"","");
        st = cn.createStatement();
        rs = st.executeQuery( sSql );
        ResultSetMetaData rsmd = rs.getMetaData();
        String s1,s2,s3,s4,s5,s6,s7,s8;
        int i=1;
        String flight_no[] = new String[100];
        //out.println("<form name='AirlLine' action='' method='post'><b>" i " . "
rs.getString(1)   "</b><br><br>");
        out.println("<h3><b>" airline  " Flights</b></h3><br>");
        out.println("<table cellspacing=20 cellpaddinng=15>");
        out.println("<tr><th>Flight
No</th><th>From</th><th>To</th><th>Departure Date</th><th>Arrival
Date</th><th>Start Time</th>");
        out.println("<th>Reach Time</th><th>Seats Remaining</th></tr>");
        while(rs.next())
        {
        s1 = rs.getString(1);
        flight_no[i] = s1;
        s2 = rs.getString(2);
        s3 = rs.getString(3);
        s4 = rs.getString(4);
        s5 = rs.getString(5);
        s6 = rs.getString(6);
        s7 = rs.getString(7);
        s8 = rs.getString(8);
        out.println("<tr><td>" s1 "</td>");
        out.println("<td>" s2 "</td>");
        out.println("<td>" s3 "</td>");
        out.println("<td>" s4 "</td>");
        out.println("<td>" s5 "</td>");
        out.println("<td>" s6 "</td>");
        out.println("<td>" s7 "</td>");
        out.println("<td>" s8 "</td></tr>");
        i  ;
        }
        out.println("</table>");
        if(i==1)
        {
          out.println("<br><center><b>Sorry!! No flights scheudle
available</b></center><br>");
        }
        else
        {
          out.println("<form name='AirlineList' action='" airline ".jsp'
method='post'>");
          out.println("<br><table><tr><td>Select Flight</td><td>");
          out.println("<select name='flight_no'>");
          for(int temp=1;temp<i;temp  )
          {
            out.println("<option>" flight_no[temp] "</option>");
          }
          out.println("</select></td></tr><tr></tr>");
          out.println("<tr><td><input type='hidden' name='no' value='" no
"'</td>");
          out.println("<td><input type='hidden' name='air_type' value='" type
"'</td></tr>");
          for(int j=1;j<=Integer.parseInt(no);j  )
          {
            out.println("<tr><th>Passenger " j  " details</th></tr>");
            out.println("<tr></tr>");
            out.println("<tr><td>Name</td><td><input type='text' name='pas" j
"_name'</td></tr>");
            out.println("<tr><td>Age</td><td><input type='text' name='pas" j
"_age'</td></tr>");
            out.println("<tr><td>Sex</td><td><input type='text' name='pas" j
"_sex'</td></tr>");
            out.println("<tr></tr>");
          }
          out.println("</table><br><br><input type='Submit' name='Submit'
value='Book Now'/></form>");
```

```
            }
            //out.println("<input name ='submit' value='Submit' type='submit'/>");
         }
         finally {
            try { if( null != rs ) rs.close(); } catch( Exception ex ) {}
            try { if( null != st ) st.close(); } catch( Exception ex ) {}
            try { if( null != cn ) cn.close(); } catch( Exception ex ) {}
               }

      %>
   </body>
</html>
```

Kingfisher.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">

<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>Kingfisher Airlines</title>
   </head>
   <body>
      <%@ page language="java" %>
      <%@ page import ="java.sql.*" %>
      <%
         String s = request.getParameter("no");
         out.println("<h1><center>Kingfisher Airlines</h1><br><h3>- Have a nice
trip!!!</h3></center><br>");
         out.println("<br><br>Your Booking Details");
         out.println("<br><br><table cellspacing=10 cellpaddin=10>");
         Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
         String sTable = request.getParameter("air_type");
         String sSql = "SELECT * FROM "   sTable   " WHERE FlightNo='"
request.getParameter("flight_no") "'";
         String sDBQ = "d:/kingfisher.mdb";

         String database = "jdbc:odbc:Driver={Microsoft Access Driver
(*.mdb)};DBQ="   sDBQ   ";DriverID=22;READONLY=true";
         Connection cn = null;
         Statement st = null;
         ResultSet rs = null;
```

```
            String
name,age,sex,flightno,from,to,departure,arrival,starttime,reachtime,seat;
            out.println("<tr><th>Name</th><th>Age</th><th>Sex</th><th>Flight
No</th><th>From</th><th>To</th><th>Departure Date</th><th>Arrival
Date</th><th>Start Time</th><th>Reach Time</th><th>Seat No</th></tr>");
            try
            {
             cn = DriverManager.getConnection( database ,"","");
             st = cn.createStatement();
             rs = st.executeQuery( sSql );
             ResultSetMetaData rsmd = rs.getMetaData();
             int seat1=1;
             while(rs.next())
             {
                 flightno = rs.getString(1);
                 from = rs.getString(2);
                 to = rs.getString(3);
                 departure = rs.getString(4);
                 arrival = rs.getString(5);
                 starttime = rs.getString(6);
                 reachtime = rs.getString(7);
                 seat = rs.getString(8);
                 seat1 = Integer.parseInt(seat);
                 for(int i=1;i<=Integer.parseInt(s);i  )
            {
                name= request.getParameter("pas" i "_name");
                age = request.getParameter("pas" i "_age");
                sex = request.getParameter("pas" i "_sex");
                flightno = request.getParameter("flight_no");

                out.println("<tr><td>" name "</td><td>" age "</td><td>" sex "</td>");
                out.println("<td>" flightno "</td><td>" from "</td><td>" to "</td><td>"
departure "</td>");


                out.println("<td>" arrival "</td><td>" starttime "</td><td>" reachtime
"</td><td>" seat1 "</td>");
                seat1--;
               }

             }
             Connection cn1 = null;
            Statement st1 = null;
            ResultSet rs1 = null;
           try
           {
            sSql = "update "   sTable   " set SeatCapacity=" seat1 " WHERE FlightNo='"
request.getParameter("flight_no") "'";
```

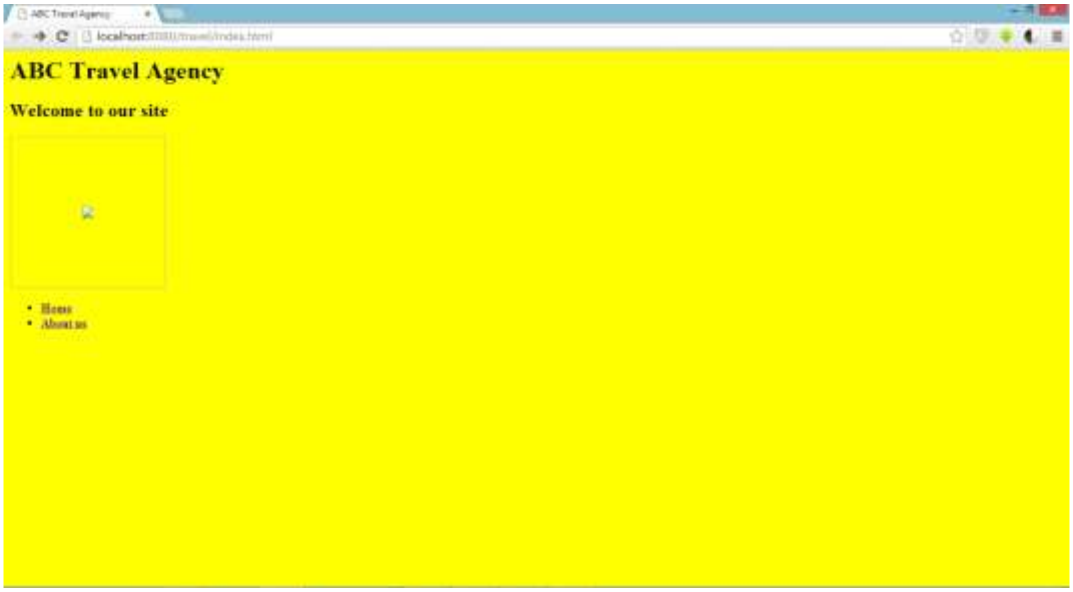
```
      rs1 = st.executeQuery( sSql );
      }
     catch(Exception e)
     {
      }

      }
      finally {
        try { if( null != rs ) rs.close(); } catch( Exception ex ) {}
        try { if( null != st ) st.close(); } catch( Exception ex ) {}
        try { if( null != cn ) cn.close(); } catch( Exception ex ) {}
          }
    %>
  </body>
</html>
```

SpiceJet.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>SpiceJet Airlines</title>
  </head>
  <body>
    <%@ page language="java" %>
    <%@ page import ="java.sql.*" %>
    <%
      String s = request.getParameter("no");
      out.println("<h1><center>SpiceJet Airlines</h1><br><h3>- Have a nice
trip!!!</h3></center><br>");
      out.println("<br><br>Your Booking Details");
      out.println("<br><br><table cellspacing=10 cellpaddin=10>");
      Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
      String sTable = request.getParameter("air_type");
      String sSql = "SELECT * FROM "   sTable   " WHERE FlightNo='"
request.getParameter("flight_no") "'";
      String sDBQ = "d:/SpiceJet.mdb";

      String database = "jdbc:odbc:Driver={Microsoft Access Driver
(*.mdb)};DBQ="   sDBQ   ";DriverID=22;READONLY=true";
      Connection cn = null;
```

```
          Statement st = null;
          ResultSet rs = null;
          String
name,age,sex,flightno,from,to,departure,arrival,starttime,reachtime,seat;
          out.println("<tr><th>Name</th><th>Age</th><th>Sex</th><th>Flight
No</th><th>From</th><th>To</th><th>Departure Date</th><th>Arrival
Date</th><th>Start Time</th><th>Reach Time</th><th>Seat No</th></tr>");
          try
          {
           cn = DriverManager.getConnection( database ,"","");
           st = cn.createStatement();
           rs = st.executeQuery( sSql );
           ResultSetMetaData rsmd = rs.getMetaData();
           int seat1=1;
           while(rs.next())
           {
              flightno = rs.getString(1);
              from = rs.getString(2);
              to = rs.getString(3);
              departure = rs.getString(4);
              arrival = rs.getString(5);
              starttime = rs.getString(6);
              reachtime = rs.getString(7);
              seat = rs.getString(8);
              seat1 = Integer.parseInt(seat);
              for(int i=1;i<=Integer.parseInt(s);i  )
          {
             name= request.getParameter("pas" i "_name");
             age = request.getParameter("pas" i "_age");
             sex = request.getParameter("pas" i "_sex");
             flightno = request.getParameter("flight_no");

             out.println("<tr><td>" name "</td><td>" age "</td><td>" sex "</td>");
             out.println("<td>" flightno "</td><td>" from "</td><td>" to "</td><td>"
departure "</td>");


             out.println("<td>" arrival "</td><td>" starttime "</td><td>" reachtime
"</td><td>" seat1 "</td>");
             seat1--;
            }

           }
           Connection cn1 = null;
          Statement st1 = null;
          ResultSet rs1 = null;
         try
         {
          sSql = "update "   sTable   " set SeatCapacity=" seat1 " WHERE FlightNo='"
request.getParameter("flight_no") "'";
          rs1 = st.executeQuery( sSql );
          }
         catch(Exception e)
         {
          }

          }
          finally {
             try { if( null != rs ) rs.close(); } catch( Exception ex ) {}
             try { if( null != st ) st.close(); } catch( Exception ex ) {}
             try { if( null != cn ) cn.close(); } catch( Exception ex ) {}
                 }
     %>
 </body>
</html>
```

INDEX.HTML

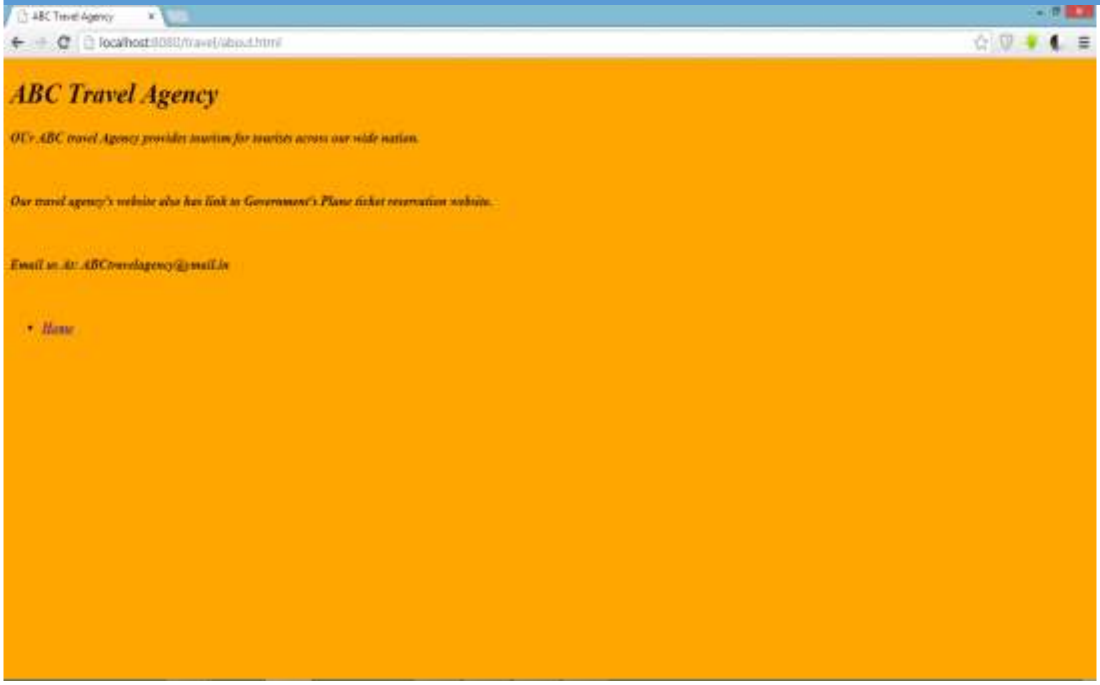


ABC Travel Agency

Welcome to our site

- Home
- About us

ABOUT.HTML



ABC Travel Agency

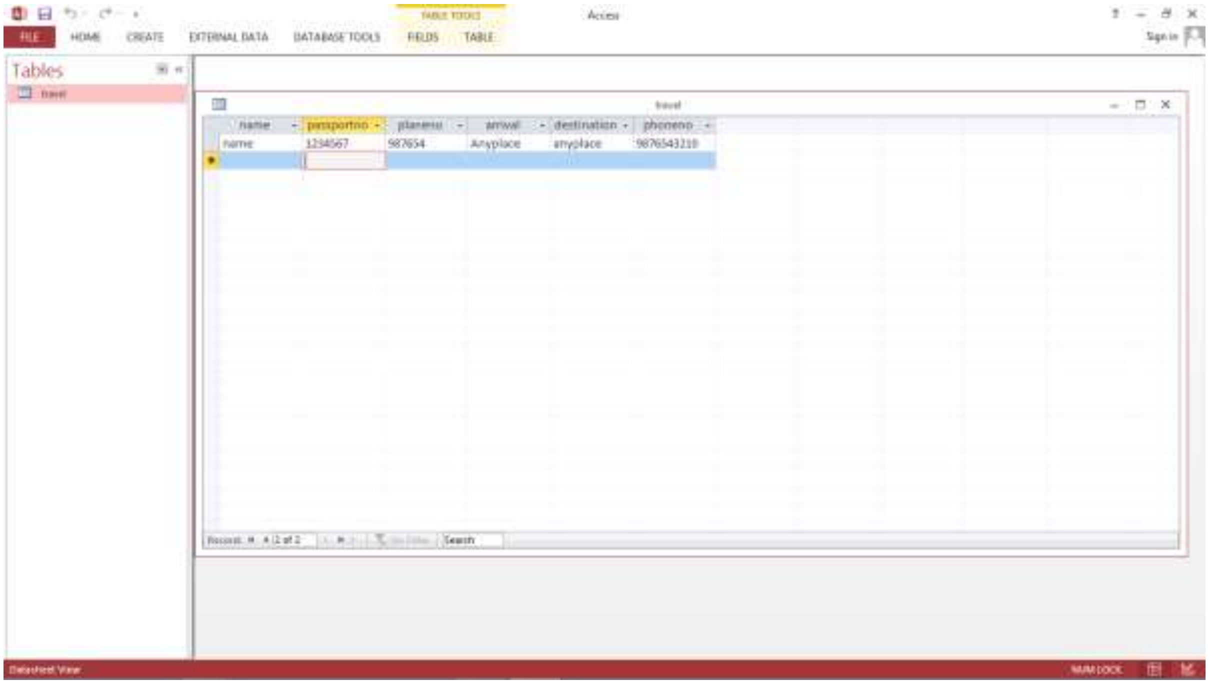
OUr ABC travel Agency provides tourism for tourists across our wide nation.

Our travel agency's website also has link to Government's Plane ticket reservation website.

Email us At: ABCtravelagency@mail.in

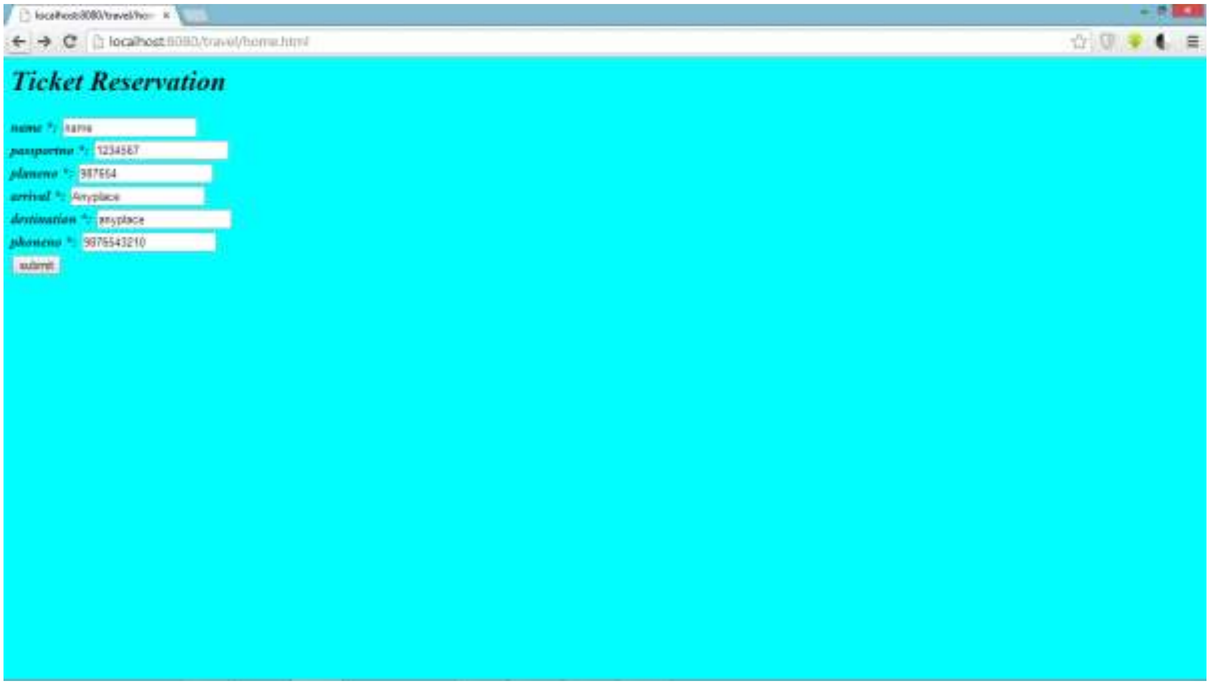
- Home

DATABASE:



name	passportno	planeno	arrival	destination	phoneno
name	1234567	987654	Anyplace	anyplace	9876543210

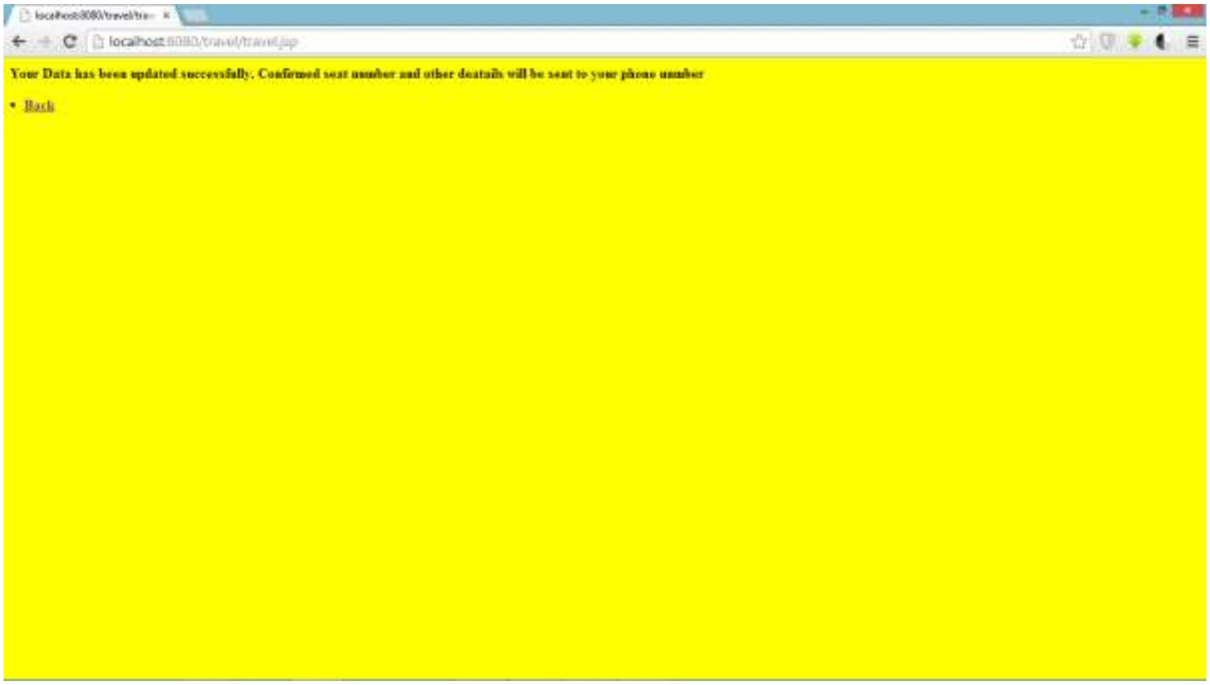
HOME.HTML



Ticket Reservation

name *: name
passportno *: 1234567
planeno *: 987654
arrival *: Anyplace
destination *: anyplace
phoneno *: 9876543210

TRAVEL.JSP



Your Data has been updated successfully. Confirmed seat number and other deatails will be sent to your phone number

- Back

EX.NO.14 WEBSERVICE FOR PRODUCT SALE

Aim

To develop a web service for product sale.

Algorithm

1. Create a home page as home.html and add all the related web pages as add.html, display.html, delete.html
2. Create the add.jsp file and create the database for user information and order information to store in the database
3. Create the display.jsp file to display the items purchased by the users.
4. Create the delete.jsp file to delete the product that the customer want to remove
5. Run the project with the home.html file in the local host with port number.

home.html

```
<html>
<body bgcolor=cyan>
   <font face=arial size=6></font>
<form method="POST" action="--WEBBOT-SELF--">
<div align=center>
<b><i> PRODUCT SALE </i></b>
<br><br><a href="add.html">Adding Details </a>
<br> <a href="delete.html">Deleting Details</a>
<br><a href="display.jsp">Display all details </a>
</div>
</form>
</body>
</html>
```

add.html

```
<html>
<body>
<b><i> ADDING THE DETAILS FOR ORDER PROCESSING </i></b>
<form  action="http://localhost:8080/WebApplication3/add.jsp">
     <table>
<tr>
          <th>NAME</th>
          <td><input type="text" name="name" ></td>
</tr>
<tr>
          <th>ORDER NO</th>
          <td><input type="text" name="orderno" ></td>
</tr>
<tr>
```

```
        <th>ITEM NAME</th>
        <td><input type="text" name="itemname" ></td>
</tr>
<tr>
        <th>QUANTITY</th>
        <td><input type="text" name="quantity" ></td>
</tr>
</table>
<input type="button" value="SAVE" >
<input type="reset" value="CLEAR">
   </form>
</body>
</html>
```

display.html

```
<html>
<body>
<p align=center>
<font face=arial size=6>
<b><i> DISPLAYING THE PARTICULAR ORDER PROCESSING </i></b>
</font></p>
<form method=post
action="http://localhost:8080/WebApplication3/display.jsp">
<div align=center>
<table>
<tr>
        <th align=center>Enter the Order No</th>
        <td><input type="text" name="orderno"></td>
</tr>
</table><br><br>
<input type=submit value="DISPLAY">
<input type=reset value="CLEAR">
</div>
</form>
</body>
</html>
```

delete.html

```
<html>
<body>
<p align=center>
<!font face=arial size=6/>
<b><i> DELETING THE DETAILS FOR ORDER PROCESSING </i></b>
</p>
```

```
<form method=post
action="http://localhost:8080/WebApplication3/delete.jsp">
<div align=center>
   <table>
<tr>
         <th align=center>Enter the Order No</th>
         <td><input type="text" name="orderno"></td>
</tr>
</table><br><br>
<input type=submit value="DELETE" onClick="alert('The Particular order
is deleted')">
<input type=reset value="CLEAR">
</div>
</form>
</body>
</html>
```

add.jsp

```
<html><body>
<%@ page language="java" import="java.sql.*" %>
<%@page import="java.io.*"%>
<%@page import="java.util.*"%>
<%
String name=request.getParameter("name");
String orderno=request.getParameter("orderno");
String itemname=request.getParameter("itemname");
String quantity=request.getParameter("quantity");
try
{
   Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
String url="jdbc:odbc:order";
Connection connect =DriverManager.getConnection(url,"","");
Statement stmt=connect.createStatement();
stmt.executeQuery("INSERT INTO Table1
VALUES('"+name+"','"+orderno+"','"+itemname+"','"+quantity+"')");
%>

<p align=center>
<br>
<b><i>
<% out.println("\n  The values stored in the database.\n"); %>
   </i></b></p>
<%
stmt.close();
connect.close();
}
catch(SQLException e){}
```

```
%><br>
<a href="home.html">GOTO HOME </a>
</body></html>
```

display.jsp

```
<html>
<body>
<%@page import="java.sql.*" %>
<%
try
{
String orderno=request.getParameter("orderno");
int a=Integer.parseInt(orderno);
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con= DriverManager.getConnection("jdbc:odbc:order","","");
Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery("select * from add where orderno="+a+"");
while(rs.next())
{
%>
<p align=center>
<br>
<b><i> The Details of Particular Order.. </i></b>
<br><br>
<table border=5 cellpading=5 cellspace=5 width=300 height=200>
<tr>
<th align=center>NAME</th>
<th align=center><%=rs.getString(1) %></td>
</tr><tr>
<th align=center>ORDER NO</th>
<th align=center><%=rs.getInt(2) %></td>
</tr><tr>
<th align=center>ITEM NAME</th>
<th align=center><%=rs.getString(3) %></td>
</tr><tr>
<th align=center>QUANTITY</th>
<th align=center><%=rs.getInt(4) %></td>
</tr>
</table>
</p>
<%
}
rs.close();
stmt.close();
con.close();
}
catch(SQLException e){}
```

```
%>
<br>
<a href="home.html">GOTO HOME </a>
</p></body></html>
```

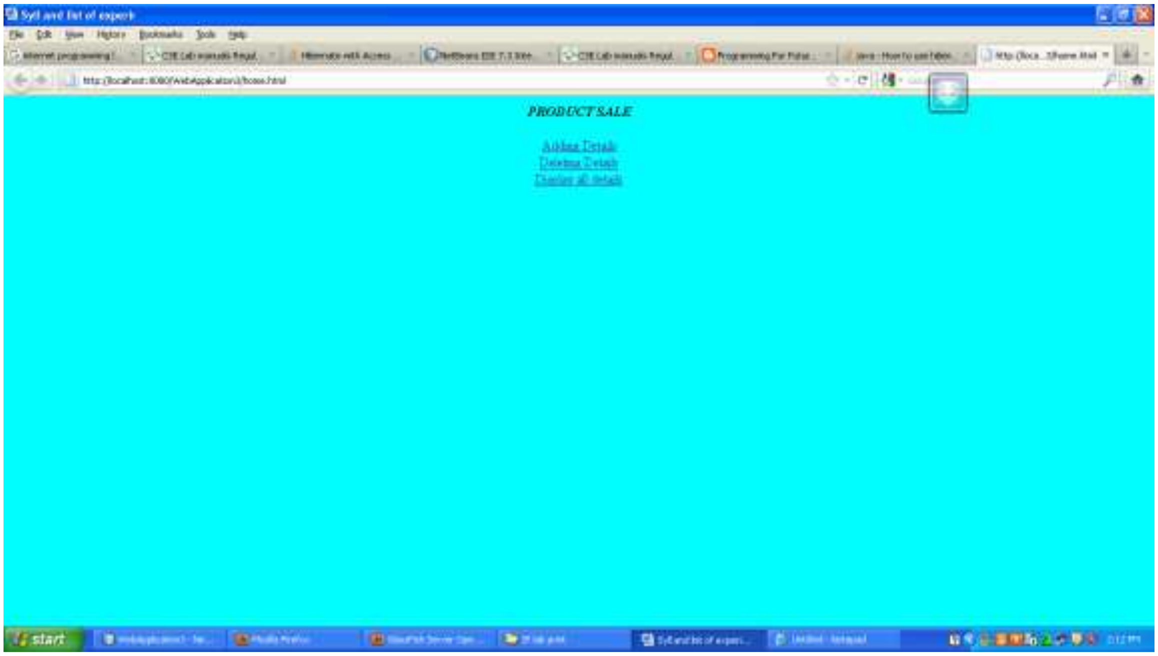
details.jsp

```
<html>
<body>
<%@ page import="java.sql.*" %>
<%
 try
 {
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con= DriverManager.getConnection("jdbc:odbc:order","","");
Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery("select * from add");
{
%>
<p align=center>
<br>
<b><i> The Details of All Order.. </i></b>
<br><br>
<table border=5 cellpading=5 cellspace=5 width=300 height=200>
<tr>
<td align=center><%= rs.getString(1) %></td>
<td align=center><%= rs.getInt(2) %></td>
<td align=center><%= rs.getString(3) %></td>
<td align=center><%=rs.getInt(4) %></td>
</tr>
<% } %>
</table>
</p>
<%
rs.close();
stmt.close();
con.close();
}
catch(SQLException e){}
%>
<br>
<a href="home.html">GOTO HOME </a>
</body></html>
```

delete.jsp

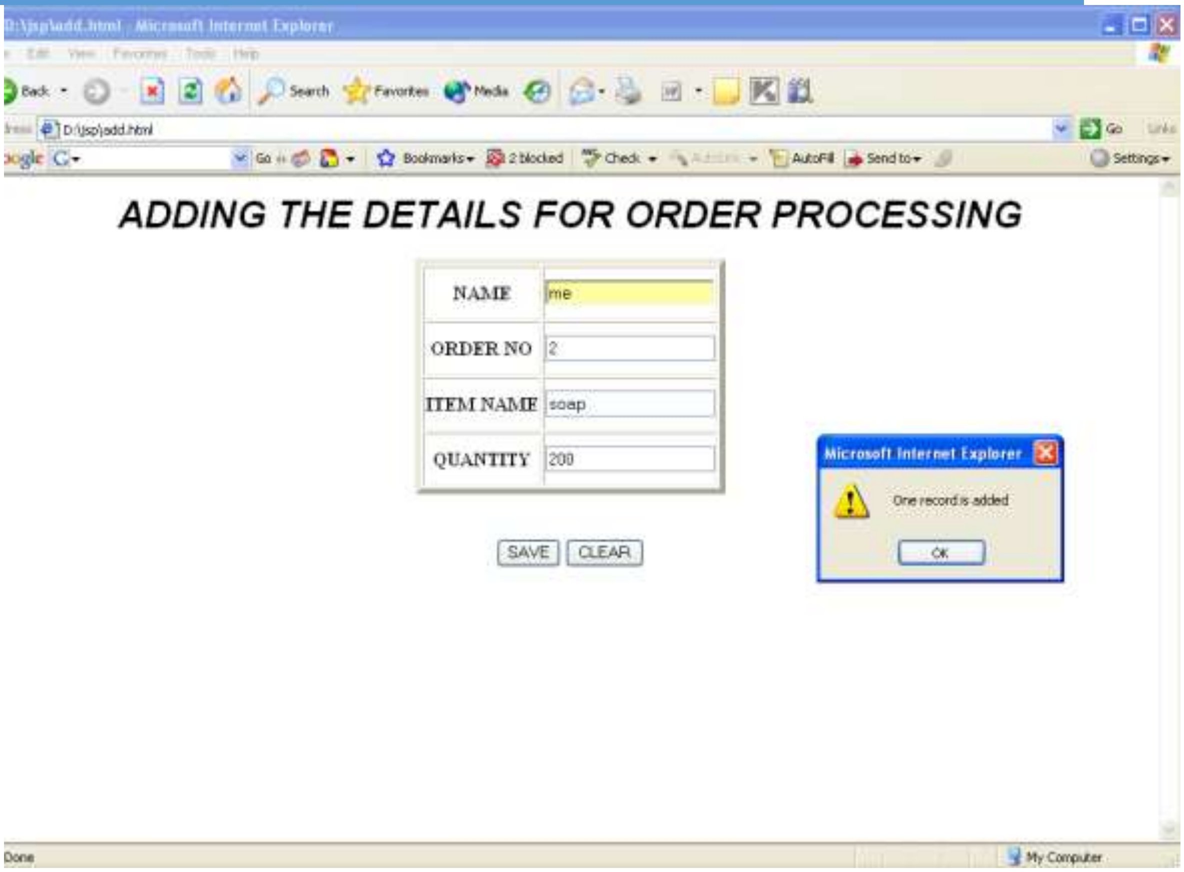
```
<html>
```

```
<body>
<%@ page import="java.sql.*" %>
<%
try
{
String orderno=request.getParameter("orderno");
int a=Integer.parseInt(orderno);
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con= DriverManager.getConnection("jdbc:odbc:order","","");
Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery("delete from add where orderno="+a+"");
rs.close();
stmt.close();
con.close();
}
catch(SQLException e){}
%>
<p align=center>
<b><i> The order deleted </i></b>
<br>
<a href="home.html">GOTO HOME </a>
</p></body></html>
```


File Edit View History Bookmarks Tools Help
http://localhost:8080/WebApplication1/home.html
start

PRODUCT SALE

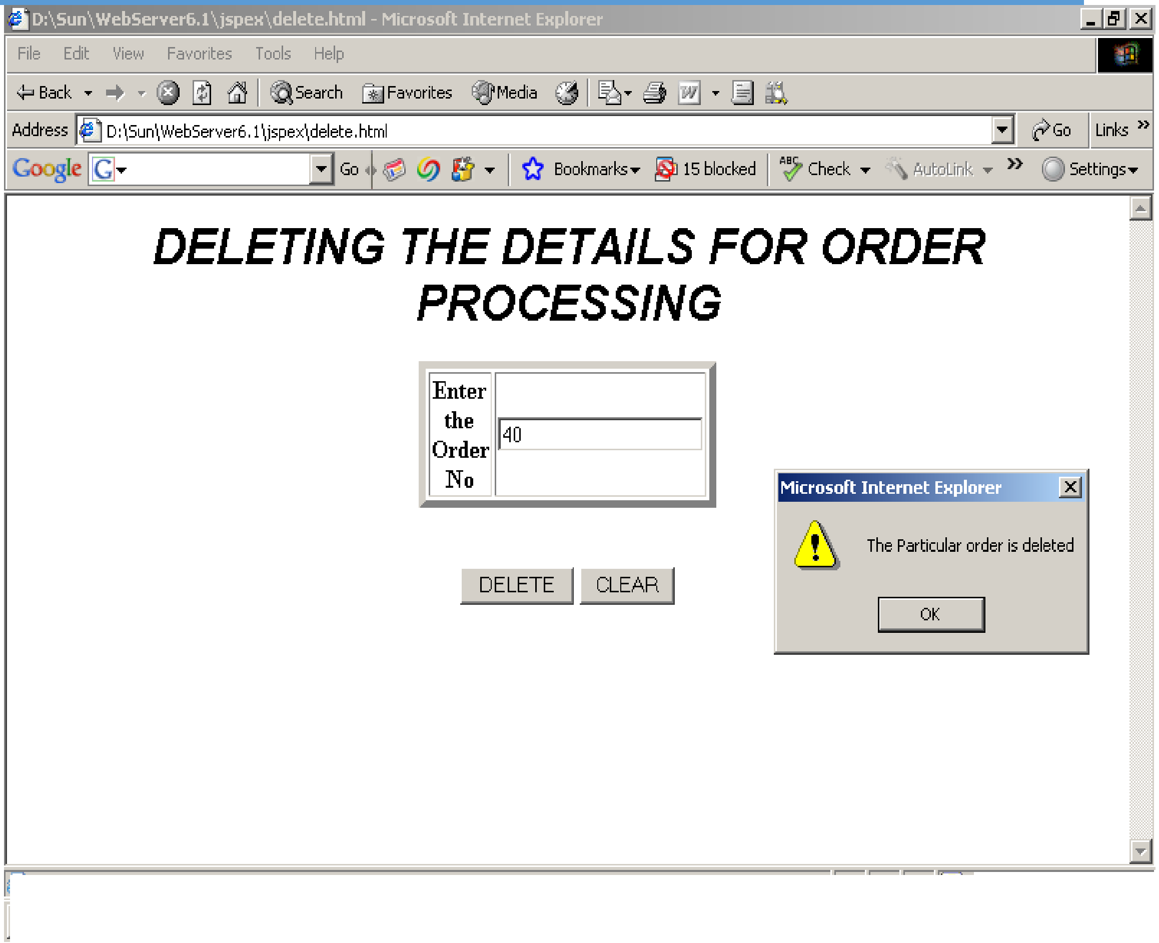
Adding Details
Deleting Details
Display all details



ADDING THE DETAILS FOR ORDER PROCESSING

NAME	me
ORDER NO	2
ITEM NAME	soap
QUANTITY	200

One record is added


D:\Sun\WebServer6.1\jspex\delete.html - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Media
Address D:\Sun\WebServer6.1\jspex\delete.html
Go
Links
Google
Go
Bookmarks
15 blocked
Check
AutoLink
Settings

DELETING THE DETAILS FOR ORDER PROCESSING

Enter the Order No	40

Microsoft Internet Explorer

The Particular order is deleted