

ST. ANNE'S
COLLEGE OF ENGINEERING AND TECHNOLOGY
ANGUCHETTYPALAYAM, PANRUTI – 607 110

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



LAB MANUAL

CS8661 – INTERNET PROGRAMMING LABORATORY

Regulation 2017

Year / Semester : III / VI

Dec 2019 – Apr 2020

PREPARED BY

Mrs. D.Pauline Freeda, M.E.,
Associate Professor / CSE

LIST OF EXPERIMENTS

1. Create a web page with the following using HTML
 - a. To embed a map in a web page
 - b. To fix the hot spots in that map
 - c. Show all the related information when the hot spots are clicked.
2. Create a web page with the following.
 - a. Cascading style sheets.
 - b. Embedded style sheets.
 - c. Inline style sheets. Use our college information for the web pages.
3. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.
4. Write programs in Java using Servlets:
 - i. To invoke servlets from HTML forms
 - ii. Session tracking using hidden form fields and Session tracking for a hit count
5. Write programs in Java to create three-tier applications using servlets for conducting online examination for displaying student mark list. Assume that student information is available in a database which has been stored in a database server.
6. Install TOMCAT web server. Convert the static web pages of programs into dynamic web pages using servlets (or JSP) and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.
7. Redo the previous task using JSP by converting the static web pages into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database.
8. Create and save an XML document at the server, which contains 10 users Information. Write a Program, which takes user Id as an input and returns the User details by taking the user information from the XML document
9.
 - i. Validate the form using PHP regular expression.
 - ii. PHP stores a form data into database.
10. Write a web service for finding what people think by asking 500 people's opinion for any consumer product.

TABLE OF CONTENTS

S.NO.	DATE	EXPERIMENT TITLE	MARKS/10	SIGN.
1		IMAGE MAPPING IN HTML		
2		STYLE SHEETS		
3		VALIDATION USING JAVASCRIPT		
4a		INVOKE SERVLETS FROM HTML FORMS		
4b		SESSION TRACKING USING HIDDEN FORM FIELDS		
4c		SESSION TRACKING FOR A HIT COUNT		
5		ON-LINE EXAMINATION USING SERVLETS		
6		CONVERSION OF STATIC WEBPAGES INTO DYNAMIC USING SERVLETS		
7		CONVERSION OF STATIC WEBPAGES INTO DYNAMIC USING JSP		
8		CREATION OF AN XML DOCUMENT		
9a		FORM VALIDATION USING PHP REGULAR EXPRESSION		
9b		STORING A FORM DATA IN PHP		
10		WEB SERVICE FOR FINDING PEOPLE'S OPINION		

AIM:

To create a web page which includes a map and display the related information when a hot spot is clicked in the map.

PROCEDURE:

1. Create a html file with map tag.
2. Set the source attribute of the img tag to the location of the image and also set the use map attribute.
3. Specify an area with name, shape and href set to the appropriate values.
4. Repeat step 3 as many hot spots you want to put in the map.
5. Create html files for each and every hot spot the user will select.

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>
```

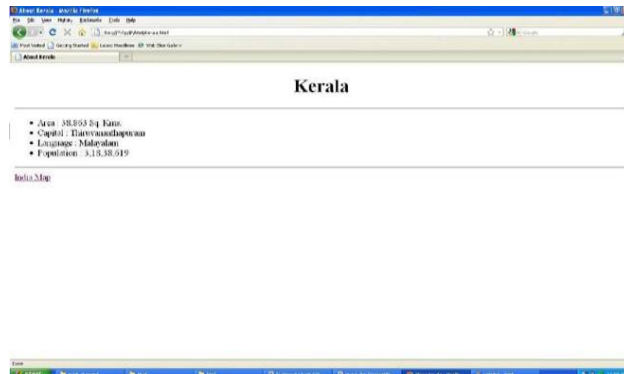
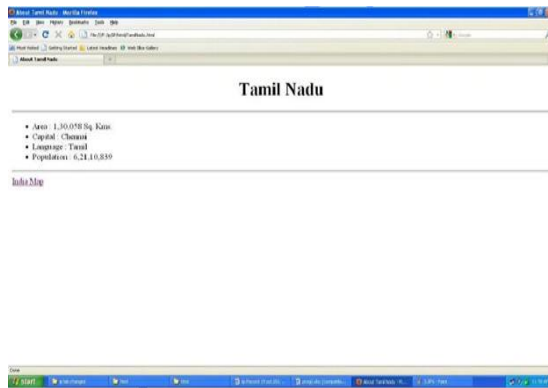
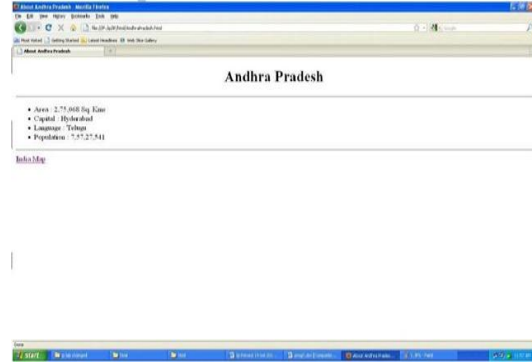
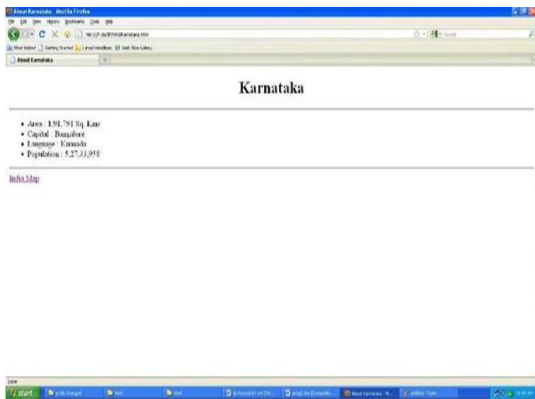
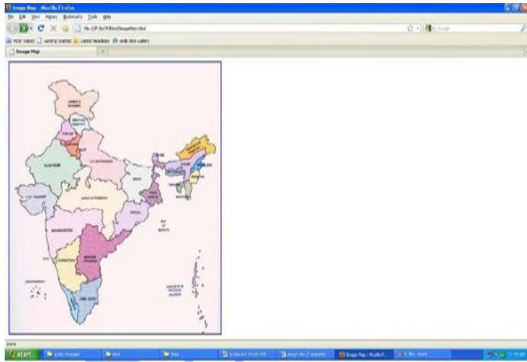
AndhraPradesh.html

```
<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.

AIM:

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

PROCEDURE:

1. Create a web page with frame sets consisting two frames
2. In the first frame include the links
3. In the second frame set display the web page of the link
4. Create a external style sheets
5. Create a embedded style sheets
6. Create a inline and internal style sheets and make it link to the external style sheets

PROGRAM:**XYZ.CSS:**

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

HTML CODE:

```
<html>
<head>
<h1>
<center>ALL STYLE SHEETS</center>
</h1>
<title>USE of INTERNAL and EXTERNAL STYLESHEETS </title>
<link rel="stylesheet" href="xyz.css" type="text/css">
<style type="text/css">
.vid
{
font-family:verdana;
font-style:italic;
color:red;
text-align:center
}
.ani
{
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
}
p
{
    font-family: georgia, serif;
    font-size: x-small;
}
hr
{
    color: #ff9900; height: 1px
}
a:hover
{
color: #ff0000;
text-decoration: none
}
</style>
</head>
<body>
    <h1 style="color:blue;margin-left:30px;">Welcome</h1> //In-line style Sheet
    <ol style="list-style-type:lower-alpha">

<b>St.Anne's College of Engineering and Technology </b>
<br>
<br>
<br>
<li> EEE</li>

<li> ECE </li>
<li> MECH</li>
<li> CSE</li>
</ol>

    <p style="font-size:20pt;color:purple">Details</p>
    <p class="ani">St.Anne's College <br>It is approved by AICTE(All India Council for
Technical Education). It is affiliated to Anna University.<br></p>
    <h2 class="vid"> St.Anne's college of engineering </h2> <br>
    <font>It is an ISO certified Institution
</font>
<br>
<br>
<font>
<h2>List of Courses offered</h2>
<ul>

```

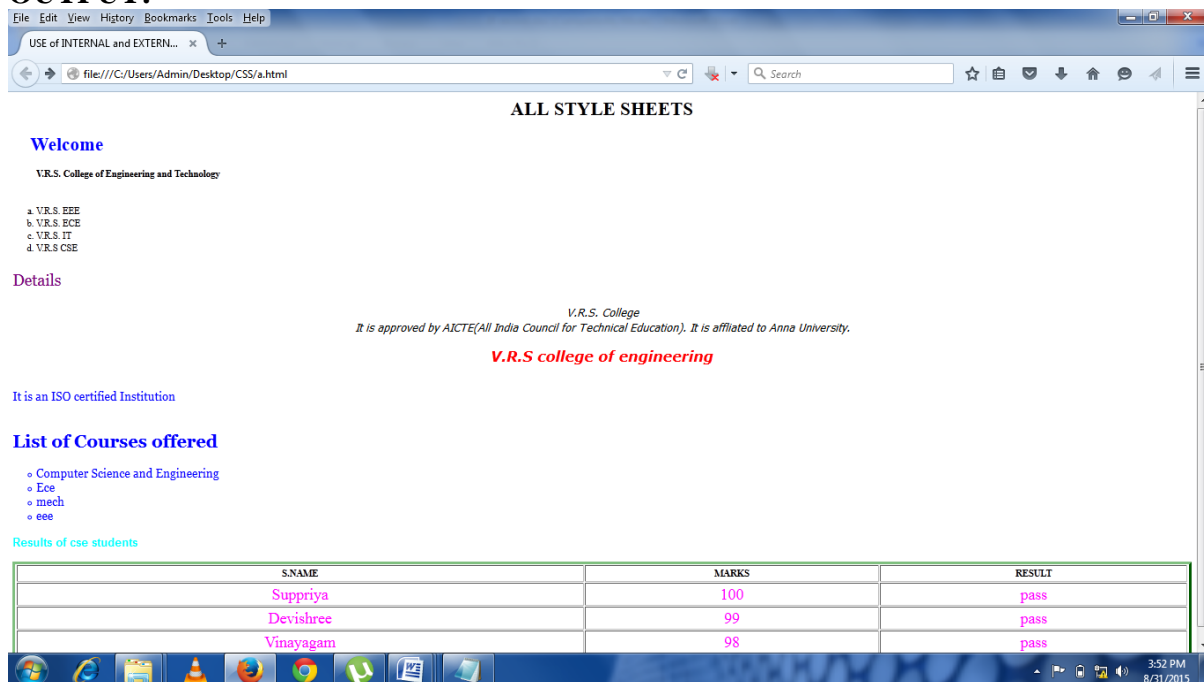


```

</li>Computer Science and Engineering</li>
</li>Ece</li>
</li>mech</li>
</li>eee</li>
</ul>
</font>
<h3>Results of cse students</h3>
<table width="100%" cellspacing="2" cellpadding="2" border="5"> <tr>
  <th>S.NAME</th> <th>MARKS</th> <th>RESULT</th>
</tr>
<tr>
  <td align="center">Suppriya</td> <td align="center">100</td>
  <td align="center">pass</td>
</tr>
<tr>
  <td align="center">Devishree</td> <td align="center">99</td>
  <td align="center">pass</td>
</tr>
<tr>
  <td align="center">Vinayagam</td> <td align="center">98</td>
  <td align="center">pass</td> </tr>
</table>
</body>
</html>

```

OUTPUT:



RESULT:

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

Date:

AIM:

To Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.

PROCEDURE:

1. Static web pages of an online Book store is developed with following pages.
 - Home page
 - Registration and user Login
 - User profile page
 - Books catalog
 - Shopping cart
 - Payment by credit card
2. Each input box in webpage is validated using java script code using <script> tag in html file.
3. Designed output is displayed.

PROGRAM AND OUTPUT :**Main.html:**

```
<frameset rows="25%, 75 %">
<frame src="top.html" name="top">
<frameset cols="25%,75%">
<frame src="left.html" name="left">
<frame src="right.html" name="right">
</frameset>
</frameset>
```

Top.html:

```
<html>
<body bgcolor="pink">
<br><br>
<marquee><h1 align="center"><b><u>ONLINE BOOK
STORAGE</u></b></h1></marquee>
</body>
</html>
```

Right.html:

```
<html>
<body bgcolor="pink">
<br><br><br><br><br>
<h2 align="center">
<b><p> welcome to online book storage. Press login if you are
having id otherwise press registration.
</p></b></h2>
</body>
</html>
```

Left.html:

```
<html>
<body bgcolor="pink">
<h3>
<ul>
<li><a href="login.html" target="right"><font color="black">
LOGIN</font></a></li><br><br>
<li><a href="reg.html" target="right"><font color="black">
```

```

REGISTRATION</font></a></li><br><br>
<li><a href="profile.html" target="right"><fontcolor="black">
USER PROFILE</font></a></li><br><br>
<li><a href="catalog.html" target="right"><fontcolor="black">
BOOKS CATALOG</font></a></li><br><br>
<li><a href="scart.html" target="right"><font color="black">
SHOPPINGCART</font></a></li><br><br>
<li><a href="payment.html" target="right"><fontcolor="black">
PAYMENT</font></a></li><br><br>
<li><a href="order.html" target="right"><font color="black">
ORDER CONFIRMATION</font></a></li><br><br>

```

ONLINE BOOK STORAGE

- [LOGIN](#)
- [REGISTRATION](#)
- [USER PROFILE](#)
- [BOOKS CATALOG](#)
- [SHOPPINGCART](#)
- [PAYMENT](#)
- [ORDER
CONFIRMATION](#)

welcome to online book storage. Press login if you are having id otherwise press registration.

Login.html:

```

<html>
<body bgcolor="pink"><br><br><br>
<script language="javascript">
function validate()
{
    var flag=1;
    if(document.myform.id.value=="||document.myform.pwd.value=="")
    {
        flag=0;
    }
    if(flag==1)
    {
        alert("VALID INPUT");
    }
    else
    {
        alert("INVALID INPUT");
        document.myform.focus();
    }
}
</script>
<form name="myform">
<div align="center"><pre>
LOGIN ID:<input type="text" name="id"><br>
PASSWORD:<input type="password" name="pwd">
</pre><br><br>
</div>
<br><br>

```

```

<div align="center">
<input type="submit" value="ok" onClick="validate()">&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<input type="reset" value="clear">
</form>
</body>
</html>.

```

Reg.html:

```

<html>
<body bgcolor="pink"><br><br>
<script language="javascript">
function validate()
{
    var flag=1;
    if(document.myform.name.value==""||
    document.myform.addr.value==""||
    document.myform.phno.value==""||
    document.myform.id.value==""||
    document.myform.pwd.value=="")
    {
        flag=0;
    }
    var str=document.myform.phno.value;
    var x;
    for(var i=0;i<str.length;i++)
    {
        x=str.substr(i,1)
        if(!(x<=9))
        {
            flag=0;
            break;
        }
    }
    if(flag==1)
    {
        alert("VALID INPUT");
    }
    else
    {
        alert("INVALID INPUT");
        document.myform.focus();
    }
}
</script>
<form name="myform">
<div align="center"><pre>
NAME :<input type="text" name="name"><br>
ADDRESS :<input type="text" name="addr"><br>
CONTACT NUMBER:<input type="text" name="phno"><br>
LOGINID :<input type="text" name="id"><br>
PASSWORD :<input type="password" name="pwd"></pre><br><br>
</div>
<br><br>
<div align="center">

```


Date:

AIM:

To write a java program to invoke servlets from HTML form.

PROCEDURE:***client.html:***

- (1) Create a web page using HTML form that contains the fields such as text, password and one submit button.
- (2) Set the URL of the server as the value of form's action attribute.
- (3) Run the HTML program.
- (4) Submit the form data to the server.

server.java:

- (1) Define the class server that extends the property of the class GenericServlet.
- (2) Handle the request from the client by using the method service() of GenericServlet class.
- (3) Get the parameter names from the HTML form by using the method getParameterNames().
- (4) Get the parameter values from the HTML forms by using the method getParameter().
- (5) Send the response to the client by using the method of PrintWriter class.

PROGRAM:**server.java:**

```
import java.io.*;
import java.util.*;
import javax.servlet.*;
public class server extends GenericServlet
{
    public void service(ServletRequest req,ServletResponse res)throws ServletException,IOException
    {
        PrintWriter pw=res.getWriter();
        pw.print("<h3>Registration Successful...</h3>");
        Enumeration e=req.getParameterNames();
        while(e.hasMoreElements())
        {
            String str1=(String)e.nextElement();
            String str2=req.getParameter(str1);
            pw.print(str1+"="+str2+"<br/>");
        }
        pw.close();
    }
}
```

web.xml:

```
<web-app>
<servlet>
    <servlet-name>Register</servlet-name>
    <servlet-class>server</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>Register</servlet-name>
    <url-pattern>/server</url-pattern>
```


Date:

AIM:

To write a Java Program for Session Tracking Using Hidden Form Fields. This servlet demonstrates session tracking using hidden form fields by displaying the shopping cart for a bookworm. Note that, if you try this servlet, the buttons at the bottom of the page it generates don't take you anywhere real.

PROCEDURE:

1. Pass Cart items in the item parameter.
2. Print the current cart items.
3. Ask if the user wants to add more items or check out.
4. Include the current items as hidden fields so they'll be passed on and submit to self.

PROGRAM:

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class ShoppingCartViewerHidden extends HttpServlet {
    public void doGet(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {
        res.setContentType("text/html");
        PrintWriter out = res.getWriter();
        out.println("<HEAD><TITLE>Current Shopping Cart Items</TITLE></HEAD>");
        out.println("<BODY>");
        // Cart items are passed in as the item parameter.
        String[] items = req.getParameterValues("item");
        // Print the current cart items.
        out.println("You currently have the following items in your cart:<BR>");
        if (items == null) {
            out.println("<B>None</B>");
        }
        else {
            out.println("<UL>");
            for (int i = 0; i < items.length; i++) {
                out.println("<LI>" + items[i]);
            }
            out.println("</UL>");
        }
        // Ask if the user wants to add more items or check out.
        // Include the current items as hidden fields so they'll be passed on.
        out.println("<FORM METHOD=GET>"); // submit to self
        if (items != null) {
            for (int i = 0; i < items.length; i++) {
                out.println("<INPUT TYPE=HIDDEN NAME=item VALUE=\"" +
                    items[i] + "\">");
            }
        }
        out.println("Would you like to<BR>");
        out.println("<INPUT TYPE=SUBMIT VALUE=\" Add More Items \">");
        out.println("<INPUT TYPE=SUBMIT VALUE=\" Check Out \">");
        out.println("</FORM>");
    }
}
```

```

    out.println("</BODY></HTML>");
}
}

```

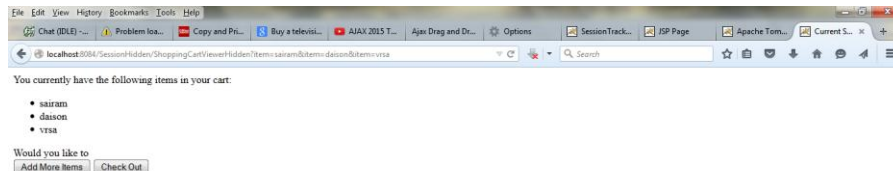


http://localhost:8084/SessionHidden/ShoppingCartViewerHidden?item=sairam&item=daison&item=vrsa

```

if (items != null) {
    for (int i = 0; i < items.length; i++) {
        out.println("<INPUT TYPE=TEXT NAME=item VALUE=\""
            + items[i] + "\">");
    }
}

```



```

if (items != null) {
    for (int i = 0; i < items.length; i++) {
        out.println("<INPUT TYPE=HIDDEN NAME=item VALUE=\""
            + items[i] + "\">");
    }
}

```

RESULT:

Thus the Java program for Session Tracking using hidden form fields has been executed successfully.

Date:

AIM:

To write a Java Program for Session tracking a hit count. This servlet uses session tracking to count the number of times a client has accessed it.

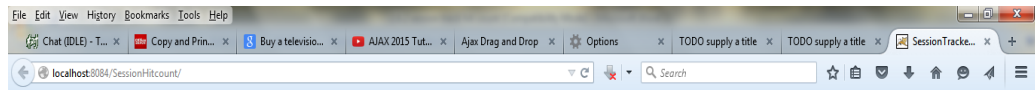
PROCEDURE:

1. Get the current session object, create one if necessary
2. Increment the hit count for this page. The value is saved in this client's session under the name "tracker.count".
3. Display the hit count for this page

PROGRAM:

```
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class SessionTracker extends HttpServlet {
    public void doGet(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {
        res.setContentType("text/html");
        PrintWriter out = res.getWriter();
        // Get the current session object, create one if necessary
        HttpSession session = req.getSession();
        // Increment the hit count for this page. The value is saved
        // in this client's session under the name "tracker.count".
        Integer count = (Integer)session.getAttribute("tracker.count");
        if (count == null)
            count = new Integer(1);
        else
            count = new Integer(count.intValue() + 1);
        session.setAttribute("tracker.count", count);
        out.println("<HTML><HEAD><TITLE>SessionTracker</TITLE></HEAD>");
        out.println("<BODY><H1>Session Tracking Demo</H1>");
        // Display the hit count for this page
        out.println("You've visited this page " + count +
            ((count.intValue() == 1) ? " time." : " times."));
        out.println("<P>");
        out.println("<H2>Here is your session data:</H2>");
        Enumeration enum = session.getAttributeNames();
        while (enum.hasMoreElements()) {
            String name = (String) enum.nextElement();
            out.println(name + ": " + session.getAttribute(name) + "<BR>");
        }
        out.println("</BODY></HTML>");
    }
}
```

OUTPUT:



Session Tracking Demo

You've visited this page 3 times.

Here is your session data:

tracker.count: 3

Servlet SessionTracker at /SessionHitcount

RESULT:

Thus the Java program for session tracking a hit count has been executed successfully.

AIM:

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

PROCEDURE:**Client:**

1. 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.
2. Create a submit button and close all the included tags.

Server:

1. Import all necessary packages
2. Define a class that extends servlet
3. 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:

1. Import necessary to java packages and javax packages and classes
2. Create a class that extends HttpServlet and implements ServletException
3. and IOException
4. 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 sqllex){ sqllex.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"+") VAL-
UES("+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;
} }

```

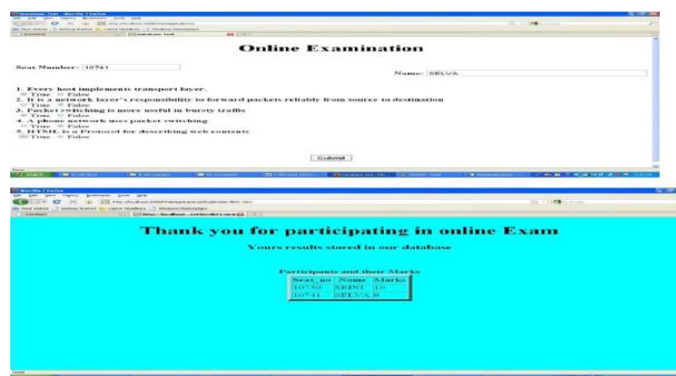
HTML Code:

```

<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.

AIM:

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

PROCEDURE:

1. We will create ahtml form for entering the user name,password and card ID.
2. 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.
3. Compile the above servlet Login servlet.java and copy its class file in tomcats folder at c:\tomcatdirectory\webapps\examples\WEB-INF\classes.
4. Then edit the web.xml in WEB-INF folder.We must store he user information such as user name,password and card id in the web.xml using init-param.
5. On successful login , the information from the cookie is checked and shopping cart page for corressponding user can be displayed.
6. Compile the above servlet LoginSuccess.java and copy its class file in the tomcat's folder at c:\tomcatdirectory\webapps\examples\WEB-INF\classes.
7. Then edit the web.xml in WEB-INF folder.
8. Start tomcat web server.Open the web browser and display the login form created in step1.

PROGRAM:**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();
}
}

```

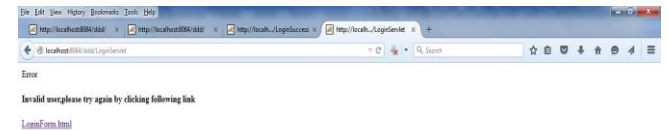
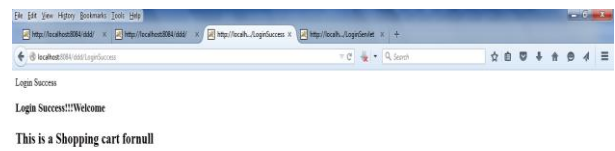
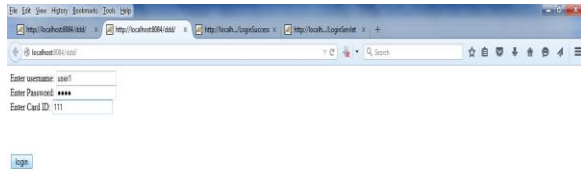
Web.xml

```

<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:



RESULT:

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


```

out.println("<<li><a href=\`catalog.html\`><fontcolor=\`black\`>BOOKS
CATALOG</font></a></li><br><br>");
out.println("<<li><a href=\`order.html\`><fontcolor=\`black\`>ORDER
CONFIRMATION</font></a></li><br><br>");
out.println("</ul>");
}
out.println("<body></html>");
%>

```

Reg.jsp:

```

%@page import="java.sql.*"%
%@page import="java.io.*"%
<%
out.println("<html><body bgcolor=\`pink\`>");
String name=request.getParameter("name");
String addr=request.getParameter("addr");
String phno=request.getParameter("phno");
String id=request.getParameter("id");
String pwd=request.getParameter("pwd");
int no=Integer.parseInt(phno);
Driver d=new oracle.jdbc.driver.OracleDriver();
DriverManager.registerDriver(d);
Connection con=
DriverManager.getConnection ("jdbc:oracle:thin:@localhost:1521:orcl","scott","tiger");
Statement stmt=con.createStatement();
String sqlstmt="select id from login";
ResultSet rs=stmt.executeQuery(sqlstmt);
int flag=0;
while(rs.next())
{
if(id.equals(rs.getString(1)))
{
flag=1;
}
}
if(flag==1)
{
out.println("SORRY LOGIN ID ALREADY EXISTS TRY AGAIN WITH NEW ID <br><br>");
out.println("<a href=\`/tr1/reg.html\`>press REGISTER to RETRY</a>");
}
else
{
Statement stmt1=con.createStatement ();
stmt1.executeUpdate ("insert into login values (+name+`,`+addr+`,`+no+`,`+id+`,`+pwd+)");
out.println ("YOU DETAILS ARE ENTERED <br><br>");
out.println ("<a href =\`/tr1/login.html\`>press LOGIN to login</a>");
}
out.println ("</body></html>");
%>

```

Profile.jsp:

```

<%@page import="java.sql.*"%>

```

```

<%@page import="java.io.*"%>
<%
out.println (“<html><body bgcolor=\”pink\”>”);
String id=request.getParameter(“id”);
Driver d=new oracle.jdbc.driver.OracleDriver();
DriverManager.regiserDriver(d);
Connection con=
DriverManager.getConnection (“jdbc:oracle:thin:@localhost:1521:orcl”,”scott”,”tiger”);
Statement stmt=con.createStatement ();
String sqlstmt=”select * from login where id=”+id+””;
ResultSet rs=stmt.executeQuery (sqlstmt);
int flag=0;
while(rs.next())
{
out.println (“<div align=\”center\”>”);
out.println (“NAME :”+rs.getString(1)+”<br>”);
out.println (“ADDRESS :”+rs.getString(2)+”<br>”);
out.println (“PHONE NO :”+rs.getString(3)+”<br>”);
out.println (“</div>”);
flag=1;
}
if(flag==0)
{
out.println(“SORRY INVALID ID TRY AGAIN ID <br><br>”);
out.println(“<a href=\”/tr1/profile.html\”>press HERE to RETRY </a>”);
}
out.println (“</body></html>”);
%>

```

Catalog.jsp:

```

<%@page import="java.sql.*"%>
<%@page import="java.io.*"%>
<%
out.println (“<html><body bgcolor=\”pink\”>”);
String title=request.getParameter (“title”);
Driver d=new oracle.jdbc.driver.OracleDriver ();
DriverManager.regiserDriver (d);
Connection con=
DriverManager.getConnection (“jdbc:oracle:thin:@localhost:1521:orcl”,”scott”,”tiger”);
Statement stmt=con.createStatement ();
String sqlstmt=”select * from book where title=”+title+””;
ResultSet rs=stmt.executeQuery (sqlstmt);
int flag=0;
while(rs.next())
{
out.println (“<div align=\”center\”>”);
out.println (“TITLE :”+rs.getString(1)+”<br>”);
out.println (“AUTHOR :”+rs.getString(2)+”<br>”);
out.println (“VERSION:”+rs.getString(3)+”<br>”);
out.println (“PUBLISHER :” +rs.getString(4)+”<br>”);
out.println (“COST :” +rs.getString(5)+”<br>”);
out.println (“</div>”);
flag=1;
}

```

```

}
if(flag==0)
{
out.println("SORRY INVALID ID TRY AGAIN ID <br><br>");
out.println("<a href='\"/tr1/catalog.html\"'>press HERE to RETRY </a>");
}
out.println("</body></html>");
%>

```

Order.jsp:

```

<%@page import="java.sql.*"%>
<%@page import="java.io.*"%>
<%
out.println("<html><body bgcolor='\"pink\"'>");
String id=request.getParameter("id");
String pwd=request.getParameter("pwd");
String title=request.getParameter("title");
String count1=request.getParameter("no");
String date=request.getParameter("date");
String cno=request.getParameter("cno");
int count=Integer.parseInt(count1);
Driver d=new oracle.jdbc.driver.OracleDriver ();
DriverManager.regiserDriver (d);
Connection con=
DriverManager.getConnection ("jdbc:oracle:thin:@localhost:1521:orcl", "scott", "tiger");
Statement stmt=con.createStatement ();
String sqlstmt="select id, password from login";
ResultSet rs=stmt.executeQuery (sqlstmt);
int flag=0,amount,x;
while(rs.next())
{
if(id.equals(rs.getString(1))&& pwd.equals(rs.getString(2)))
{
flag=1;
}
}
if(flag==0)
{
out.println("SORRY INVALID ID TRY AGAIN ID <br><br>");
out.println("<a href='\"/tr1/order.html\"'>press HERE to RETRY </a>");
}
else
{
Statement stmt2=con.createStatement();
String s="select cost from book where title='"+title+"'";
ResultSet rs1=stmt2.executeQuery(s);
int flag1=0;
while(rs1.next())
{
flag1=1;
x=Integer.parseInt(rs1.getString(1));
amount=count*x;
}
}
}
%>

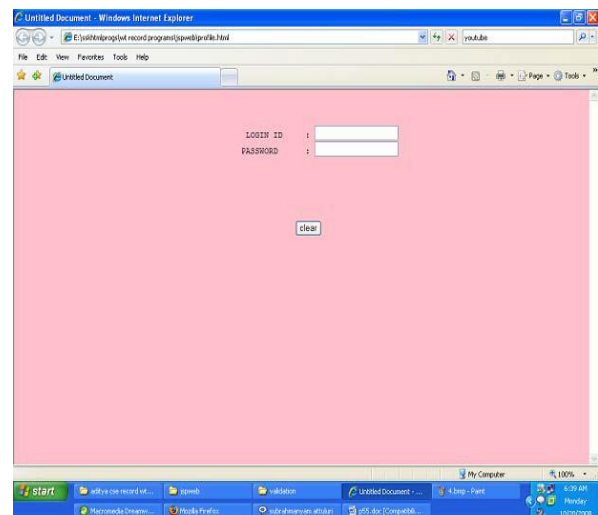
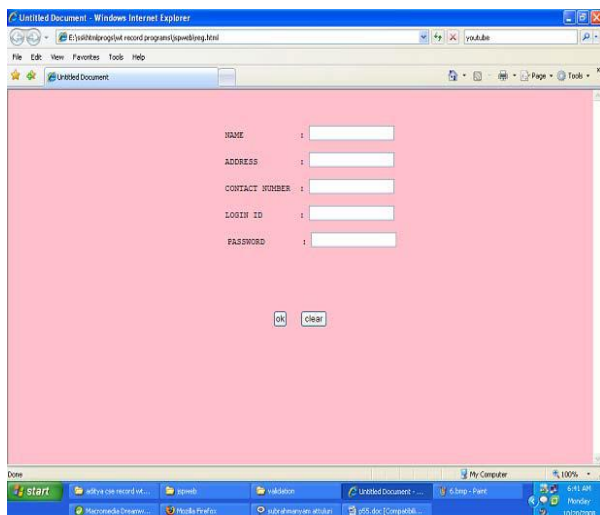
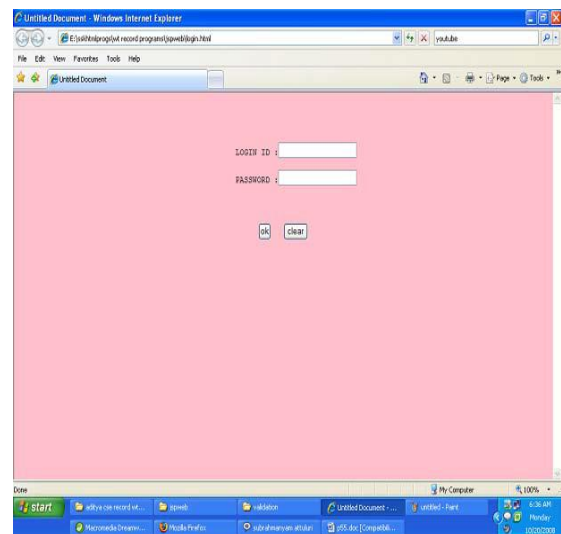
```

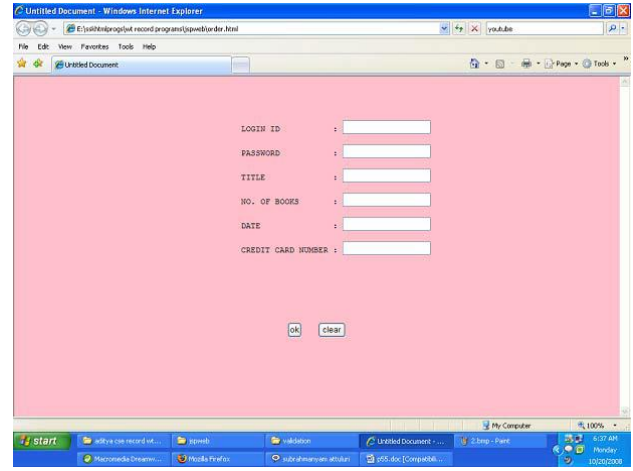
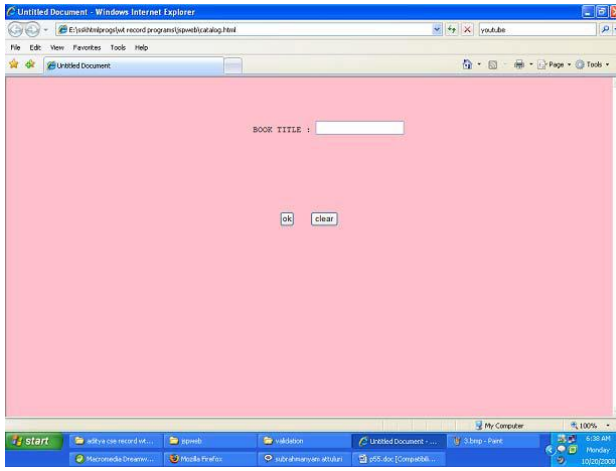
```

out.println("AMOUNT :"+amount+"<br><br><br><br>");
Statement stmt1=con.createStatement ();
stmt1.executeUpdate ("insert into details ("+"id+"","+title+"","+amount+"","+date+"","+cno+"");");
out.println ("YOU ORDER HAS TAKEN<br>");
}
if(flag1==0)
{
out.println("SORRY INVALID BOOK TRY AGAIN <br><br>");
out.println("<a href='\"/tr1/order.html\"'>press HERE to RETRY </a>");
}
}
}
out.println ("</body></html>");%>

```

OUTPUT:





RESULT:

The static web pages are converted into dynamic web pages using JSP for book catalogue.

Date :

AIM:

To create and save an XML document at the server, which contain some users information. To develop Java Program takes user id as an input and returns the user details by taking the user information from the XML document.

PROCEDURE:

1. Save Students information in the XML file on the specific location.
2. Create and Establish the server connection between html file and XML file in the host
3. Get the user ID as input
4. Display the student's information.

PROGRAM :**StudentDetails.Java :**

```
import javax.servlet.*;
import java.util.*;
import java.io.*;
import javax.xml.parsers.*;
import org.w3c.dom.*;

public class StudentDetails implements Servlet
{
    private DocumentBuilderFactory fact;
    private DocumentBuilder builder;
    private Document doc;
    private NodeList list,childs;
    private Node node,parent,child;
    private String str;
    private String hallTicket;
    private ServletConfig sc;
    public void init(ServletConfig sc)
    {
        try
        {
            this.sc=sc;
            str="h://Details.xml";
            fact=DocumentBuilderFactory.newInstance();
            builder=fact.newDocumentBuilder();
            doc=builder.parse(str);
            System.out.println("In the Init Method");
        }
        catch(Exception e)
        {
            System.out.println("Error in the Init Method"+e.getMessage());
        }
    }
    public void service(ServletRequest req, ServletResponse res)throws ServletException,IOException
    {
        hallTicket=req.getParameter("hall");
        res.setContentType("text/html");
    }
}
```

```

PrintWriter pw=res.getWriter();
list=doc.getElementsByTagName("HallTicketNo");
pw.print("<center><h1>Welcome To Student Details</center></h1>");
for(int i=0;i<list.getLength();i++)
{
    node=list.item(i);
    if(node.getTextContent().equals(hallTicket))
    {
        parent=node.getParentNode();
        childs=parent.getChildNodes();
        for(int j=1;j<childs.getLength()-1;j=j+2)
        {
            child=childs.item(j);
            pw.print("<center>"+child.getNodeName()+" "+child.getTextContent());
        }
        break;
    }//if
}//for
}//service
public ServletConfig getServletConfig()
{
return sc;
}
public String getServletInfo()
{
return;
}
public void destroy()
{
}
}

```

LogIn.html :

```

<html>
<head><title>StudentDetails</title>
</head>
    <body style="height: 100%;
    width:100%;
    margin: 0;
    padding: 0;
    overflow-y:hidden;">
<form method="post" action="http://localhost:8080/StudentDetails/MyServletEx">
<div style=    "position:absolute;
    top:0;
    left:0;
    width:100%;
    height:100%;
    margin:0;
    padding:0;
    z-index:0;">
</div>
<div style=    "position:absolute;
    top:4%;

```

```

        left:25%;
        z-index:1" align="center" >
<font size="6" color="red" >ST.ANNE'S College of Engineering and Technololy</font>
<br/>
<font size=4 color="red">(Affiliated to Anna University Approved By AICTE)</font>
<br/>
<font size=3 color="red">PANRUTI</font>
<br/>
<font size=4 color="red">STUDENT'S INFORMATION RETRIEVAL SYSTEM</font>
<br/>
</div>
<div style=    "position:absolute;
        top:60%;
        right:5%;
        z-index:2">
<input type="text" SIZE="10" name="hall"><br/>
<input type="submit" value="Submit">
</div>
</form>
</body>
</html>

```

web.xml :

```

<web-app>
<servlet>
    <servlet-name>Student</servlet-name>
    <servlet-class>StudentDetails</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>Student</servlet-name>
    <url-pattern>/MyServletEx</url-pattern>
</servlet-mapping>
</web-app>

```

Details.xml output :

```

- <StudentDetails>
- <Details>
  <HallTicketNo>05e21a0501</HallTicketNo>
  <Name>Shravya</Name>
  <Education>B.Tech</Education>
  <Specialization>CSE</Specialization>
  <Year>IV</Year>
  <Semester>I</Semester>
  <Ambition>SoftwareEngineer</Ambition>
  <Hobby>Reading Books</Hobby>
</Details>
- <Details>
  <HallTicketNo>05e21a0502</HallTicketNo>
  <Name>UshaSingh</Name>
  <Education>B.Tech</Education>
  <Specialization>CSE</Specialization>
  <Year>IV</Year>
  <Semester>I</Semester>

```

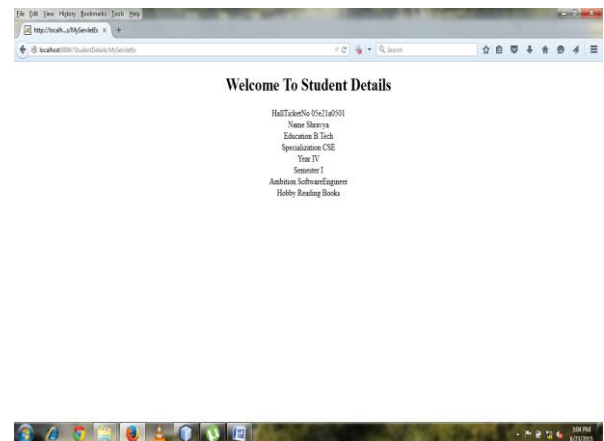
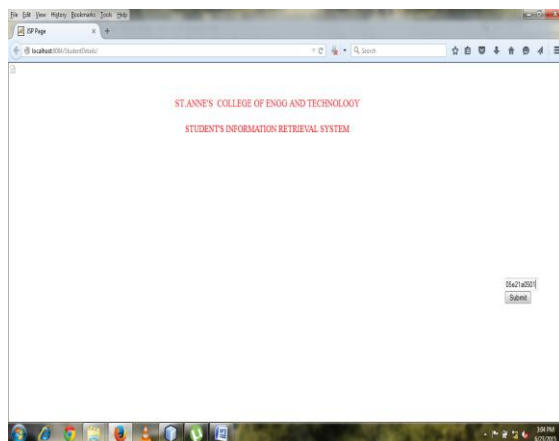


```

<Ambition>SoftwareEngineer</Ambition>
<Hobby>Dance</Hobby> </Details>-
<Details>
<HallTicketNo>05e21a0503</HallTicketNo> <Name>Archana</Name>
<Education>B.Tech</Education>
<Specialization>CSE</Specialization> <Year>IV</Year> <Semester>I</Semester>
<Ambition>SoftwareEngineer</Ambition>
<Hobby>Foot Boll</Hobby> </Details>-
<Details>
<HallTicketNo>05e21a0504</HallTicketNo> <Name>Pavani</Name>
<Education>B.Tech</Education>
<Specialization>CSE</Specialization>
<Year>IV</Year>
<Semester>I</Semester>
<Ambition>SoftwareEngineer</Ambition>
<Hobby>Cricket</Hobby>
</Details>- <Details> <HallTicketNo>05e21a0505</HallTicketNo>
<Name>Vishnu Vardhan</Name>
<Education>B.Tech</Education> <Specialization>CSE</Specialization> </Details>
</StudentDetails>

```

OUTPUT:



RESULT:

Thus the Java 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.

Date:

AIM:

To validate the form using PHP regular expression.

PROCEDURE:

1. Form is created for class registration with fields.
2. These fields are validated using PHP code.
3. Form is displayed.

PROGRAM:

```

<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"]);
}
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"]);
}
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 $data;
}
?>
<h2>Absolute classes registration</h2>
<p><span class = "error">* required field.</span></p>
<form method = "post" action = "<?php
echo htmlspecialchars($_SERVER["PHP_SELF"]);?>">
<table>
<tr>
<td>Name:</td>
<td><input type = "text" name = "name">
<span class = "error">* <?php echo $nameErr;?></span>
</td>
</tr>
<tr>
<td>E-mail: </td>
<td><input type = "text" name = "email">
<span class = "error">* <?php echo $emailErr;?></span>
</td>
</tr>
<tr>
<td>Time:</td>
<td> <input type = "text" name = "website">
<span class = "error"><?php echo $websiteErr;?></span>
</td>
</tr>
<tr>
<td>Classes:</td>
<td> <textarea name = "comment" rows = "5" cols = "40"></textarea></td>
</tr>
<tr>
<td>Gender:</td>
<td>
<input type = "radio" name = "gender" value = "female">Female
<input type = "radio" name = "gender" value = "male">Male
<span class = "error">* <?php echo $genderErr;?></span>
</td>
</tr>
<td>
<input type = "submit" name = "submit" value = "Submit">
</td> </table> </form>
<?php
echo "<h2>Your given values are as:</h2>";
echo $name;
echo "<br>";
echo $email;
echo "<br>";
echo $website;
echo "<br>";
echo $comment;
echo "<br>";

```

```
echo $gender;
?> </body></html>
```

OUTPUT:

Absolute classes registration

* required field.

Name: *

E-mail: *

Time:

Classes:

Gender: Female Male *

Your given values are as :

RESULT:

The form is validated using PHP regular expression.

AIM:

To store a PHP form data into database.

PROCEDURE:

1. Form is created with fields name, mail id, contact and address.
2. Table is created in mysql.
3. The input data entered into the fields are stored using PHP code.

PROGRAM:**MySQL Code:**

```
CREATE DATABASE IF NOT EXISTS colleges;
```

```
CREATE TABLE students(  
student_name varchar(255) NOT NULL,  
student_email varchar(255) NOT NULL,  
student_contact varchar(255) NOT NULL,  
student_address varchar(255) NOT NULL  
)
```

HTML Code :**insert.php**

```
<!DOCTYPE html>  
<html>  
<head>  
<title>PHP insertion</title>  
<link href="css/insert.css" rel="stylesheet">  
</head>  
<body>  
<div class="maindiv">  
<!--HTML Form -->  
<div class="form_div">  
<div class="title">  
<h2>Insert Data In Database Using PHP.</h2>  
</div>  
<form action="insert.php" method="post">  
<!-- Method can be set as POST for hiding values in URL-->  
<h2>Form</h2>  
<label>Name:</label>  
<input class="input" name="name" type="text" value="">  
<label>Email:</label>  
<input class="input" name="email" type="text" value="">  
<label>Contact:</label>  
<input class="input" name="contact" type="text" value="">  
<label>Address:</label>  
<textarea cols="25" name="address" rows="5"></textarea><br>  
<input class="submit" name="submit" type="submit" value="Insert">  
</form>  
</div>
```

```
</div>
</body>
</html>
```

Insert form information into database:

PHP Code segment:

```
<?php
$connection = mysql_connect("localhost", "root", ""); // Establishing Connection with Server
$db = mysql_select_db("colleges", $connection); // Selecting Database from Server
if(isset($_POST['submit'])){ // Fetching variables of the form which travels in URL
$name = $_POST['name'];
$email = $_POST['email'];
$contact = $_POST['contact'];
$address = $_POST['address'];
if($name != "" || $email != ""){
//Insert Query of SQL
$query = mysql_query("insert into students(student_name, student_email, student_contact,
student_address) values ('$name', '$email', '$contact', '$address')");
echo "<br/><br/><span>Data Inserted successfully...!!</span>";
}
else{
echo "<p>Insertion Failed <br/> Some Fields are Blank....!!</p>";
}
}
mysql_close($connection); // Closing Connection with Server
?>
```

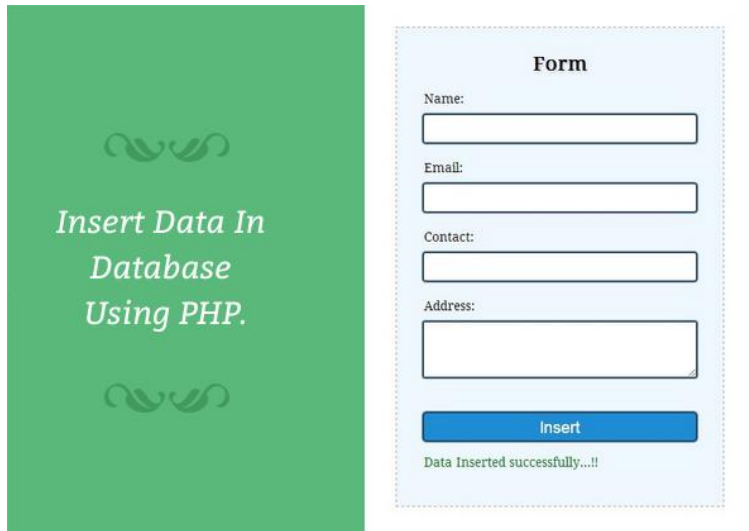
Includes basic styling of HTML elements:

CSS File – insert.css

```
@import "http://fonts.googleapis.com/css?family=Droid+Serif";
/* The Above Line Is To Import Google Font Style */
.maindiv {
margin:30px auto;
width:980px;
height:500px;
background:#fff;
padding-top:20px;
font-family:'Droid Serif',serif;
font-size:14px
}
.title {
width:500px;
height:70px;
text-shadow:2px 2px 2px #cfcfcf;
font-size:16px;
text-align:center
}
.form_div {
width:70%;
float:left
}
form {
width:300px;
```

```
border:1px dashed #aaa;
padding:10px 30px 40px;
margin-left:70px;
background-color:#f0f8ff
}
form h2 {
text-align:center;
text-shadow:2px 2px 2px #cfcfcf
}
textarea {
width:100%;
height:60px;
border-radius:1px;
box-shadow:0 0 1px 2px #123456;
margin-top:10px;
padding:7px;
border:none
}
.input {
width:100%;
height:30px;
border-radius:2px;
box-shadow:0 0 1px 2px #123456;
margin-top:10px;
padding:7px;
border:none;
margin-bottom:20px
}
.submit {
color:#fff;
border-radius:3px;
background:#1F8DD6;
padding:5px;
margin-top:40px;
border:none;
width:100%;
height:30px;
box-shadow:0 0 1px 2px #123456;
font-size:18px
}
p {
color:red;
text-align:center
}
span {
text-align:center;
color:green
}
```

OUTPUT:



The screenshot displays two main components side-by-side. On the left is a green rectangular box with white text that reads "Insert Data In Database Using PHP." The text is centered and flanked by decorative green leaf-like icons. On the right is a light blue form titled "Form" with a dashed border. The form contains four input fields labeled "Name:", "Email:", "Contact:", and "Address:". Below these fields is a blue "Insert" button. At the bottom of the form, a message states "Data Inserted successfully...!!".

RESULT:

The PHP form data is stored into the database.

AIM:

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

PROCEDURE:

1. Open the home page.
2. Enter the login ID and type the comments then submit.
3. Retrieve comments with post id
4. 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
        // retrieve post
        include('config.php');
        include ('function.php');
        dbConnect();

        $query = mysql_query(
            'SELECT *
            FROM post
            WHERE post_id = 1');
        $row = mysql_fetch_array($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
        $comment_query = mysql_query(
            "SELECT *
            FROM comment
            WHERE post_id = {$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

```

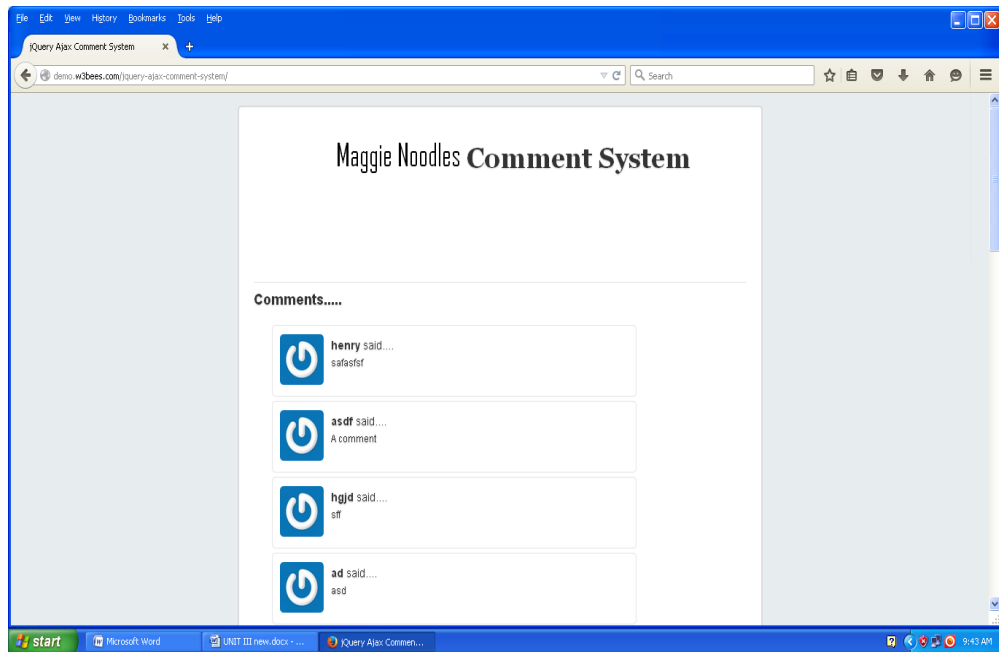
$(document).ready(function(){
  var form = $('form');
  var submit = $('#submit');

  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.

VIVA QUESTIONS AND ANSWERS

1. **What are the three major components of HTML document?**
 - HTML declaration
 - The head
 - The body
2. **List out some of the HTML tags?**
, <DIR>, <DL>, <DT>, <DD>, <MENU>, ,
3. **What kind of tag is HTML?**
Every HTML tag is either a container tag or standalone tag.
4. **How to format tags?**
HTML provides a host of tags that we can use to change how text is displayed on a browser screen.
5. **Define HTML.**
It is a simple page description language, which enables document creation for the web.
6. **List out the document structure tags.**
<HTML>, <HEAD>, <BASE>, <ISINDEX>, <META>, <LINK>, <SCRIPT>, <STYL>, <TITLE>, <BDO>, and <BODY>.
7. **What is CSS?**
CSS stands for Cascading Style Sheets. Styles define how to display HTML elements.
8. **List the features of CSS?**
 - Text handling including fonts, sizing, styles and spacing.
 - Printing specific features, downloadable fonts
 - Support for vertical running text, multicolumn layout facilities.
9. **What are the three main parts of CSS syntax?**
The CSS syntax is made up of three parts: a selector, a property and a value.
10. **Selector {property: value}**
 - The selector is normally the HTML element/tag we wish to define.
 - The property is the attribute we wish to change and each property can take a value.
 - The property and value are separated by a colon and surrounded by curly braces.
11. **How to insert a style sheet?**
When a browser reads a style sheet, it will format the document according to it. There are three ways of inserting a style sheet:
 - External Style Sheet
 - Internal Style Sheet
 - Inline styles
 - Multiple style sheets
12. **List out the basic Internet Protocols.**
TCP/IP, POP3, MIME, IMAP
13. **What are Web Servers?**
A Web Server is software that accepts HTTP requests from web clients and returns an appropriate resource in the HTTP response.
14. **Define Web Clients.**
 - A Web Client is software that accesses a web server by sending an HTTP request message and processing the resulting HTTP response.
15. **How multi-tier architecture is differed from MVC architecture.**
In multi tier architecture client tier never communicates directly with the data tier. All communications must pass through business tier. So it is called linear architecture. Where as in MVC, the view sends updates to controller, then controller updates model. So it is a triangular architecture.
16. **What is Session ID?**

A session ID is a unique identification string usually a long, random and alpha-numeric string, that is transmitted between the client and the server. Session IDs are usually stored in the cookies, URLs (in case url rewriting) and hidden fields of Web pages.

17. What are the common mechanisms used for session tracking?

- Cookies
- SSLsessions
- URL- rewriting

18. Define protocol.

A protocol is a formal set of rules that must be followed in order to communicate.

19. What is the role of server?

- Manages application tasks
- Handles storage
- Handles security
- Provides scalability
- Handles accounting and distribution

20. Define internet.

Network is an interconnection of systems to share data and information. Internet is network of network or collection of heterogeneous networks.

21. Define XML.

XML is a meta-markup language that provides a format for describing structured data. This facilitates more structured declarations of content and more meaningful search result across multiple platforms.

22. Define DTD.

- A DTD is a set of rules that specifies how to use XML markup. It contains specifications for each element, including what the element's attributes are, what values the attributes can take on and what elements can be contained in others.

23. What are the XML rules for distinguishing between the content of a document and the Xml markup element?

- The start of XML markup elements is identified by either the less than symbol (<), the apostrophe or single quote (') and the double quotation marks (") are used by XML for markup.
- To use these special characters as content within your document, you must use the corresponding general XML entity.

24. List some examples of web services.

SMS: <http://www.websvcicex.net/sendsmsworld.asmx2>.

Whois: <http://www.websvcicex.net/whois.asmx?op=GetWhoIS>

25. List out some web service technologies?

- XML
- SOAP
- WSDL

26. What is SOAP?

SOAP - Simple Object Access Protocol - protocol specification for exchanging structured information in the implementation of Web Services in computer networks. - relies on Extensible Markup Language (XML) for its message format, and usually relies on other Application Layer protocols, most notably Hypertext Transfer Protocol (HTTP) and Simple Mail Transfer Protocol (SMTP), for message negotiation and transmission.