



SRI SHANMUGHA COLLEGE OF ENGINEERING AND TECHNOLOGY

(APPROVED BY AICTE, NEW DELHI & AFFILIATED TO ANNA UNIVERSITY AND ACCREDITED BY NAAC & NBA(ECE,CSE,MECH)

**Tiruchengode-Sankari Main Road, Pullipalayam, Morur(Po),
Sankari (Tk), Salem (Dt) Pin: 637 304**



RECORD NOTE BOOK

(CS8661)INTERNET PROGRAMMING LABORATORY

NAME :

REG NO :

YEAR :



SRI SHANMUGHA COLLEGE OF ENGINEERING AND TECHNOLOGY

(APPROVED BY AICTE, NEW DELHI & AFFILIATED TO ANNA UNIVERSITY AND ACCREDITED BY NAAC & NBA(ECE,CSE,MECH)

Tiruchengode-Sankari Main Road, Pullipalayam, Morur(Po),

Sankari (Tk), Salem (Dt) Pin: 637 304



RECORD NOTE BOOK

REG NO.

Certified that this is a bonafide record of Practical work done by
Mr/Ms of the
Semester Branch during the Academic year
in the Laboratory

Staff-in-charge

Head of the Department

Submitted for the Anna University Practical Examination held on.....

Internal Examiner

External Examiner

LIST OF EXPERIMENTS

Ex.No	Date	Name Of The Experiment	Page No	Staff Signature	Remarks
1		Web Page with Maps and Links			
2		Web Page with CSS			
3		Form Validation using Java Script			
4		Invoke Servlet from HTML Forms			
5		Session tracking using Hidden Fields			
6		Converting Static web pages to dynamic web pages			
7		THREE-TIER APPLICATIONS USING SERVLETS			
8		RETRIEVING USER INFORMATION FROM XML DOCUMENT			
9		Validate the form using PHP Regular Expression			
10		PHP Stores form data into Database			
11		Web Service for Storing Peoples Opinion			

Ex. No: 1	Web Page with Maps and Links
Date :	

Aim

To create a web page with following features,

- To embed a map in a web page
- To fix the hot spots in that map
- Show all the related information when the hot spots are clicked.

Algorithm

Step 1: create a file with .html extension

Step 2: Link the India map image with the tag

Step 3: Use <area> and <map> tag for creating the links

Step 4: Link the appropriate page with the image segregation

Step 4: Load the web page in browser to see the output.

Program

ImageMap.html

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Image Map</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
 <map name="metroid" id="metroid">
```

```
<area href='TamilNadu.html' shape='circle' coords='175,495,30'
```

```
title='TamilNadu'/>
```

```
<area href = "Karnataka.html" shape = "rect" coords = "100,400,150,450" title =
```

```
"Karnataka" /> <area href = "AndhraPradesh.html" shape = "poly" coords = "150,
```

```
415, 175,348,265,360,190,420,190,440" title = "Andhra Pradesh" />
```

```
<area href = "Kerala.html" shape = "poly" coords =
```

```
"108,455,150,515,115,490,148,495,110,448,155,501"
```

```
title = "Kerala" /> </map>
```

```
</BODY>
```

```
</HTML>
```

TamilNadu.html

```
<HTML>
<HEAD>
<TITLE>About Tamil Nadu</TITLE>
</HEAD>
<BODY>
<CENTER><H1>Tamil Nadu</H1></CENTER> <HR>
<UL>
    <LI>Area : 1,30,058 Sq. Kms.</LI>
    <LI>Capital : Chennai</LI>
    <LI>Language : Tamil</LI>
    <LI>Population : 6,21,10,839</LI> </UL><hr>
<a href='ImageMap.html'>India Map</a>
</BODY>
</HTML>
```

Karnataka.html

```
<HTML>
<HEAD>
<TITLE>About Karnataka</TITLE> </HEAD>
<BODY>
<CENTER><H1>Karnataka</H1></CENTER>
<HR>
<UL>
    <LI>Area : 1,91,791 Sq. Kms</LI>
    <LI>Capital : Bangalore</LI>
    <LI>Language : Kannada</LI>
    <LI>Population : 5,27,33,958</LI>
</UL>
<hr>
<a href='ImageMap.html'>India Map</a>
</BODY>
</HTML>
```

AndhraPradesh.html

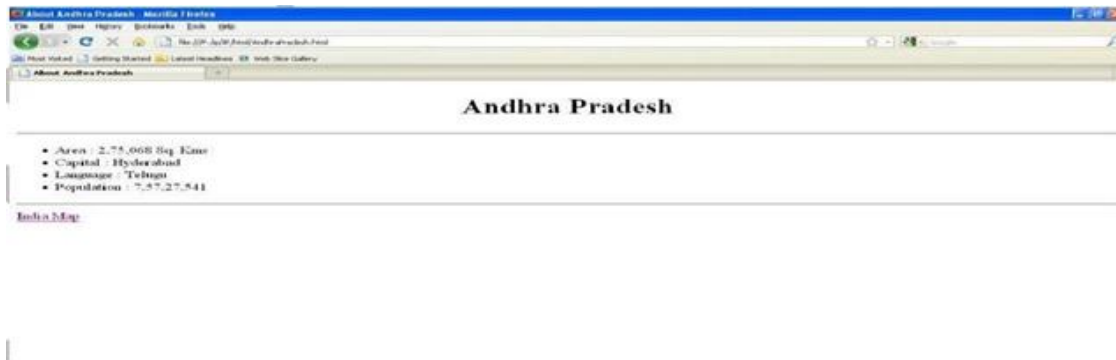
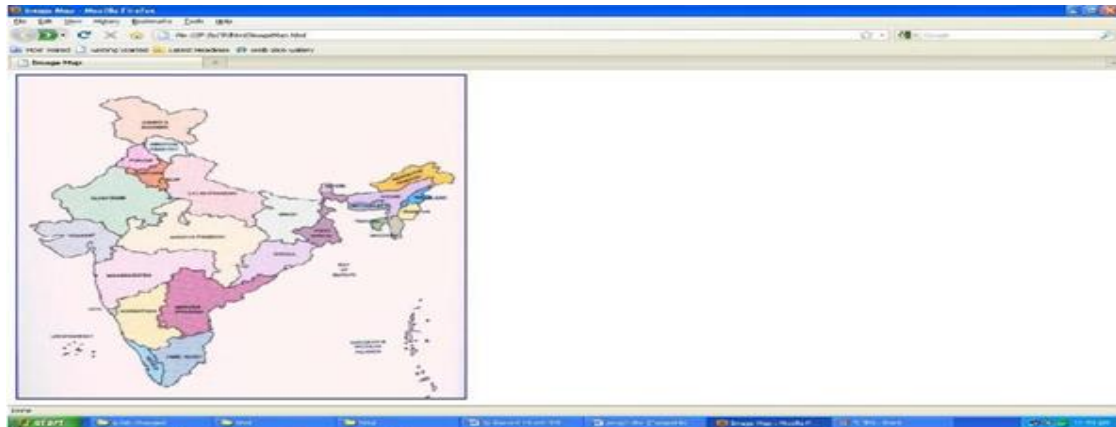
```
<HTML>
<HEAD>
<TITLE>About Andhra Pradesh</TITLE> </HEAD>
<BODY>
<CENTER><H1>Andhra Pradesh</H1></CENTER> <HR>
<UL>
```

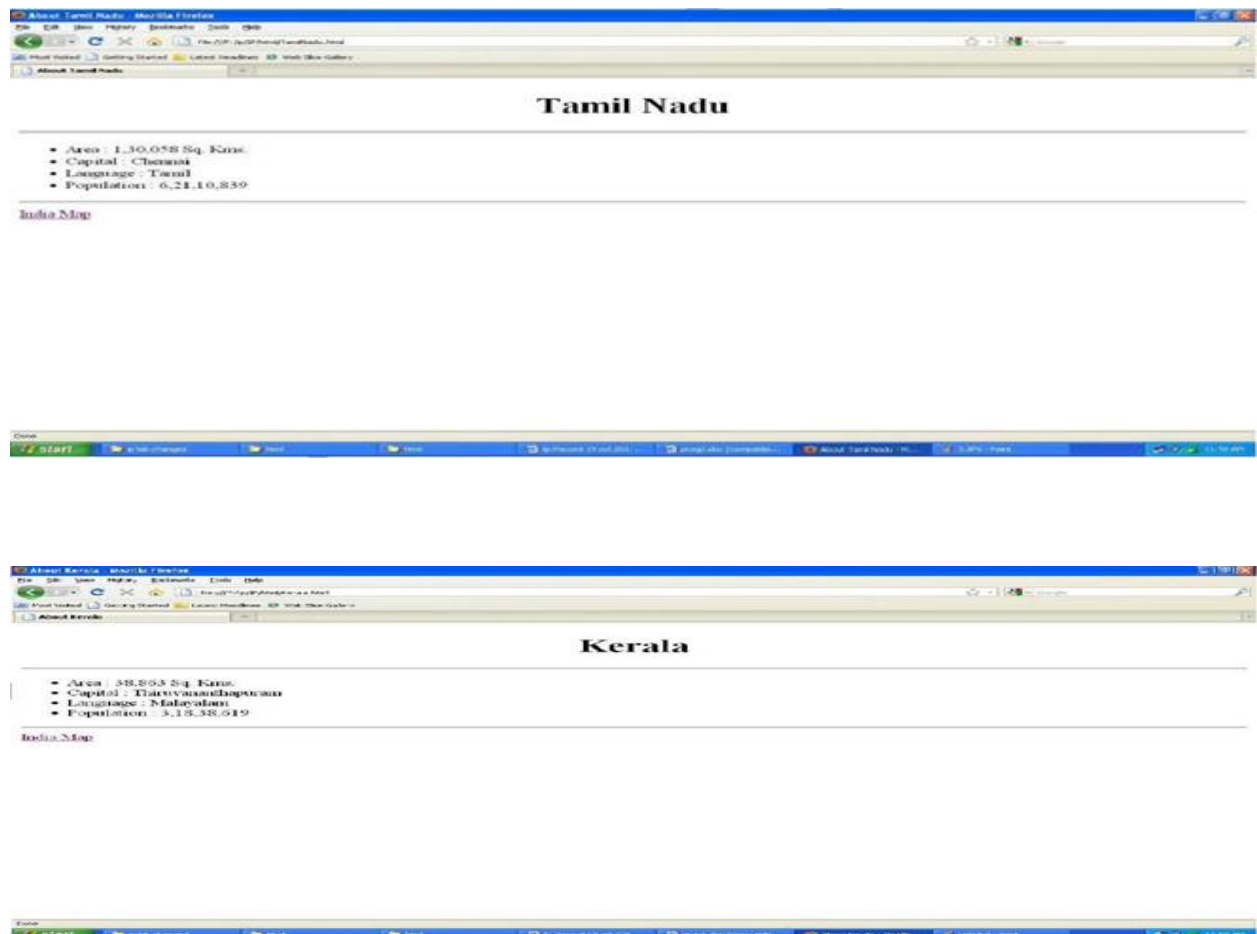
```
<LI>Area : 2,75,068 Sq. Kms</LI>
<LI>Capital : Hyderabad</LI>
<LI>Language : Telugu</LI>
<LI>Population : 7,57,27,541</LI>
</UL>
<hr>
<a href='ImageMap.html'>India Map</a>
</BODY>
</HTML>
```

Kerala.html

```
<HTML>
<HEAD>
<TITLE>About Kerala</TITLE>
</HEAD>
<BODY>
<CENTER>
<H1>Kerala</H1></CENTER>
<HR>
<UL>
<LI>Area : 38,863 Sq. Kms.</LI>
<LI>Capital : Thiruvananthapuram</LI>
<LI>Language : Malayalam</LI>
<LI>Population : 3,18,38,619</LI>
</UL>
<hr>
<a href='ImageMap.html'>India Map</a>
</BODY>
</HTML>
```

Output:





Result:

Thus the creation of a web page which includes a map and display the related in-formation when a hot spot is clicked in the map was executed successfully

Ex. No.: 2	Web Page with CSS
Date :	

Aim:

To create a web page that displays college information using various style sheet

Procedure:

- Create a web page with frame sets consisting two frames
- In the first frame include the links
- In the second frame set display the web page of the link
- Create a external style sheets
- Create a embedded style sheets
- Create a inline and internal style sheets and make it link to the external style sheets

Program:

CSS CODE:

External.css

```
h3{ font-family:arial;font-size:20;color:cyan}
table{border-color:green}
td{ font-size:20pt;color:magenta}
```

HTML CODE:

Style.html

```
<html>
<head><h1><center>ALL STYLE SHEETS</center></h1>
<title>USE of STYLESHEETS
</title>
<link rel="stylesheet" href="External.css" type="text/css"> <!-- External Style
Sheet -->
<style type="text/css">                                <!-- Internal Style Sheet -
->
.S1{font-family:verdana; font-style:italic; color:red; text-align:center}
.S2{font-family:tahoma; font-style:italic; font-size:20; text-align:center;}
font{font-family:georgia; color:blue; font-size:20}
ul{list-style-type:circle}
</style>
</head>
```

```

<body>
<ol style="list-style-type:lower-alpha">
<b> Anna University </b><br><br><br>
<li> University College of Engineering, Nagercoil
<li> University College of Engineering, Nellai
<li> University College of Engineering, Tuticorin
</ol>

<p style="font-size:20pt;color:purple">
  University College of Engineering</p>      <!-- InlineStyle Sheet -->
<p class="S2"> Run by Anna University, Chennai<br>
It is approved by AICTE.
<br>
</p>

<h2 class="S1"> University College of Engineering</h2>
<br>
<font>Located in Konam, Nagercoil</font><br>
<br>
<font>
<h2>List of Courses offered</h2>
<ul>
<li>CSE</li>
<li>IT</li>
<li>ECE</li>
<li>EEE</li>
<li>MECH</li>
<li>Civil</li>
</ul>
</font>
<h3>Pass percentage in year 2015</h3>
<table width="100%" cellpadding="2" cellspacing="2" border="5">
<tr>
<th>Sl.No</th>
<th>Dept</th>
<th>Pass Percentage</th>
</tr>
<tr>
<td align="center">1</td>

```

```

<td align="center">CSE</td>
<td align="center">80</td>
</tr>
<tr>
<td align="center">2</td>
<td align="center">ECE</td>
<td align="center">78</td>
</tr>
<tr>
<td align="center">3</td>
<td align="center">Mech</td>
<td align="center">75</td>
</tr>

</table>
</body>
</html>

```

Output:

USE of STYLESHEETS

file:///C:/Users/SK%20Singh/Desktop/CSS/style.html

ALL STYLE SHEETS

Anna University

a. University College of Engineering, Nagercoil
b. University College of Engineering, Nellore
c. University College of Engineering, Tuticorin

University College of Engineering

Run by Anna University, Chennai
It is approved by AICTE(All India Council for Technical Education). It is affiliated to Anna University.

University College of Engineering

Located in Konam, Nagercoil

List of Courses offered

- CSE
- IT
- ECE
- EEE
- MECH
- Civil

Pass percentage in year 2015

SLNo	Dept	Pass Percentage
1	CSE	80
2	ECE	78
3	Mech	75

Windows taskbar: 12:47 PM 7/18/2016

Result:

Thus the creation of a web page that displays college information using various style sheet was successfully executed and verified.

Ex. No.: 3	Form Validation using Java Script
Date :	

Aim

To validate a HTML form using java script and react to the user accordingly.

Algorithm

Program

Step 1: Create a form with text field and submit button.

Step 2: Validate the form with java script

Step 2.1: Write a javascript to call the function and to check the value of text field is empty, if empty return false for the form submission or else true.

Step 2.2: Check the entered text is a number or not. If it is number then display “Input ok” or else display “Input not valid”

Step 2.3: Create auto validation code with form.

Step 3: Run the code in browser and see the output.

CheckingNull.html

<!DOCTYPE html>

<html>

<head>

<script>

function validateForm() {

var x = document.forms["myForm"]["fname"].value;

if (x == "") {

alert("Name must be filled out");

return false;

}

}

`</script>`

`</head>`

`<body>`

`<form name="myForm" action="/action_page.php" onsubmit="return validateForm()" method="post">`

`Name: <input type="text" name="fname">`

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

`</form>`

`</body>`

`</html>`

CheckingNumericInput.html

`<!DOCTYPE html>`

`<html>`

`<body>`

`<h2>JavaScript Can Validate Input</h2>`

`<p>Please input a number between 1 and 10:</p>`

`<input id="numb">`

`<button type="button" onclick="myFunction()">Submit</button>`

`<p id="demo"></p>`

`<script>`

`function myFunction() {`

`var x, text;`

```
// Get the value of the input field with id="numb"

x = document.getElementById("numb").value;

// If x is Not a Number or less than one or greater than 10

if (isNaN(x) || x < 1 || x > 10) {

    text = "Input not valid";

} else {

    text = "Input OK";

}

document.getElementById("demo").innerHTML = text;

}

</script>

</body>

</html>
```

AutomaticValid.html

```
<!DOCTYPE html>
<html>
<body>
<form action="/action_page.php" method="post">
  <input type="text" name="fname" required>
  <input type="submit" value="Submit">
</form>
<p>If you click submit, without filling out the text field,
your browser will display an error message.</p>
</body>
</html>
```

Output:

The screenshot shows a web browser window with the URL `w3schools.com/js/tryit.asp?filename=tryjs_functions`. A modal alert box is displayed in the center, stating "An embedded page on this page says Name must be filled out" with an "OK" button. In the background, a form titled "Name:" is visible with a text input field and a "Submit" button. The code editor on the left shows the following HTML and JavaScript:

```
<!DOCTYPE html>
<html>
<head>
<script>
function validateForm() {
  var x = document.forms["myForm"]["fname"].value;
  if (x == "") {
    alert("Name must be filled out");
    return false;
  }
}
</script>
</head>
<body>

<form name="myForm" action="/action_page.php" onsubmit="return validateForm()"
method="post">
  Name: <input type="text" name="fname">
  <input type="submit" value="Submit">
</form>

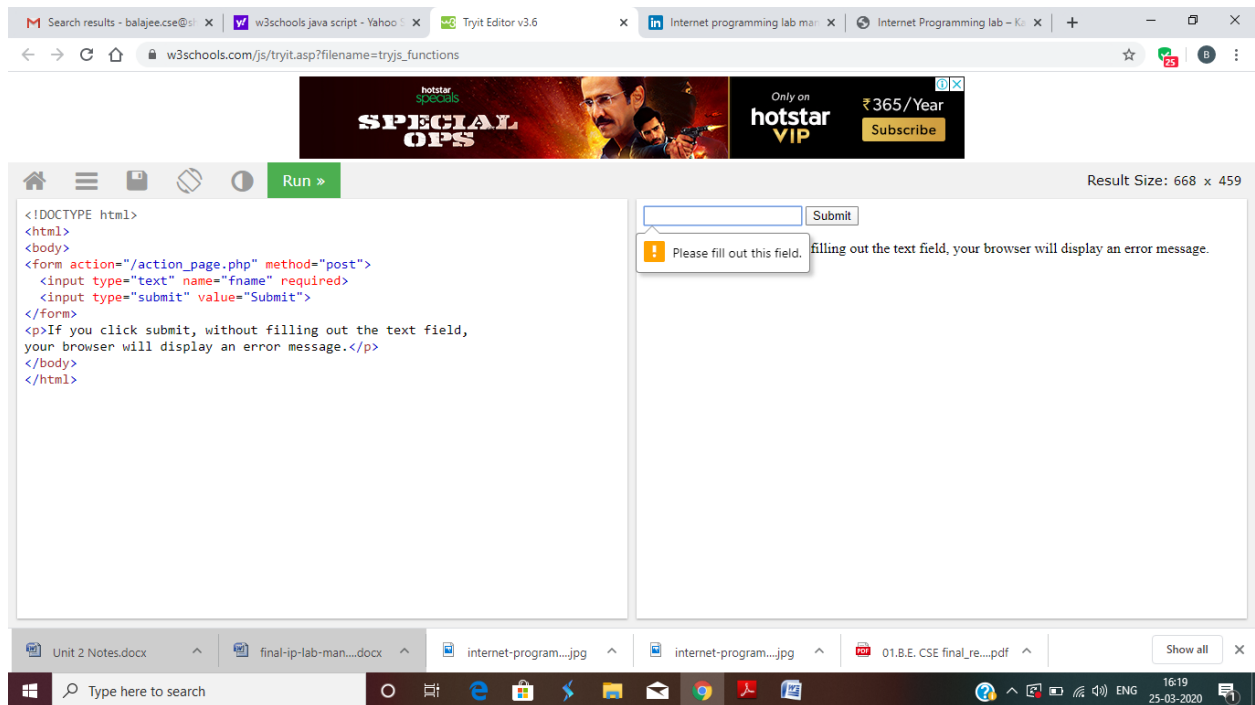
</body>
</html>
```

The Windows taskbar at the bottom shows the search bar and various application icons. The system clock indicates 16:14 on 25-03-2020.

The screenshot shows the same web browser window, but the alert box is gone. The form now displays the result of the validation. The code editor on the left shows the following HTML and JavaScript:

```
<!DOCTYPE html>
<html>
<body>
<h2>JavaScript Can Validate Input</h2>
<p>Please input a number between 1 and 10:</p>
<input id="numb">
<button type="button" onclick="myFunction()">Submit</button>
<p id="demo"></p>
<script>
function myFunction() {
  var x, text;
  // Get the value of the input field with id="numb"
  x = document.getElementById("numb").value;
  // If x is Not a Number or less than one or greater than 10
  if (isNaN(x) || x < 1 || x > 10) {
    text = "Input not valid";
  } else {
    text = "Input OK";
  }
  document.getElementById("demo").innerHTML = text;
}
</script>
</body>
</html>
```

The form output shows "Please input a number between 1 and 10:" followed by an input field containing the letter 'a' and a "Submit" button. Below the input field, the text "Input not valid" is displayed. The Windows taskbar at the bottom shows the search bar and various application icons. The system clock indicates 16:17 on 25-03-2020.



Result:

Thus the form validation to check null value and alphabetic character using java script is done successfully.

Ex. No: 4	Invoke Servlet from HTML Forms
Date:	

Aim

To write a HTML script, web.xml mapping with Java servlet class file to invoke servlet from html forms.

Algorithm

Step 1: Create a HTML file with forms to submit

Step 2: Create a java file with the name “HelloForm.java” and compile it for creating the class file “HelloForm.class”

Step 3: Write a web.xml file for mapping the URL pattern with the java class file.

Step 4: Run the apache tomcat for making the system as a server

Step 5: Run” localhost:8080” in browser and check the tomcat running

Step 6: Run html file and submit the form to see the results of java class file.

Program

Form.html

```
<html>
<body>
  <form action = "HelloForm" method = "GET">
    First Name: <input type = "text" name = "first_name">
    <br />
    Last Name: <input type = "text" name = "last_name" />
    <input type = "submit" value = "Submit" />
  </form>
</body>
</html>
```

HelloForm.java

```
// Import required java libraries
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

// Extend HttpServlet class
public class HelloForm extends HttpServlet {
```

```

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

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

    PrintWriter out = response.getWriter();
    String title = "Using GET Method to Read Form Data";
    String docType =
        "<!doctype html public \"-//w3c//dtd html 4.0 \" + \"transitional//en\">\n";

    out.println(docType +
        "<html>\n" +
        "<head><title>" + title + "</title></head>\n" +
        "<body bgcolor = \"#f0f0f0\">\n" +
        "<h1 align = \"center\">" + title + "</h1>\n" +
        "<ul>\n" +
        "  <li><b>First Name</b>: "
        + request.getParameter("first_name") + "\n" +
        "  <li><b>Last Name</b>: "
        + request.getParameter("last_name") + "\n" +
        "</ul>\n" +
        "</body>" +
        "</html>"
    );
}
}

```

Web.xml

```

<servlet>
  <servlet-name>HelloForm</servlet-name>
  <servlet-class>HelloForm</servlet-class>
</servlet>

<servlet-mapping>
  <servlet-name>HelloForm</servlet-name>
  <url-pattern>/HelloForm</url-pattern>
</servlet-mapping>

```

Output

Html Form

First Name: Last Name:

Java File

Using GET Method to Read Form Data

First Name: ZARA

Last Name: ALI

Result:

Thus the java servlet output is successfully shown in the browser by invoking it from the html forms.

Ex. No.: 5	Session tracking using Hidden Fields
Date :	

Aim:

To perform session tracking using hidden form fields.

Procedure:

1. Create a html file which contains user information.
2. The first servlet will receive these values.
3. First servlet stores them in the hidden form fields.
4. The second servlet will display the user information with greeting message.

Program:

HTML CODE

```
<html>
<head>
<title>Session Tracking Using Hidden Form Field</title>
</head>
<body>
<form action="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

```
import java.io.*;
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=\"Servlet2\">");
        out.print("<input type='hidden' name='uname' value="+n+">");
        out.print("<input type='submit' value='Just click here'>");
        out.print("</form>");
        out.close();
    }
    catch(Exception e)
    {
        System.out.println(e);
    }
}
}

```

SERVLET2

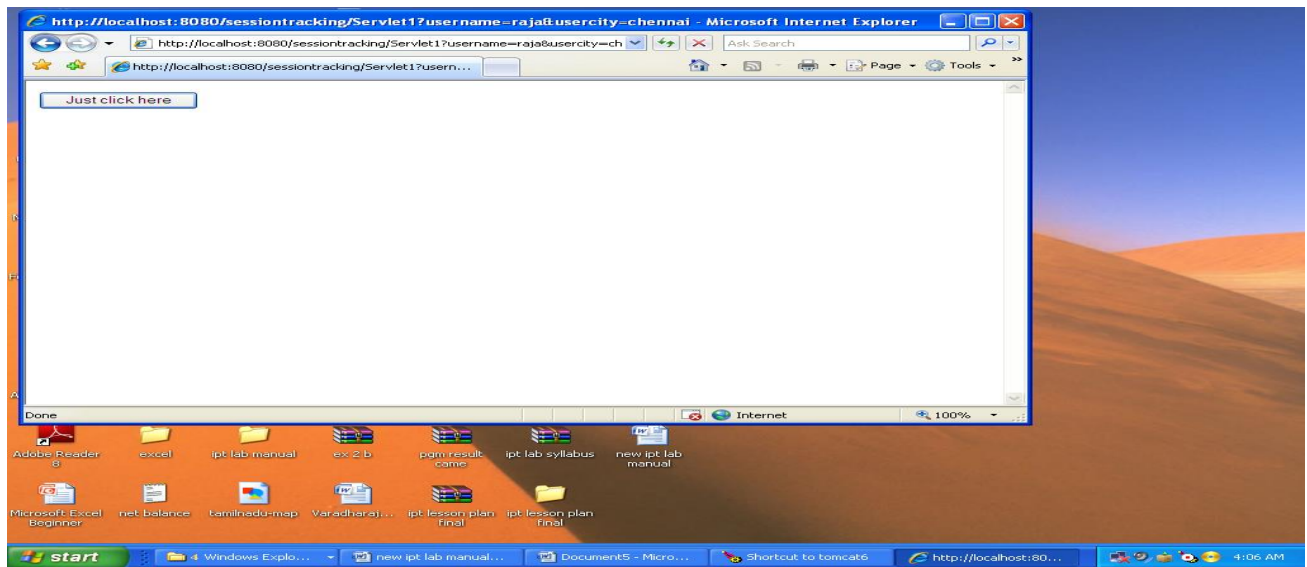
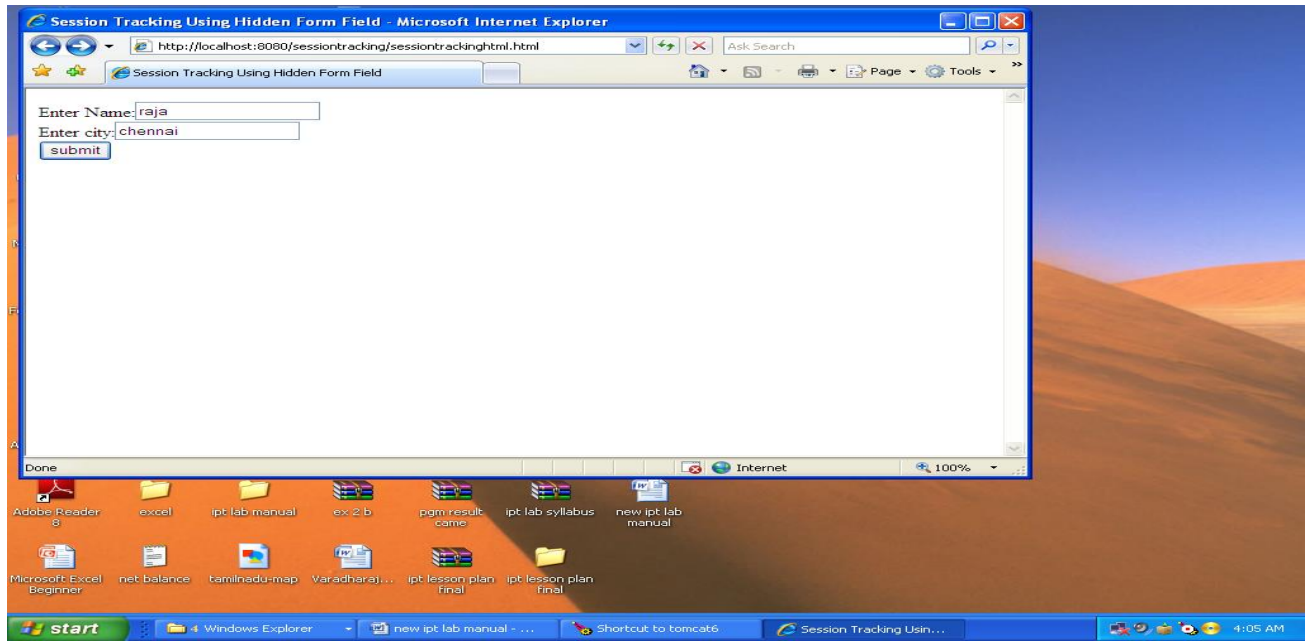
```

import java.io.*;
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("uname");
            out.println("Hello"+n);
            String c=request.getParameter("usercity");
            out.print("you are from"+c);
            out.close();
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}

```

Output:



Result:

Thus the Java Program for Session Tracking Using Hidden Form Fields has been executed successfully.

Ex. No: 6	Converting Static web pages to dynamic web pages
Date :	

Aim:

To convert the static web pages into dynamic web pages using servlets (or JSP) and cookies.

Procedure:

step1: we will create ahtml form for entering the user name,password and card ID.

Step2:From the above HTML form, the servlet program is invoked in which the validity of the user name,password and card id is 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 we set the cookies in which the current user name is stored.

Step3: compile the above servlet Login servlet.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.We must store the user information such as user name,password and card id in the web.xml using init-param.

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

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

Then edit the web.xml in WEB-INF folder.

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

Index.jsp

```
<html>
<head>
<body>
<form action="http://localhost:8084/ddd/LoginServlet" method="post">
Enter username:
```

```

<input type="text" value="" name="user">
<br>
Enter Password:
<input type="password" value="" name="password">
<br>
Enter Card ID:
<input type="text" value="" name="cardID">
<br>
<br> <br> <br>
<input type="submit" value="login">
</form>
</body>

```

Loginservlet.html

```

import java.io.*;
import java.net.*;

import javax.servlet.*;
import javax.servlet.http.*;

public class LoginServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {

            String usr=request.getParameter("user");
            String pwd=request.getParameter("password");
            String card=request.getParameter("cardID");
            boolean flag=true;

            String[] userID=getInitParameter("usernames").split(",");
            String[] password=getInitParameter("passwords").split(",");
            String[] cardids=getInitParameter("cardIDs").split(",");

```



```

        int i;
        for(i=0;i<userID.length;i++)
        {

if(userID[i].equals(usr)&&password[i].equals(pwd)&&cardids[i].equals(card))
        {
            flag=false;
            Cookie MyCookie=new Cookie("CurrentUser", usr);
            MyCookie.setMaxAge(60*60);
            response.addCookie(MyCookie);
            response.sendRedirect("http://localhost:8084/ddd/LoginSuccess");
        }
    }

        if(flag==true)
        {
            out.print("Error");
            out.println("<h4>Invalid user,please try again by clicking following
link</h4>");
            out.println("<a href='http://localhost:8084/ddd/'>"+ "LoginForm.html");
        }
    }
finally {
    out.close();
}
}

```

LoginSuccess.java

```

import java.io.*;
import java.net.*;

import javax.servlet.*;
import javax.servlet.http.*;

public class LoginSuccess extends HttpServlet {protected void
doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

```

```

Cookie[] my_cookies=request.getCookies();
response.setContentType("text/html");
PrintWriter out=response.getWriter();
out.print("Login Success");
out.println("<b>");
String userName=null;
if(my_cookies!=null)
{
    for(Cookie cookie:my_cookies)
    {
        if(cookie.getName().equals("currentUser"))
            userName=cookie.getValue();
    }
}
out.print("<h3>Login Success!!!Welcome</h3>");
out.print("<h2>This is a Shopping cart for"+userName+"</h2>");
out.close();

}
}

```

Web.xml

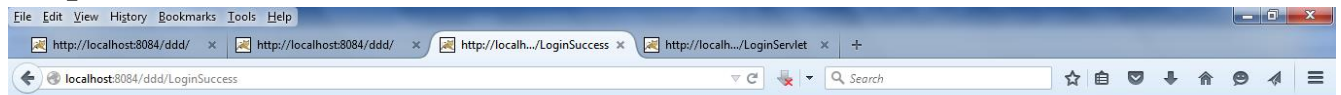
```

<servlet>
<servlet-name>LoginServlet</servlet-name>
<servlet-class>LoginServlet</servlet-class>
<init-param>
    <param-name>usernames</param-name>
    <param-value>user1,user2,user3</param-value>
</init-param>
<init-param>
    <param-name>passwords</param-name>
    <param-value>pwd1,pwd2,pwd3</param-value>
</init-param>
<init-param>
    <param-name>cardIDs</param-name>

```

```
        <param-value>111,222,333</param-value>
    </init-param>
</servlet>
<servlet>
    <servlet-name>LoginSuccess</servlet-name>
    <servlet-class>LoginSuccess</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>LoginServlet</servlet-name>
    <url-pattern>/LoginServlet</url-pattern>
</servlet-mapping>
<servlet-mapping>
    <servlet-name>LoginSuccess</servlet-name>
    <url-pattern>/LoginSuccess</url-pattern>
</servlet-mapping>
<session-config>
    <session-timeout>
        30
    </session-timeout>
</session-config>
<welcome-file-list>
    <welcome-file>index.jsp</welcome-file>
</welcome-file-list>
</web-app>
```

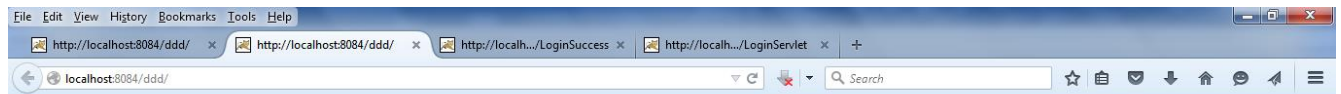
Output:



Login Success

Login Success!!!Welcome

This is a Shopping cart fornull



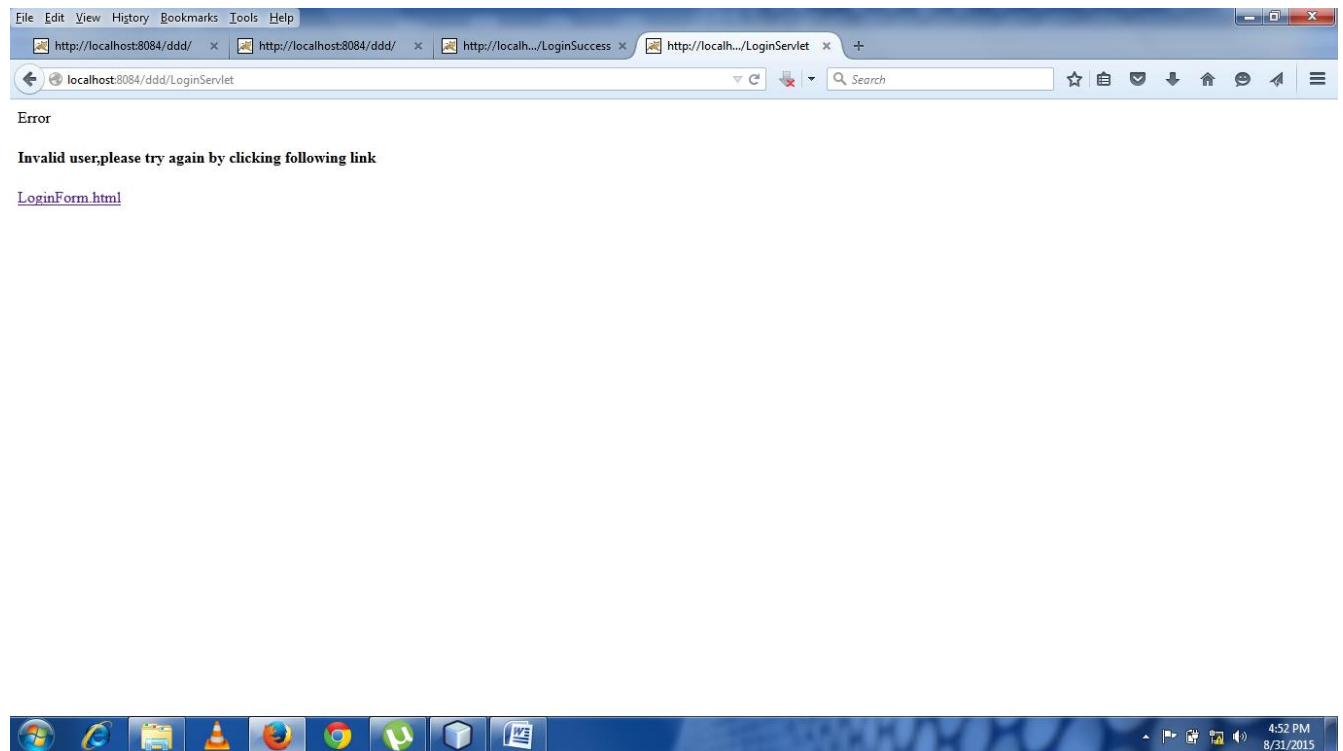
Enter username: user1

Enter Password: ****

Enter Card ID: 222

login





Result:

Thus the conversion of the static web pages into dynamic web pages using servlets cookies has been executed successfully.

Ex. No: 7	THREE-TIER APPLICATIONS USING SERVLETS
Date :	

Aim:

To write java servlet programs to conduct online examination and to display student mark list available in a database

Procedure:

Client:

- In index.html on the client side declare the contents that you like to transfer to the server using html form and input type tags.
- Create a submit button and close all the included tags.

Server:

- Import all necessary packages
- Define a class that extends servlet
- In the doPost() method, do the following:
 - i) Set the content type of the response to "text/html"
 - ii) Create a writer to the response
 - iii) Get a parameter from the request
 - iv) If its value is equal to right answer then add 5 to mark variable
 - v) Similarly repeat step
 - vi) for all parameters
 - vii) Display the result in an html format using the writer

Student Mark List Database:

- Import necessary to java packages and javax packages and classes
- Create a class that extends HttpServlet and implements ServletException and IOException
- In the doGet() method, do the following:
 - i) Create a PrintWriter object
 - ii) Open a connection with the data source name
 - iii) Write a sql query and execute to get the resultset
 - iv) Display the resultset information in html form

Program:

SERVLET CODE:

```
import java.io.*;
import java.sql.*;
```

```

import javax.servlet.*;
import javax.servlet.http.*;
public class StudentServlet3 extends HttpServlet
{
String message,Seat_no,Name,ans1,ans2,ans3,ans4,ans5;
int Total=0;
Connection connect;
Statement stmt=null;
ResultSet rs=null;
public void doPost(HttpServletRequest request,HttpServletResponse response)
throws ServletException,IOException
{
try
{
String url="jdbc:odbc:NEO";
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
connect=DriverManager.getConnection(url," "," ");
message="Thank you for participating in online Exam";
}

catch(ClassNotFoundException cnfex)
{ cnfex.printStackTrace();}
catch(SQLException sqlex)
{ sqlex.printStackTrace();}
catch(Exception excp)
{ excp.printStackTrace();}
Seat_no=request.getParameter("Seat_no");
Name=request.getParameter("Name");
ans1=request.getParameter("group1");
ans2=request.getParameter("group2");
ans3=request.getParameter("group3");
ans4=request.getParameter("group4");
ans5=request.getParameter("group5"); if(ans1.equals("True"))
Total+=2;
if(ans2.equals("False"))
Total+=2;
if(ans3.equals("True"))
Total+=2;
if(ans4.equals("False"))

```

```

Total+=2;
if(ans5.equals("False"))
Total+=2; try
{
Statement stmt=connect.createStatement();
String query="INSERT INTO student("+"Seat_no,Name,Total"+" ) VALUES
("+"Seat_no+"","+"Name+"","+"Total+"");
int result=stmt.executeUpdate(query);
stmt.close();
}
catch(SQLException ex)
{}
response.setContentType("text/html");
PrintWriter out=response.getWriter();
out.println("<html>");
out.println("<head>");
out.println("</head>");
out.println("<body bgcolor=cyan>");
out.println("<center>");
out.println("<h1>"+message+"</h1>\n");
out.println("<h3>Yours results stored in our database</h3>");
out.print("<br><br>");
out.println("<b>"+ "Participants and their Marks"+"</b>");
out.println("<table border=5>");
try
{
Statement stmt=connect.createStatement();
String query="SELECT * FROM student";
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.print("<td>"+rs.getInt(1)+"</td>");
out.print("<td>"+rs.getString(2)+"</td>");
out.print("<td>"+rs.getString(3)+"</td>");
out.println("</tr>");
}
}

```



```

out.println("</table>");
}
catch(SQLException ex){ } finally
{
try
{
if(rs!=null)
rs.close();
if(stmt!=null)
stmt.close();
if(connect!=null)
connect.close();
}
catch(SQLException e){ }
}
out.println("</center>");
out.println("</body></html>");
Total=0;
} }

```

HTML CODE:

```

<html>
<head><title>Database Test</title></head>
<body>
<center><h1>Online Examination</h1> </center>
<form action="StudentServlet3.view" method="POST">
  <div align="left"><br></div>
  <b>Seat Number:</b> <input type="text" name="Seat_no">
  <div align="Right">
  <b>Name:</b> <input type="text" name="Name" size="50"><br>
  </div>
  <br><br>
  <b>1. Every host implements transport layer.</b><br>
  <input type="radio" name="group1" value="True">True
  <input type="radio" name="group1" value="False">False<br>
  <b>2. It is a network layer's responsibility to forward packets reliably from source
  to destina-tion</b><br>
  <input type="radio" name="group2" value="True">True
  <input type="radio" name="group2" value="False">False<br>
  <b>3. Packet switching is more useful in bursty traffic</b><br>

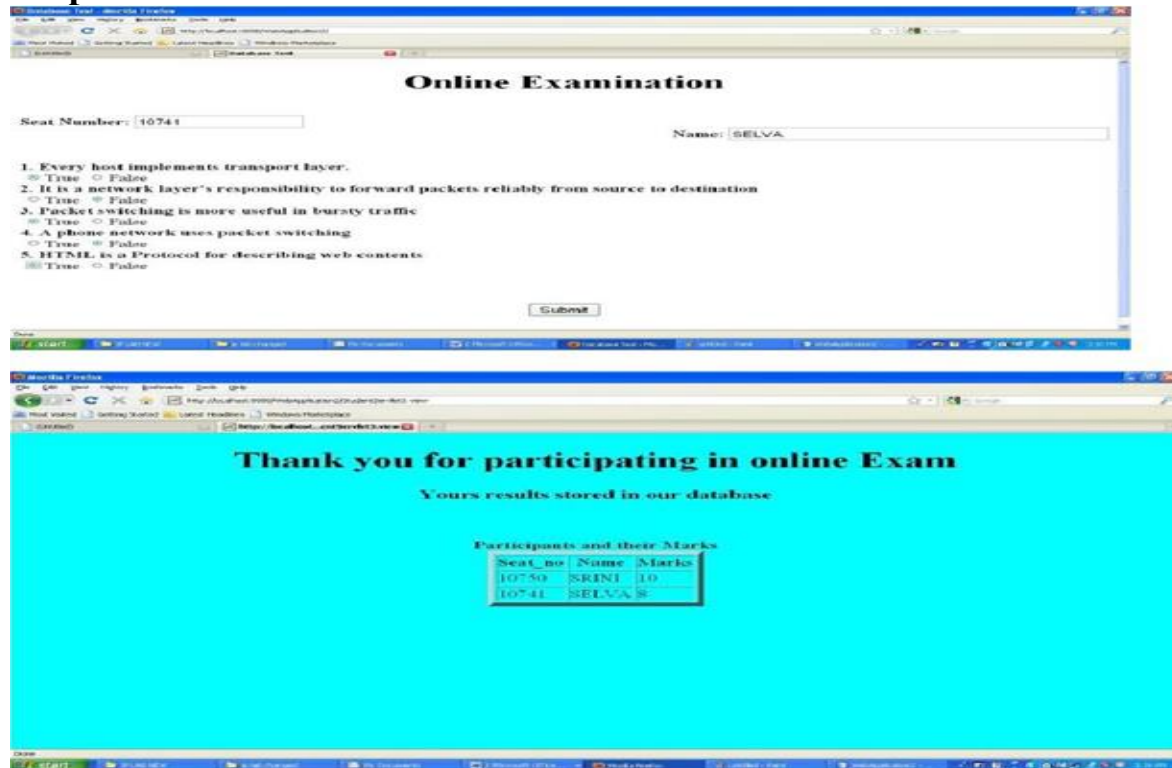
```

```

<input type="radio" name="group3" value="True">True
<input type="radio" name="group3" value="False">False<br>
<b>4. A phone network uses packet switching</b><br>
<input type="radio" name="group4" value="True">True
<input type="radio" name="group4" value="False">False<br>
<b>5. HTML is a Protocol for describing web contents</b><br>
<input type="radio" name="group5" value="True">True
<input type="radio" name="group5" value="False">False<br> <br><br><br>
<center>
<input type="submit" value="Submit"><br><br> </center>
</form>
</body>
</html>

```

Output:



Result:

Thus to write java servlet programs to conduct online examination and to display student mark list available in a database was successfully executed and verified.

Ex. No: 8	RETRIEVING USER INFORMATION FROM XML DOCUMENT
Date:	

Aim:

To create and save an XML document at the server, which contain ten users information. 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.

Procedure:

- Save Students information in the XML file on the specific location.
- Create and establish the connection between html file and XML file.
- Get the user ID as input
- Display the student's information.

StudentDetails.xml :

```
<?xml version="1.0" encoding="UTF-8"?>
<Student>
<PersonDetails>
  <id>101</id>
  <name>Anand</name>
  <city>Madurai</city>
  <Branch>CSE</Branch>
  <Year>I</Year>
</PersonDetails>
<PersonDetails>
  <id>102</id>
  <name>Anu</name>
  <city>Konam</city>
  <Branch>CSE</Branch>
  <Year>II</Year>
</PersonDetails>
<PersonDetails>
  <id>103</id>
  <name>Archana</name>
  <city>Madurai</city>
  <Branch>CSE</Branch>
  <Year>I</Year>
</PersonDetails>
<PersonDetails>
```

```
<id>104</id>
<name>Monica</name>
<city>Nellai</city>
<Branch>CSE</Branch>
<Year>III</Year>
</PersonDetails>
</Student>
```

LogIn.html :

```
<!DOCTYPE html>

<html>

<head>

</head>

<body>

<script type="text/javascript">

function Display()

{

if(window.XMLHttpRequest)

{

    xmlhttp=new XMLHttpRequest();

}

xmlhttp.open("GET","UserInfo.xml",false);

xmlhttp.send();

xmlDoc=xmlhttp.responseXML;

var x=xmlDoc.getElementsByTagName("PersonDetails");

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> <hr> Registration ID=");
document.write(x[j].getElementsByTagName("id")[0].childNodes[0].nodeValue);
document.write("</br> Name=");

document.write(x[j].getElementsByTagName("name")[0].childNodes[0].nodeValue);
document.write("</br> City=");

document.write(x[j].getElementsByTagName("city")[0].childNodes[0].nodeValue);
;
document.write("</br> Branch=");

document.write(x[j].getElementsByTagName("Branch")[0].childNodes[0].nodeValue);
document.write("</br> Year=");

document.write(x[j].getElementsByTagName("Year")[0].childNodes[0].nodeValue);
document.write("</br> ");    }

</script>

<form name='myform'>

Enter ID:

<input type='text' id='key'/><br/>

```

```
<input type='button' value='submit' onclick='Display()'/>
```

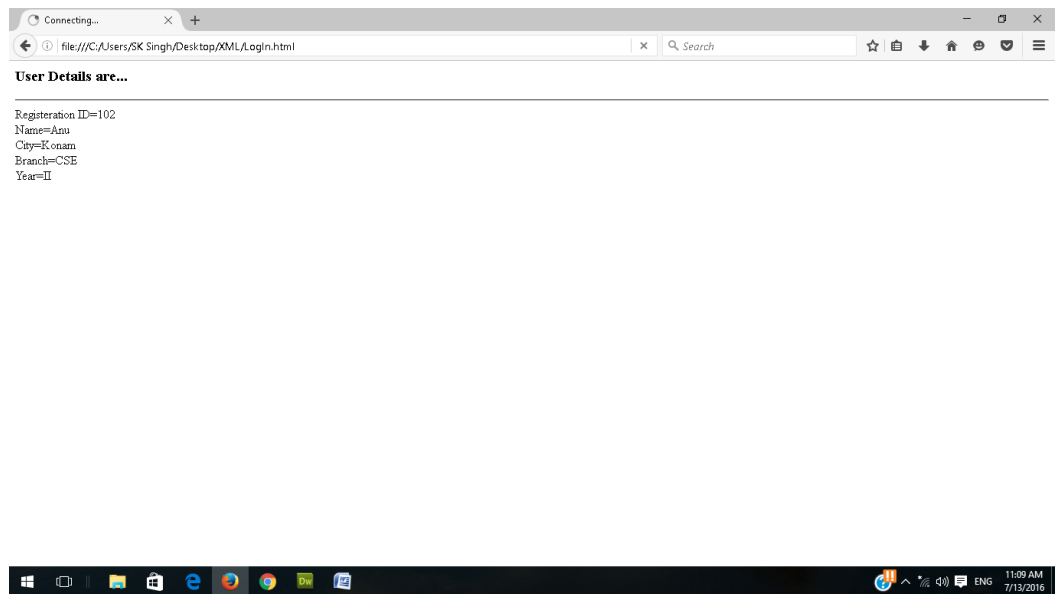
```
</form>
```

```
</body>
```

```
</html>
```

Output:





Result:

Thus the Program takes user id as an input and returns the user details by taking the user information from the XML document has been executed successfully.

Ex. No: 9	Validate the form using PHP Regular Expression
Date:	

Aim

To create a HTML form to validate with PHP regular expression and to say it is valid or not.

Algorithm

Step 1: Create a HTML file with following input fields,

Name, Email, Website, Comment, Gender and Submit button

Step 2: Write php functions with regular expression to validate the form

Step 3: The name test box should allow only letters and white space

Step 4: Email should contain @ and . symbols

Step 5: Web site should be cross checked for URL format

Step 6: Comment is optional

Step 7: Gender is mandatory

Step 8: Execute and see the result

Program

Validate.php

```

<!DOCTYPE HTML>
<html>
<head>
<style>
.error {color: #FF0000;}
</style>
</head>
<body>

```



```

<?php
// define variables and set to empty values
$nameErr = $emailErr = $genderErr = $websiteErr = "";
$name = $email = $gender = $comment = $website = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if (empty($_POST["name"])) {
        $nameErr = "Name is required";
    } else {
        $name = test_input($_POST["name"]);
        // check if name only contains letters and whitespace
        if (!preg_match("/^[a-zA-Z ]*$/", $name)) {
            $nameErr = "Only letters and white space allowed";
        }
    }
}

if (empty($_POST["email"])) {
    $emailErr = "Email is required";
} else {
    $email = test_input($_POST["email"]);
    // check if e-mail address is well-formed
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $emailErr = "Invalid email format";
    }
}

if (empty($_POST["website"])) {
    $website = "";
} else {
    $website = test_input($_POST["website"]);
    // check if URL address syntax is valid
    if (!preg_match("/^b(?:(:https?|ftp):\\\/|www\\.)[-a-z0-9+&@#\\/%?~_!|:,;]*[-a-z0-9+&@#\\/%?~_~_]/i", $website)) {
        $websiteErr = "Invalid URL";
    }
}

```

```

}

if (empty($_POST["comment"])) {
    $comment = "";
} else {
    $comment = test_input($_POST["comment"]);
}

if (empty($_POST["gender"])) {
    $genderErr = "Gender is required";
} else {
    $gender = test_input($_POST["gender"]);
}
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>

```

<h2>PHP Form Validation Example</h2>

<p>* required field</p>

<form method="post" action="<?php echo htmlspecialchars(\$_SERVER["PHP_S
ELF"]);?>">

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

* <?php echo \$nameErr;?>

E-mail: <input type="text" name="email">

* <?php echo \$emailErr;?>

Website: <input type="text" name="website">

<?php echo \$websiteErr;?>

Comment: <textarea name="comment" rows="5" cols="40"></textarea>

Gender:

<input type="radio" name="gender" value="female">Female

<input type="radio" name="gender" value="male">Male

<input type="radio" name="gender" value="other">Other

* <?php echo \$genderErr;?>

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

</form>

<?php

echo "<h2>Your Input:</h2>";

echo \$name;

echo "
";

echo \$email;

echo "
";

echo \$website;

echo "
";

echo \$comment;

echo "
";

echo \$gender;

?>

</body>

</html>

Output:

PHP Form Validation Example

* required field

Name: * Only letters and white space allowed

E-mail: * Invalid email format

Website: Invalid URL

Comment:

Gender: ☐ Female ☐ Male ☐ Other *

Your Input:

Testcase 1
test@
abc
abc
male

Result:

Thus the HTML form is been created and validated using PHP regular expressions.

Ex. No: 10	PHP Stores form data into Database
Date:	

Aim

To create a PHP code to retrieve the data from HTML forms and store it in database.

Algorithm

Step 1: Create a HTML page with a form

Step 2: Create a form with First Name, Last Name and email fields to submit to PHP page

Step 3: Receive the data in PHP page and create variables to store those data.

Step 4: Initialize connection variable with server name, user name and password.

Step 5: Use the existing database

Step 6: Insert the records into sql table

Step 7: Show the status of record insertion

Program

Form.html

```
<html>

<head>

<title> form page </title>

</head>

<body>

<form action = "storage.php" method = "post">

<b>Submit Data for Storage</b> </br> </br>
```

First Name: <input type = "text" name = "fn" /> </br> </br>

Last Name: <input type = "text" name = "ln" /> </br> </br>

e-mail: <input type = "text" name = "email" /> </br> </br>

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

</form>

</body>

</html>**Storage.php**

<?php

\$servername = "localhost";

\$username = "username";

\$password = "password";

\$dbname = "myDB";

\$fn = \$_POST["fn"];

\$ln = \$_POST["ln"];

\$email = \$_POST["email"];

// Create connection

\$conn = mysqli_connect(\$servername, \$username, \$password, \$dbname);

// Check connection

\$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('\$fn', '\$ln', '\$email')";

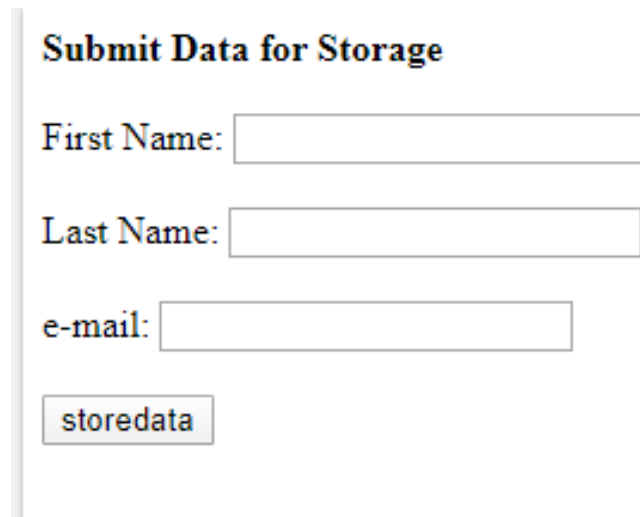
```
if (mysqli_query($conn, $sql)) {  
    echo "New record created successfully";  
} else {  
    echo "Error: " . $sql . "<br>" . mysqli_error($conn);  
}
```

```
mysqli_close($conn);
```

```
?>
```

Output:

HTML File



The screenshot shows a web form with a title "Submit Data for Storage". It contains three text input fields labeled "First Name:", "Last Name:", and "e-mail:". Below these fields is a button labeled "storedata". The form is displayed within a browser window, indicated by a vertical scrollbar on the left.

PHP File

New record created successfully

Result:

Thus the html form is created to submit the data and it is stored in database using PHP file successfully.

| | |
|------------|--|
| Ex. No: 11 | Web Service for Storing Peoples Openion |
| Date: | |

Aim:

To Write a web services for finding what people think by asking 500 people's opinion for any consumer product

Procedure:

- Open the home page.
- Enter the login ID and type the comments then submit.
- Retrieve comments with post id
- Display the comments.

Program:

Index.php

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <title>jQuery Ajax Comment System - Demo</title>
    <link rel="stylesheet" href="css/style.css">
    <script src=http://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js>
    </script>
    <script src="js/script.js"></script>
</head>
<body>
    <div class="wrap">
        <h1> Maggy Noodles Comment System</h1>
        <?php
            // retrive post
            include('config.php');
            include ('function.php');
            dbConnect();

```



```

1');
$query = mysql_query('SELECT * FROM post WHERE post_id =
$row = mysql_fetch_array($query);
?>
<div class="post">
<h2><?php echo $row['post_title']?></h2>
<p><?php echo $row['post_body']?></p>
</div>

<?php
// retrieve comments with post id
$query = mysql_query(
    "SELECT * FROM comment WHERE post_id =
    {$row['post_id']} ORDER BY comment_id DESC LIMIT 15");
?>

<h2>Comments.....</h2>
<div class="comment-block">
<?php while($comment = mysql_fetch_array($comment_query)): ?>
<div class="comment-item">
<div class="comment-avatar">

</div>
<div class="comment-post">
<h3><?php echo $comment['name'] ?> <span>said....</span></h3>
<p><?php echo $comment['comment']?></p>
</div>
</div>
<?php endwhile?>
</div>
<h2>Submit new comment</h2>
<!--comment form -->
<form id="form" method="post">
<!-- need to supply post id with hidden field -->

```

```

        <input type="hidden" name="postid" value="<?php echo
$row['post_id']?>">
        <label>
        <span>Name *</span>
        <input type="text" name="name" id="comment-name"
placeholder="Your name here...." required>
        </label>
        <label>
        <span>Email *</span>
        <input type="email" name="mail" id="comment-mail"
placeholder="Your mail here...." required>
        </label>
        <label>
        <span>Your comment *</span>
        <textarea name="comment" id="comment" cols="30" rows="10"
placeholder="Type your comment here...." required></textarea>
        </label>
        <input type="submit" id="submit" value="Submit Comment">
        </form>
    </div>
</body>
</html>

```

Ajax_Comment.php

```

<?php
if (isset( $_SERVER['HTTP_X_REQUESTED_WITH'] )):
    include('config.php');
    include('function.php');
    dbConnect();

    if (!empty($_POST['name']) AND !empty($_POST['mail']) AND
!empty($_POST['comment']) AND !empty($_POST['postid'])) {
        $name = mysql_real_escape_string($_POST['name']);
        $mail = mysql_real_escape_string($_POST['mail']);
        $comment = mysql_real_escape_string($_POST['comment']);
    }
}

```

```

$postId = mysql_real_escape_string($_POST['postid']);

mysql_query("
    INSERT INTO comment
    (name, mail, comment, post_id)
    VALUES('{ $name}', '{ $mail}', '{ $comment}', '{ $postId}'))";

}
?>

```

```

<div class="comment-item">
    <div class="comment-avatar">
        
    </div>
    <div class="comment-post">
        <h3><?php echo $name ?> <span>said....</span></h3>
        <p><?php echo $comment?></p>
    </div>
</div>

```

```

<?php
    dbConnect(0);
endif?>

```

Config.php

```

<?php
# db configuration
define('DB_HOST', 'localhost');
define('DB_USER', 'root');
define('DB_PASS', 'root');
define('DB_NAME', 'dbname');
?>

```

Function.php

```

<?php

```

```

/**
 * Connect to mysql server
 * @param bool
 * @use true to connect false to close
 */
function dbConnect($close=true){

    if (!$close) {
        mysql_close($link);
        return true;
    }
    $link = mysql_connect(DB_HOST, DB_USER, DB_PASS) or die('Could
not connect to MySQL DB ') . mysql_error();
    if (!mysql_select_db(DB_NAME, $link))
        return false;
}
/**
 * gravatar Image
 * @see http://en.gravatar.com/site/implement/images/
 */
function avatar($mail, $size = 60){
    $url = "http://www.gravatar.com/avatar/";
    $url .= md5( strtolower( trim( $mail ) ) );
    // $url .= "?d=" . urlencode( $default );
    $url .= "&s=" . $size;
    return $url;
}
?>

```

Style.CSS

```

/* general styling */
*{
    margin: 0;
    padding: 0;
    box-sizing: border-box;

```

```
-webkit-box-sizing: border-box;
-moz-box-sizing: border-box;
-webkit-font-smoothing: antialiased;
-moz-font-smoothing: antialiased;
-o-font-smoothing: antialiased;
font-smoothing: antialiased;
text-rendering: optimizeLegibility;
}
body{
  font: 12px Arial,Tahoma,Helvetica,FreeSans,sans-serif;
  text-transform: inherit;
  color: #333;
  background: #e7edee;
  width: 100%;
  text-shadow: 0 1px 1px rgba(0, 0, 0, 0.2)
}
.wrap{
  width: 720px;
  margin: 15px auto;
  padding: 15px 20px;
  background: white;
  border: 2px solid #DBDBDB;
  -webkit-border-radius: 5px;
  -moz-border-radius: 5px;
  border-radius: 5px;
  overflow: hidden;
}

a{ text-decoration: none; color: #333}
h1{
  font-family: Georgia, "Times New Roman", Times, serif;
  font-size: 2.8em;
  text-align: center;
  margin: 25px 0;
}
```

```
h2{font-size: 1.5em; margin: 8px 0}
h3{
    font-size: 1.2em;
    margin: 5px 0;
}
h3 span{
    font-weight: normal;
    font-size: 1em;
}
.item{
    clear: both;
    margin:0;
    padding: 10px;
    overflow: hidden;
    border-top: 1px solid #DBDBDB;
}
.item:last-child{border-bottom:1px solid #DBDBDB}
.item:hover{background: #f9f9f9}
.post{
    padding: 10px 0;
    border-bottom: 1px solid #E6E6E6;
}
.comment-block{
    margin: 20px 0 20px 20px;
}
.comment-item{
    overflow: hidden;
    width: 500px;
    clear: both;
    padding: 10px;
    border: 1px solid #E6E6E6;
    border-radius: 5px;
    margin: 5px;
}
.comment-avatar{
```

```
        width: 60px;
        float: left;
    }
    .comment-avatar img{
        width: 60px;
        height: 60px;
        border-radius: 5px;
    }
    .comment-post{
        width: 400px;
        float: left;
        padding: 0 5px 0 10px;
    }
    #form{
        clear: both;
        margin: 10px;
        width: 500px;
    }
    /* form styling */
    input[type="text"],
    input[type="email"],
    input[type="tel"],
    input[type="url"],
    textarea {
        width:100%;
        background: #fff;
        border: 1px solid #ddd;
        font-size: 13px;
        line-height: 20px;
        margin: 0;
        padding: 7px 10px;
        box-shadow: inset 0 1px 2px #eee;
        border: 1px solid #CCC;
        margin:0 0 5px;
        border-radius:5px;
```

```
}
```

```
textarea {
```

```
    height:100px;
```

```
    max-width:100%;
```

```
}
```

```
input[type="submit"] {
```

```
    cursor:pointer;
```

```
    width:100%;
```

```
    border:none;
```

```
    background:#991D57;
```

```
    background-image:linear-gradient(bottom, #8C1C50 0%, #991D57 52%);
```

```
    background-image:-moz-linear-gradient(bottom, #8C1C50 0%, #991D57  
52%);
```

```
    background-image:-webkit-linear-gradient(bottom, #8C1C50 0%, #991D57  
52%);
```

```
    color:#FFF;
```

```
    margin:0 0 5px;
```

```
    padding:10px;
```

```
    border-radius:5px;
```

```
}
```

```
input[type="submit"]:hover {
```

```
    background-image:linear-gradient(bottom, #9C215A 0%, #A82767 52%);
```

```
    background-image:-moz-linear-gradient(bottom, #9C215A 0%, #A82767  
52%);
```

```
    background-image:-webkit-linear-gradient(bottom, #9C215A 0%, #A82767  
52%);
```

```
    -webkit-transition:background 0.3s ease-in-out;
```

```
    -moz-transition:background 0.3s ease-in-out;
```

```
    transition:background-color 0.3s ease-in-out;
```

```
}
```

```
input[type="submit"]:active {
```

```
    box-shadow:inset 0 1px 3px rgba(0,0,0,0.5);
```

```
}
```

```
input:focus,
```



```
textarea:focus {
    outline:0;
    border:1px solid #999;
}
label{
    display: block;
    margin: 5px 0;
    font-weight: 900;
    cursor: pointer;
}

.alert{
    display: none;
    padding: 8px 35px 8px 14px;
    margin: 20px 0;
    text-shadow: 0 1px 0 rgba(255, 255, 255, 0.5);
    color: #468847;
    background-color: #dff0d8;
    border-color: #d6e9c6;
    -webkit-border-radius: 4px;
    -moz-border-radius: 4px;
    border-radius: 4px;
}
```

Script.js

```
$(document).ready(function(){
    var form = $('form');
    var submit = $('#submit');

    form.on('submit', function(e) {
        // prevent default action
        e.preventDefault();
        // send ajax request
        $.ajax({
            url: 'ajax_comment.php',
            type: 'POST',
            cache: false,
```

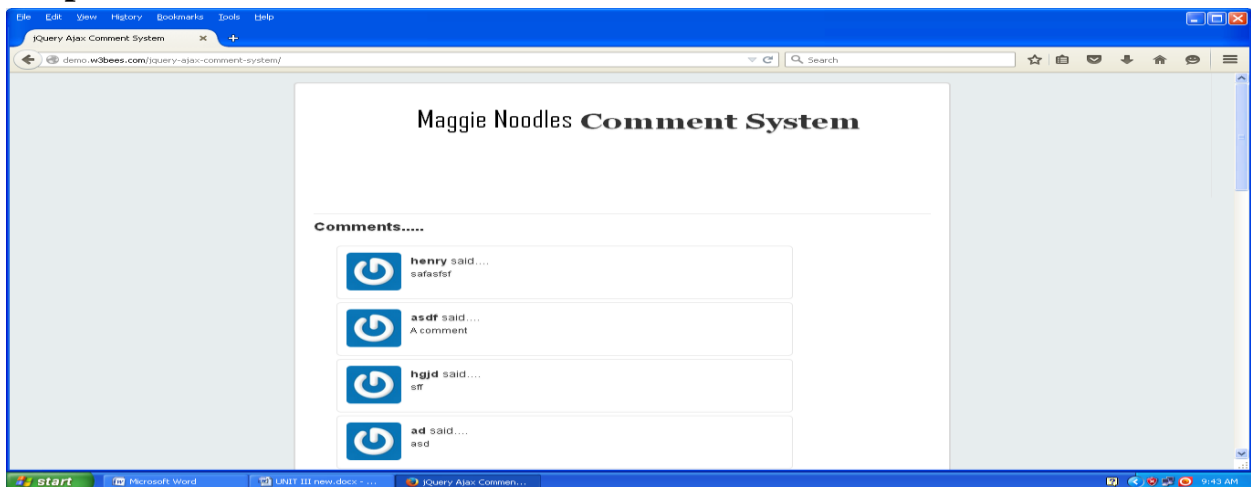
```

data: form.serialize(), //form serizlize data
beforeSend: function(){
    // change submit button value text and disabled it
    submit.val('Submitting...').attr('disabled', 'disabled');
},
success: function(data){
    // Append with fadeIn see
    http://stackoverflow.com/a/978731
    var item = $(data).hide().fadeIn(800);
    $('.comment-block').append(item);

    // reset form and button
    form.trigger('reset');
    submit.val('Submit Comment').removeAttr('disabled');
},
error: function(e){
    alert(e);
}
});
});

```

Output:



Result:

Thus a web services for finding what people think by asking 500 people's opinion for any consumer product has been executed successfully.