

## LIST OF EXPERIMENTS

Ex.No.	TITLE OF THE EXPERIMENT	Page No.
1	CREATING A WEBPAGE USING IMAGE MAP	2
2	CREATING A WEBPAGE WITH CASCADING STYLE SHEETS	7
3	CREDIT CARD VALIDATION USING JAVASCRIPT	12
4 (a)	TO INVOKE SERVLETS FROM HTML FORMS	24
4 (b)	SESSION TRACKING USING HIDDEN FORM FIELDS	28
4 (c)	SESSION TRACKING FOR A HIT COUNT	30
5	THREE-TIER ARCHITECTURE – ONLINE EXAMINATION	32
6	CONVERTING STATIC WEB PAGES INTO DYNAMIC WEB PAGES USING SERVLETS	37
7	CONVERTING STATIC WEB PAGES INTO DYNAMIC WEB PAGES USING JSP	41
8	PARSING XML DOCUMENT	45
9	VALIDATE THE FORM USING PHP REGULAR EXPRESSION	48
10	IMPLEMENTING AN APPLICATION WITH WEB SERVICES	50

<b>Ex No: 1</b>	<b>CREATING A WEBPAGE USING IMAGE MAP</b>
<b>Date:</b>	

### AIM:

To create a web page with the following using HTML:

- i) To embed an image map in a web page.
- ii) To fix the hot spots.
- iii) Show all the related information when the hot spots are clicked.

### ALGORITHM:

Step 1: Open notepad and type the HTML coding for homepage home.html which has an image map using <MAP> tag and create some hotspots

Step 2: Hotspots are created by including a link at required coordinate position using <a> tag which directs to its corresponding webpages

Step 3: Write the coding for all the link webpages

Step 4: Run the home.html in suitable web browser

Step 5: Display output.

### PROGRAM:

#### HOME.HTML

```
<html>
<head>
<title>Home - States of India!!!</title>
</head>
<body bgcolor="gold">
<h1><u><center>Republic of India</center></u></h1>
<p>
India is the Seventh Largest country in the world by geographical area, the second most
Populous country with over 1.3 billion people, India is a vast South Asian country with diverse
terrain – from Himalayan peaks to Indian Ocean coastline – and history reaching back 5
millennia. . India is a federal constitutional republic with a parliamentary democracy consisting
of 28 states and 7 Union Territories.
</p>
<center>

<map name="india">
<area shape="circle" coords="100,200,10" href="ANDHRAPRADESH.html" alt="Learn about
andra">
<area shape="circle" coords="70,275,10" href="KERALA.html" alt="Learn about kerala">
<area shape="circle" coords="70,210,20" href=" KARNATAKA.html" alt="Learn about
karnataka">
<area shape="circle" coords="100,250,20" href="TAMILNADU.html" alt="Learn about
tamilnadu">
</map></center>
```

## <h2> Features</h2>

<ul>

<li><b>Population</b> - 1.252 billion (2013).

<li><b>Capital</b> - New Delhi

<li><b>Largest City</b> - Mumbai

<li><b>Currency</b> - Indian Rupee

<li><b>Time Format</b> - IST (UTC + 5:30)

<li><b>National Sport</b> - Hockey

<li><b>Current PM</b> - Narendra Modi

<li><b>Current President</b> - Pranab Mukherjee

</li>

</ul>

<h2><b>To view details of southern states please click on the specified area in the map!</b></h2>

</h2>

</body>

</html>

## TAMILNADU.HTML

<html>

<head><title>Tamil Nadu - India</title></head>

<body bgcolor="palegreen">

<h1><center>Tamil Nadu</center></h1>

<h3>is one of the 28 states of India. Its capital and largest city is Chennai.

Tamil Nadu lies in the southernmost part of the Indian Peninsula and

It is bordered by the States of Puducherry, Kerala, Karnataka, Andhra Pradesh

</h3><h3>

<ul>

<li>Districts<i> - 32</i>

<li>Capital City<i> - Chennai</i>

<li>Largest City<i> - Chennai</i>

<li>Governor<i> - Konijeti Rosaiah</i>

<li>Chief Minister<i> - Jayalalithaa</i>

<li>Population<i> - 72,138,958</i>

<li>Tourist spots<i> - Mamallapuram, Ooty, Kodaikanal, Marina, Mudurai Meenakshi Amman Temple, Thanjavur etc.,</i>

</ul>

<a href="Home.html">back</a>

</body>

</html>

## ANDRAPRADESH.HTML

<html>

<head><title>Andhra Pradesh - India</title></head>

<body bgcolor="tan">

<h1><center>Andhra Pradesh</center></h1>

<h3>A.P., is a state situated on the southeastern coast of India. It is

India's fourth largest state by area and fifth largest by population.</h3>

```
<h3>
<ul>
<li>Districts<i> - 23</i>
<li>Capital City<i> - Hyderabad</i>
<li>Largest City<i> - Hyderabad</i>
<li>Governor<i> - E. S. L. Narasimhan</i>
<li>Chief Minister<i> - N. Kiran Kumar Reddy</i>
<li>Population<i> - 78,323,330</i>
<li>Tourist spots<i> - Tirumala Tirupati, Charminar, Golconda Fort,
Chandragiri, Chowmahalla Place, Falaknuma Palace etc.,</i>
</ul>
<a href="Home.html">back</a>
</body>
</html>
```

## KARNATAKA.HTML

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

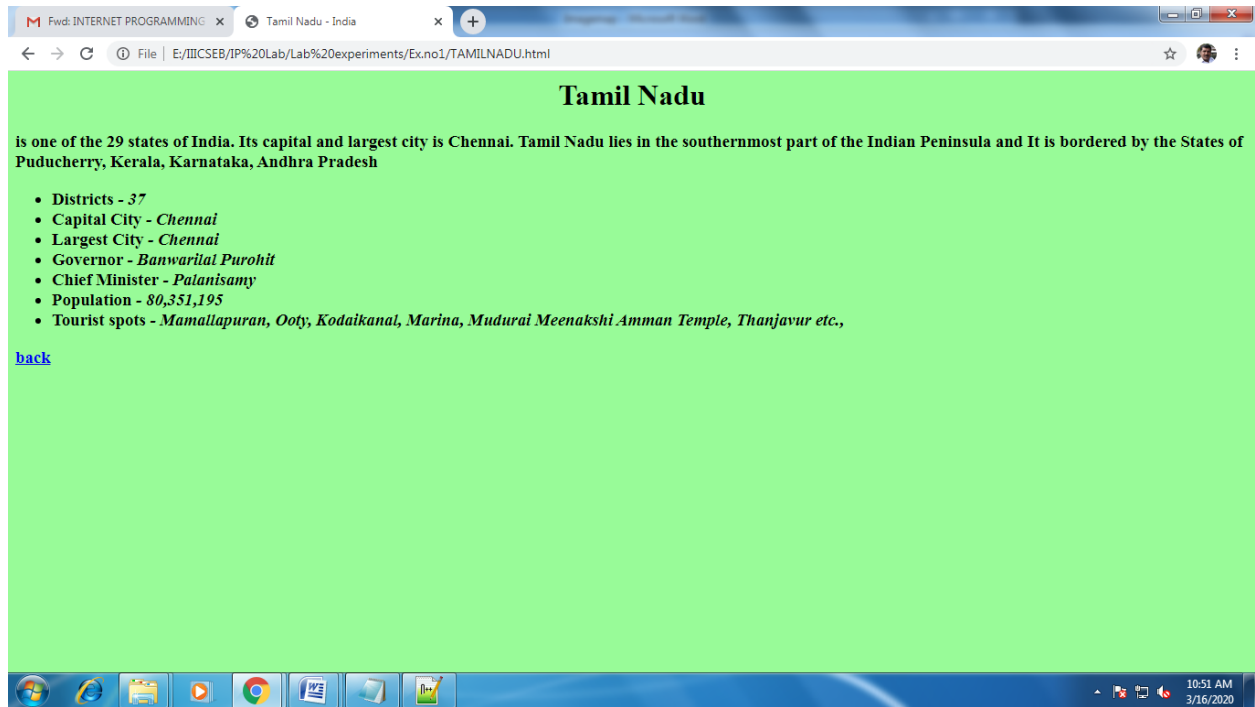
## KERALA.HTML

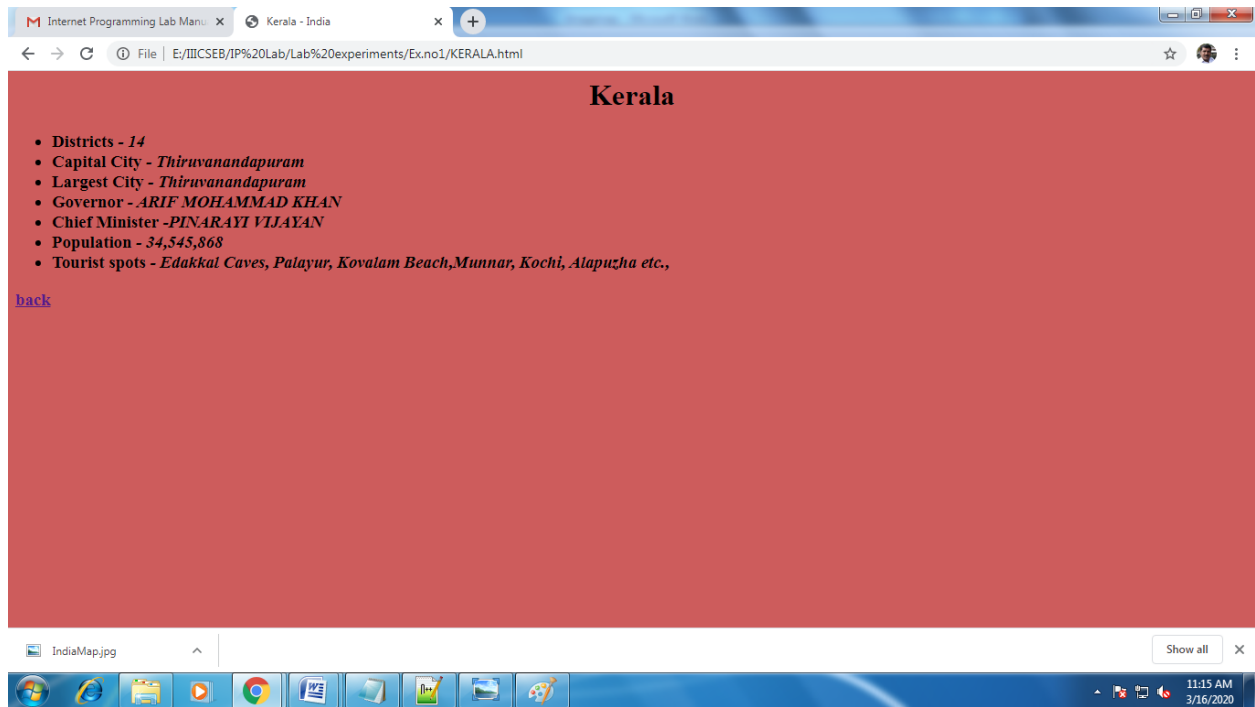
```
<html>
<head><title>Kerala - India</title></head>
<body bgcolor="indianred">
<h1><center>Kerala</center></h1>
<h3>
<ul>
<li>Districts<i> - 14</i>
<li>Capital City<i> - Thiruvananthapuram</i>
<li>Largest City<i> - Thiruvananthapuram</i>
<li>Governor<i> - P.Sathasivam</i>
<li>Chief Minister<i> - Oommen Chandy </i>
```

```
</li>Population</li> - 33,387,677</li>
</li>Tourist spots</li> - Edakkal Caves, Palayur, Kovalam Beach,Munnar, Kochi, Alapuzha
etc.,</li>
</ul>
<a href="Home.html">back</a>
</h3>
</body>
</html>
```

## OUTPUT:







## RESULT:

Thus a webpage with given specifications was created and its output was verified.

<b>Ex No: 2</b>	<b>CREATING A WEBPAGE WITH CASCADING STYLE SHEETS</b>
<b>Date:</b>	

### AIM:

Create a web page with the following:

- Cascading style sheets.
- Embedded style sheets.
- Inline style sheets.

### ALGORITHM:

#### *For Internal Style sheets,*

Use <style> tag with required styling options of CSS defined within the <head> tag of the main page.

Use selectors when multiple styling is to be applied on same element at different places.

Save and run the html page defined with internal CSS.

#### *For External Style sheets,*

Open notepad and type CSS code with appropriate styling and save the file with .css extension

Use anchor tag <a> to link to the created *css file* in <head> field of html page.

Save and run the html page defined to link to external style sheet.

### HTML CODING:

#### Homepage.html

```
<html>
<head><h1><center>VELAMMAL GROUP OF INSTITUTIONS</center></h1>
<title>
</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}
</style>
</head>
<body bgcolor="cyan">
<ol style="list-style-type:lower-alpha">
<b>VELAMMAL INSTITUTION</b><br><br><br>
<li>Velammal Engineering College
<li>Velammal Institute of Technology
<li>Velammal College of Enginnering and Technology
</ol>
<p style="font-size:20pt;color:purple">VELAMMAL INSTITUTE OF TECHNOLOGY</p>
<br><br>
<p class="ani">Approved by AICTE and Affiliated to AnnaUniversity.<br><br></p>
```



## About Us



**Velammal Institute of Technology,** brain child of our Honorable Chairman bloomed on 14th August 2008 intended to provide the world class technical education to conquer tomorrow's technology at global level with human values, so as to make our student community succeed in life.

The institution has developed the state-of-art infrastructure within the short span of time with dedicated and determined staff members.

Velammal Institute of Technology (VITech) is committed to be a leader in innovative work force development and a world class learning venue for the ever changing needs of Business and Industry.

## Academics Overview

### **UG B.E/B.TECH PROGRAM**

- Computer Science and Engineering

- Electronics and Communication Engineering

- Mechanical Engineering

- Information Technology

- Electrical and Electronics Engineering

### **PG PROGRAMME:**

||
||
||

<b>PG PROGRAMME:</b>
----------------------

||
||
||

<b>MBA (Master of Business Administration)</b>
--

<b>2yrs</b>
-------------

||
||
||

<b>M.E Computer Science and Engineering</b>
---

<b>2yrs</b>
-------------

||
||
||

<b>M.E Embedded System Technologies</b>
---

<b>2yrs</b>
-------------

||
||
||

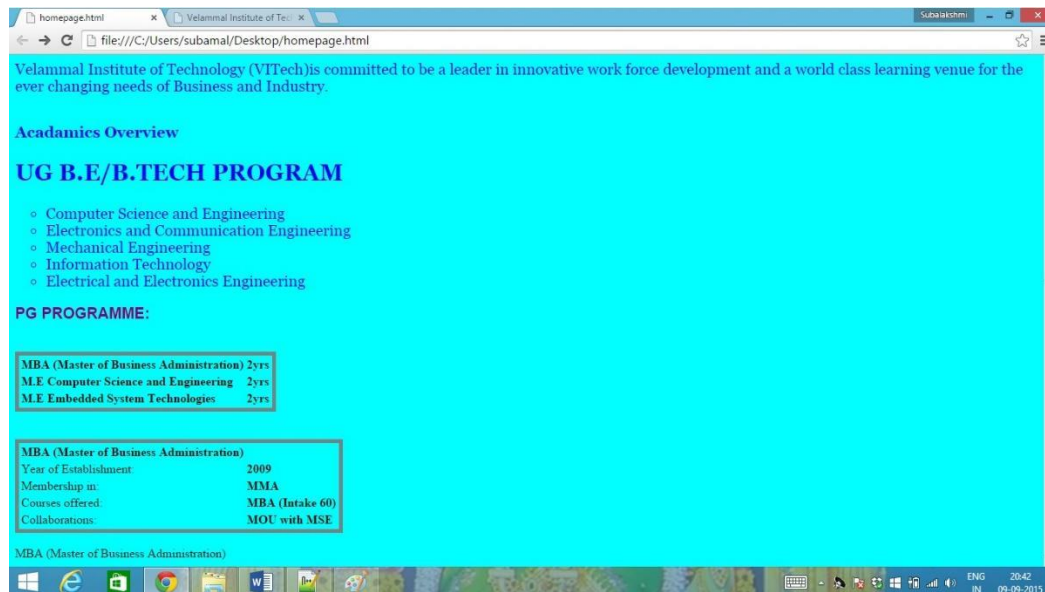
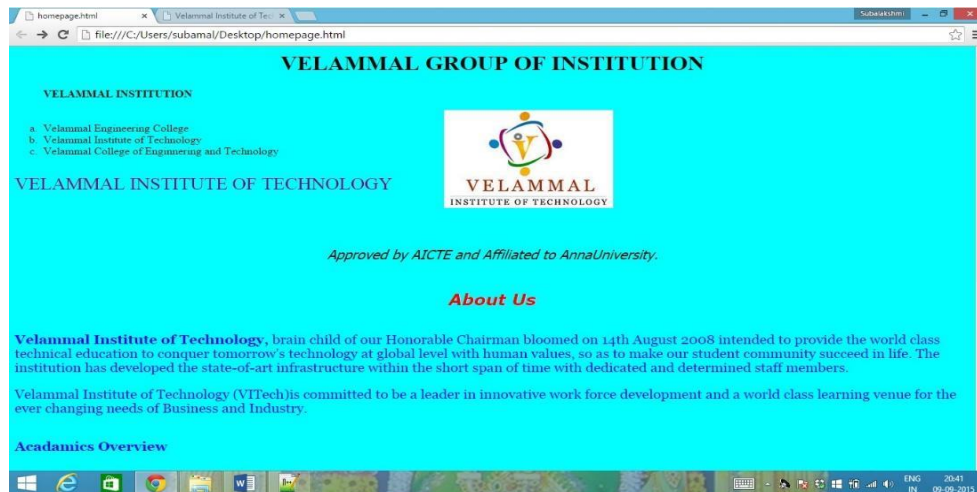
<b>MBA (Master of Business Administration)</b>
--

```

</tr>
<tr>
<td>Year of Establishment:</td>
<td><strong>2009</strong></td>
</tr>
<tr>
<td>Membership in:</td>
<td><strong>MMA</strong></td>
</tr>
<tr>
<td>Courses offered:</td>
<td><strong>MBA
(Intake 60)</strong></td>
</tr>
<tr>
<td>Collaborations:</td>
<td><strong>MOU with MSE</strong></td>
</tr>
</table>
<br/>
MBA (Master of Business Administration)
</table>
</body>
</html>

```

## OUTPUT:



## RESULT:

Thus a webpage with following specifications are developed using different style sheet and its output was verified.

<b>Ex No: 3</b>	<b>CREDIT CARD VALIDATION USING JAVASCRIPT</b>
<b>Date:</b>	

### AIM:

To write a program to validate the Registration, user login, user profile and payment by credit card pages using JavaScript.

### PROCEDURE:

#### ● Home page:

##### Main.html:

```
<html>
<frameset rows="25%,*">
<frame src="top.html" name="top" scrolling="no" frameborder="0">
<frameset cols="25%,75%">
<frame src="left.html" name="left" scrolling="no" frameborder="0">
<frame src="right.html" name="right" scrolling="auto" frameborder="0">
</frameset>
</frameset>
</html>
```

##### 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>
<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="profile.html" target="right"><font color="black"> USER
PROFILE</font></a></li><br><br>
<li><a href="catalog.html" target="right"><font color="black"> BOOKS
CATALOG</font></a></li><br><br>
<li><a href="scart.html" target="right"><font color="black"> SHOPPING
CART</font></a></li><br><br>
<li><a href="payment.html" target="right"><font color="black">
PAYMENT</font></a></li><br><br>
<br><br>
</ul>
</body>
</html>
```

## Login.html

14

## User Profile Page

### Profile.html

```
<html>

<body bgcolor="pink"><br><br>

<script type="text/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;
}

alert("Enter all the details");
var str=document.myform.phno.value;
var x=new RegExp("\\d","g");
if(!(str.match(x)))
{
if(!(str.length==10))
flag=0;
}

var str1=document.myform.id.value;
var x1=new RegExp("^[A-Z][a-zA-Z]+$","g");

if(!(str1.match(x1)))
{
flag=0;
alert("Invalid UserID");
}

var str1=document.myform.pwd.value;

var x1=new RegExp("^[A-Z][a-zA-Z- Z]+$","g"); if(!(str1.match(x1)))
{
```

```
if(flag==1)
{
flag=0;
alert("Invalid password");
}
alert("VALID INPUT");
window.self.location.href="login.html";
}
else
{
alert("INVALID INPUT");
document.myform.focus();
<br><br>
}
}

</script>

<form name="myform">
<div align="center"><pre>
NAME:<input type="text" name="name"><br>
ADDRESS :<input type="type" 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>

<div align="center">

<input type="button" value="ok" onClick="validate()">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~
<input type="reset" value="clear">

</form></body></html>
```



## Books Catalog

### Scart.html

```
<html>

<body bgcolor="pink"><br><br><br>

<script
language="javascript">
function validate()
{
var flag=1; if(document.myform.title.value=="")
{
flag=0;
}
str=document.myform.title.value;
if(str=="c"||str=="C")
{ document.myform.t1.value="C"; document.myform.t2.value=444;
}
else if(str=="jsp"||str=="JSP")
else
{ document.myform.t1.value="JSP"; document.myform.t2.value=555;
}
{
flag=0;
}
if(flag==1)
{
alert("VALID INPUT");
}
else
{
alert("INVALID INPUT");
document.myform.focus();
}
}
```



[illegible]

## Payment of Credit Card

### Payment.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=="||
document.myform.amount.value=="|| document.myform.num.value=="")
{
flag=0;
}

var str=document.myform.amount.value; var x=new RegExp("\\d","g");
if(!(str.match(x)))
{

flag=0;
}

var str1=document.myform.num.value; var
x1=new RegExp("\\d","g");

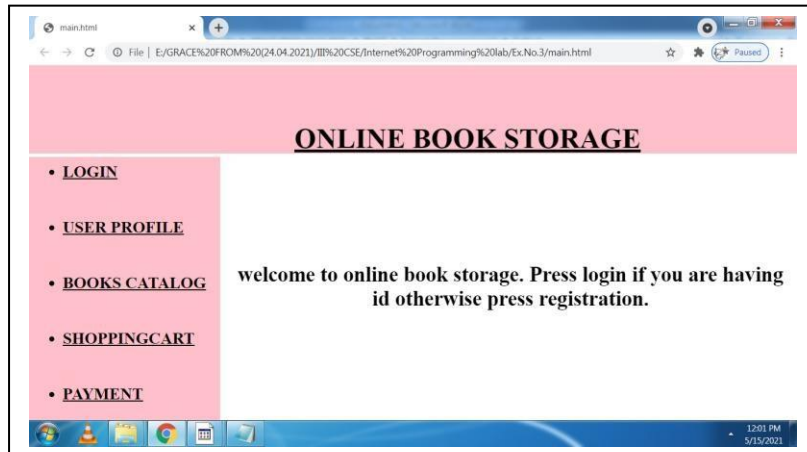
if(!(str.match(x1)))
{
flag=0;
}

if(flag==1)
{
alert("VALID INPUT");
window.self.location.href="order.html";
}
else
{
alert("INVALID INPUT");
document.myform.focus();
}
```

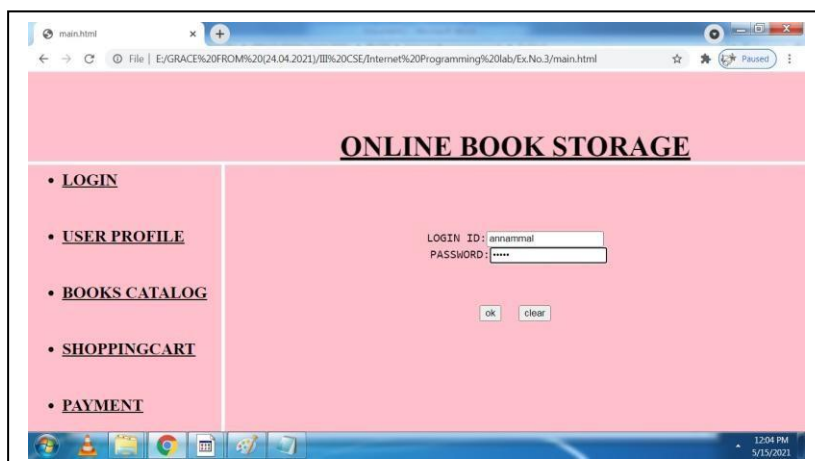


## OUTPUT

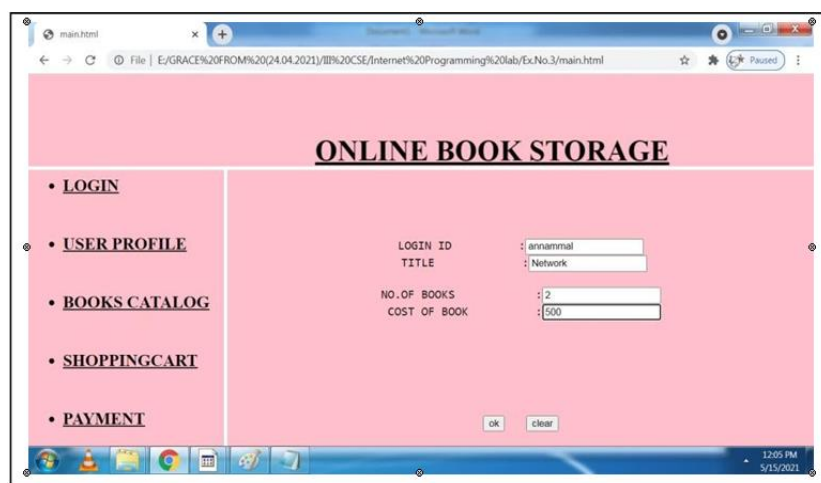
### Main.html



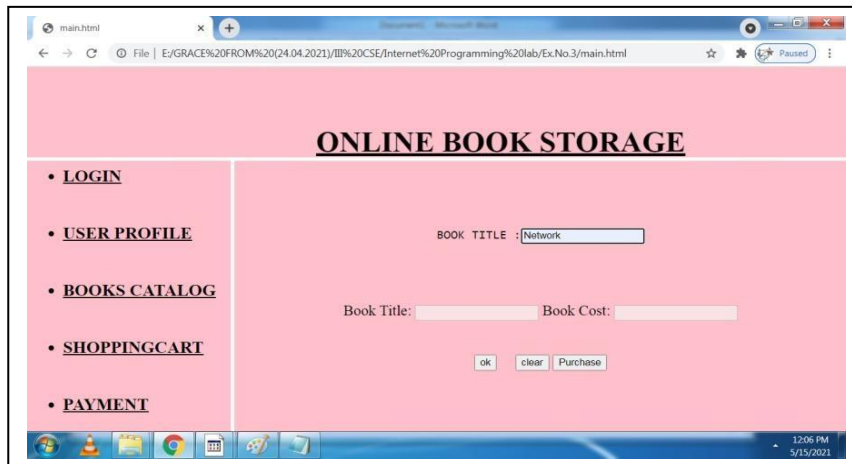
### Login.html



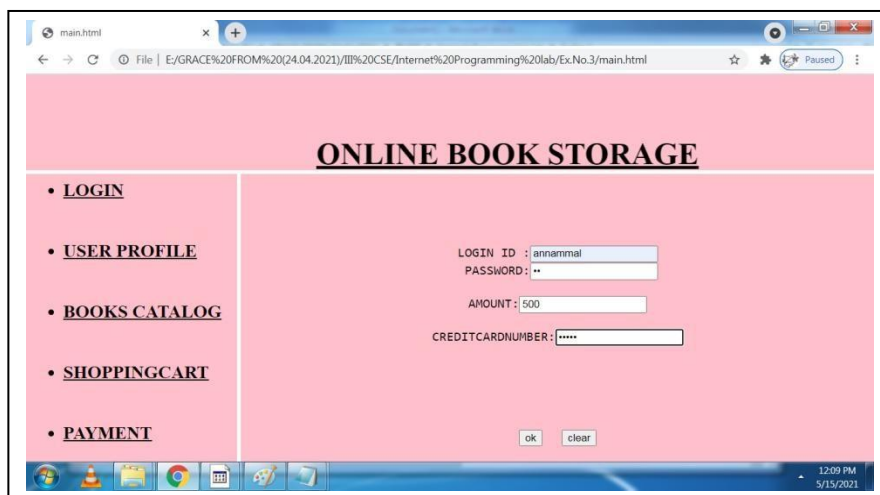
### Catalog.html



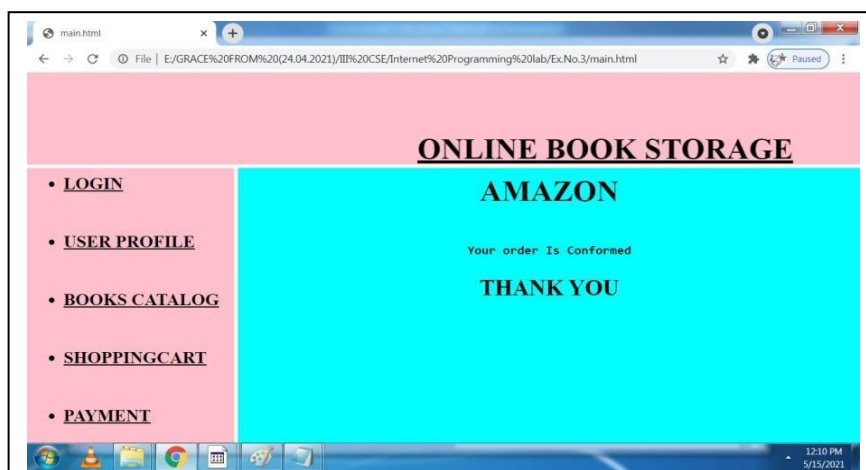
## Scart.html



## Payment.html



## Order.html



## RESULT:

Thus the program was executed successfully.

<b>Ex.No.4 (a)</b>	<b>INVOKING SERVLETS FROM HTML FORM</b>
<b>Date:</b>	

### AIM:

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

### ALGORITHM:

#### 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.

#### 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 out=res.getWriter();

        Enumeration en=req.getParameterNames();
        while(en.hasMoreElements())
        {
            String name_received=(String)en.nextElement();
            out.println(name_received + "=");
            String value_received=req.getParameter(name_received);
            out.println(value_received);
            out.println(" ");
        }
        out.close();
    }
}
```



### **web.xml:**

```
<servlet>
    <servlet-name>server</servlet-name>
    <servlet-class>server</servlet-class>
</servlet>

<servlet-mapping>
    <servlet-name>server</servlet-name>
    <url-pattern>/servlet/server</url-pattern>
</servlet-mapping>
```

### **Client.HTML:**

```
<html>
<head>
<title>Invoking Servlet From HTML</title>
<title> Student Information Form </title>
</head>
<body bgcolor="violet">

<center>
<form name="form1" method="post" action="http://localhost:8080/examples/servlet/server">

<h3>Enter Student Information </h3>
<table>
<tr>
<td><b> Roll Number</b> </td>
<td><input type="text" name="Roll number" size="25" value=" "><br/>
</td>
</tr>

<tr>
<td><b> Student Name</b> </td>
<td><input type="text" name="Student Name" size="25" value=" "><br/>
</td>
</tr><br>

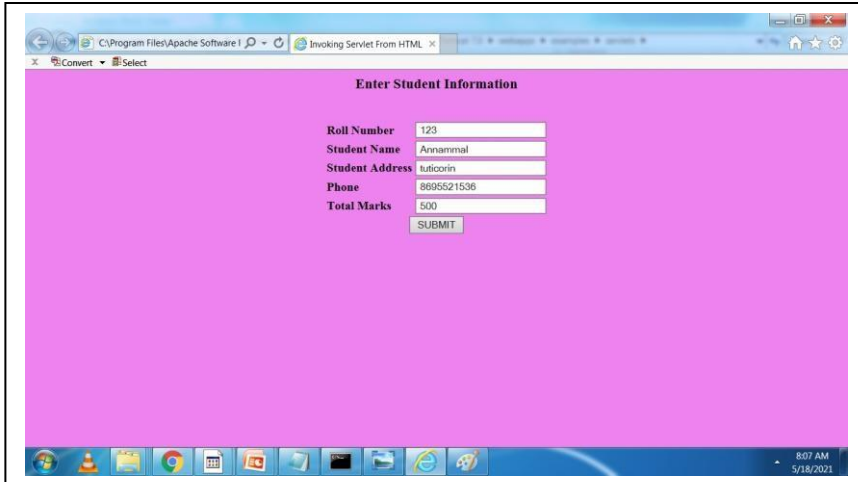
<tr>
<td><b> Student Address</b> </td>
<td><input type="text" name="Address" size="25" value=" "><br/>
</td>
</tr>

<tr>
<td><b> Phone</b> </td>
<td><input type="text" name="Phone" size="25" value=" "><br/>
</td>
</tr>
```

```
<tr>
<td><b> Total Marks</b> </td>
<td><input type="text" name="Total Marks" size="25" value=" "><br/>
</td>
</tr>
</table>
```

```
<input type="submit" Value="SUBMIT">
</form>
</center>
</body>
</html>
```

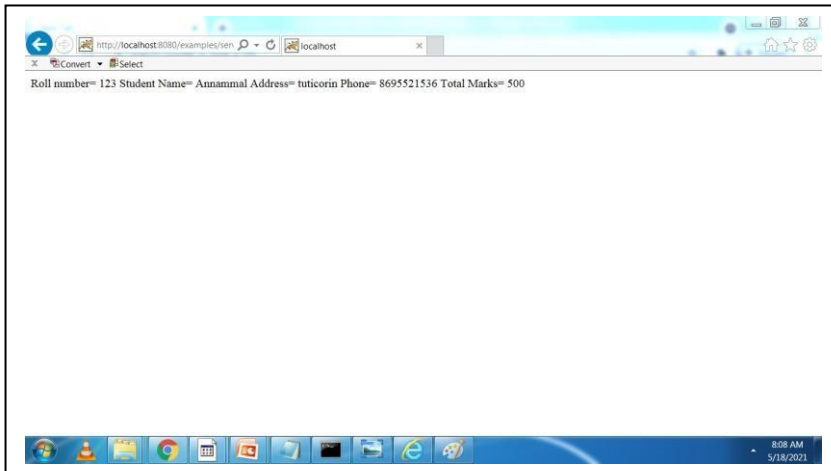
## OUTPUT:



The screenshot shows a web browser window with the title bar 'C:\Program Files\Apache Software... Invoking Servlet From HTML'. The address bar shows 'http://localhost:8080/examples/ven'. The page content is a form titled 'Enter Student Information' with a pink background. The form contains the following fields and values:

Field	Value
Roll Number	123
Student Name	Annammal
Student Address	tuticorin
Phone	8695521536
Total Marks	500

Below the fields is a 'SUBMIT' button. The Windows taskbar at the bottom shows the time as 8:07 AM on 5/18/2021.



The screenshot shows the same web browser window after the form has been submitted. The address bar remains 'http://localhost:8080/examples/ven'. The page content now displays the output of the servlet:

Roll number= 123 Student Name= Annammal Address= tuticorin Phone= 8695521536 Total Marks= 500

The Windows taskbar at the bottom shows the time as 8:08 AM on 5/18/2021.

## RESULT:

Thus the program to invoke servlets from HTML form was successfully executed.

<b>Ex.No.4 (b)</b>	<b>SESSION TRACKING USING HIDDEN FORM FIELDS SESSION TRACKING FOR A HIT COUNT</b>
<b>Date:</b>	

### AIM:

To write a program to implement session tracking in servlets using hidden form fields.

### ALGORITHM:

1. Create a servlet program using http for session tracking using hidden as an input type in hidden form fields.
2. Set classpath where servlet-api.jar file resides.
3. Compile the servlet program using javac programname.java
4. Place the class file ...\\Tomcat 5.5\\webapps\\ROOT\\WEB-INF\\classes\\ folder.
5. Modify the web.xml file using your servletClassName.
6. Invoke the class file using http://localhost:8080/servletClassName from your browser

### 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>");
```

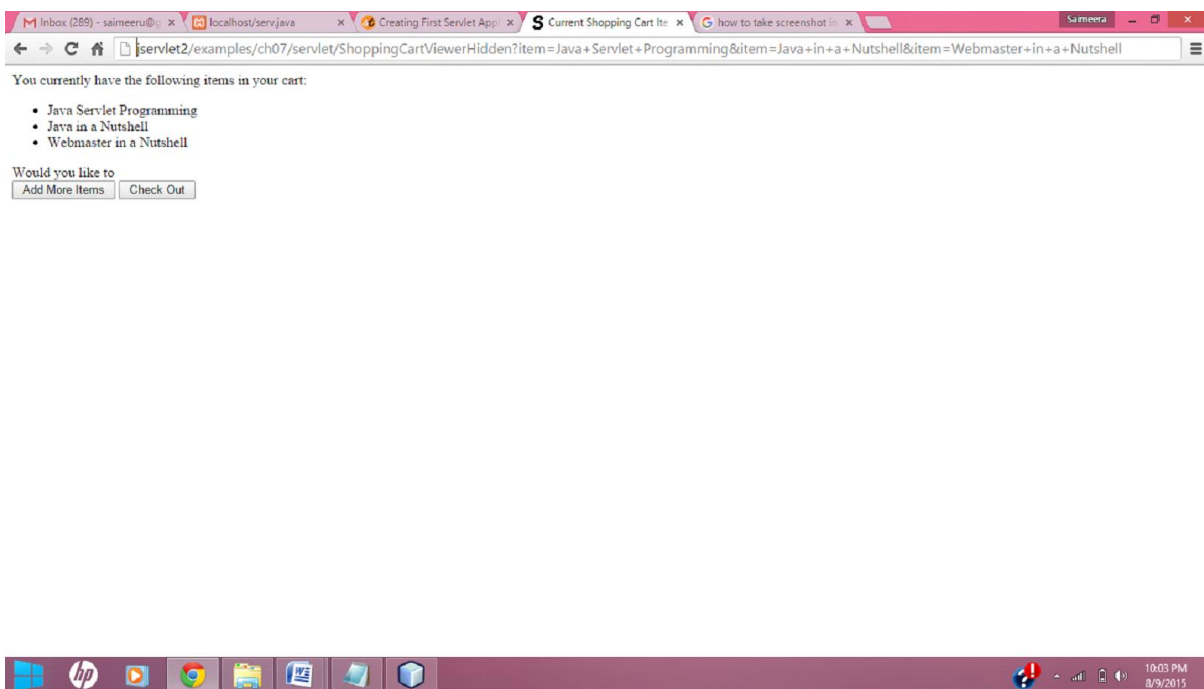
```
        }
```

```
        // 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>");
}
}
```

## OUTPUT:



## RESULT:

Thus a program was implemented for session tracking using hidden form fields

<b>Ex.No.4 (b)</b>	<b>SESSION TRACKING USING HIT COUNT</b>
<b>Date:</b>	

### AIM:

To write a program to implement session tracking in servlets using hit count.

### ALGORITHM:

1. Create a servlet program using http in which a count is maintained for session tracking.
2. Set classpath where servlet-api.jar file resides.
3. Compile the servlet program using javac programname.java
4. Place the class file ...\\Tomcat 5.5\\webapps\\ROOT\\WEB-INF\\classes\\ folder.
5. Modify the web.xml file using your servletClassName.
6. .Invoke the class file using http://localhost:8080/servletClassName from your browser

### 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();
```

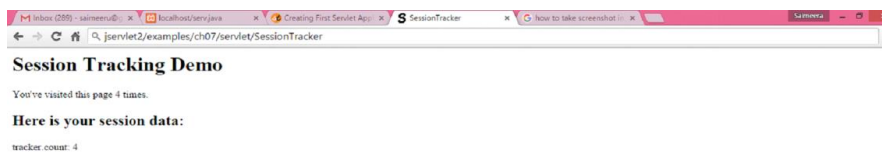
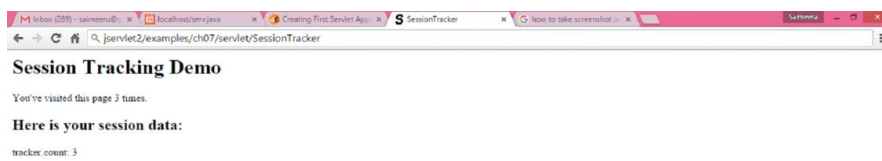
```
    out.println(name + ": " + session.getAttribute(name) + "<BR>");
```

```
}
```

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

```
}}
```

## OUTPUT:



## RESULT:

Thus a program was implemented for session tracking using hit count.

<b>Ex No: 5</b>	<b>THREE-TIER ARCHITECTURE – ONLINE EXAMINATION</b>
<b>Date:</b>	

### AIM:

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

### ALGORITHM:

1. Create a HTML index page which accepts the Seat number and name followed by the answers for a set of questions.
2. Invoke the servlet when user presses submit button.
3. In servlet enable JDBC connection wherein you have already created an SQL table called student with seatno, name and total as its column.
4. Insert the entry of user by query and execute query update, create another query to display all the entries in the table and Execute that query to display the final output.
5. Include appropriate exception handlers for each key steps involved.

### 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,"root ","saphira ");
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){
System.out.println("Query not executed");
}
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){
System.out.println("Query didnt select any row!");
} 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:

```

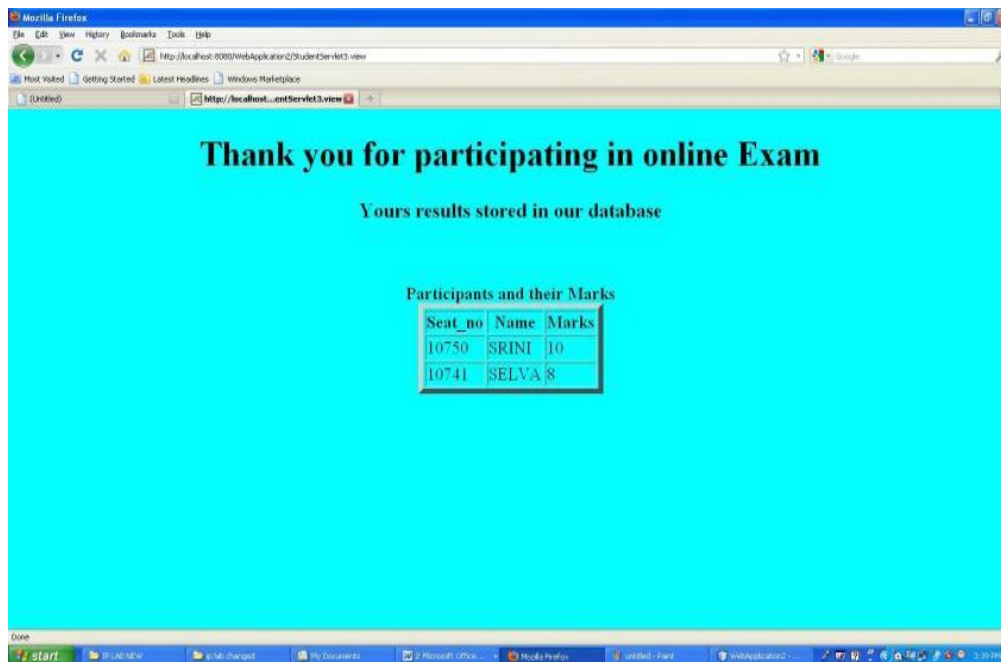
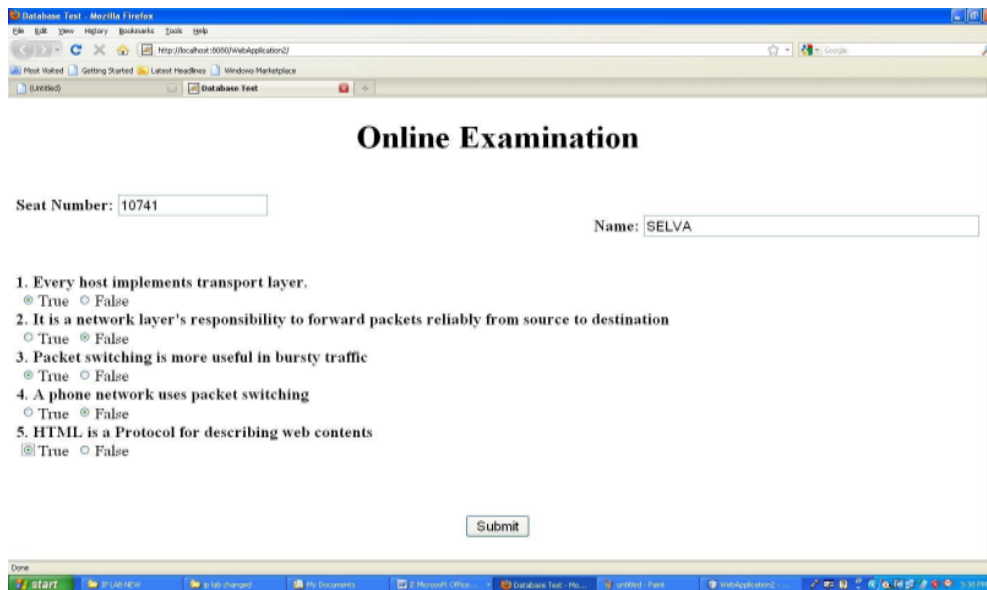
<html>
<head><title>Database Test</title></head><body>
<center>
<h1>Online Examination</h1></center>
<form action="StudentServlet3.view" method="POST">
<div align="left"><br></div>
<b>Seat Number:</b><input type="text" name="Seat_no"><div align="Right">
<b>Name:</b><input type="text" name="Name" size="50"><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 destination
</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 a java servlet program to conduct online examination was successfully executed and the student marks are saved in database.

<b>Ex No: 6</b>	<b>CONVERTING STATIC WEB PAGES INTO DYNAMIC WEB PAGES USING SERVLETS</b>
<b>Date:</b>	

### **AIM:**

To convert the static web pages into dynamic web pages using servlets and cookies.

### **PROCEDURE:**

Step 1: we will create a html form for entering the user name,password and card ID.

Step 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.

Step 3: compile the above servlet Login servlet.java and copy its class file in tomcat folder at c:\tomcat directory\webapps\examples\WEB-INF\classes. 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.

Step 4: On successful login, the information from the cookie is checked and shopping cart page for corresponding user can be displayed.

Step 5: 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.

Step 6: Start tomcat web server. Open the web browser and display the login form created in step1.

## Server.java

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

public class server 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:8080/examples/servlet/success");
                }
            }
            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:8080/examples/servlet'>"+ "in.html");
            }
        }

        finally{
            {
                out.close();
            }
        }
    }
}
```

## Success.java

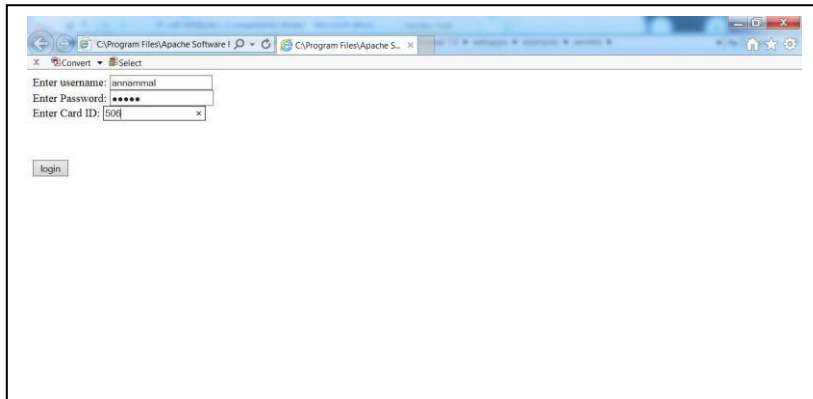
```
import java.io.*;
import java.net.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class success 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();
    }
}
```

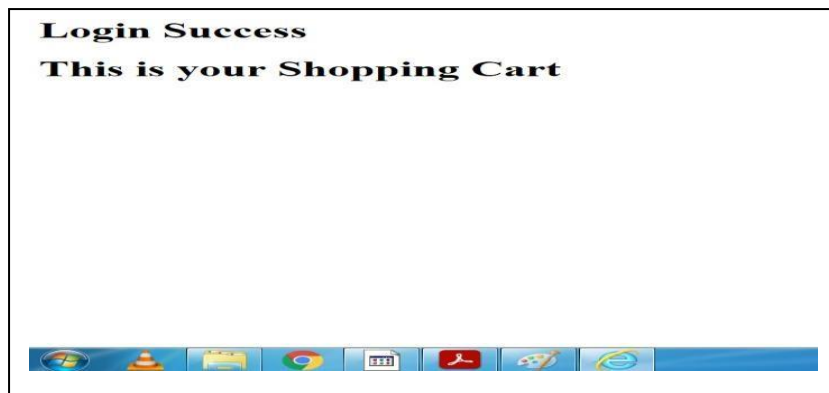
## In.html

```
<html>
<head>
<body>
<form action="http://localhost:8080/examples/servlet/ser" 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>
```

## OUTPUT



A screenshot of a web browser window. The address bar shows the path "C:\Program Files\Apache Software F...". The page contains a login form with three input fields: "Enter username:" with the text "lanname1", "Enter Password:" with masked characters "\*\*\*\*\*", and "Enter Card ID:" with the text "500". Below these fields is a "login" button.



## RESULT:

Thus the program to convert the static web pages into dynamic web pages using servlets and cookies was successfully executed.



<b>Ex No: 7</b>	<b>CONVERTING STATIC WEB PAGES INTO DYNAMIC WEB PAGES USING JSP</b>
<b>Date:</b>	

### AIM:

To create a database with user information and books information. The books catalogue should be dynamically loaded from the database.

### PROCEDURE:

- 1) Create your own directory under tomcat/webapps (e.g. tr1)
- 2) Copy the html files in tr1
- 3) Copy the jsp files also into tr1
- 4) Start tomcat give the following command  
Catalina.bat run  
At install-dir/bin
- 5) give url as <http://localhost:8081/tr1/main.html>

### Program

#### Register.java

```
package demotest;
import java.io.IOException;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException; import
javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest; import
javax.servlet.http.HttpServletResponse;

public class register extends HttpServlet {
private static final long serialVersionUID = 1L;
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    // TODO Auto-generated method stub
    String first_name = request.getParameter("first_name");
    String last_name = request.getParameter("last_name");
    String username = request.getParameter("username");
    String password = request.getParameter("password");
    String address = request.getParameter("address");
    String contact = request.getParameter("contact");
    if(first_name.isEmpty() || last_name.isEmpty() || username.isEmpty() || password.isEmpty() ||
    address.isEmpty() || contact.isEmpty())
    {

    RequestDispatcher req =request.getRequestDispatcher("aa.jsp");
    req.include(request, response);
    }
    else
    {
    RequestDispatcher req = request.getRequestDispatcher("bb.jsp");
    req.forward(request, response);
    }}}
}
```

## aa.jsp

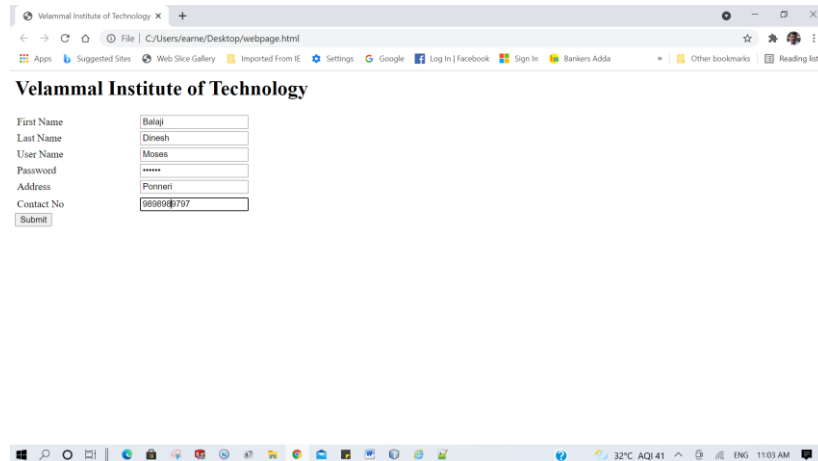
```
<% @ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"% >
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Velammal Institute of Technology</title>
</head>
<body>
<h1> Velammal Institute of Technology </h1>
<form method="post">
<table style="width: 50%">
<tr>
<td>First Name</td>
<td><input type="text" name="first_name"/></td>
</tr>
<tr>
<td>Last Name</td>
<td><input type="text" name="last_name"/></td>
</tr>
<tr>
<td>User Name</td>
<td><input type="text" name="username" /> </td></td>
</tr>
<tr>
<td>Password</td>
<td><input type="password" name="password"></td>
</tr>
<tr>
<td>Address</td>
<td><input type="text" name="address" /></td>
</tr>
<tr>
<td>Contact No</td>
<td><input type="text" name="contact" /></td>
</tr>
</table>
<input type="submit" value="Submit" /></form>
</body>
</html>
```

## bb.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title> Success Page</title>
</head>
<body>
    <a><b>Welcome User!!!!</b></a>
</body>
</html>
```

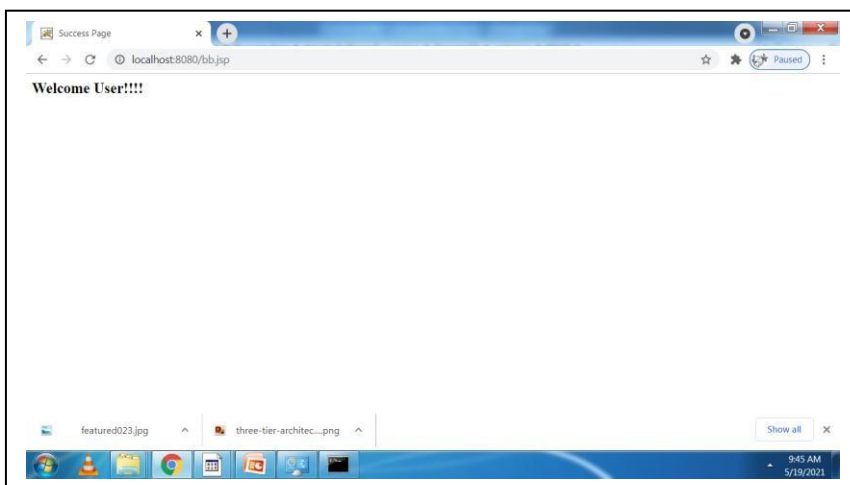
## OUTPUT



The screenshot shows a web browser window with the title "Velammal Institute of Technology". The address bar displays "C:/Users/earne/Desktop/webpage.html". The page contains a login form with the following fields and values:

Field	Value
First Name	Balaji
Last Name	Dinesh
User Name	Moses
Password	*****
Address	Ponneri
Contact No	989898797

A "Submit" button is located below the form fields. The browser's taskbar at the bottom shows the system clock as 11:03 AM on 5/19/2021.



## RESULT:

Thus the program was executed successfully

<b>Ex No:8</b>	<b>PARSING XML DOCUMENT</b>
<b>Date:</b>	

### 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

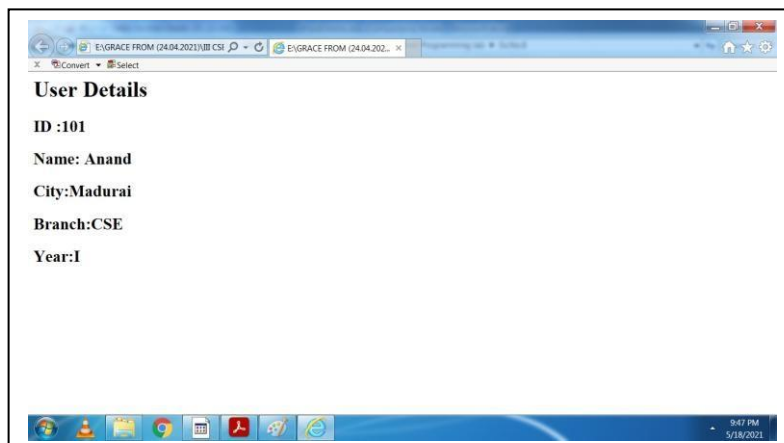
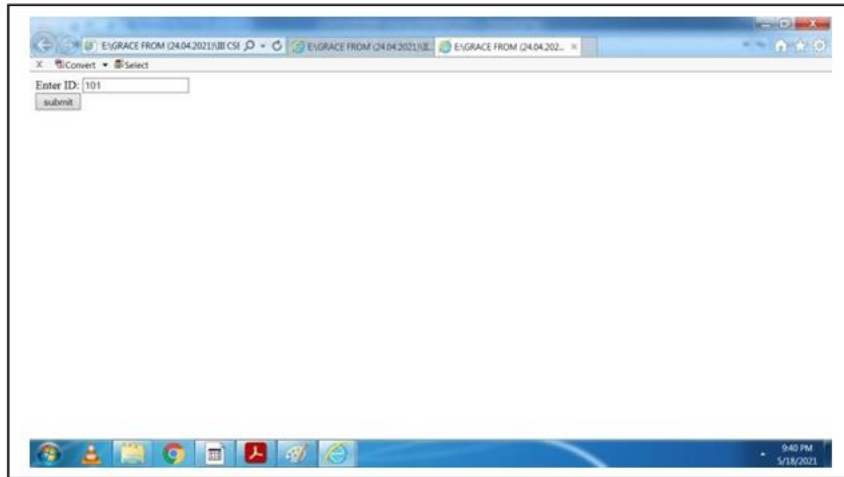
### *Student.xml*

```
<?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","student.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 to create and save an XML document at the server was executed successfully.

<b>Ex No: 9</b>	<b>VALIDATE THE FORM USING PHP REGULAR EXPRESSION</b>
<b>Date:</b>	

**AIM:**

To write a PHP program to validate the form using PHP regular expression.

**ALGORITHM:**

Step1 : PHP validates the data at the server-side, submitted by HTML form

Step2 : Create the Php program and store the file in the xampp→ htdocs

Step3: Validate Empty String, Validate String, Validate Numbers, Validate Email Validate URL, input length.

Step4: select Start →Run and type http://localhost/hello.php to execute the php program.

Step 5 : Stop the program.

**PROGRAM:**

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

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $name = test_input($_POST["name"]);
    $email = test_input($_POST["email"]);
    $website = test_input($_POST["website"]);
    $comment = test_input($_POST["comment"]);
    $gender = test_input($_POST["gender"]);
}
```

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

<h2>PHP Form Validation </h2>

```
<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>">
    Name: <input type="text" name="name">
    <br><br>
```



```

E-mail: <input type="text" name="email">
<br><br>
Website: <input type="text" name="website">
<br><br>
Comment: <textarea name="comment" rows="5" cols="40"></textarea>
<br><br>
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
<br><br>
<input type="submit" name="submit" value="Submit">
</form>

<?php
echo "<h2>Your Input:</h2>"; echo
$name;
echo "<br>"; echo
$email; echo
"<br>"; echo
$website; echo
"<br>"; echo
$comment; echo
"<br>"; echo
$gender;
?>
</body>
</html>

```

## OUTPUT

**PHP Form Validation**

Name:

E-mail:

Website:

Comment:

Gender: ☒ Female ☐ Male ☐ Other

**Your Input:**

Annammal  
jmjannie7@gmail.com  
www.abcd.com  
female

## RESULT:

Thus the program the form using PHP regular expression was successfully executed.

<b>Ex No: 10</b>	<b>IMPLEMENTING AN APPLICATION WITH WEB SERVICES</b>
<b>Date:</b>	

### AIM:

To write a web services for predicting for any product sales.

### PROCEDURE:

- Open the Home page.
- View the 2 products.
- Put the rating by user.
- Display the high rating product on the top of the page.

### PROGRAM:

#### *Page.html*

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html lang="en">
<head>
<title>AJAX 5 Star Rating</title>
<script src="http://code.jquery.com/jquery-latest.js"></script>
<script>
// This is the first thing we add
$(document).ready(function() {
$('.rate_widget').each(function(i) { var
widget = this;
var out_data = {
widget_id : $(widget).attr('id'),
fetch: 1
};
$.post( 'ratings.php',
out_data,
function(INFO) {
$(widget).data( 'fsr', INFO );
set_votes(widget);
},
'json'
);
});
$('.ratings_stars').hover(
// Handles the mouseover
function() {
$(this).prevAll().andSelf().addClass('ratings_over');
$(this).nextAll().removeClass('ratings_vote');
},
// Handles the mouseout
function() {
$(this).prevAll().andSelf().removeClass('ratings_over');
// can't use 'this' because it wont contain the updated data set_votes($(this).parent());
}
}
```

```

);

// This actually records the vote
$('.ratings_stars').bind('click', function() { var
star = this;
var widget = $(this).parent(); var
clicked_data = { clicked_on :
$(star).attr('class'),
widget_id : $(star).parent().attr('id')
};
$.post( 'ratings.php',
clicked_data,
function(INFO) {
widget.data( 'fsr', INFO );
set_votes(widget);
},
'json'
);
});
});

function set_votes(widget) {
var avg = $(widget).data('fsr').whole_avg;
var votes = $(widget).data('fsr').number_votes; var exact
= $(widget).data('fsr').dec_avg;
window.console && console.log('and now in set_votes, it thinks the fsr is ' +
$(widget).data('fsr').number_votes);
$(widget).find('.star_' + avg).prevAll().andSelf().addClass('ratings_vote');
$(widget).find('.star_' + avg).nextAll().removeClass('ratings_vote');
$(widget).find('.total_votes').text( votes + ' votes recorded (' + exact + ' rating'
);
}

// END FIRST THING
</script>
<style>
.rate_widget {
border: 1px solid #CCC;
overflow: visible;
padding: 10px;
position: relative;
width: 180px;
height: 32px;
}
.ratings_stars {
background: url('star_empty.png') no-repeat; float:
left;
height: 28px;

```

```
padding: 2px;
width: 32px;
}
.ratings_vote {
background: url('star_full.png') no-repeat;
}

.ratings_over {
background: url('star_highlight.png') no-repeat;
}
.total_votes {
background: #eaeaea;
top: 58px;
left: 0; padding:
5px;
position: absolute;
}
.movie_choice {
font: 10px verdana, sans-serif;
margin: 0 auto 40px auto; width:
180px;
}
h1 {
text-align: center;
width: 400px; margin:
20px auto;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1> Rate the following movies! </h1>
```

```
<div class='movie_choice'> Rate:
```

```
Raiders of the Lost Ark
```

```
<div id="r1" class="rate_widget">
```

```
<div class="star_1 ratings_stars"></div>
```

```
<div class="star_2 ratings_stars"></div>
```

```
<div class="star_3 ratings_stars"></div>
```

```
<div class="star_4 ratings_stars"></div>
```

```
<div class="star_5 ratings_stars"></div>
```

```
<div class="total_votes">vote data</div>
```

```
</div>
```

```
</div>
```

```
<div class='movie_choice'> Rate:
```

```
The Hunt for Red October
```

```
<div id="r2" class="rate_widget">
```

```
<div class="star_1 ratings_stars"></div>
```

```
<div class="star_2 ratings_stars"></div>
```

```
<div class="star_3 ratings_stars"></div>
```

```

<div class="star_4 ratings_stars"></div>
<div class="star_5 ratings_stars"></div>
<div class="total_votes">vote data</div>
</div>
</div>
</body>
</html>

```

### ***Ratings.php***

```

<?php
$rating = new ratings($_POST['widget_id']); isset($_POST['fetch']) ?
$rating->get_ratings() : $rating->vote(); class ratings {
var $data_file = './ratings.data.txt';
private $widget_id;
private $data = array(); function_
_____construct($wid) {
$this->widget_id = $wid;
$all = file_get_contents($this->data_file);
if($all) {
$this->data = unserialize($all);
}
}
public function get_ratings() { if($this-
>data[$this->widget_id]) {
echo json_encode($this->data[$this->widget_id]);
}
else {
$data['widget_id'] = $this->widget_id;
$data['number_votes'] = 0;
$data['total_points'] = 0;
$data['dec_avg'] = 0;
$data['whole_avg'] = 0; echo
json_encode($data);
}}
public function vote() {
# Get the value of the vote
preg_match('/star_([1-5]){1})/', $_POST['clicked_on'], $match);
$vote = $match[1];
$ID = $this->widget_id;
# Update the record if it exists
if($this->data[$ID]) {
$this->data[$ID]['number_votes'] += 1;
$this->data[$ID]['total_points'] += $vote;
}
# Create a new one if it doesn't else
{

```

```

$this->data[$ID]['number_votes'] = 1;
$this->data[$ID]['total_points'] = $vote;
}
$this->data[$ID]['dec_avg'] = round( $this->data[$ID]['total_points'] / $this->
data[$ID]['number_votes'], 1 );
$this->data[$ID]['whole_avg'] = round( $this->data[$ID]['dec_avg'] );
file_put_contents($this->data_file, serialize($this->data));
$this->get_ratings();
}
# ---
# end class
}

```

## OUTPUT:



## Result:

Thus the web service for predicting for any product sales has been executed successfully.

