MVGR College Of Engineering (Autonomous)
Department of Computer Science and Engineering

WEB TECHNOLOGIES

LAB MANUAL FOR CSE & IT

INDEX

1. Introduction

Web technologies lab covers the design and development of dynamic web pages. And the student will learn to develop the web application using the concepts in this lab. As part of this lab student will learn the html, java script, css, xml, java visual beans, servers, servlets and jsps. These technologies will cover the design, development and deploying the application.

Students will learn the importance of IDE's. And get the exposure on eclipse IDE. How to use the servers, And integrate the servers to the IDE. As part of this lab students will learn the eclipse IDE. Students will get the exposure on database and integrating the windows and web applications to the database.

HTML is a language for describing web pages. HTML stands for Hyper Text Markup Language. HTML is a markup language. A markup language is a set of markup tags. The tags describe document content. HTML documents contain HTML tags and plain text. HTML documents are also called web pages. HTML markup tags are usually called HTML tags. HTML tags are keywords (tag names) surrounded by angle brackets like <html>. HTML tags normally come in pairs like and . The first tag in a pair is the start tag, the second tag is the end tag. The end tag is written like the start tag, with a slash before the tag name. Start and end tags are also called opening tags and closing tags.

JavaScript is one of 3 languages all web developers MUST learn: HTML to define the content of web pages. CSS to specify the layout of web pages. JavaScript to program the behavior of web pages.

XML stands for EXtensible Markup Language. XML was designed to describe data. XML is a software- and hardware-independent tool for carrying information. XML is easy to learn.

There are several common scripting solutions to create Web applications. These are scripts that are run on the server before the page is sent to the user. The different server side scripting are like CGIs, Servlets, JSPs and PHPs.

Java servlets are a powerful alternative to CGI programs and scripting languages. Servlets are extremely similar to proprietary server APIs (application programming interfaces), but because they are written in the Java programming language, they can be easily ported to any environment that supports the Java Servlet API. Since they run in the Java Virtual Machine (JVM), they bypass the security problems that affect the server APIs.

JSP technology is used to create web application just like Servlet technology. It can be thought of as an extension to Servlet because it provides more functionality than servlet such as expression language, JSTL, etc. A JSP page consists of HTML tags and

JSP tags. The JSP pages are easier to maintain than Servlet because we can separate designing and development.

JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of JavaSE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database.

2. Lab Objectives:

1. Students will learn the design and development of html pages.
2. Students will gain the knowledge of JavaScript concepts.
3. Students will gain the knowledge of XML concepts.
4. Students will understand platform independent java bean components.
5. Students will understand server-side technology that is servlets and JSP.
6. Students will understand the database connectivity using JDBC.

3. Lab Resource requirements

Hardware Requirements:

- Desktops with 1GB RAM, 200GB Hard Disk, Pentium+ Processor.

Software Requirements:

- Apache tomcat webserver
- MYSQL DB
- WAMP SERVER
- BROWSER
- Eclipse
- NOTEPAD++ with XML plugin
- Stylus studio

4. Outcomes:

1. Able to design a static web page using the html components.
2. Able to design a dynamic web page using the html, css and java script.
3. Able to use the XML and visual beans in developing the web application.
4. Able to configure the apache tomcat web server.
5. Able to use the servlets and JSP concepts
6. Able to develop servlets and JSP application using JDBC Connectivity.

5. Web Technologies Syllabus Programs

S.No	Programs
1, 2	Develop static pages (using only HTML) of an online Book store. Thepages should resemble: www.amazon.com The website should consist the following pages. Home page, Registration and user Login, User profile page, Books catalog, Shopping cart, Payment By credit card, order confirmation.
3	Develop and demonstrate the usage of inline, internal and external style sheet using CSS. Design a web page using CSS which includes the following: **1)** Use different font styles **2)** Control the repetition of image with background-repeat and no-repeat property **3)** Define style for links as a: link, a: active, a: hover, a: visited **4)** Add customized cursors for links.
4	Develop and demonstrate JavaScript with POP-UP boxes and functions for the following problems: **a)** Input: Click on Display Date button using onclick() function Output: Display **date** in the textbox **b)** Input: A number n obtained using **prompt** Output: **Factorial** of n number using **alert** **c)** Input: A number n obtained using **prompt** Output: A **multiplication table** of numbers from 1 to 10 of n using **alert** **d)** Input: A number n obtained using **prompt** and add another number using **confirm** Output: **Sum** of the entire n numbers using **alert**
5	Write *JavaScript* to validate the following fields of the Registration page. **1.** First Name (Name should contains alphabets and the length should not be less than 6 characters). **2.** Password (Password should not be less than 6 characters length). **3.** E-mail id (should not contain any invalid and must follow the standard pattern

	name@domain.com) 4. Mobile Number (Phone number should contain 10 digits only). 5. Last Name and Address (should not be Empty).
6, 7	Validate the registration, user login, user profile and payment by creditcard pages using JavaScript.
8	Write an XML file which will display the Book information which includes the following: 1) Title of the book 2) Author Name 3) ISBN number 4) Publisher name 5) Edition 6) Price Write a Document Type Definition (DTD) to validate the above XMLfile. Display the XML file as follows. The contents should be displayed in a table. The header of the table should be in color GREY. And the Author names column should be displayed in one color and should be capitalized and in bold. Use your own colors for remaining columns. Use XML schemas XSL and CSS for the above purpose.
9	**VISUAL BEANS:** Create a simple visual bean with a area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click.
10	1) Install TOMCAT web server. While installation assign port number 8080. Make sure that these ports are available i.e., no other process is using this port. 2) Access the above developed static web pages for books web site, using these servers by Putting the web pages developed in week-1 and week-2 in the document root. Access the pages by using the URLs : **http://localhost:8080/rama/books.html**

11	User Authentication: Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 And, pwd4 respectively. Write a servlet for doing the following. Create a Cookie and add these four-user ids and passwords to this Cookie. Read the user id and passwords entered in the Login form (week1) and authenticatewith the values (user id and passwords) available in the cookies. If he is a valid user (i.e., user-name and password match) you should welcome him by name(user-name) else you should display "You are not an authenticated user ".
12	Install a database (MySQL or Oracle). Create a table which should contain at least the following fields: name, password, email-id, phone number (these should hold the data from the registration form). Practice 'JDBC' connectivity. Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Experiment with various SQL queries. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).
13	Write a JSP which does the following job: Insert the details of the 3 or 4 users who register with the web site (week 12) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database.

6. Lab Manual

WEEK- 1, 2:

Develop static pages (using only HTML) of an online Book store. The pages should resemble: www.amazon.com. The website should consist the following pages.

- **Home page**
- **Registration and user Login**
- **User profile page**
- **Books catalog**
- **Shopping cart**
- **Payment by credit card Order Conformation**

PROCEDURE:

Home page

Main.html:

```
<html>
<head>
<title>
Amazon</titl
e>
</head>
<body bgcolor="cyan"> <center>
<strong><h1>Welcome to AMAZON</h1></strong>
<form method="post" action="login.html" target=_blank >
<h4>for books</h4><input type="submit" value="click here">
</form>
</center>
</body>
</html>
```

- **Registration and user Login**

Login.html:

```
<html>
<head>
<title>
login page</title>
</head>
<body bgcolor="cyan"> <center>
<strong><h1> AMAZON </h1></strong></center>
<right>
<table align="right">

<tr>
<td><h4>user name</td>
<td><input type="text" ></td>
<td></td>
</tr>

<tr>
<td><h4>password</td>
<td><input type="password"></td>
<td></td>
</tr>

<tr>
<td>
<form method="post" action="catalog.html"  >
<input type="submit" value="submit" >
</form>
</td>
<td>
<form method="post" action="userpro.html"  >
<input type="submit"
value="register" >&amp;nbsp;&amp;nbsp;
<input type="reset" value="reset"></form></td>
</tr>

</table>

</body>
</html>
```

- **User profile page**

Userpro.html:

```
<html>
<head>
<title>
login page</title>
</head>
<body bgcolor="cyan">
<center><strong><h1> AMAZON </h1></strong></center>
<form method="post" action="catalog.html" >
<right>
<table align="left">
<tr>
<td><h4>user name</td>
<td><input type="text" ></td>
<tr>
<tr>
<td><h4>password</td>
<td><input type="password"></td>
</tr>
<tr>
<td><h4>confirm password</td>
<td><input type="password"></td>
</tr>
<tr>
<td><h4>male   
<option >
<input type="radio" name="sex" id="male"></td>
<td><h4>female    
<input type="radio" name="sex" id="female" ></td>
</option>
</tr>
<tr>
<td>Address</td>
<td><textarea name="address" rows=5 cols=19>
</textarea>
</td>
<tr>
<td>
<input type="submit" value="submit" ></td>
<td>
<input type="reset" value="reset"></td>
</tr>
```

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

- **Books catalog**

Catalog.html:

```
<html>
<head>
<title>
books catalog</title>
</head>
<body bgcolor="cyan">
<center><h1>AMAZON</h1></center>
<form method="post" action="shopping.html">
<left>
<table>
<tr>
 <td><b><h3>frontend books</td>
<td></td></tr>
<tr>
<td></td>
 <td><h4>C&Ds</td>
</tr>
<tr>
<td></td>
 <td><h4>Ads</td>
</tr>
<tr>
<td></td>
 <td><h4>JAVA
 </td></tr>
<tr>
 <td><b><h3>backend books</td>
<td></td>
</tr>
<tr>
<td></td>
 <td><h4>Oracle</td>
</tr>
<tr>
<td></td>
<td><h4>Ms SQL Server
</td></tr>
<tr>
```

```
<td></td>
<td><h4>MySql </td>
</tr>
</table>
</h4>
<center>
<b>for buy one of these books
<br>
</b><input type="submit" value="click here">
</center>
</form>
</body>
</html>
```

Shopping cart

Shopping.html:

```
<html>
<head><title>shopping cart</title>
</head>
<body bgcolor="cyan">
<center><h1>
Shopping Cart</h1></center>
<br><br><br><br><br>
<table align="center">
<tr>
<td>Text Books</td>
<td>
<select >
<optgroup label="select the book">
<option value="C&Ds">C&Ds
<option value="Ads">Ads
<option value="Java">Java
<option value="Oracle">Oracle
<option value="Ms SQL Server">Ms SQL Server
<option value="MySql">MySql
</optgroup>
</select>
</td></tr>
<tr>
<td>
 Quantity</t
 d>
<td>
<input type="text" id="q">
```

```
</td></tr>
<tr>
<td></td>
<td>
<form method=post action="payment.html">
<input type="submit" value=ok />
</form>
</td></tr>
</table>
<center>
<pre>Cost of one book is"500" + shipping "100"</pre>
</center>
<body>
</html>
```

Payment by credit card

Payment.html:

```
<html>
<head><title>payment</title></head>
<body bgcolor="cyan">
<center><h1>Payment By Credit Card</h1></center>
<form method=post action="ordrconform.html">
<br><br><br><br><br>
<table align="center">
<tr>
<td>
<h4>Total Amount</h4></td>
<td><input type="text">
</td>
</tr>
<tr>
<td><h4>Credit Card Number</td>
<td><input type="text"></td>
</tr>
<tr>
<td>
</td>
<td><input type="submit" value=OK>
</td>
</tr>
</table>
</form></body>
</html>
```

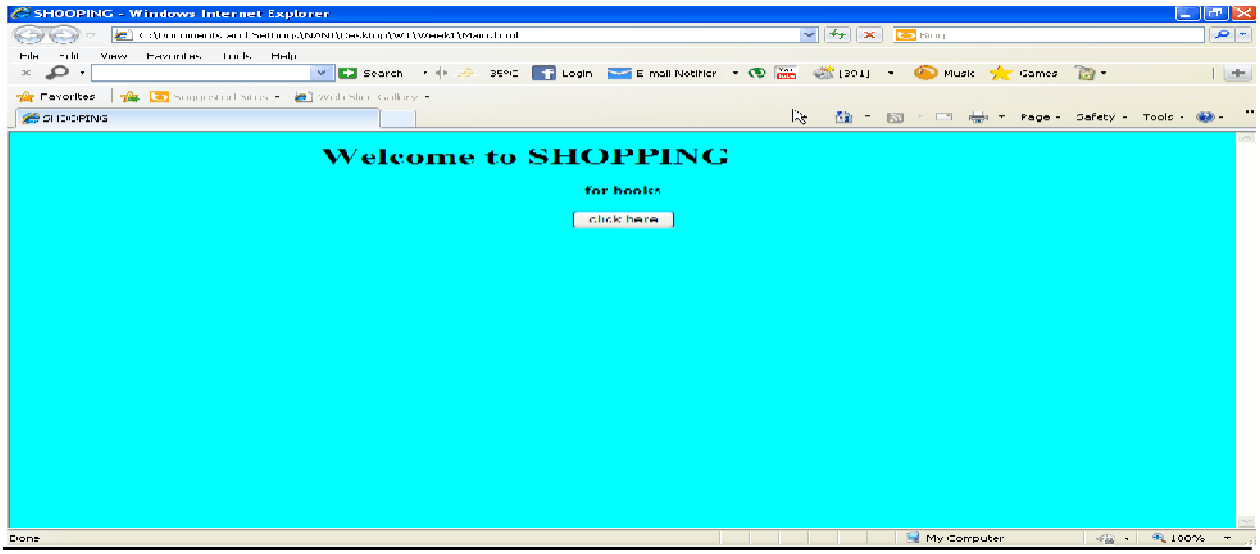
Order Conformation

Ordrconform:

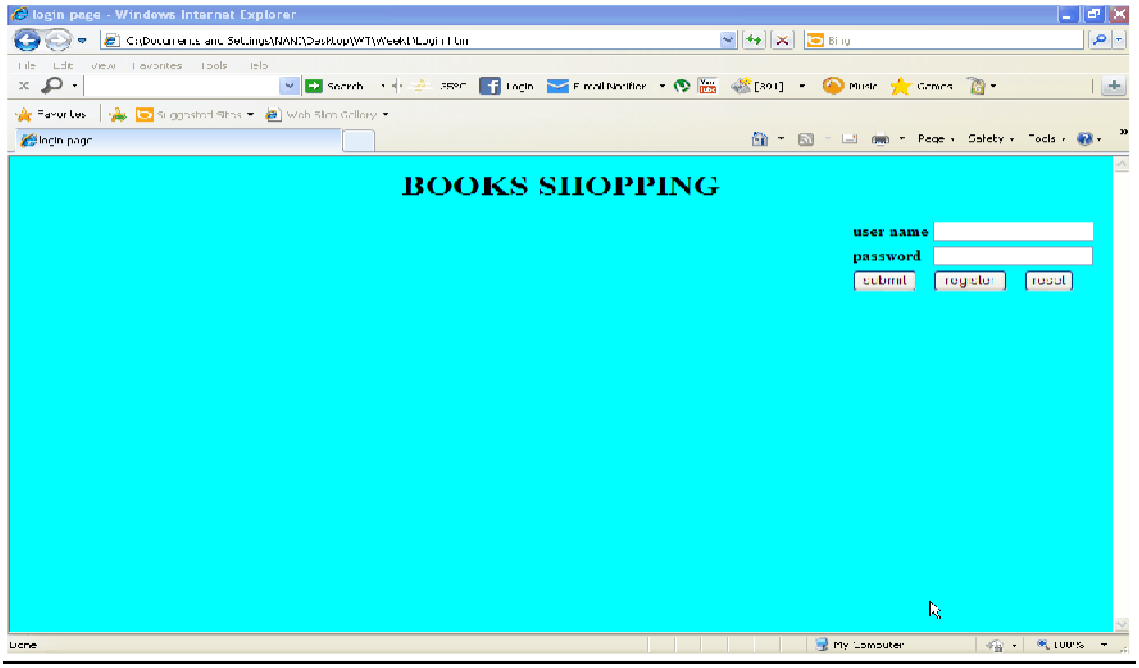
```
<html>
<head><title>order conformation</title><M/head>
<body bgcolor="cyan">
<center>
<h1><b>BOOK SHOPPING</h1>
<pre><strong>
<b>Your order Is Conformed
</strong></pre>
<h2><b>THANK YOU</h2>
</center>
</body></html>
```

OUTPUT:

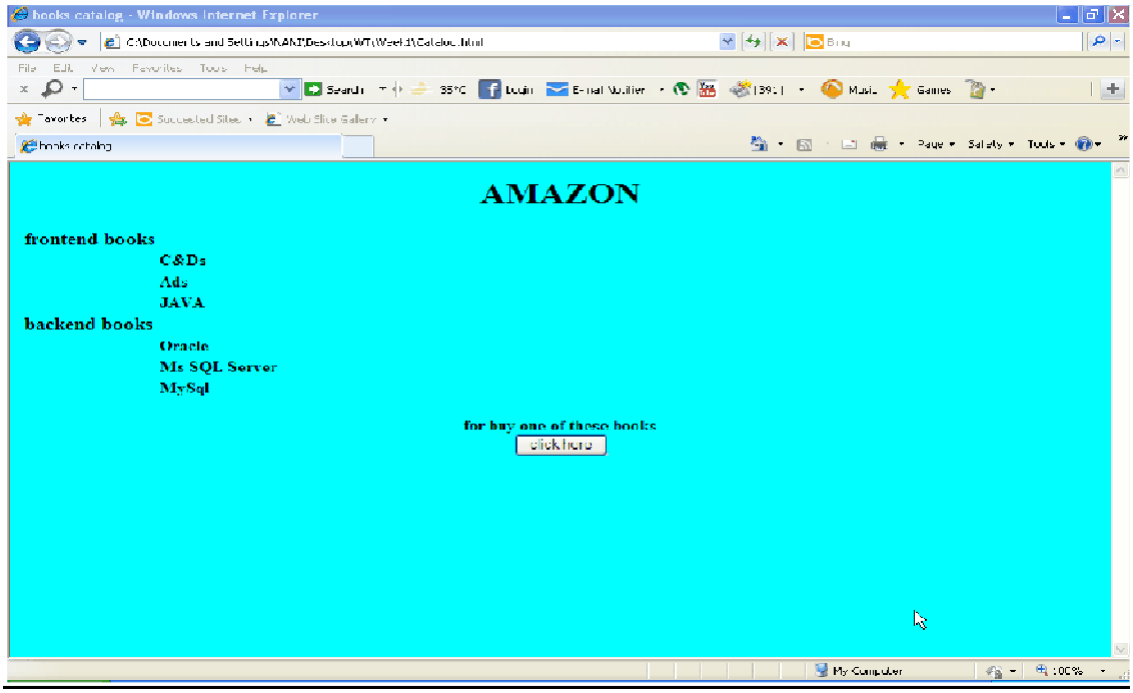
Main.html:



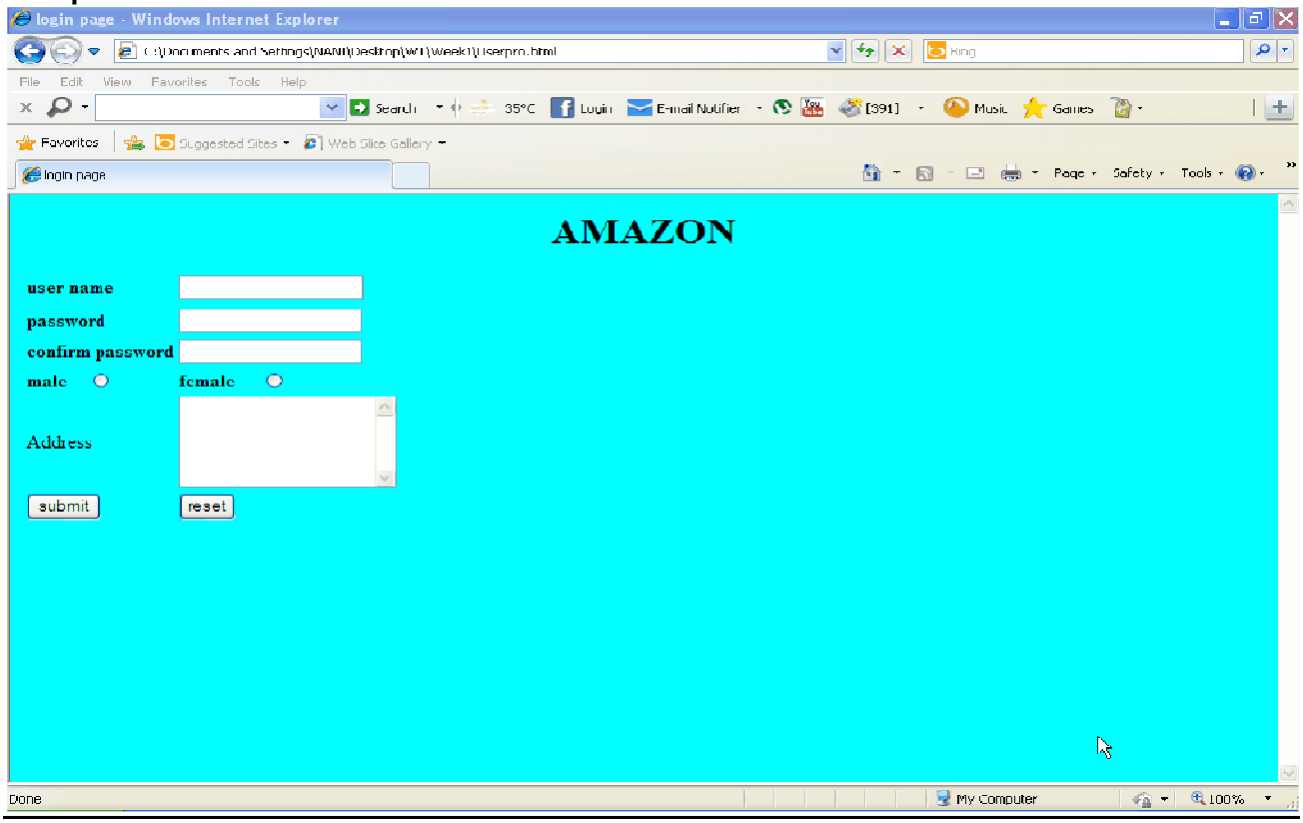
Login.html:



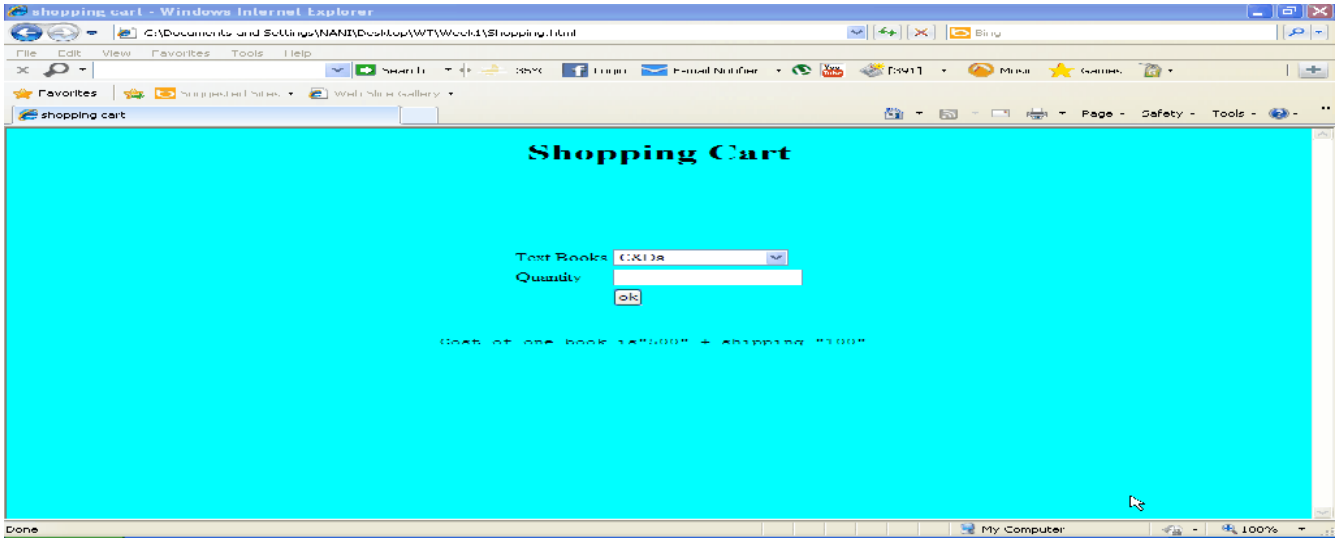
Catalog.html:



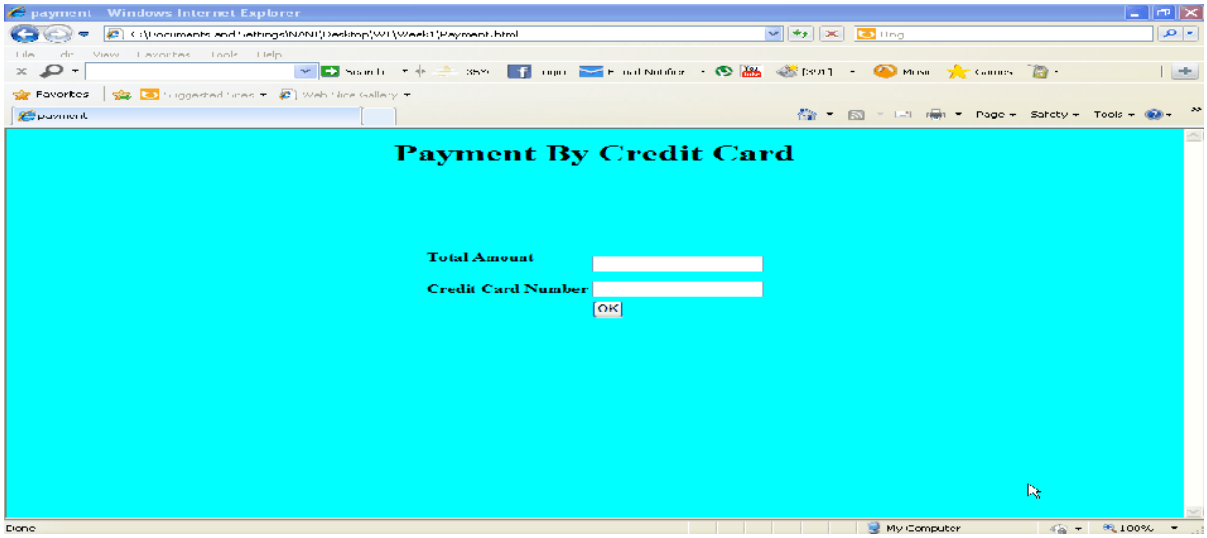
Userpro.htm



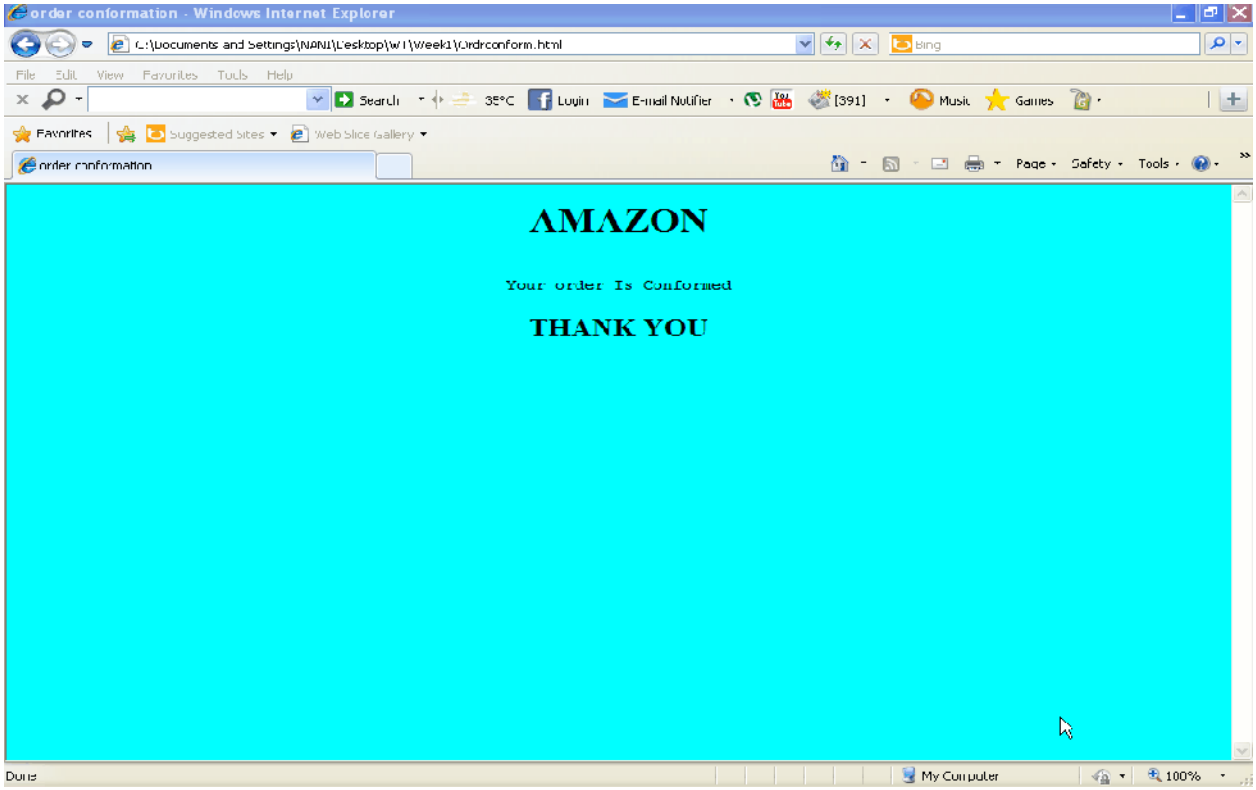
Shooping.html:



Payment.html:



Ordrconform.html:



WEEK 3

Develop and demonstrate the usage of inline, internal and external style sheet using CSS.

Design a web page using CSS which includes the following:

5) Use different font styles
6) Control the repetition of image with background-repeat and no-repeat property
7) Define style for links as a: link, a: active, a: hover, a: visited
8) Add customized cursors for links.

PROGRAM:

style.css

```
p.left
{
text-
align:l
eft;
color:b
lue;
font-
family:Cambr
ia; font-
size:large;
text-indent:20px;
}
p.center
{
text-align:center;
text-
decoration:underline;
text-
transform:uppercase;
letter-spacing:-3px;
word-
spacing:20px;
font-size:larger;
}
p.right
{
text-
align:right;
color:red;
```

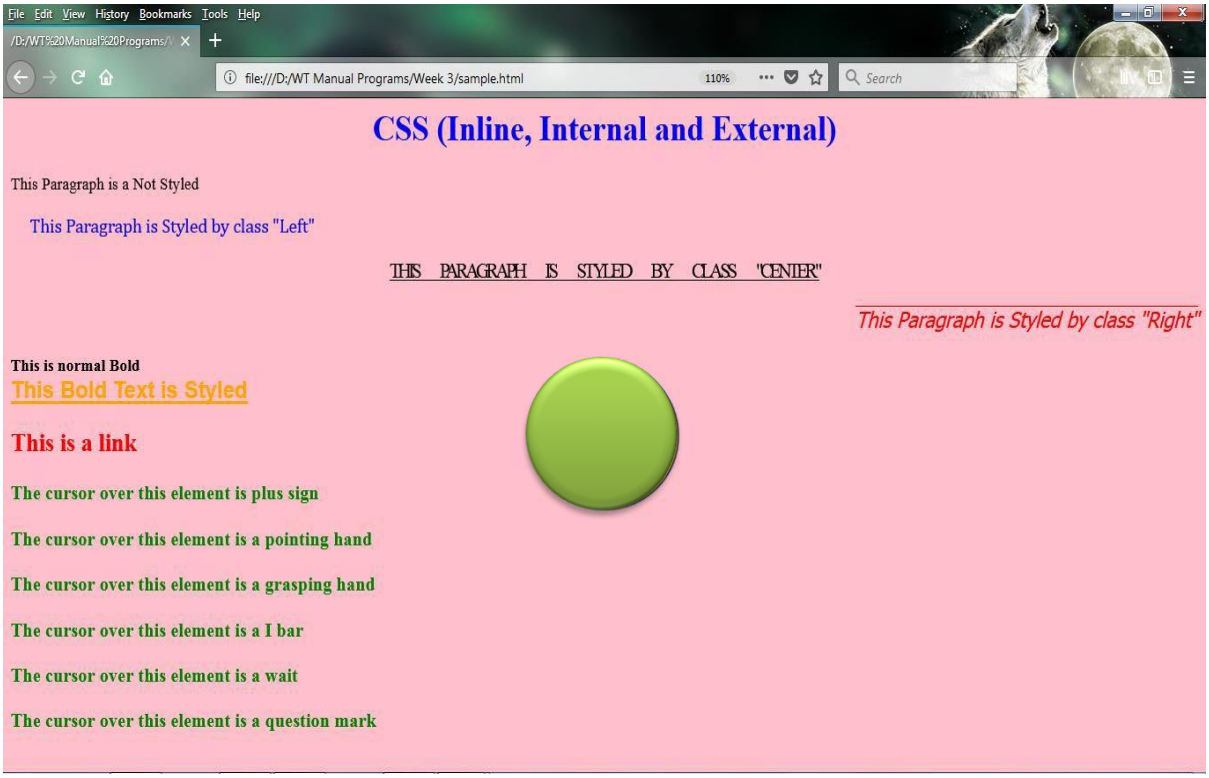
```
font-
family:Tahoma;
font-size:15pt;
text-
decoration:overline;
font-style:italic;
      }
      b#headline
      {
color:orange;
font-size:22px;
font-
family:arial;
text-decoration:underline;
      }
```

sample.html

```
      <html>
 <head>
  <style
   type="text/css">
   body
      {
      background-
      image:url('images/cse.png');
      background-repeat:no-repeat;
      background-position:center center;
      background-attachment:fixed;
      background-color:pink;
      }
      a:link { text-decoration:none;color:orange; }
      a:visited { text-decoration:none;color:red; }
      a:hover { text-
      decoration:underline;color:blue; } a:active {
      text-decoration:underline;color:purple; } h3
      { color:green; }
      .c1{cursor:crosshair}
      .c2{cursor:pointer}
      .c3{cursor:move}
      .c4{cursor:text}
      .c5{cursor:wait}
      .c6{cursor:help}
  </style>
 <link rel="stylesheet" type="text/css" href="style.css">
</head>
      <body bgcolor="cyan">
 <h1 style="color:blue;text-align:center;"> CSS (Inline, Internal and External) </h1>
```

```
<p>This Paragraph is a Not Styled</p>
<p class="left">This Paragraph is Styled by class "Left"</p>
<p class="center">This Paragraph is Styled by class "Center"</p>
<p class="right">This Paragraph is Styled by class "Right"</p>
  <b>This is normal Bold</b> <br>
  <b id="headline">This Bold Text is Styled </b>

<h2><b><a href=" ">This is a link</a></b></h2>

<h3 class="c1">The cursor over this element is plus sign</h3>
<h3 class="c2">The cursor over this element is a pointing hand</h3>
<h3 class="c3">The cursor over this element is a grasping hand</h3>
<h3 class="c4">The cursor over this element is a I bar</h3>
<h3 class="c5">The cursor over this element is a wait</h3>
<h3 class="c6">The cursor over this element is a question mark</h3>
    </html>
```

OUTPUT 1:



WEEK 4

Develop and demonstrate JavaScript with POP-UP boxes and functions for the following problems:

e) Input: Click on Display Date button using onclick( ) function Output: Display **date** in the textbox

f) Input: A number n obtained using **prompt** Output: **Factorial** of n number using **alert**

g) Input: A number n obtained using **prompt** Output: A **multiplication table** of numbers from 1 to 10 of n using **alert**

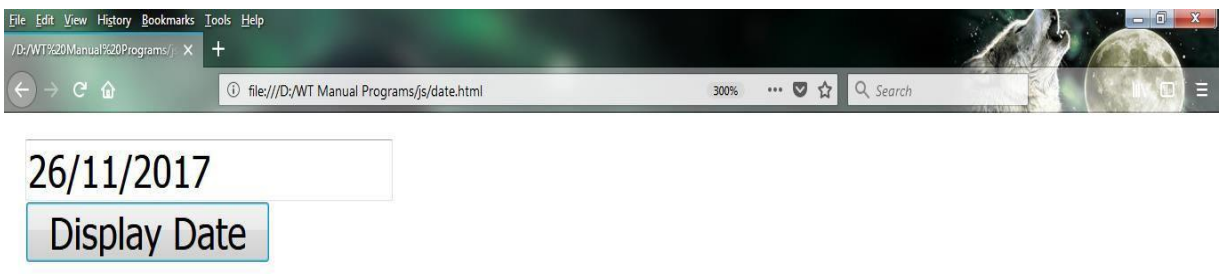
h) Input: A number n obtained using **prompt** and add another number using **confirm** Output: **Sum** of the entire n numbers using

alert PROGRAM:

a) date.html

```
<html>
<body>
<script>
function display(){
var x="You have
clicked"; var d=new
Date();
var date=d.getDate();
var
month=d.getMonth();
month++;
var year=d.getFullYear();
document.getElementById("dis").value=date+"/"+month+"/"+
year;
}
</script>
<form>
<input type="text" id="dis" /><br />
<input type="button" value="Display Date" onclick="display()" />
</form>
<body>
</html>
```

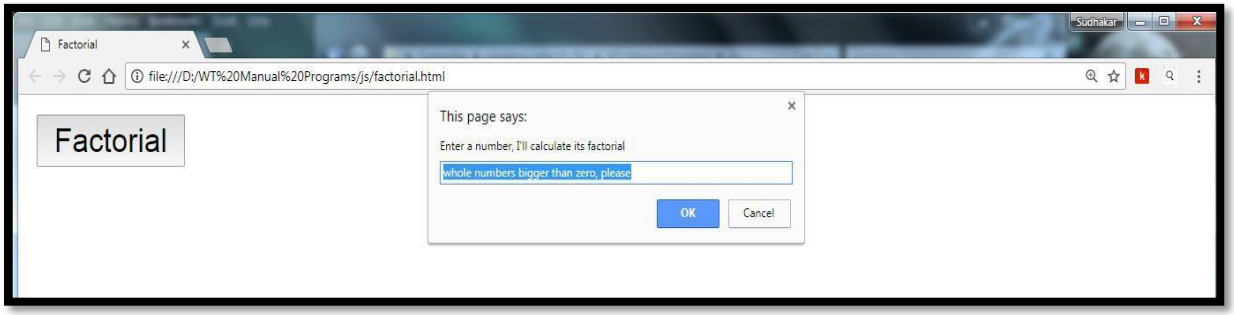
OUTPUT:

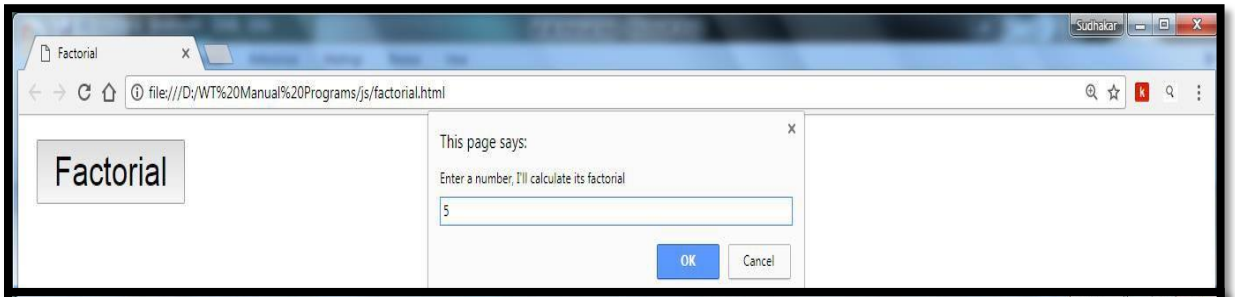


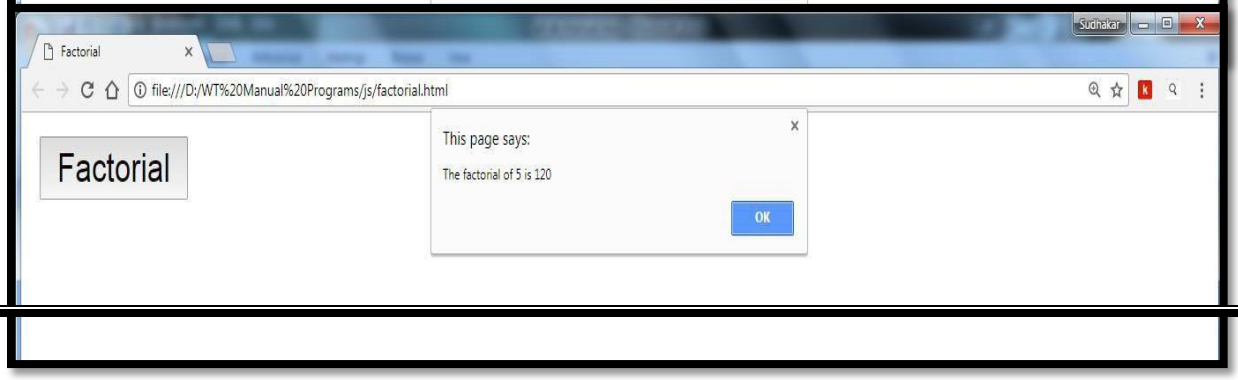
b) factorial.html

```
<html>
<head>
<title>factorial</title>
<script language='javascript'>
function factorialcalc()
{
    number = parseint(prompt("enter a number, i'll calculate its
             factorial", "whole numbers bigger than zero, please"))
    factorial = 1
    for (i=1; i <= number; i++)
    {
        factorial = factorial * i
    }
    alert("the factorial of " + number + " is " + factorial)
}
</script>
</head>
<body><form name=frm>
<input type=button value='factorial' onclick="factorialcalc();">
</form>
</body>
</html>
```

OUTPUT:





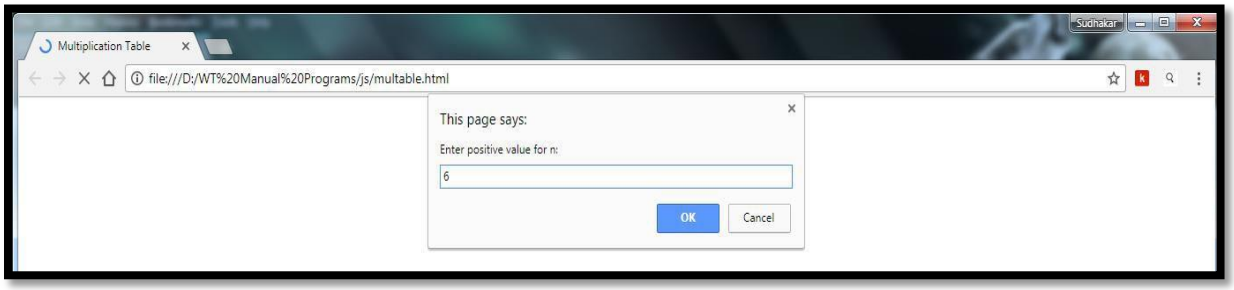


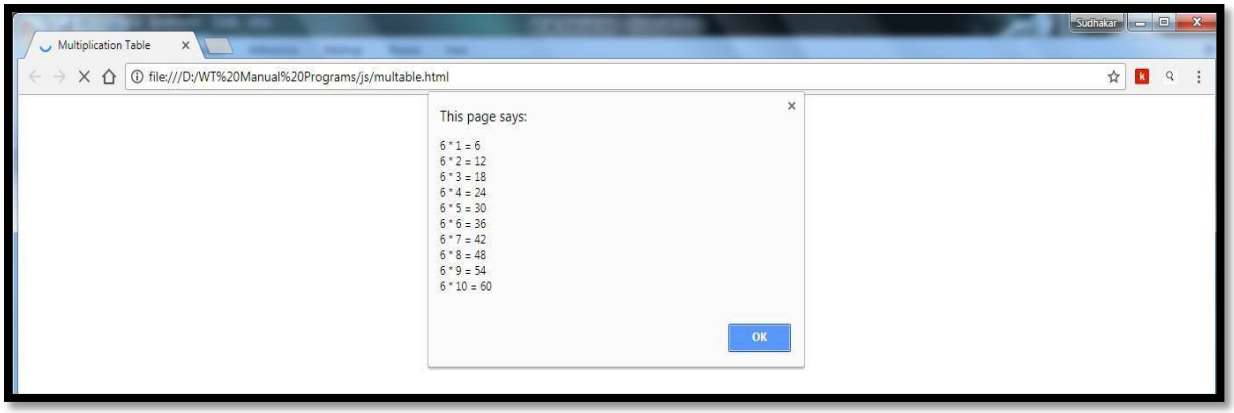
a) multable.html

```
<html>
<head><title> Multiplication Table </title></head>
<body>
<script type="text/javascript">
<!--
var n=prompt("Enter positive value for n:
"," "); if(!isNaN(n)) {
        var table="";
        var
        number="";
        for(i=1;i<=10;i++) {
                number = n * i;
                table += n + " * " + i + " = " + number + "\n";
        }
        alert(table);
        }
 else {
        alert("Enter positive value");
         n=prompt("Enter positive value for n:
}        "," ");
```

```
        document.write(n+" table values displayed using alert ..<br />");
    // -->
</script>
</body>
</html>
```

OUTPUT:



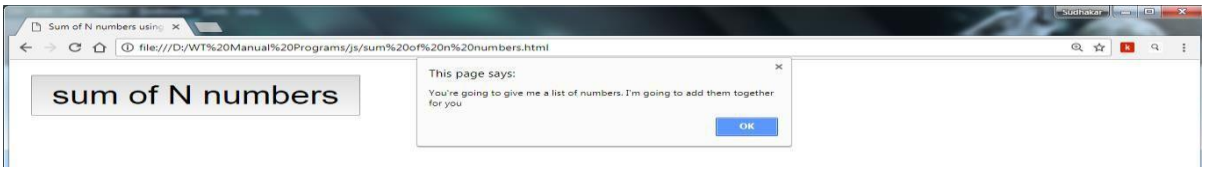


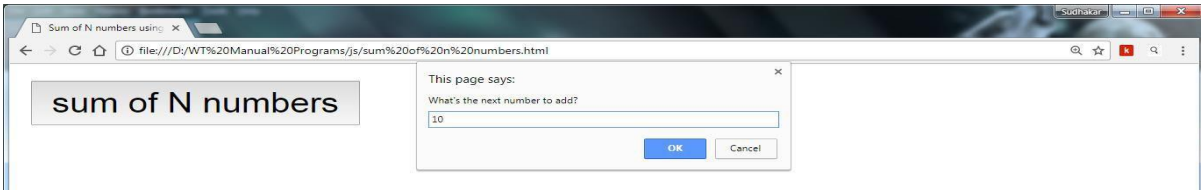
a) sum of n numbers.html

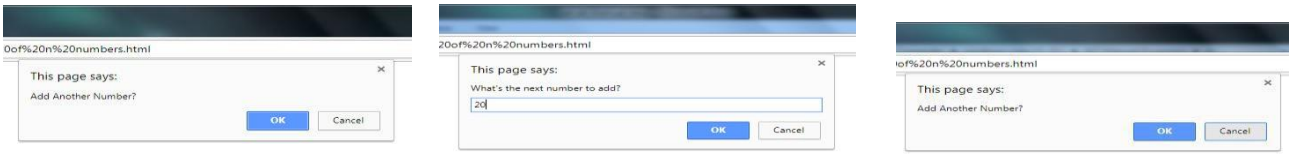
```
        <html>
  <head><title>sum of n numbers using popup boxes</title>
 <script
  language='javascript'>
  function addsum()
   {
   alert("you're going to give me a list of numbers. i'm going to add them together
        for you"); var keepgoing = true
        var sumofnums =
        0 while
        (keepgoing) {
        sumofnums = sumofnums + parseint(prompt("what's the next number to
        add?","")) keepgoing = confirm("add another number?")
        }
  alert("the sum of all your numbers is " + sumofnums)
   }
      </script>
      </head>
```

```
    <body>
<form name=frm>
 <input type=button value='sum of n numbers' onclick="addsum();">
</form>
    </body>
    </html>
```

OUTPUT:







WEEK 5

Write *JavaScript* to validate the following fields of the Registration page.

6. **First Name** (Name should contains alphabets and the length should not be less than 6 characters).
7. **Password** (Password should not be less than 6 characters length).
8. **E-mail id** (should not contain any invalid and must follow the standard pattern name@domain.com)
9. **Mobile Number** (Phone number should contain 10 digits only).
10. **Last Name and Address** (should not be Empty).

AIM: To validate the fields of registration page using JavaScript

DESCRIPTION: In order to validate the fields of login and registration pages JavaScript is used. JavaScript is programming code that can be inserted into HTML pages. JavaScript inserted into HTML pages, can be executed by all modern web browsers. JavaScript is mainly used for validating the elements in a form submitted by the user. This JavaScript code can react to user events.

PROGRAM: After clicking OK button the page is redirected to success.html

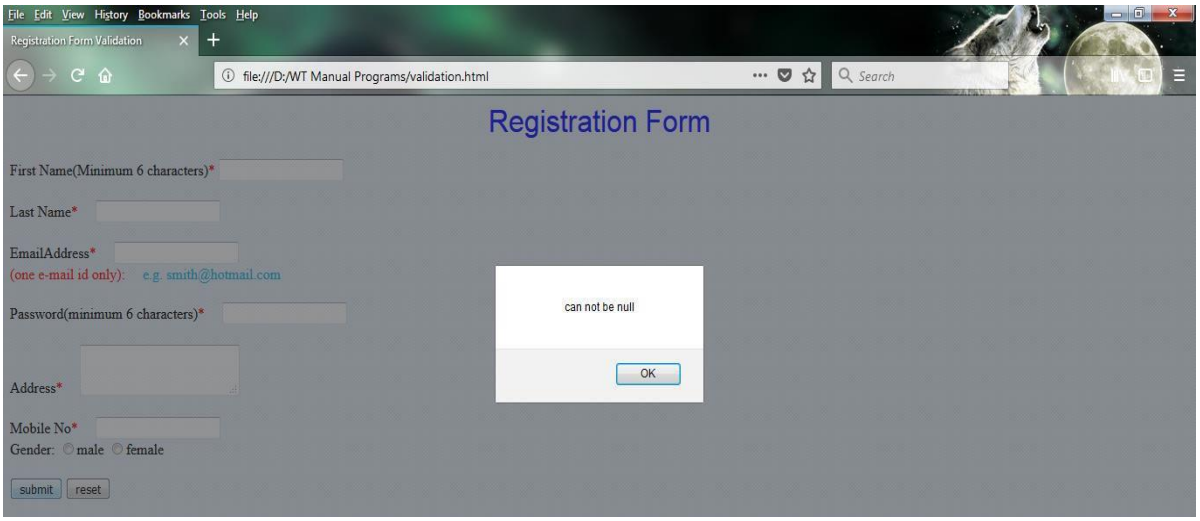
```
<html>
<head><title>Registration Form Validation</title></head>
<body bgcolor="#E4F0F8">
<script
type='text/javascript'>
function formValidator()
{
// Make quick references to our fields
var firstname =
document.getElementById('firstname'); var
lastname =
document.getElementById('lastname'); var email
= document.getElementById('email');
var pass =
document.getElementById('pass'); var
addr =
document.getElementById('addr');
var mobileno = document.getElementById('mobileno');

// Check each input in the order that it appears in
the form! if(notEmpty(firstname, "can not be null")){
if(isAlphabet(firstname, "Please enter only letters for your
Firstname")){ if(lengthRestriction(firstname, 6)){
if(isAlphabet(lastname, "Please enter only letters for your
Lastname")){ if(emailValidator(email, "Please enter a valid
email address")){
if(lengthRestriction(pass, 6)){
if(isAlphanumeric(pass, "please enter Numbers and Letters Only for
password")){ if(notEmpty(addr, "please enter the address")){
if(isNumeric(mobileno, "Please enter a valid
mobileno")){ if(lengthRestriction1(mobileno, 10 ,
10)){
return true;
} } } }
}
}
}
} } }
return false;
}
function notEmpty(elem, helperMsg){
if(elem.value.length ==
0){
alert(helperMsg);
elem.focus(); // set the focus to
this input return false;
}
return true;
}
function isNumeric(elem, helperMsg){
var numericExpression = /^[0-9]+$/;
if(elem.value.match(numericExpressi
```

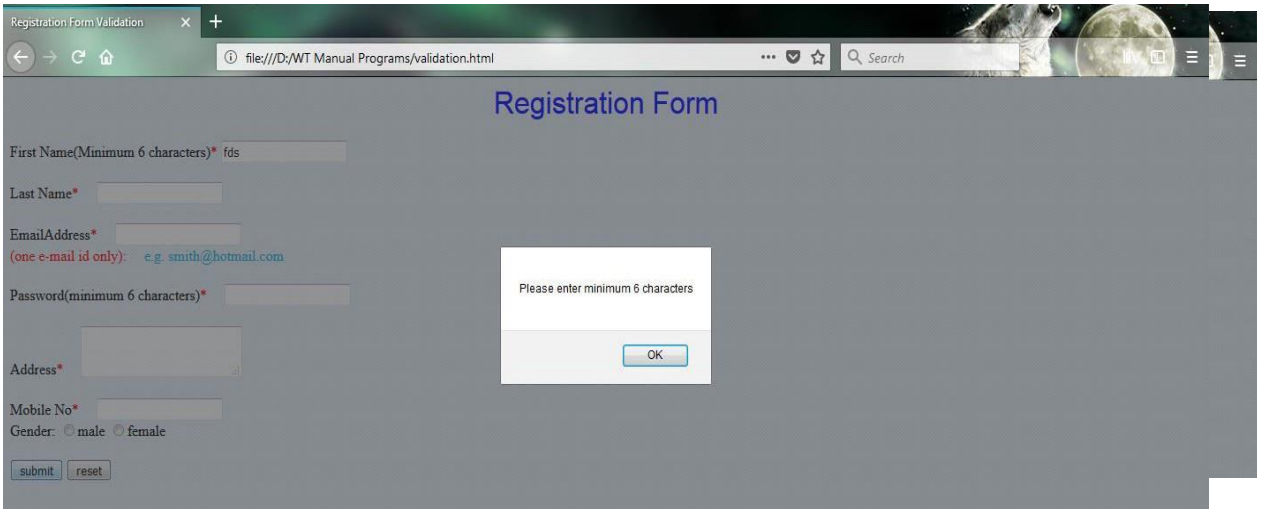
```
        on)){ return true;
        }else{
                alert(helperMsg
                ); elem.focus();
                return false;
        }
    }
function isAlphabet(elem,
        helperMsg){ var alphaExp =
        /^[a-zA-Z]+$/;
        if(elem.value.match(alphaEx
        p)){
                return true;
        }else
        {       alert(helperMsg
                ); elem.focus();
                return false;

        }
     }
function isAlphanumeric(elem,
        helperMsg){ var alphaExp = /^[0-
        9a-zA-Z]+$/;
        if(elem.value.match(alphaExp)){
                return true;
        }else
        {       alert(helperMsg
                ); elem.focus();
                return false;

        }
      }
function lengthRestriction(elem,
        min){ var uInput =
        elem.value;
        if(uInput.length >= min){
                return true;
        }else
        {       alert("Please enter minimum " +min+ "
                characters"); elem.focus();
                return false;

        }
    }
```

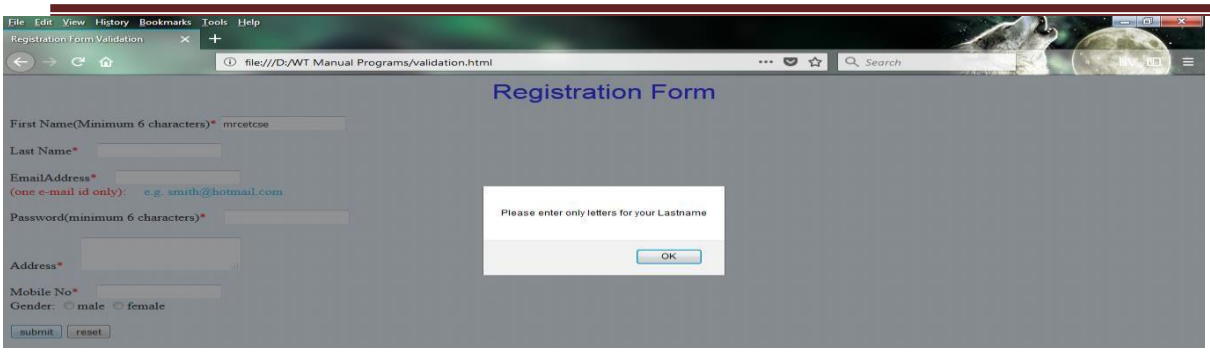
```
  function emailValidator(elem, helperMsg)
{
    var emailExp = /^[\w\-\.\+]+\@[a-zA-Z0-9\.\-]+\.[a-zA-z0-
    9]{2,4}$/; if(elem.value.match(emailExp))
    {
            return true;
    }
    else{
            alert(helperMsg
            ); elem.focus();
            return false;
          }
}
  function lengthRestriction1(elem, min, max)
{
    var uInput = elem.value;
    if(uInput.length >= min && uInput.length <= max)
    {
                                                return true;
    }
     else
     {      alert("Please enter 10 numbers
            only"); elem.focus();
            return false;

          }
}
```

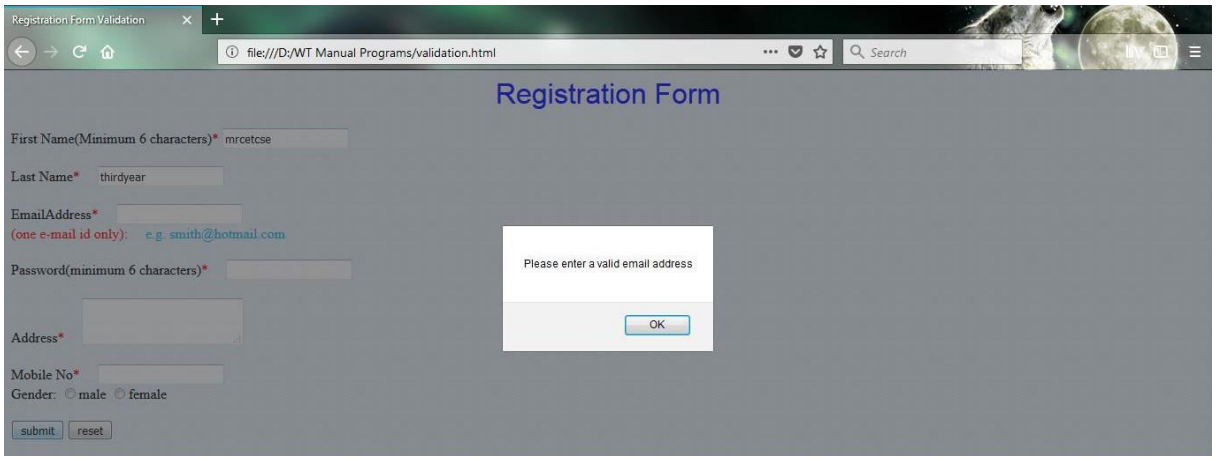
```
    </script>
      <center><font color="blue" size="6" face="arial">Registration
      Form</font></center><br />
<form onsubmit='return formValidator()'
  action="right.html"> First Name(Minimum 6
  characters)<font color="red">* </font>
       <input type='text' id='firstname' /><br /><br />
  Last Name<font color="red"><font color="red">* </font> </font>    
      <input type='text' id='lastname' /><br /><br />
  Email Address<font color="red">* </font>    
      <input type='text' id='email' /><br />
      <font color="red">(one e-mail id only):</font>    
      <font color="redblue">e.g. smith@hotmail.com</font><br /><br/>
  Password(minimum 6 characters)<font color="red">* </font>
     
      <input type='password' id='pass'><br /><br/>
  Address<font color="red">* </font>
     
      <textarea rows="2" cols="20" id='addr' /></textarea> <br />
  <br/> Mobile No<font color="red">* </font>    
      <input type='text' id='mobileno' /><br />
  Gender: <input type='radio'
  name="gender">male
          <input type='radio' name="gender">female<br/><br />
      <input type='Submit' value='submit' />
      <input type='Reset' value='reset' />
    </form>
  </body>
      </html>
```

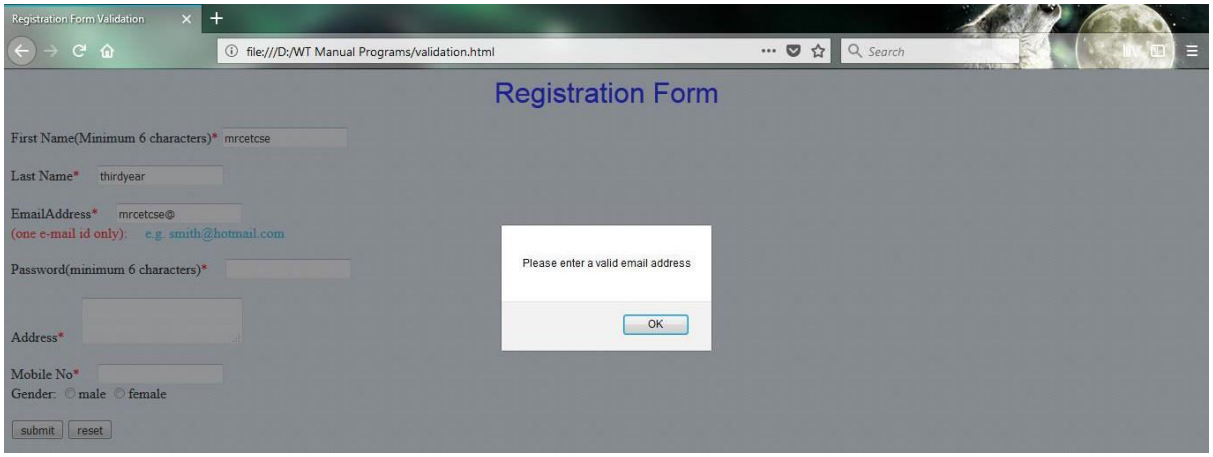
OUTPUT:

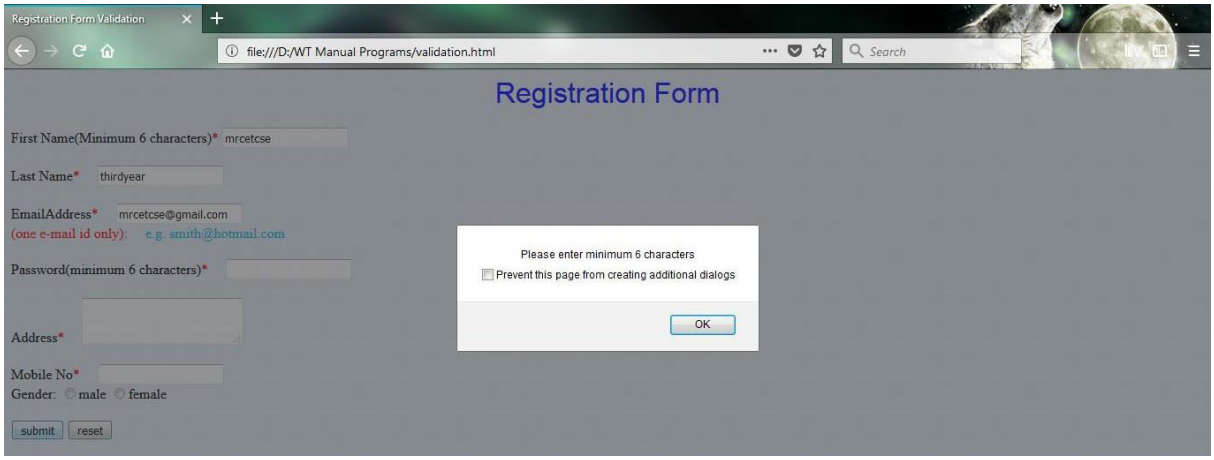


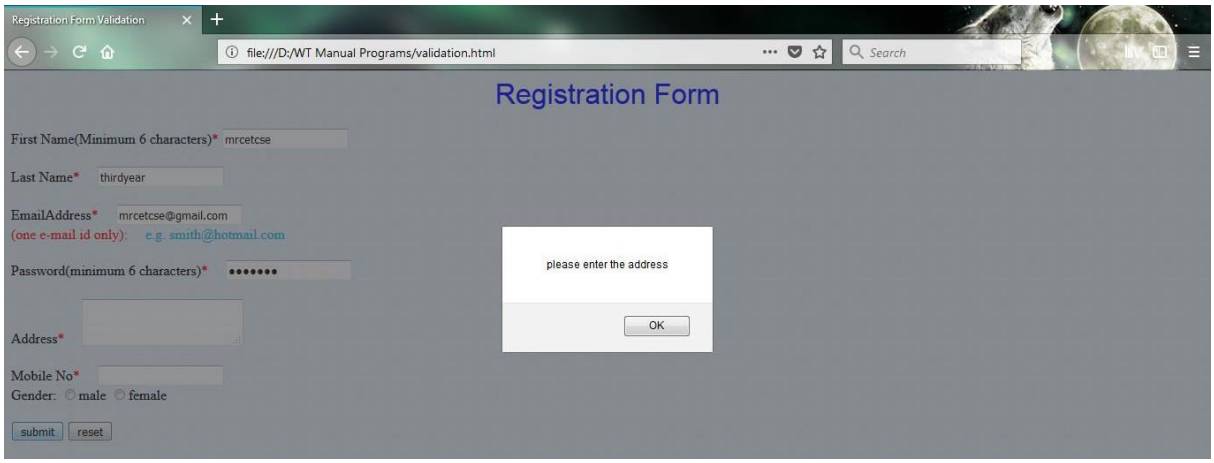

Registration Form Validation
file:///D:/WT Manual Programs/validation.html
Search
Registration Form
First Name(Minimum 6 characters)* fds
Last Name*
EmailAddress*
(one e-mail id only): e.g. smith@hotmail.com
Password(minimum 6 characters)*
Address*
Mobile No*
Gender: male female
submit reset
Please enter minimum 6 characters
OK

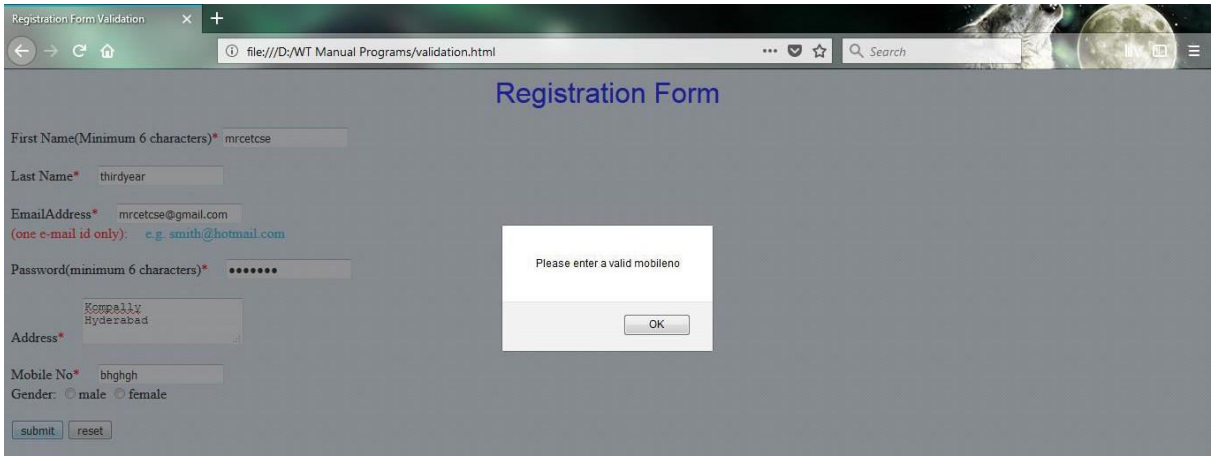

File Edit View History Bookmarks Tools Help
Registration Form Validation
file:///D:/WT Manual Programs/validation.html
Search
Registration Form
First Name(Minimum 6 characters)* mrcetcse
Last Name*
EmailAddress*
(one e-mail id only): e.g. smith@hotmail.com
Password(minimum 6 characters)*
Address*
Mobile No*
Gender: male female
submit reset
Please enter only letters for your Lastname
OK

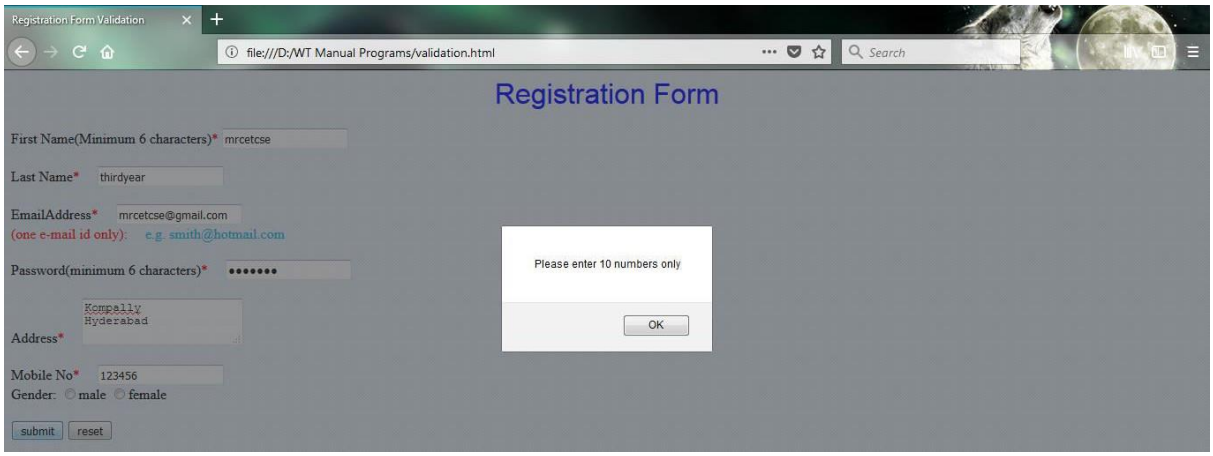

Registration Form Validation
file:///D:/WT Manual Programs/validation.html
Search
Registration Form
First Name(Minimum 6 characters)* mrcetcse
Last Name* thirdyear
EmailAddress*
(one e-mail id only): e.g. smith@hotmail.com
Password(minimum 6 characters)*
Address*
Mobile No*
Gender: male female
submit reset
Please enter a valid email address
OK

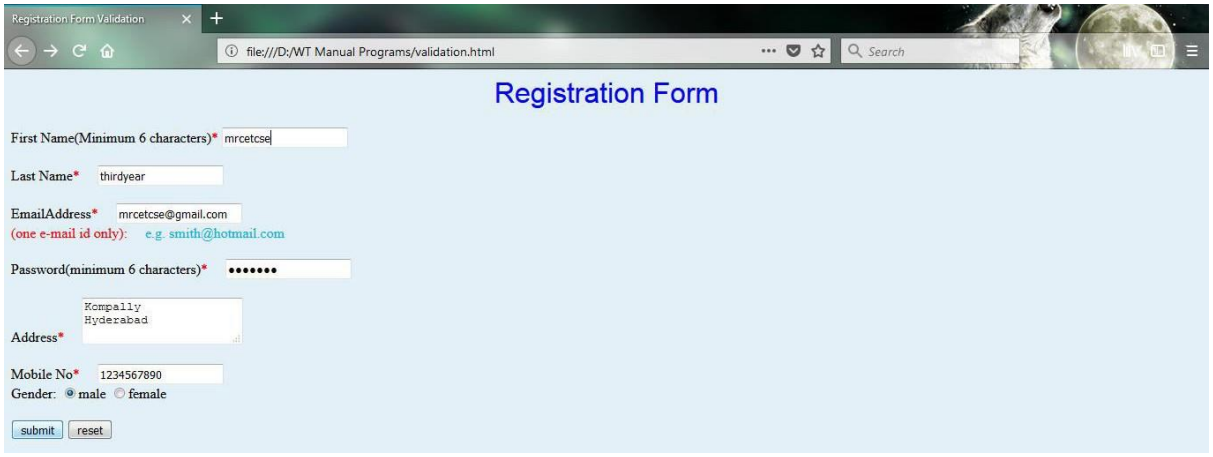

Registration Form Validation
file:///D:/WT Manual Programs/validation.html
Search
Registration Form
First Name(Minimum 6 characters)* mrcetcse
Last Name* thirdyear
EmailAddress* mrcetcse@
(one e-mail id only): e.g. smith@hotmail.com
Password(minimum 6 characters)*
Address*
Mobile No*
Gender: male female
submit reset
Please enter a valid email address
OK


Registration Form Validation
file:///D:/WT Manual Programs/validation.html
Search
Registration Form
First Name(Minimum 6 characters)* mrcetcse
Last Name* thirdyear
EmailAddress* mrcetcse@gmail.com
(one e-mail id only): e.g. smith@hotmail.com
Password(minimum 6 characters)*
Address*
Mobile No*
Gender: male female
submit reset
Please enter minimum 6 characters
Prevent this page from creating additional dialogs
OK


Registration Form Validation
file:///D:/WT Manual Programs/validation.html
Search
Registration Form
First Name(Minimum 6 characters)* mrcetcse
Last Name* thirdyear
EmailAddress* mrcetcse@gmail.com
(one e-mail id only): e.g. smith@hotmail.com
Password(minimum 6 characters)*
Address*
Mobile No*
Gender: male female
submit reset
please enter the address
OK


Registration Form Validation
file:///D:/WT Manual Programs/validation.html
Search
Registration Form
First Name(Minimum 6 characters)* mrcetcse
Last Name* thirdyear
EmailAddress* mrcetcse@gmail.com
(one e-mail id only): e.g. smith@hotmail.com
Password(minimum 6 characters)* •••••••
Kompally
Hyderabad
Address*
Mobile No* bhghgh
Gender: male female
submit reset
Please enter a valid mobileno
OK


Registration Form Validation
file:///D:/WT Manual Programs/validation.html
Search
Registration Form
First Name(Minimum 6 characters)* mrcetcse
Last Name* thirdyear
EmailAddress* mrcetcse@gmail.com
(one e-mail id only): e.g. smith@hotmail.com
Password(minimum 6 characters)* •••••••
Kompally
Hyderabad
Address*
Mobile No* 123456
Gender: male female
submit reset
Please enter 10 numbers only
OK


Registration Form Validation
file:///D:/WT Manual Programs/validation.html
Search
Registration Form
First Name(Minimum 6 characters)* mrcetcse
Last Name* thirdyear
EmailAddress* mrcetcse@gmail.com
(one e-mail id only): e.g. smith@hotmail.com
Password(minimum 6 characters)* •••••••
Kompally
Hyderabad
Address*
Mobile No* 1234567890
Gender: male female
submit reset


HTML Color Names
order conformation
file:///D:/WT%20Manual%20Programs/success.html?gender=on
Successfully Registered....

WEEK-6,7:

Validate the Registration, user login, user profile and payment bycredit 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
  arehaving 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">SHOPPINGCART</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>
```

Registration and user Login

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=
="")
{
alert("LoginId and Password must be
filled")flag=0;
}
if(flag==1)
{
alert("VALID INPUT");
window.open("catalog.html","right");
}
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"><br><br>
</pre>
<input type="button" value="ok" onClick="validate()">    
<input type="reset" value="clear" >
</div>

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

- **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==
        "")
        {
                        alert("Enter all the details");
          flag=0;
        }
         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]+$","g");if(!(str1.match(x1)))
                        {
                                flag=0;
                                alert("Invalid UserID");
                                }
                                var
                        str1=document.myform.pwd.value; var
                        x1=new RegExp("^[A-Z][a-zA-Z]+$","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();
                    }
                    }
                </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>
<br><br
>
<div align="center">
<input type="button" value="ok" onClick="validate()">&amp;nbsp;&amp;nbsp;&amp;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")
```

```
                {
                 document.myform.t1.value="JSP";
                 document.myform.t2.value=555;
                 }
            else
                {
                 flag=0;
                }

              if(flag==1)
                  {
                   alert("VALID INPUT");
                    }
                      else
                     {
                          alert("INVALID INPUT");
                          document.myform.focus();
                       }
                       }
                </script>
<form name="myform" action="payment.html" target="right">
<div align="center"><pre>
BOOK TITLE :<input type="text" name="title"><br>
</pre><br><br>
Book Title:    <input type="text" name="t1"
disabled>Book Cost:      <input type="text"
name="t2" disabled>
</div>
<br><br>
<div align="center">
<input type="submit" value="ok" onClick="validate()">&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;
<input type="reset" value="clear">
<input type="submit" value="Purchase">
</form>
</body>
</html>
```

- **Shopping cart:**

Catalog.html:

```
<html>
<body bgcolor="pink"><br><br><br>
<script
   language="javascript">
   function validate()
    {
      var flag=1;
       if(document.myform.id.value==
       ""||
          document.myform.title.value==""||
          document.myform.no.value==""||
          document.myform.cost.value=="")
           {
             flag=0;
          }

       str=document.myform.title.value;
       var str1=document.myform.cost.value;
       if(!((str=="c"&& str1==444) || (str=="jsp" && str1==555)))
                  {
                   flag=0;
                   }

      if(flag==1)
                   {
                     alert("VALID INPUT");

                      }
          else
                  {
                    alert("INVALID INPUT");
                    document.myform.focus();
                     }
     }
</script>
<form name="myform"  action="scart.html" target="right">
<div align="center"><pre>
LOGIN ID          :<input type="text" name="id"><br>
TITLE            :<input type="text" name="title"><br>
NO.OF BOOKS    :<input type="text" name="no"><br>
```

```
COST OF BOOK        :<input type="text"name="cost"><br>
</pre><br><br>
```

```
</div>
<br><br>
<div align="center">
<input type="submit" value="ok"  onClick="validate()">     
<input type="reset" value="clear">
</form>
</body>
</html>
```

- **Payment by 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();
}
}
</script>
<form name="myform">
<div align="center"><pre>
LOGIN ID            :<input type="text" name="id"><br>
PASSWORD            :<input type="password" name="pwd"><br>
AMOUNT                    :<input type="text"
name="amount"><br>
CREDITCARDNUMBER        :<input type="PASSWORD" name="num"><br></pre><br><br>
</div>
<br><br>
<div align="center">
<input type="button" value="ok" onClick="validate()">&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;
<input type="reset" value="clear" >
</form>
</body>
</html>
```

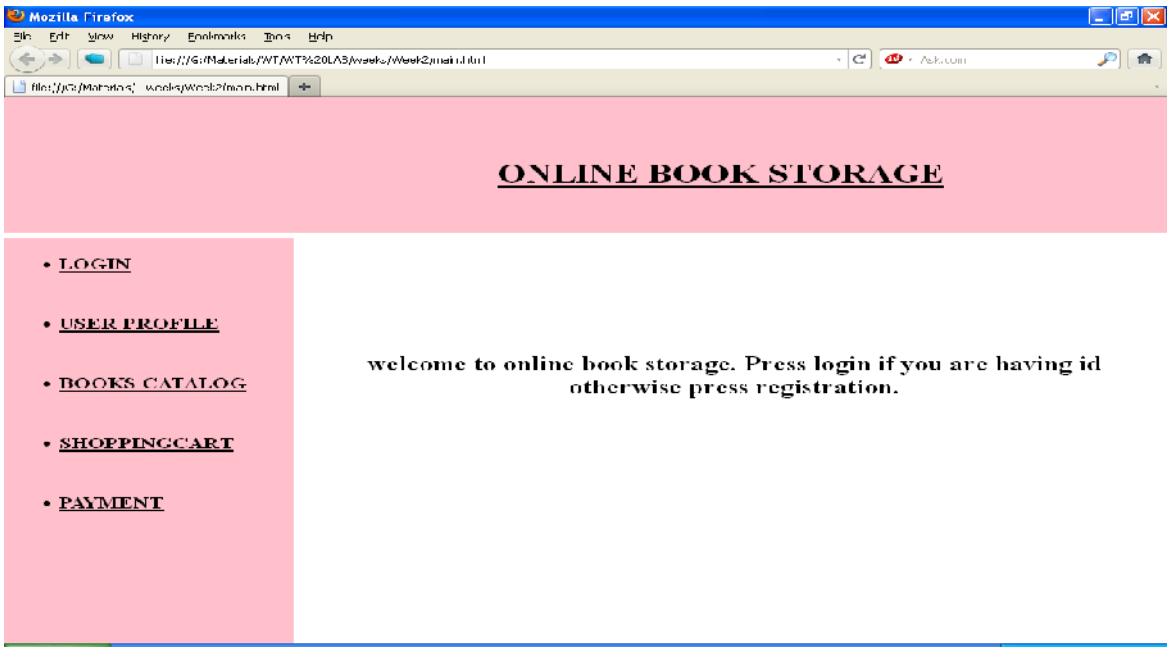
- **Order Conformation**

Order.html:

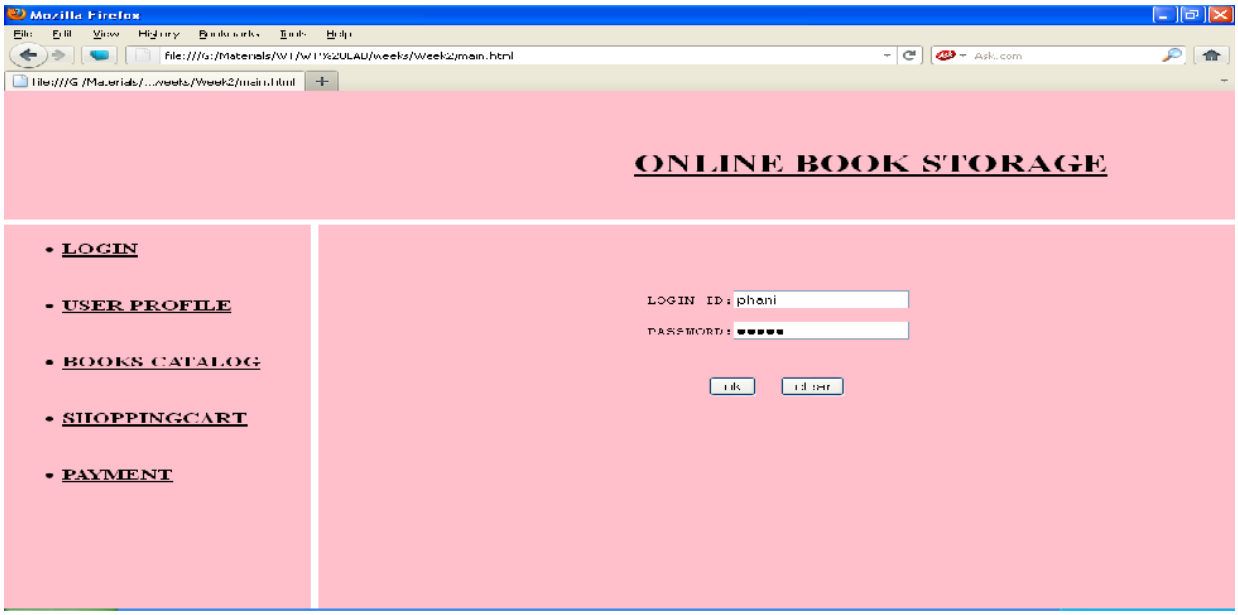
```
<html
>
<head><title>order conformation</title><M/head>
<body bgcolor="cyan">
<center>
<h1><b>AMAZON</h1>
<pre><strong>
<b>Your order Is Conformed
</strong></pre>
<h2><b>THANK YOU</h2>
</center>
</body>
</html>
```

OUTPUT:

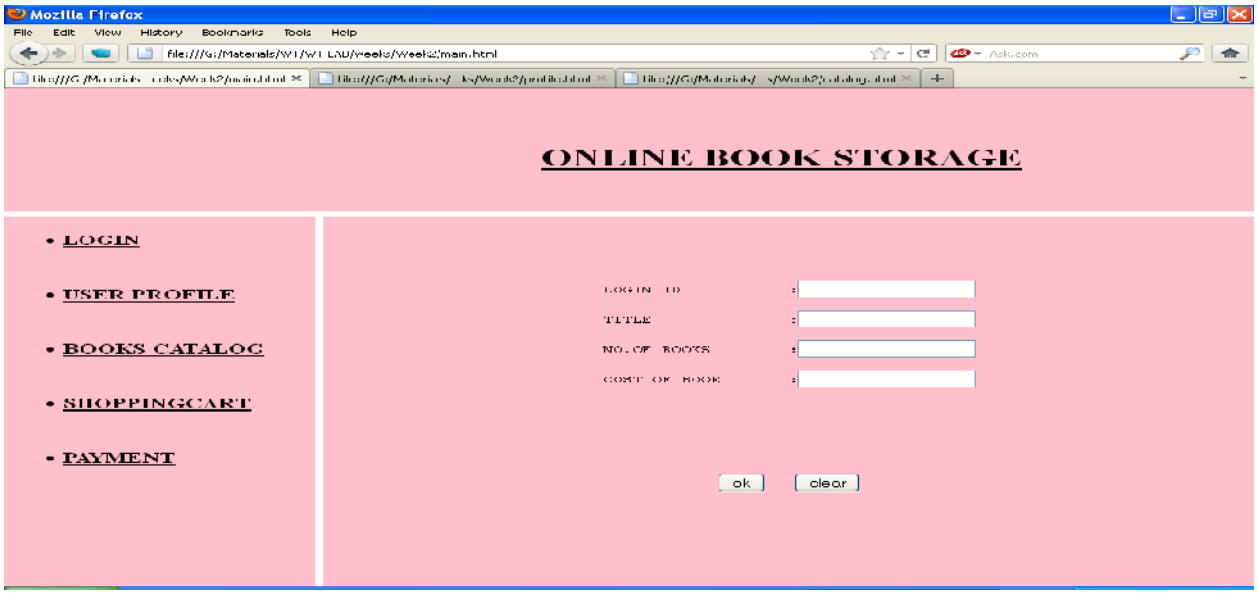
Main.html



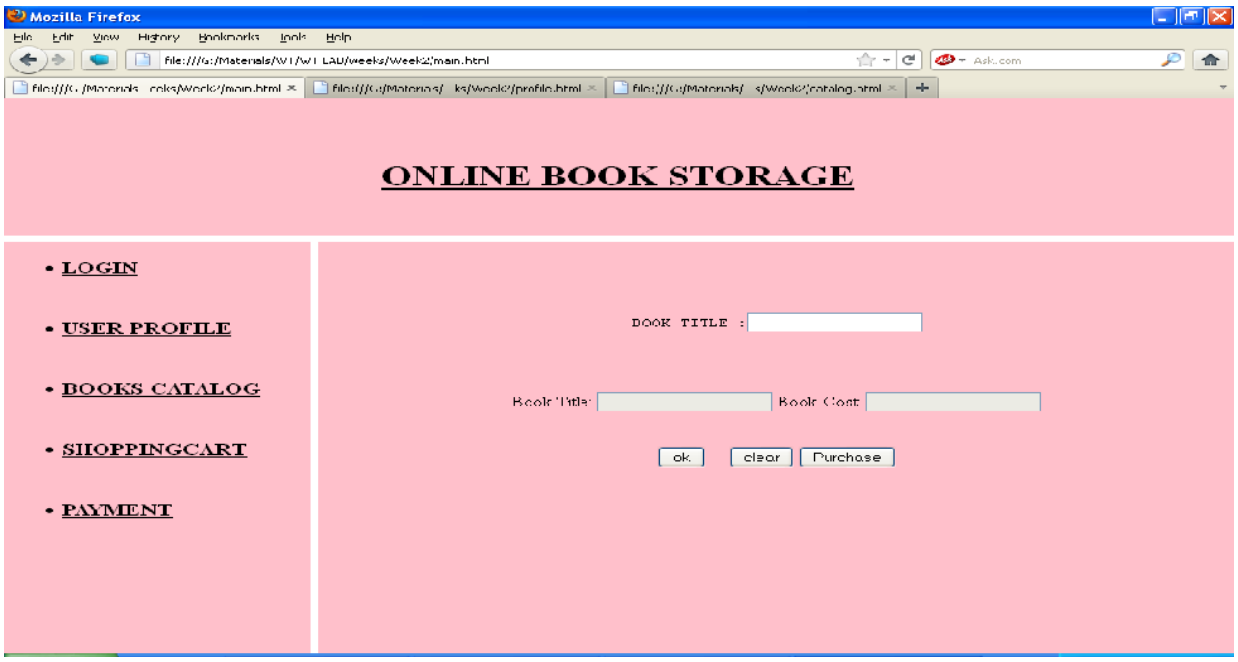
Login.html:



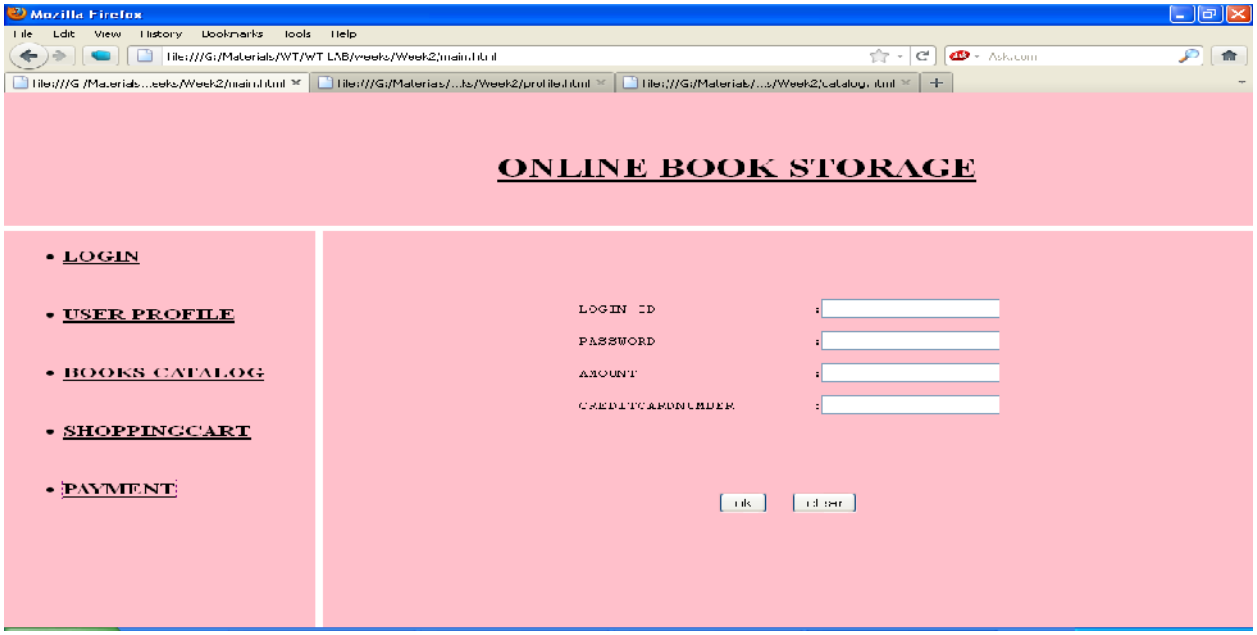
Catalog.html:



Scart.html:



Payment.html:



Order.html

WEEK 8:

AIM: Write an XML file which will display the Book information which includes the following:
1) Title of the book
2) Author Name
3) ISBN number
4) Publisher name
5) Edition
6) Price
Write a Document Type Definition (DTD) to validate the above XML

file.Display the XML file as follows.

The contents should be displayed in a table. The header of the table should be in color GREY. And the Author names column should be displayed in one color and should be capitalized and inbold. Use your own colors for remaining columns. Use XML schemas XSL and CSS for the above purpose.

1. Books.DTD:

```
<!ELEMENT details (title, author, ISBN_Number, publisher, edition, price) >
<!ELEMENT title (#PCDATA)>
<!ELEMENT author (#PCDATA)>
<!ELEMENT ISBN_Number (#PCDATA)>
<!ELEMENT  publisher (#PCDATA)>
<!ELEMENT edition (#PCDATA)>
<!ELEMENT price (#PCDATA)>
```

2. Th.CSS

```
.thb
{
background-color:gray;
}
.bg
{
background-color:red;
}
```

3. Books.XML:

```
<?xml version="1.0"?>
<!DOCTYPE book SYSTEM "books.dtd">
<book>
	<details>
		<title> C</title>
		<author> BalaGuru Swami</author>
		<ISBN_Number>2536</ISBN_Number>
		<publisher>pearson</publisher>
		<edition>2</edition>
		<price>255/-</price>
	</details>
	<details>
		<title> C++</title>
		<author> BalaGuru Swami</author>
		<ISBN_Number>5236</ISBN_Number>
		<publisher>pearson</publisher>
		<edition>2</edition>
		<price>315/-</price>
	</details>

	<details>
		<title> E-Commerce</title>
		<author> Kalakata</author>
		<ISBN_Number>8562</ISBN_Number>
		<publisher>pearson</publisher>
		<edition>5</edition>
		<price>300/-</price>
	</details>
	<details>
		<title> CO</title>
		<author> Marris </author>
		<ISBN_Number>4578</ISBN_Number>
		<publisher>Dream Tech</publisher>
		<edition>5</edition>
		<price>270/-</price>
	</details>
	<details>
		<title> Web Technologies</title>
		<author> Kumar </author>
		<ISBN_Number>5423</ISBN_Number>
		<publisher>Willay</publisher>
		<edition>6</edition>
```

```
            <price>500/-</price>
        </details>
        <details>
            <title> Web Programming</title>
            <author> Kumar </author>
            <ISBN_Number>1258</ISBN_Number>
            <publisher>Willay</publisher>
            <edition>6</edition>
            <price>500/-</price>
        </details>
</book>
```

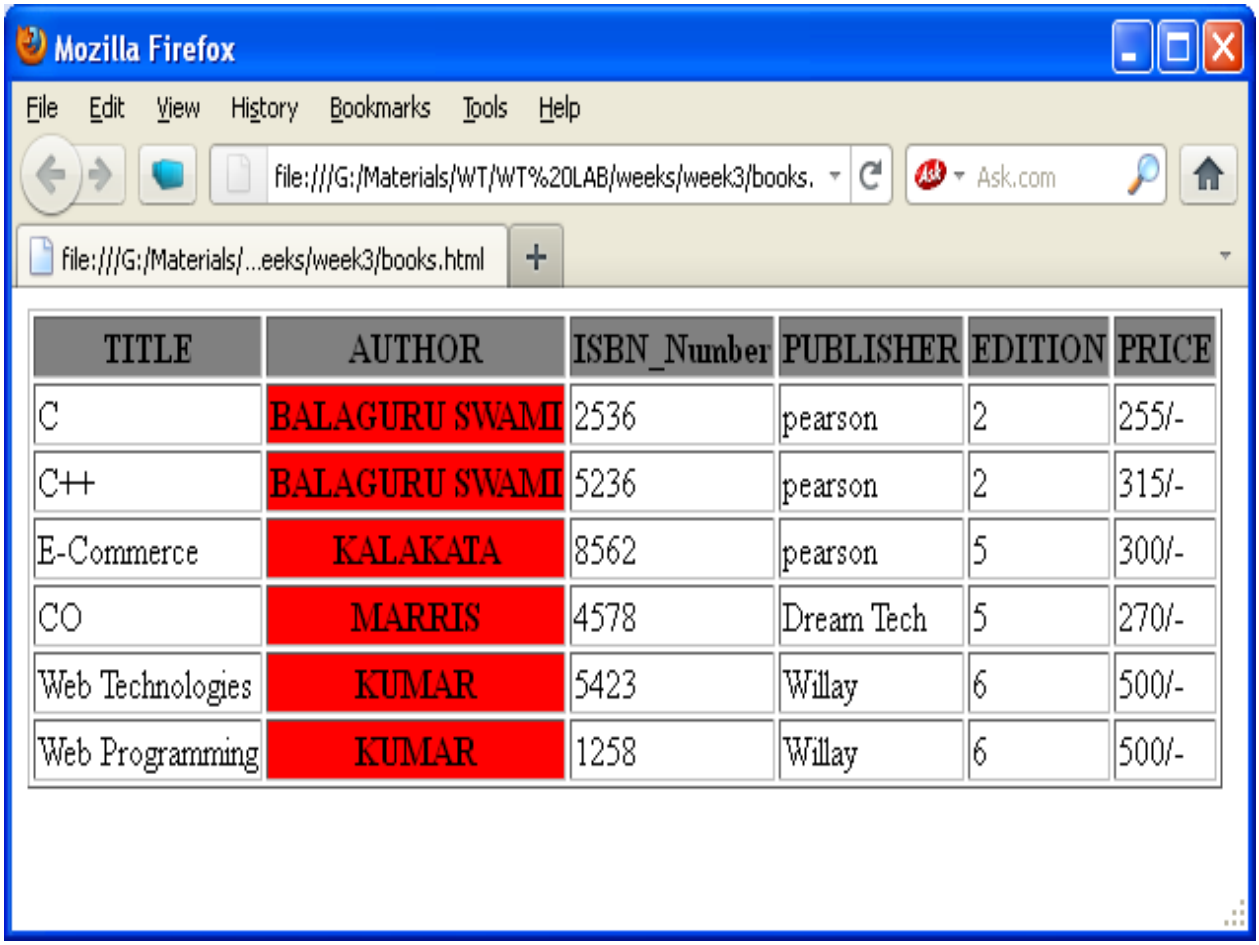
4. **Books.HTML:**

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="th.css">
</head>
<body>

<script

type="text/javascript">if

(window.XMLHttpRequest)
 {// code for IE7+, Firefox, Chrome, Opera, Safari
 xmlhttp=new XMLHttpRequest();
 }
else
 {// code for IE6, IE5
 xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
 }
xmlhttp.open("GET","books.xml",false);
xmlhttp.send();
xmlDoc=xmlhttp.responseXML;

document.write("<table border='1'>");
var x=xmlDoc.getElementsByTagName("details")
document.write("<tr><th class='thb'>");
document.write("TITLE</th><th class='thb'>AUTHOR</th><th
class='thb'>ISBN_Number</th><th
class='thb'>PUBLISHER</th><thclass='thb'>EDITION</th><th
class='thb'>PRICE</th></tr>");
for (i=0;i<x.length;i++)
 {
 document.write("<tr><td>");
```

```
document.write(x[i].getElementsByTagName("title")[0].childNodes[0].nodeValue);
document.write("</td><th class='bg'>");
```

```
document.write(x[i].getElementsByTagName("author")[0].childNodes[0].nodeValue.toU
pperCase());
 document.write("</th><td>");

document.write(x[i].getElementsByTagName("ISBN_Number")[0].childNodes[0].n
odeV alue);
 document.write("</td><td>");

document.write(x[i].getElementsByTagName("publisher")[0].childNodes[0].nodeValue);
 document.write("</td><td>");
 document.write(x[i].getElementsByTagName("edition")[0].childNodes[0].nodeValue);
 document.write("</td><td>");
 document.write(x[i].getElementsByTagName("price")[0].childNodes[0].nodeValue);
 document.write("</td></tr>");

 }
document.write("</table>");
</script>

</body>
</html>
```

OUTPUT:



TITLE	AUTHOR	ISBN_Number	PUBLISHER	EDITION	PRICE
C	BALAGURU SWAMI	2536	pearson	2	255/-
C++	BALAGURU SWAMI	5236	pearson	2	315/-
E-Commerce	KALAKATA	8562	pearson	5	300/-
CO	MARRIS	4578	Dream Tech	5	270/-
Web Technologies	KUMAR	5423	Willay	6	500/-
Web Programming	KUMAR	1258	Willay	6	500/-

WEEK 9

VISUAL BEANS:
Create a simple visual bean with a area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click.
Process:

Create a New Bean

Here are the steps that you must follow to create a new Bean:
1. Create a directory for the new Bean.
2. Create the Java source file(s).
3. Compile the source file(s).
4. Create a manifest file.
5. Generate a JAR file.
6. Start the BDK.
7. Test.

The following sections discuss each of these steps in detail.

Create a Directory for the New Bean

You need to make a directory for the Bean. To follow along with this example, create **colorsdirectory**. Then change to that directory.

Create the Source File for the New Bean

The source code for the **Colors** component is shown in the following listing. It is located in the file **Colors.java**.

The color of the component is determined by the private **Color** variable **color**, and its shape is determined by the private **boolean** variable **rectangular**. The constructor defines an anonymous inner class that extends **MouseAdapter** and overrides its **mousePressed( )** method. The **change( )** method is invoked in response to mouse presses. The component is initialized to a rectangular shape of 200 by 100 pixels.

The **change( )** method is invoked to select a random color and repaint the component. The **getRectangular( )** and **setRectangular( )** methods provide access to the one property of this Bean. The **change( )** method calls **randomColor( )** to choose a color and then calls **repaint( )** to make the change visible. Notice that the **paint( )** method uses the **rectangular** and **color** variables to determine how to present the Bean.

SOURCE CODE:- Colors.java:

```
import java.awt.*;
import
java.awt.event.*;
public class Colors extends Canvas
{
transient private Color
color;private boolean
rectangular; public
Colors()
{
addMouseListener(new MouseAdapter() {
public void mousePressed(MouseEvent
me) {change();
}
});
rectangular =
false;
setSize(200,
100); change();
}
public boolean getRectangular()
{
return rectangular;
}
public void setRectangular(boolean flag)
{
this.rectangular =
flag;repaint();
}
public void change()
{
color =
randomColor();
repaint();
}
private Color randomColor()
{
int r =
(int)(255*Math.random()); int
g = (int)(255*Math.random());
int b =
(int)(255*Math.random());
```

```
return new Color(r, g, b);
}
public void paint(Graphics g)
{
Dimension d =
getSize();int h =
d.height;
```

```
int w =
d.width;
g.setColor(color
);
if(rectangular)
{
g.fillRect(0, 0, w-1, h-1);
}
else
{
g.fillOval(0, 0, w-1, h-1);
}
}
}
```

Compile the Source Code for the New Bean

Compile the source code to create a class file. Type the following:
javac Colors.java.

Create a Manifest File

You must now create a manifest file. Put the source code for your manifest file in the file
colors.mft.
It is shown here:

```
Manifest-Version:
1.0Name:
Colors.class Java-
Bean: True
```

This file indicates that there is one **.class** file in the JAR file and that it is a Java Bean. Notice that the **Colors.class** file in the current directory.

Generate a JAR File

Beans are included in the ToolBox window of the BDK only if they are in JAR files in thedirectory **c:\bdk\jars**. These files are generated with the jar utility.
Enter the following:

jar cvfm colors.jar colors.mft *.class

This command creates the file **colors.jar.**

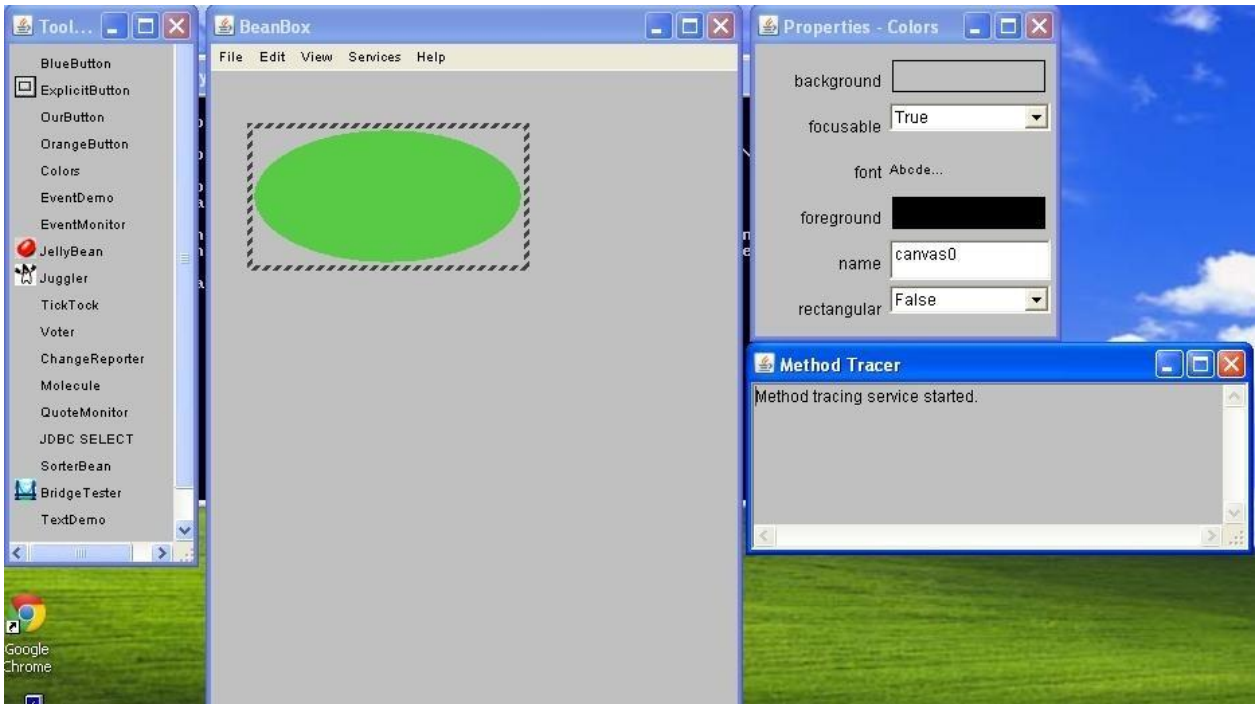
Start the BDK

Change to the directory **c:\bdk\beanbox** and type **run**. This causes the BDK to start. You shouldsee three windows, titled ToolBox, BeanBox, and Properties. The ToolBox window should include an entry labeled “Colors” for your new Bean.

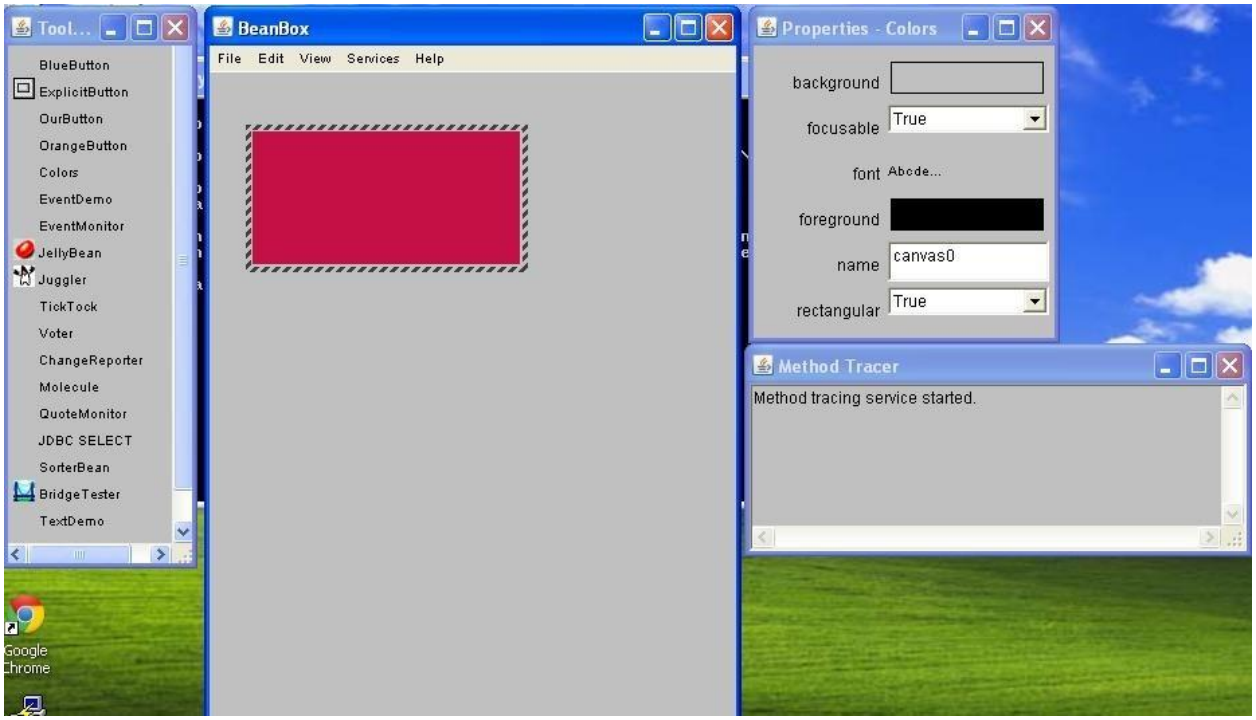
Create an Instance of the Colors Bean

After you complete the preceding steps, create an instance of the **Colors** Bean in the BeanBox window. Test your new component by pressing the mouse anywhere within its borders. Its colorimmediately changes. Use the Properties window to change the **rectangular** property from **false**to **true**. Its shape immediately changes.

Screen shot for circle:



Screen shot for triangle:



WEEK 10

1) Install TOMCAT web server. While installation assign port number 8080. Make sure thatthese ports are available i.e., no other process is using this port.
2) Access the above developed static web pages for books web site, using these servers byPutting the web pages developed in week-1 and week-2 in the document root.
Access the pages by using the urls : http://localhost:8080/rama/books.html

1. Install the TOMCAT web server:

Step 1:

Installation of JDK:

Before beginning the process of installing Tomcat on your system, ensure first the availability ofJDK on your system program directory. Install it on your system if not already installed (becauseany version of tomcat requires the Java 1.6 or higher versions) and then set the class path (environment variable) of JDK. To set the **JAVA_HOME Variable:** you need to specify the location of the java run time environment to support the Tomcat else Tomcat server can not run.

This variable contains the path of JDK installation directory.

```
set JAVA_HOME=C:\Program Files\Java\jdk1.6
```

Note: it should not contain the path up to bin folder. Here, we have taken the URL pathaccording to our installation convention.
For Windows OS, go through the following steps:

First, right click on the

```
My Computer->properties->advance->Environment
Variables->New->set the Variable name =
JAVA_HOME and variable value = C:\Program
Files\Java\jdk1.6
```

Now click on all the subsequent ok buttons one by one. It will set the JDK path.

Step 2:

For setting the class path variable for JDK, do like this:

First, right click on the

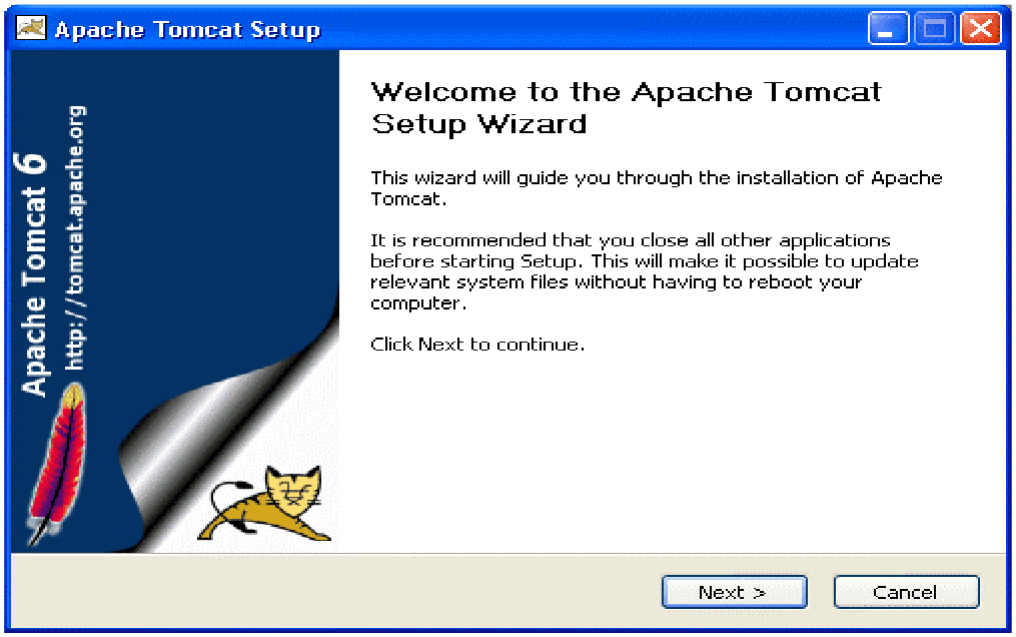
My Computer->properties->advance->Environment Variables->path.

Now, set bin directory path of JDK in the path variable

Step 3:

The process of installing Tomcat 6.0 begins here from now. It takes various steps for installingand configuring the Tomcat 6.0.

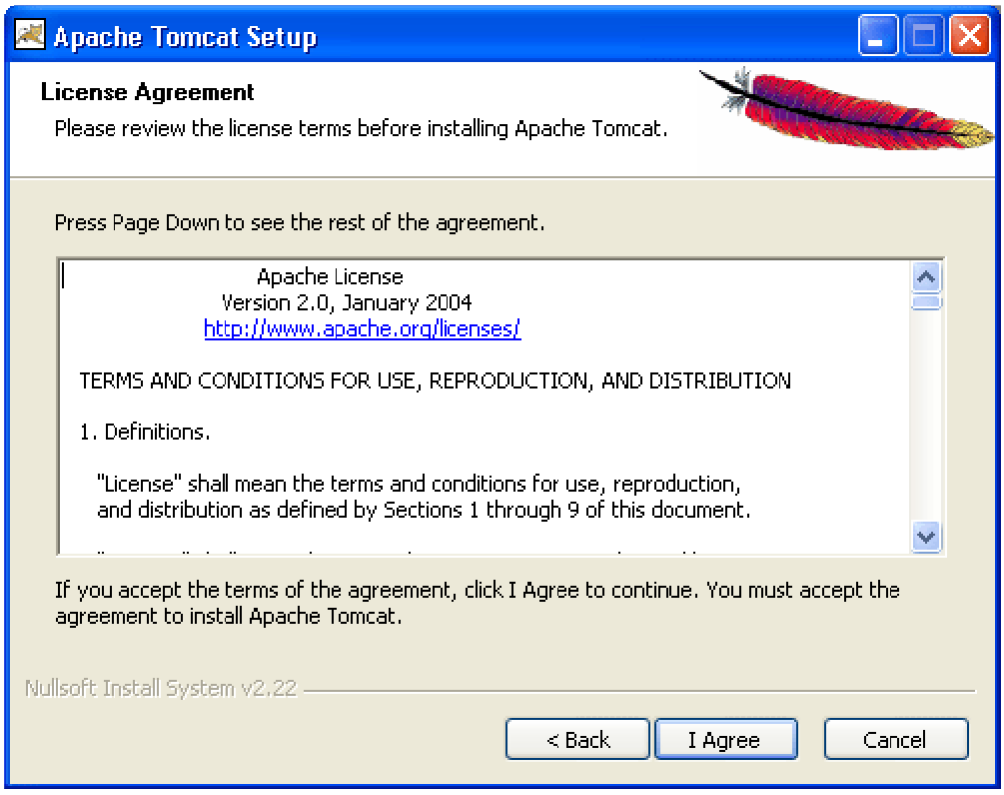
For Windows OS, Tomcat comes in two forms: .zip file and .exe file (the Windows installer file).Here we are exploring the installation process by using the .exe file. First unpack the zipped file and simply execute the '.exe' file.



A Welcome screen shot appears that shows the beginning of installation process. Just click onthe 'Next' button to proceed the installation process.

Steps 4:

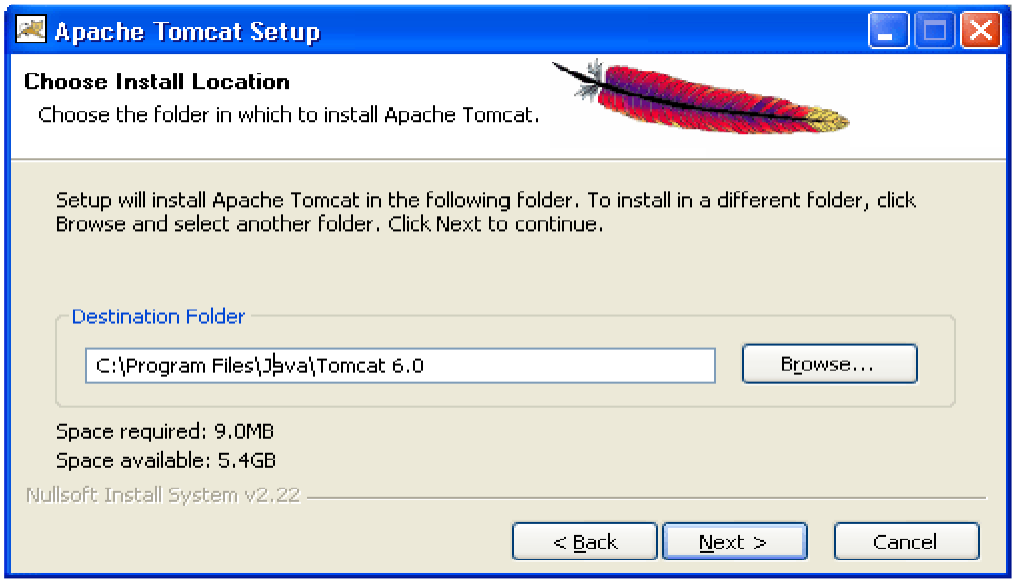
A screen of 'License Agreement' displays.



Click on the 'I Agree' button.

Step 5:

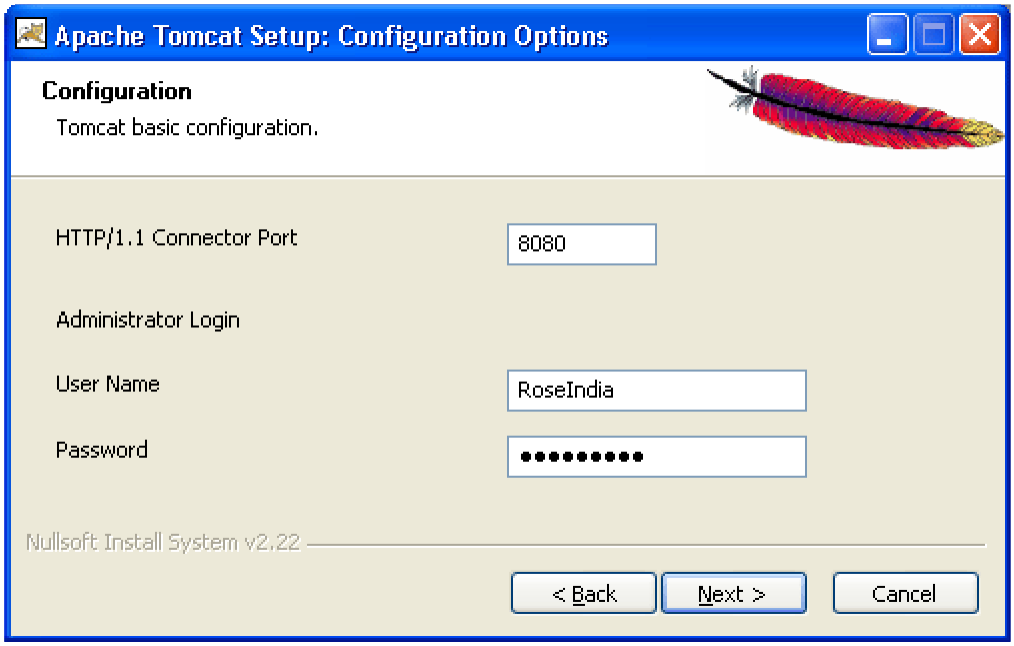
A screen shot appears asking for the 'installing location'



Choose the default components and click on the 'Next' button.

Step 6:

A screen shot of 'Configuration Options' displays on the screen. Choose the location for theTomcat files as per your convenience. You can also opt the default Location



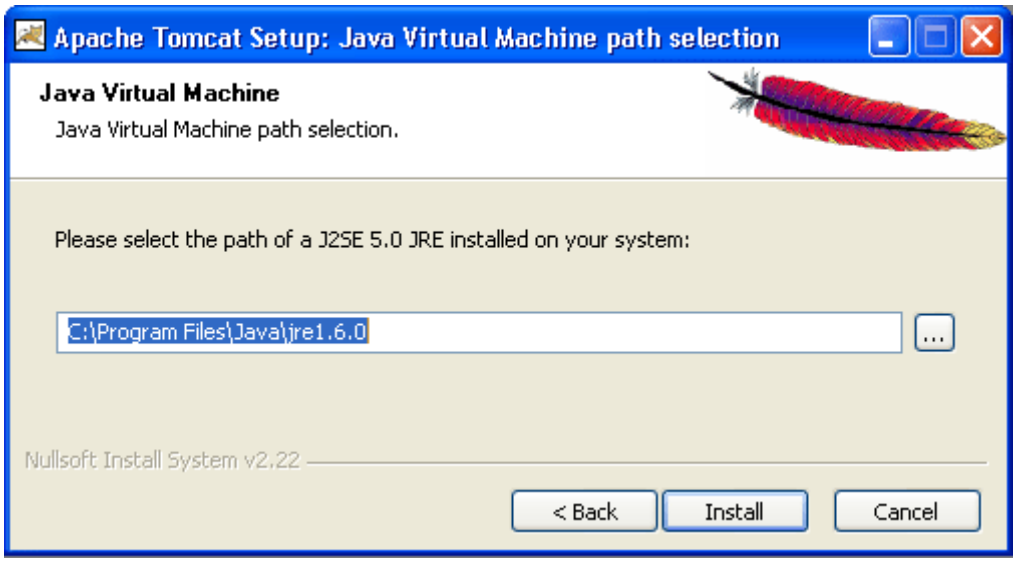
The port number will be your choice on which you want to run the tomcat server. The port number 8080 is the default port value for tomcat server to proceed the HTTP requests. The usercan also change the 'port number' after completing the process of installation; for this, users haveto follow the following tips.

Go to the specified location as " **Tomcat 6.0 \conf \server.xml** ". Within the server.xml filechoose "Connector" tag and change the port number.

Now, click on the 'Next' button to further proceed the installation process.

Step 7:

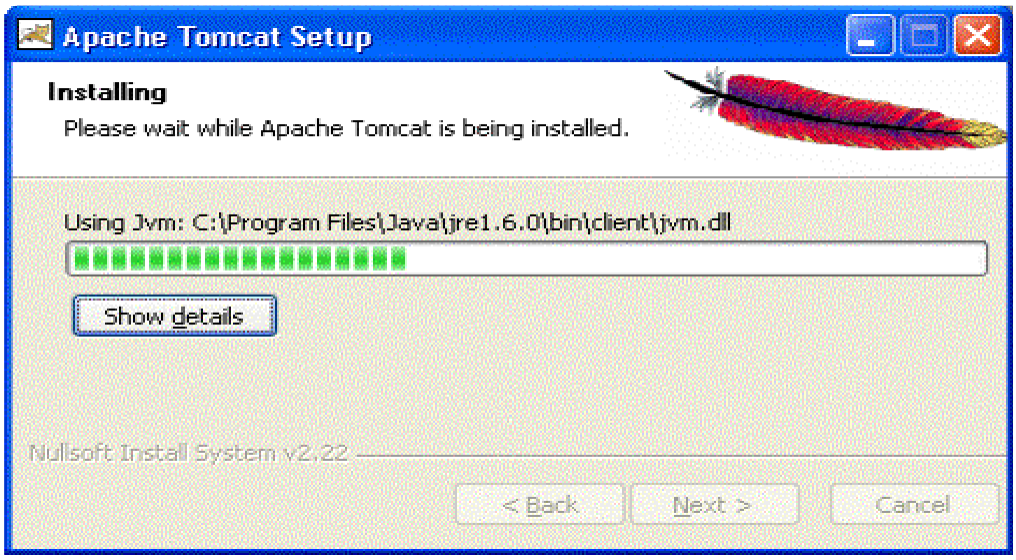
A Window of Java Virtual Machine displays on the screen



This window asks for the location of the installed Java Virtual Machine. Browse the location of the JRE folder and click on the Install button. This will install the Apache tomcat at the specifiedlocation.

Step 8:

A processing window of installing displays on the screen.



To get the information about installer click on the "Show details" button

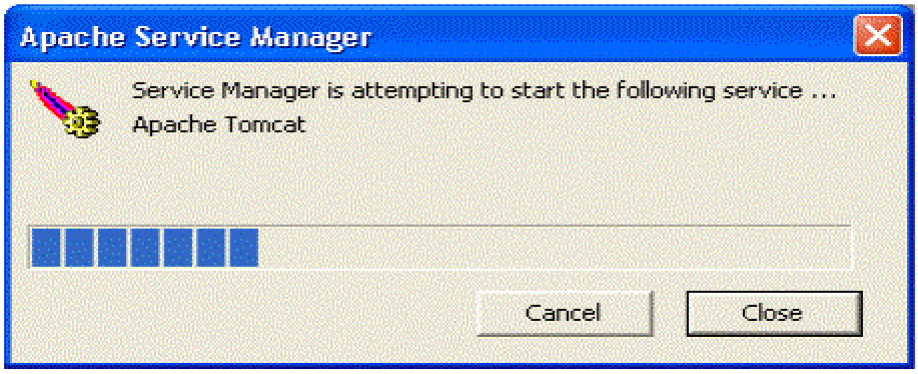
Step 9:

A screen shot of 'Tomcat Completion' displays on the screen.

Click on the 'Finish' button.

Step 10:

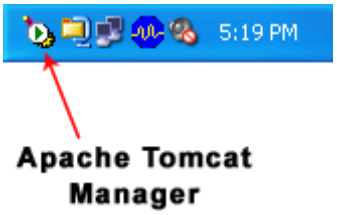
A window of Apache Service Manager appears with displaying the running process.



Let the running process goes on.

Step 11:

After completing the installation process, the Apache Tomcat Manager appears on the toolbarpanel like shown in the below picture.



Start the Tomcat Server:

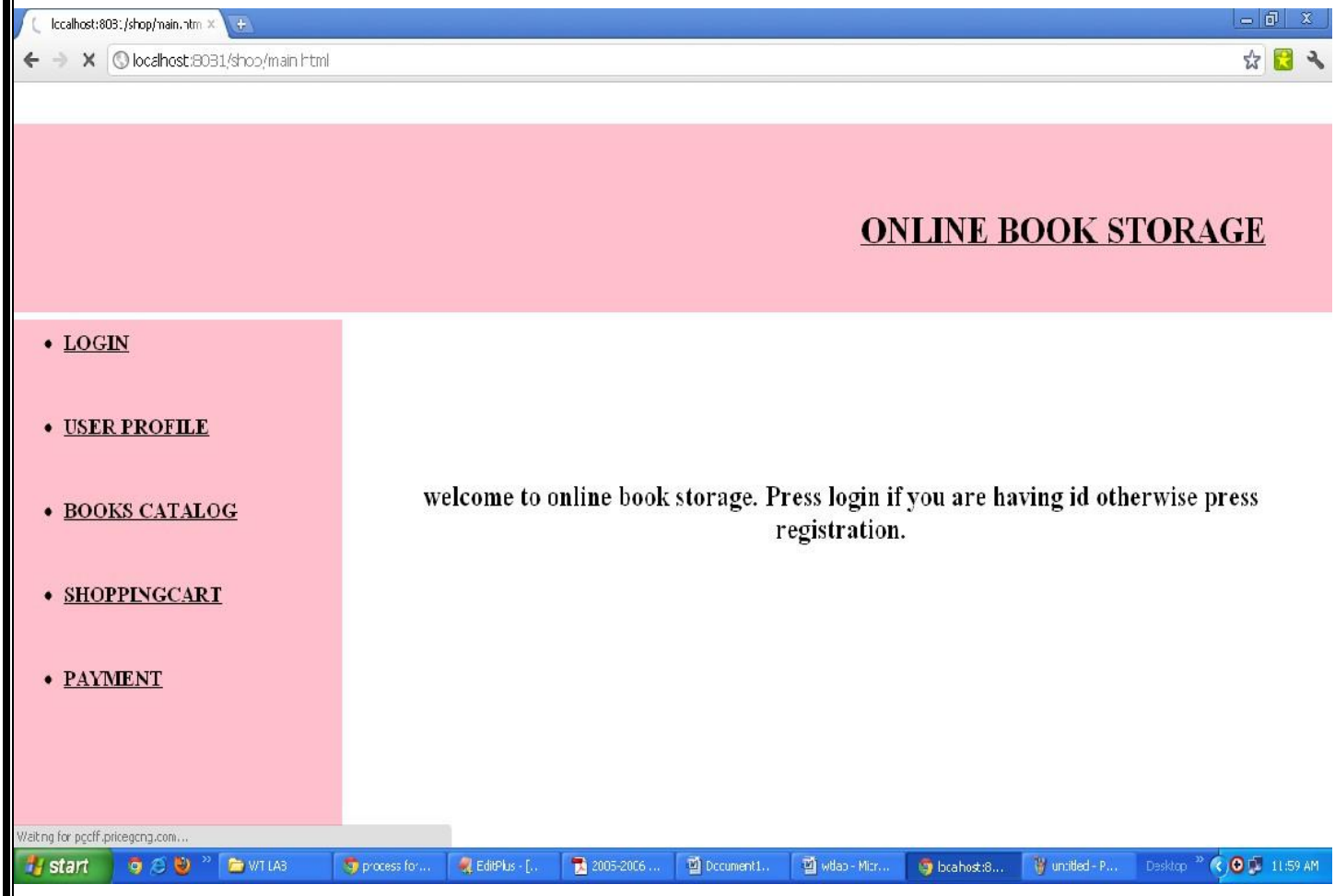
1. Start the tomcat server from the bin folder of Tomcat 6.0 directory by double clicking the"tomcat6.exe" file.
OR create a shortcut of this .exe file at your desktop.
2. Now Open web browser and type URL http://localhost:8080 in the address bar to test theserver
3. To Stop the Tomcat Server: Stop the server by pressing the "Ctrl + c" keys.

The screen of Apache Tomcat software looks like this:

PROCEDURE:

1. First install the tomcat into the system.
2. Then make a sub directory(eg., books) in the \tomcat\webapps.
3. Under books create WEB-INF directory and also place week1 programs in this booksdirectory only.
4. After this start tomcat by giving the following command at the instll_dir>tomcat>bin
5. Catalina.bat run
6. At the I.E(web browser) give the url as http://localhost:8080/ books /main.html
7. Port no 8080 is assigned for the tomcat.

Screen shot:



WEEK 11

User Authentication :

Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 And, pwd4 respectively. Write a servelet for doing the following.

1. Create a Cookie and add these four user id's and passwords to this Cookie.
2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.

If he is a valid user(i.e., user-name and password match) you should welcome him by name(user-name) else you should display " You are not an authenticated user ".

PROCEDURE:

1. First install the tomcat into the system.
2. Then make a subdirectly(eg., tr) in the \tomcat\webapps.
3. Under tr create WEB-INF directory and also place the html files in this tr directory only.
4. Next under WEB-INF create two subclasses lib,classes and web.xml
5. Next place all the class files under the classes and jar files(servlet-api.jar,classes12.jaretc…) under lib subdirectories.
6. After this start tomcat by giving the following command at the instll_dir>tomcat>bin
7. Catalina.bat run
8. At the I.E(web browser) give the url as http;//localhost:8080//tr/htmlfile or servlet urlpattern
9. Portno 8080 is assigned for the tomcat.

Login.html:

```
<html>
<body bgcolor="pink">
<form action="show" method="get">
<center>User Name: <input type="test"
name="name"><br>Password:<input type="password"
```

```
name="pass"><br>
<input type="submit" name="b1">
<input type="Button" name="b2" value="Reset">
</center>
</form>
</body>
</html>
```

Login.java:

```
import java.io.*;
import
javax.servlet.*;
import javax.servlet.http.*;
/** Example using servlet initialization. Here, the message
 *  to print and the number of times the message should be
 *  repeated is taken from the init parameters.
 */
public class login extends HttpServlet
{
        public void doPost(HttpServletRequest request, HttpServletResponse
         response)throws ServletException, IOException
        {
         response.setContentType("text/html");
          String
          na=request.getParameter("name");
          String
          pa=request.getParameter("pass");
          PrintWriter out =
          response.getWriter(); Cookie
          nam1=new Cookie("user1","pace");
          Cookie nam2=new
          Cookie("user2","phani"); Cookie
          nam3=new Cookie("user3","cse"); Cookie
          nam4=new Cookie("user4","ece"); Cookie
          pas1=new Cookie("pwd1","college");
          Cookie pas2=new
          Cookie("pwd2","kumar"); Cookie
```

```
pas3=new Cookie("pwd3","it"); Cookie
pas4=new Cookie("pwd4","eee");
int flag=0;
String
nam[]={nam1.getValue(),nam2.getValue(),nam3.getValue(),nam4.getValue()};
String
pas[]={pas1.getValue(),pas2.getValue(),pas3.getValue(),pas4.getValue()};
```

```
        for(int i=0;i<4;i++)
         {
                 if(nam[i].equals(na)&&pas[i].equals(pa))
                {
                 flag=1;
                }
        }
        if(flag==1)
        {
                  out.println("<title>The ShowMessage Servlet</title>");
                out.println("<BODY BGCOLOR=\"#FDF5E6\">\n" +"<H1 ALIGN=CENTER>
                WELCOME <br/>TO</br> " +na.toUpperCase() +
                "</H1>");out.println("</BODY></HTML>");
         }
         els
         e
         {       out.println("<title>The ShowMessage Servlet</title>");
                out.println("<BODY BGCOLOR=\"#FDF5E6\">\n" + "<H1 ALIGN=CENTER>
                User is invalid </H1>");
                 out.println("</BODY></HTML>");

        }

   }
}
```

Web.xml:

```
<web-app>
        <servlet>
                <servlet-name>log </servlet-name>
                <servlet-class>login </servlet-class>
        </servlet>
        <servlet-mapping>
                <servlet-name>log </servlet-name>
                <url-pattern> /show </url-pattern>
        </servlet-mapping>
</web-app>
```

Create a directory:

Create a directory “cookies”, in that directory copy login.html file and create a directory a”WEB-INF”. In that WEB-INF directory again create directory “classes” and copy web.xml file. **Compile the servlet:**

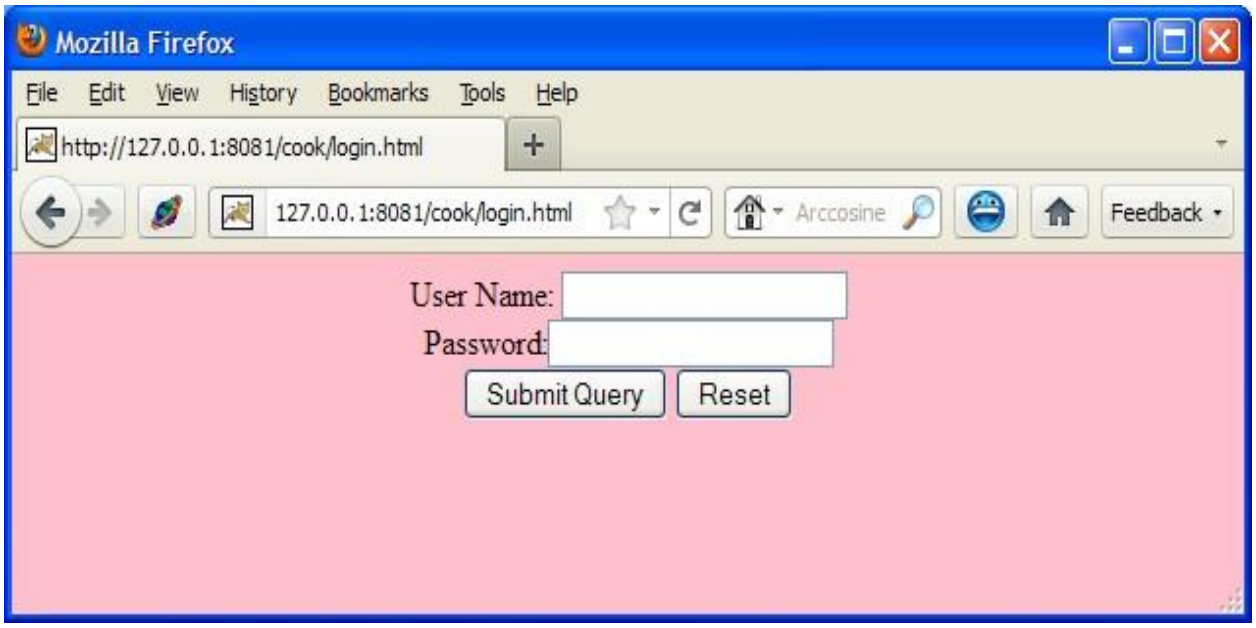
Compile the servlet file then copy the class file of that servlet into the directory **C:\Program Files\Apache Software Foundation\Tomcat 6.0\cookies\WEB-INF\classes.**

Open the server:

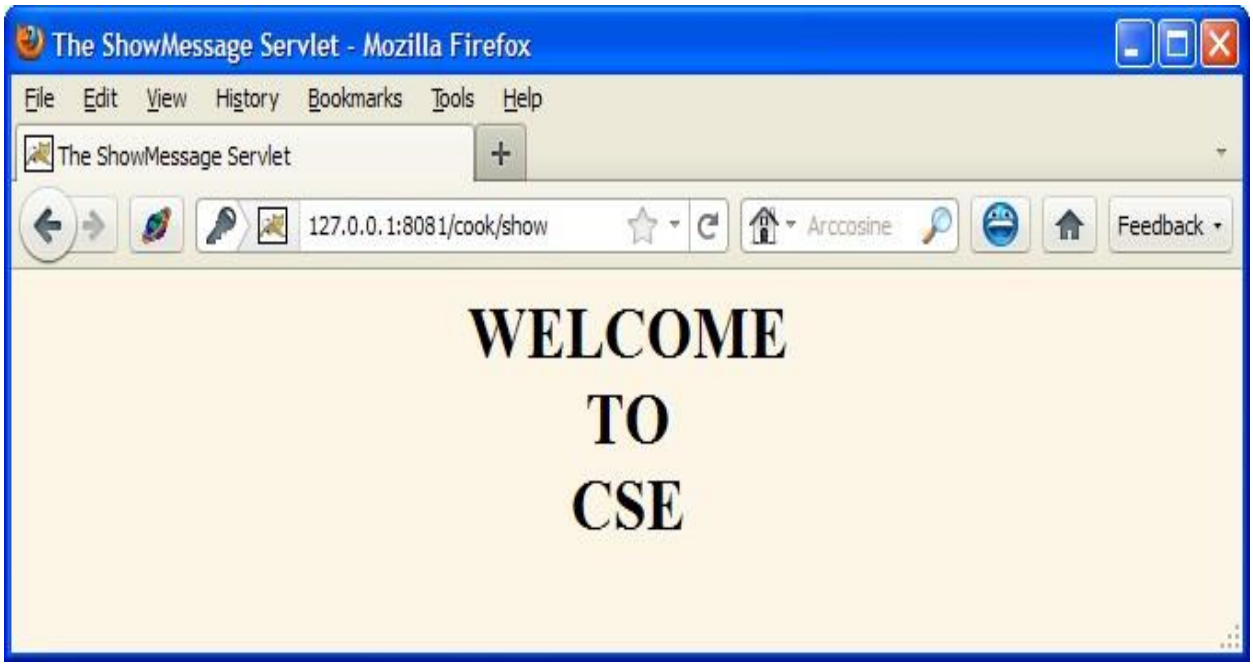
1. Start tomcat by giving the following command at the instll_dir>tomcat>binCatalina.bat run
2. At the I.E(web browser) give the url as http://localhost:8080/ cookies /login.html

OUTPUT:

Login.html:



Login.java:



WEEK-12:

Install a database(Mysql or Oracle).

Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form).

Practice 'JDBC' connectivity.

Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Experiment with various SQL queries. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

Main.html:

```
<html>
<body>
<br /><br /><br /><br />
<h1 align="center"><U>ONLINE BOOK STORAGE</U></h1><br/><br /><br />
<h2 align="center"><pre>
<b>Welcome to online book
storage.Press LOGIN if you are
having id otherwise press
REGISTRATION
</b></pre></h2>
<br /><br /><pre>
<div align="center"><a href="login.html">LOGIN</a> <a
href="reg.html">REGISTRATION</a></div></pre>
</body>
</html>
```

login.html:

```
<html>
<body><br /><br /><br />
<form name="myform" method="post" action="login">
<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"/>
```

```
&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;<input type="reset" value="clear"
/>
</div>
</form>
</body>
</html>
```

Profile.html

```
<html>
<body><br /><br /><br />
<form name="myform" method="post" action="profile">
<div align="center"><pre>
LOGIN ID :<input type="text" name="id" /><br />
</pre><br /><br />
</div>
<br /><br />
<div align="center">
<input type="submit" value="ok" onclick="validate()" />
&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;<input type="reset" value="clear" />
</div>
</form>
</body>
</html>
```

Userinfo.html

```
<html>
<head>
<title>User Info Entry Form</title>
</head>
<body bgcolor="white">
<form action="userinfo1.jsp" method="post">
<table>
<tr>
<td>Name:</td>
<td><input type="text" name="userName" >
</td>
</tr>
<tr>
<td>Sex:</td>
<td><input type="text" name="sex" >
</td>
<td>(Male or female)</td>
</tr>
<tr>
<td colspan=2><input type="submit"></td>
</tr>
</table>
</form>
</body>
</html>
```

Reg.html

```
<html>
<body><br /><br />
<form name="myform" method="post" action="reg">
<table align="center" >

  <tr>
    <td>NAME</td>
    <td>:<input type="text" name="name" /></td>
  </tr>
  <tr>
    <td>ADDRESS</td>
    <td>:<input type="text" name="addr" /></td>
  </tr>
  <tr>
    <td>CONTACT NUMBER</td>
    <td>:<input type="text" name="phno" /></td>
  </tr>
  <tr>
    <td>LOGINID</td>
    <td>:<input type="text" name="id" /></td>
  </tr>
  <tr>
    <td>PASSWORD</td>
    <td>:<input type="password" name="pwd" /></td>
  </tr>
</table>
<br /><br />
<div align="center">
<input type="submit" value="ok" onclick="validate()" />
      <input type="reset" value="clear" />
</div>
</form>
</body>
</html>
```

login.java

```
import java.sql.*;
import java.io.*;
import
javax.servlet.*;
import javax.servlet.http.*;
public class login extends HttpServlet
{
        public void service(HttpServletRequest
        req,HttpServletResponse resp)throws
        ServletException,IOException
        {
```

```
PrintWriter
pw=resp.getWriter();
pw.println("<html><body>");
```

```
String id=req.getParameter("id");
String
pwd=req.getParameter("pwd");
String s1="",s2="";
try
{
        Class.forName("oracle.jdbc.driver.OracleDrive
        r");Connection con=
        DriverManager.getConnection("jdbc:oracle:thin:@localhost:1522:XE","sys
        tem", "tiger");
        Statement
        stmt=con.createStatement(); String
        sqlstmt="select * from login";
        ResultSet
        rs=stmt.executeQuery(sqlstmt);int
        flag=0;
        while(rs.next())
        {
                s1=rs.getString(4
                        );
                s2=rs.getString(5
                        );
        }

                if(id.equals(s1)&&pwd.equals(s2))
                {
                        flag=1;
                }

        if(flag==0)
        {
        pw.println("<br><br>SORRY INVALID ID TRY AGAIN ID<br><br>");
        pw.println("<a href=\"login.html\">press LOGIN to RETRY</a>");
        }
        else
        {
        pw.println("<br><br>WELCOME TO
        "+id.toUpperCase()+"<br><br>");pw.println("<h3><ul>");
        pw.println("<li><a
                href=\"profile.html\"><fontcolor=\"black\">USER
                PROFILE</font></a></li><br><br>");
        pw.println("<li><a href=\"catalog.html\"><fontcolor=\"black\">BOOKS
        CATALOG</font></a></li><br><br>");
        pw.println("<li><a href=\"order.html\">
        <fontcolor=\"black\">ORDER
        CONFIRMATION</font></a></li></ul><br><br>");
        }
        pw.println("</body></html>");
}
catch(Exception e)
```

```
                {
                        resp.sendError(500,e.toString());
                }
        }
}
```

reg.java:

```
import java.sql.*;
import java.io.*;
import
javax.servlet.*;
import javax.servlet.http.*;
public class  reg extends HttpServlet
{
        public void service(HttpServletRequest req,
        HttpServletResponse resp)throws ServletException,IOException
        {
                PrintWriter
                pw=resp.getWriter();
                resp.setContentType("text/ht
                ml");
                pw.println("<html><body>");
                String
                name=req.getParameter("name");
                String
                addr=req.getParameter("addr");
                String
                phno=req.getParameter("phno");
                String id1=req.getParameter("id");
                String
                pwd1=req.getParameter("pwd");

                tr
                y
                {       Class.forName("oracle.jdbc.driver.OracleDriver");
                        Connection con=DriverManager.getConnection
                                ("jdbc:oracle:thin:@localhost:1522 :XE","system","tiger");
                        Statement
                        stmt=con.createStatement(); String
                        sqlstmt="select * from login";
                        ResultSet
                        rs=stmt.executeQuery(sqlstmt);int
                        flag=0;
                        while(rs.next())
                        {


                                if(id1.equals(rs.getString(4))&&pwd1.equals(rs.getString(5)))
                                {
                                        flag=1;
                                }
                        }
                        if(flag==1)
                        {
```

```
pw.println("<br><br>SORRY INVALID ID ALREADY EXITS
                        TRY AGAIN WITH NEW ID<br><br>");
pw.println("<a href=\"reg.html\">press REGISTER to RETRY</a>");
}
else
{
Statement
stmt1=con.createStatement();
stmt1.executeUpdate("insert into login
values
                ("'+name+"',"'+addr+"',"'+phno+"',"'+id1+"',"'+pwd1+"')");
```

```
                pw.println("<br><br>YOUR DETAILS ARE ENTERED<br><br>");
                pw.println("<a href=\"login.html\">press LOGIN to login</a>");
                }
                pw.println("</body></html>");
            }
            catch(Exception e)
            {
                resp.sendError(500,e.toString());
            }
        }
}
```

Catalog.java:

```
import java.sql.*;
import java.io.*;
import
javax.servlet.*;
import javax.servlet.http.*;
public class catalog extends HttpServlet
{
    public void service(HttpServletRequest
    req,HttpServletResponse resp)throws
    ServletException,IOException
    {
        PrintWriter
        pw=resp.getWriter();
        pw.println("<html><body>");
        String
        title=req.getParameter("title");
        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
Connection
            con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1522:XE"
            ,"system","tiger");Statement stmt=con.createStatement();
            String sqlstmt="select * from book where
            title=\'"+title+"\'";ResultSet
            rs=stmt.executeQuery(sqlstmt);
            int flag=0;
            while(rs.next(
            ))
            {
                pw.println("<div align=\"center\">");
                pw.println("TITLE
                                        :"+rs.getString(1)+"<br>");
                pw.println("AUTHOR      :"+rs.getString(2)+"<br>");
                pw.println("VERSION     :"+rs.getString(3)+"<br>");
```

```
            pw.println("PUBLISHER
                                                :"+rs.getString(4)+"<br>");
            pw.println("COST
                                                :"+rs.getString(5)+"<br>");
            pw.println("</div>");
            flag=1;
    }
    if(flag==0)
    {
    pw.println("<br><br>SORRY INVALID TITLE TRY AGAIN <br><br>");
```

```
                        pw.println("<a href=\"catalog.html\">press HERE to RETRY</a>");
                        }
                        pw.println("</body></html>");
                }
                catch(Exception e)
                {
                        resp.sendError(500,e.toString());
                }
        }
}
```

profile.java:

```
import java.sql.*;
import java.io.*;
import
javax.servlet.*;
import javax.servlet.http.*;
public class profile extends HttpServlet
{
        public void service(HttpServletRequest
        req,HttpServletResponse resp)throws
        ServletException,IOException
        {
                PrintWriter
                pw=resp.getWriter();
                pw.println("<html><body>");
                String
                id=req.getParameter("id");try
                {
                        Class.forName("oracle.jdbc.driver.OracleDriver");
                        Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1522:XE","system","tiger");
                        Statement stmt=con.createStatement();
                        String sqlstmt="select * from login where
                        id="'"+id+"'";ResultSet
                        rs=stmt.executeQuery(sqlstmt);
                        int flag=0;
                        pw.println("<br><br><br>"
                        );while(rs.next())
                        {
                                pw.println("<div align=\"center\">");
                                pw.println("NAME
                                                        :"+rs.getString(1)+"<br>"
                                );pw.println("ADDRESS
                                :"+rs.getString(2)+"<br>");
                                pw.println("PHONE NO
                                                                :"+rs.getString(3)+"<br>"
                                );pw.println("</div>");
                                flag=1;
                        }
```

```
            if(flag==0)
            {
            pw.println("<br><br>SORRY INVALID ID TRY AGAIN ID<br><br>");
            pw.println("<a href=\"profile.html\">press HERE to RETRY</a>");
            }
            pw.println("</body></html>");
    }
```

```
                catch(Exception e)
                {
                        resp.sendError(500,e.toString());
                }
        }
}
```

Order.java:

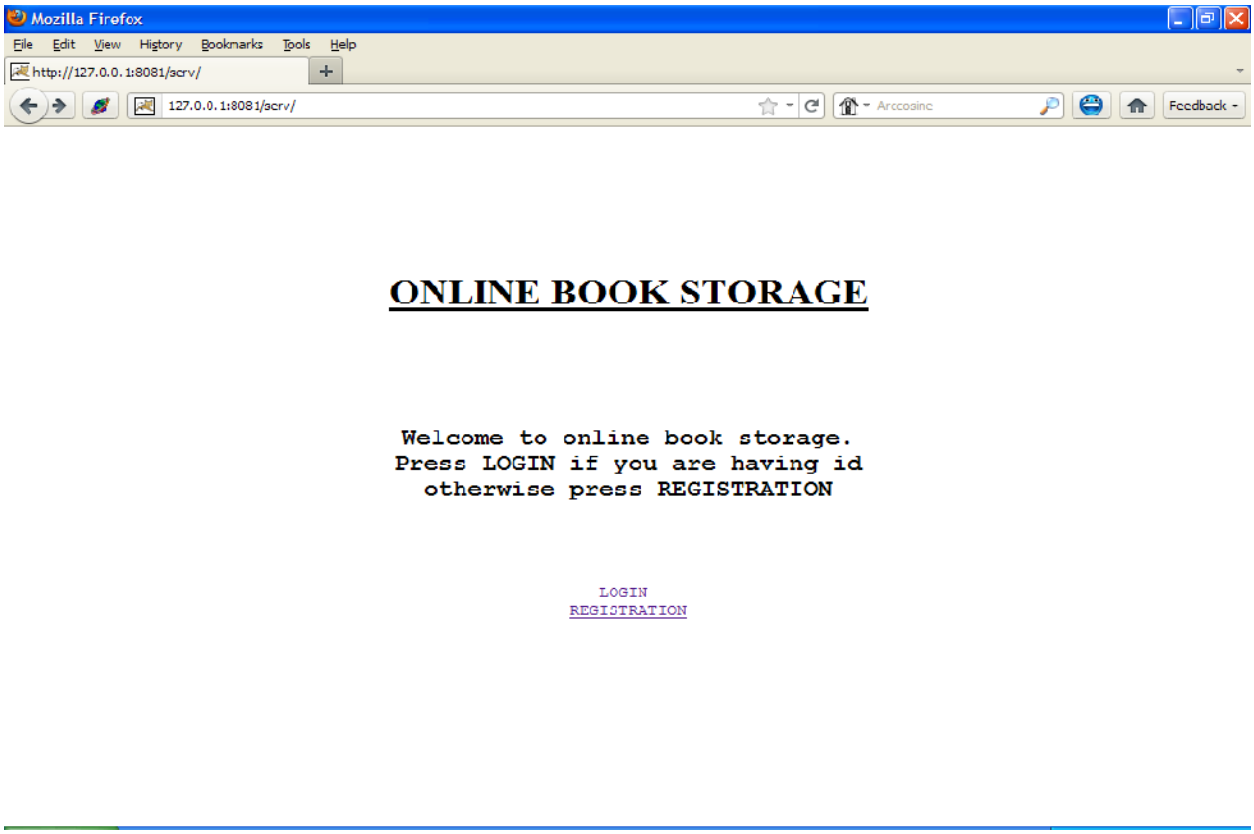
```
import java.sql.*;
import java.io.*;
import
javax.servlet.*;
import javax.servlet.http.*;
public class order extends HttpServlet
{
        public void service(HttpServletRequest
        req,HttpServletResponse resp)throws
        ServletException,IOException
        {
        int count;
                PrintWriter
                pw=resp.getWriter();
                pw.println("<html><body>");
                String
                id=req.getParameter("id");
                String
                pwd=req.getParameter("pwd");
                String
                title=req.getParameter("title");
                String
                count1=req.getParameter("no");
                String
                date=req.getParameter("date");
                String
                cno=req.getParameter("cno"); try
                {
                        count=Integer.parseInt(count1);
                        Class.forName("oracle.jdbc.driver.OracleDriver");
Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1522:XE","system","tiger");
                        Statement stmt=con.createStatement();
                        String sqlstmt="select * from login";
                        ResultSet
                        rs=stmt.executeQuery(sqlstmt);int
                        flag=0,amount,x;
                        while(rs.next())
                        {
                                if(id.equals(rs.getString(4))&&pwd.equals(rs.getString(5)))
                                {
                                        flag=1;
```

```
        }
}
if(flag==0)
{
pw.println("<br><br>SORRY INVALID ID TRY AGAIN ID<br><br>");
pw.println("<a href= \" order.html \" >press HERE to RETRY</a>");

}
else
{
```

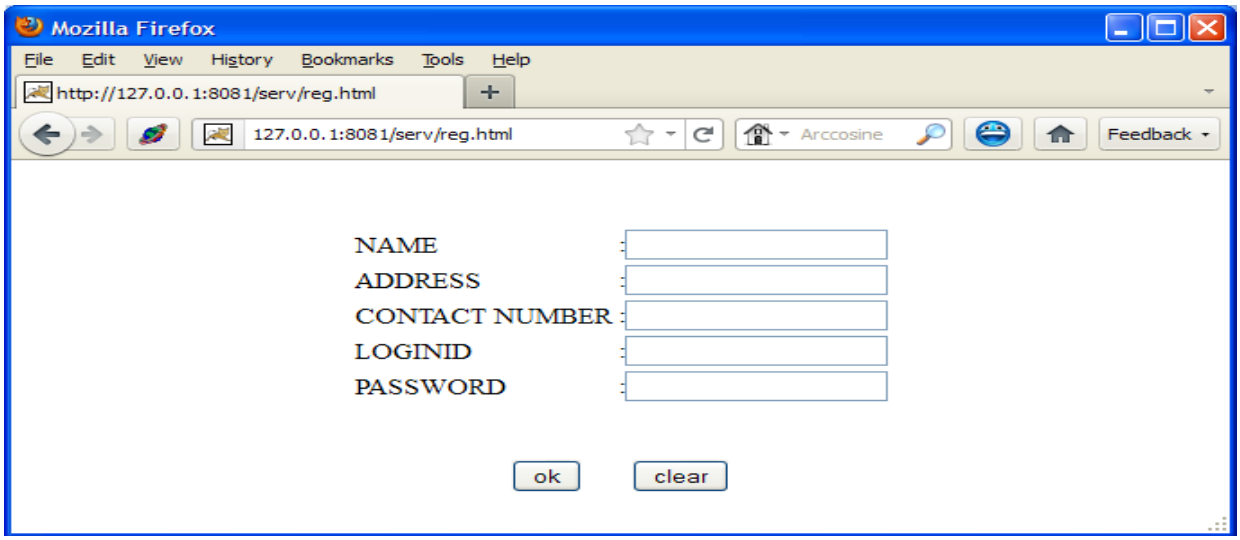
```
                                Statement stmt2=con.createStatement();
                                String s="select cost from book where title='"+title+"'";
                                ResultSet rs1=stmt2.executeQuery(s);
                                int flag1=0;
                                while(rs1.next(
                                ))
                                {
                                        flag1=1;
                                        x=Integer.parseInt(rs1.getString(
                                        1));amount=count*x;

        pw.println("<br><br>AMOUNT:"+amount+"<br><br><br><br>")
        ; Statement stmt1=con.createStatement();
        stmt1.executeUpdate("insert into details values('"+id+"','"+title+"','"+amount+"','"+cno+"')");
        pw.println("<br>YOUR ORDER has taken<br>");
                                }
                                if(flag1==0)
                                {
                        pw.println("<br><br><br>SORRY INVALID ID TRY AGAIN ID<br><br>");
                        pw.println("<a href=\"order.html\">press HERE to RETRY</a>");

                    }
                }

                  pw.println("</body></html
                >");con.close();
          }
          catch(Exception e)
          {
             resp.sendError(500,e.toString());
        }
    }
}
```

OUTPUT:

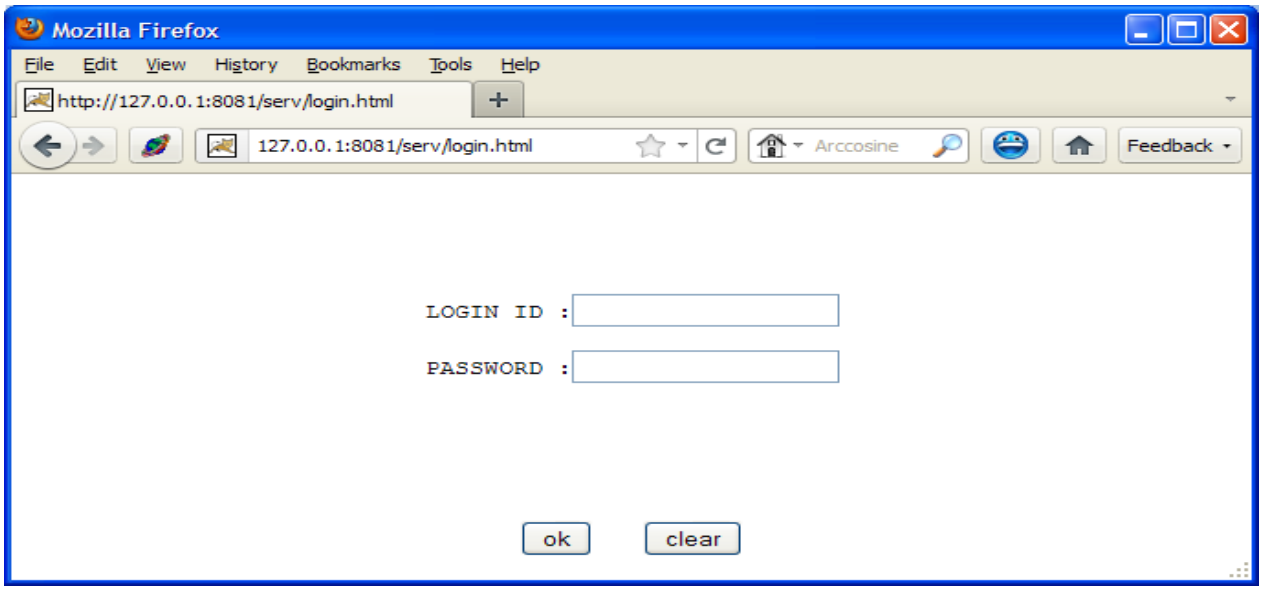
Main.html:



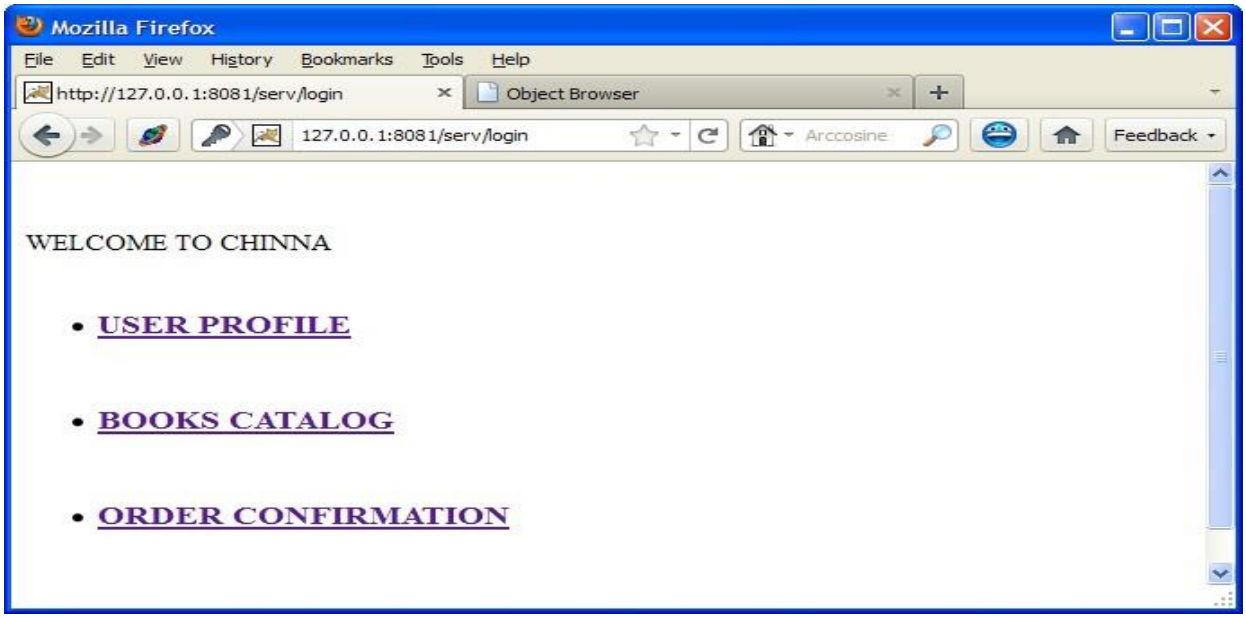
Registration:



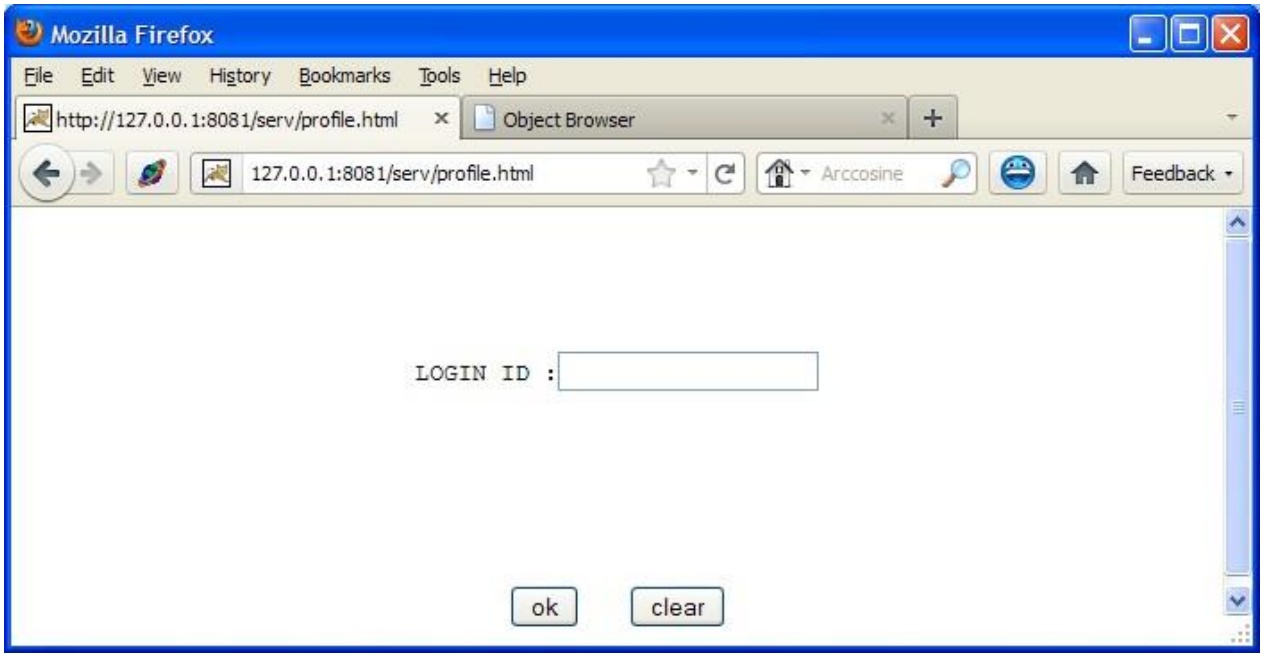
Login Page:



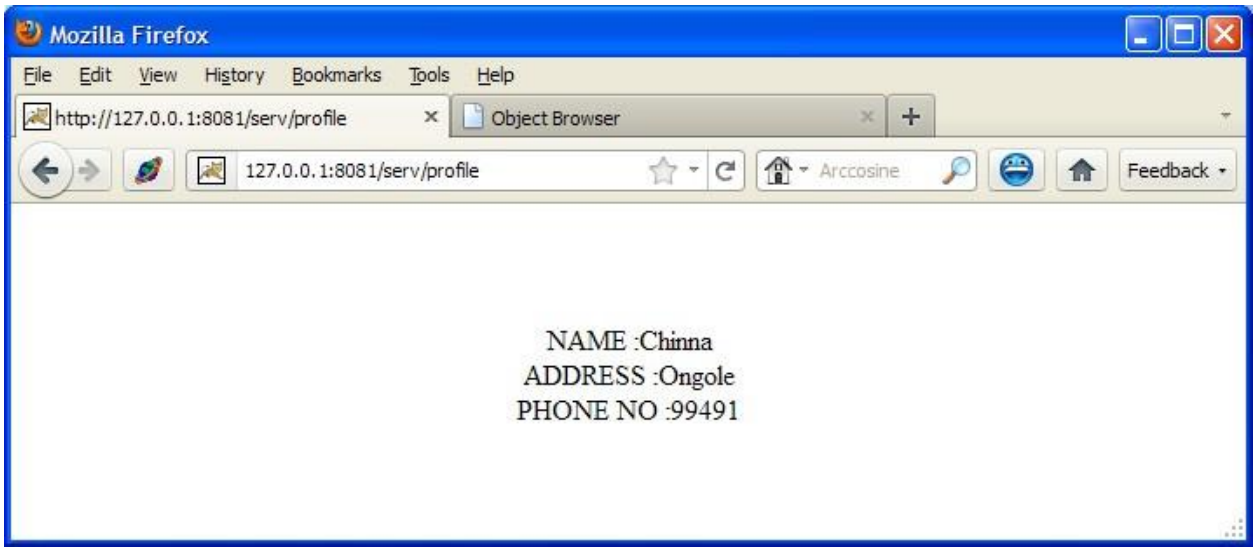
Login Servlet page:



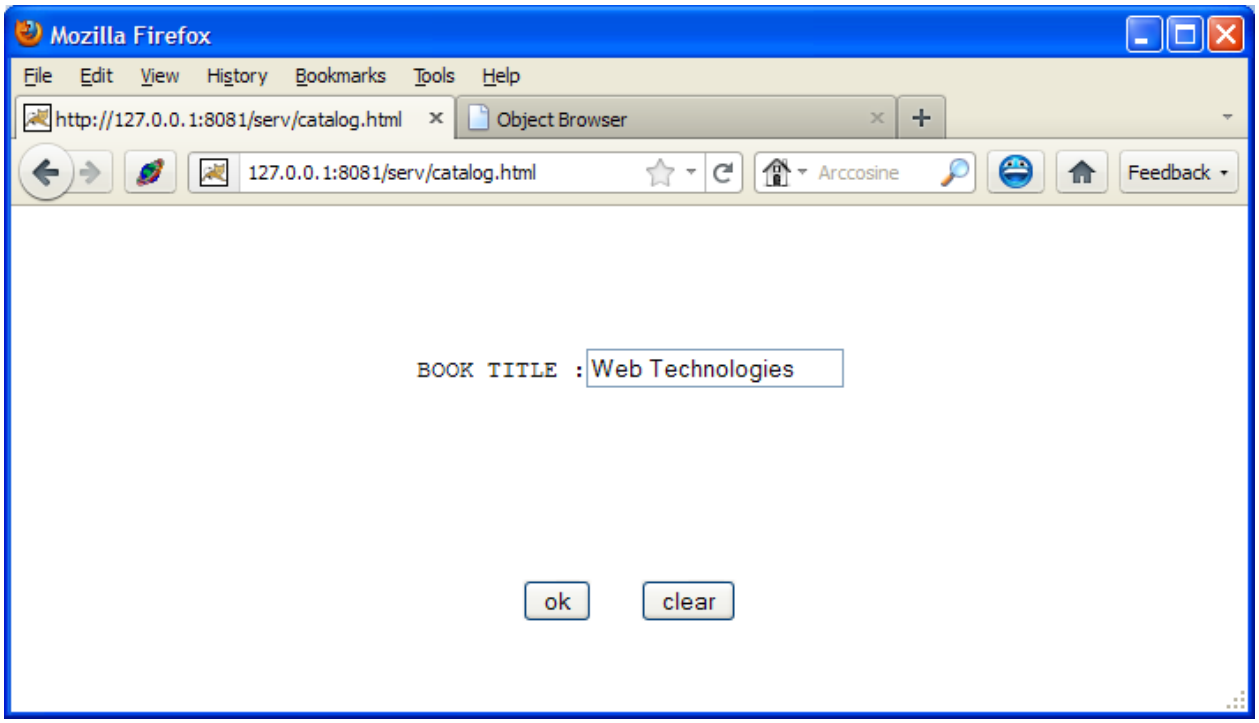
Profile page:



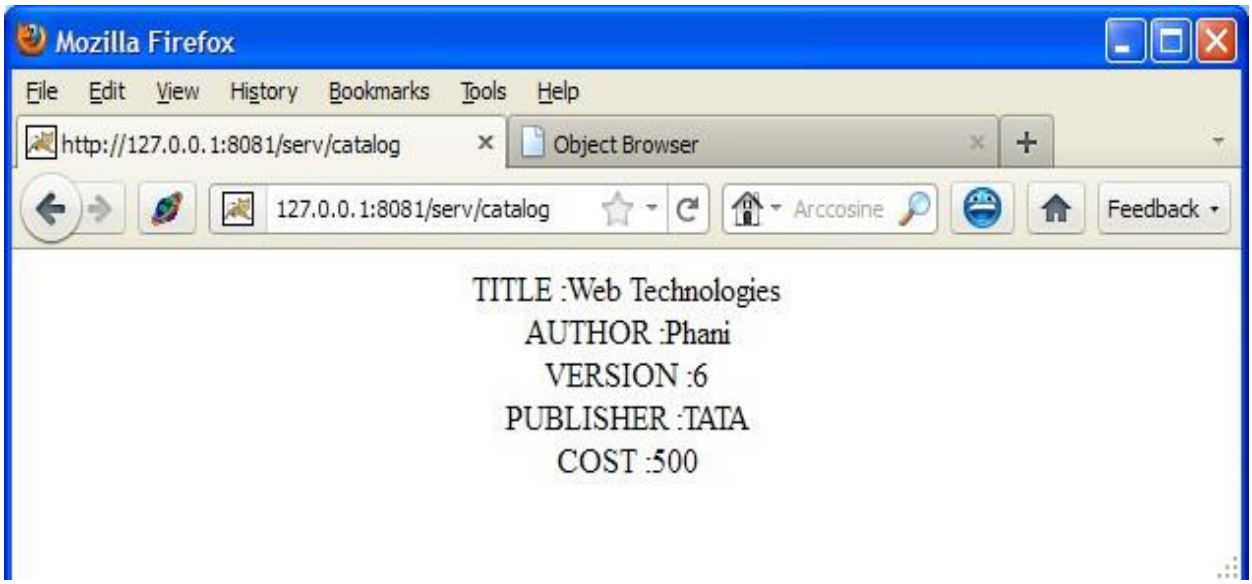
Profile Servlet page:



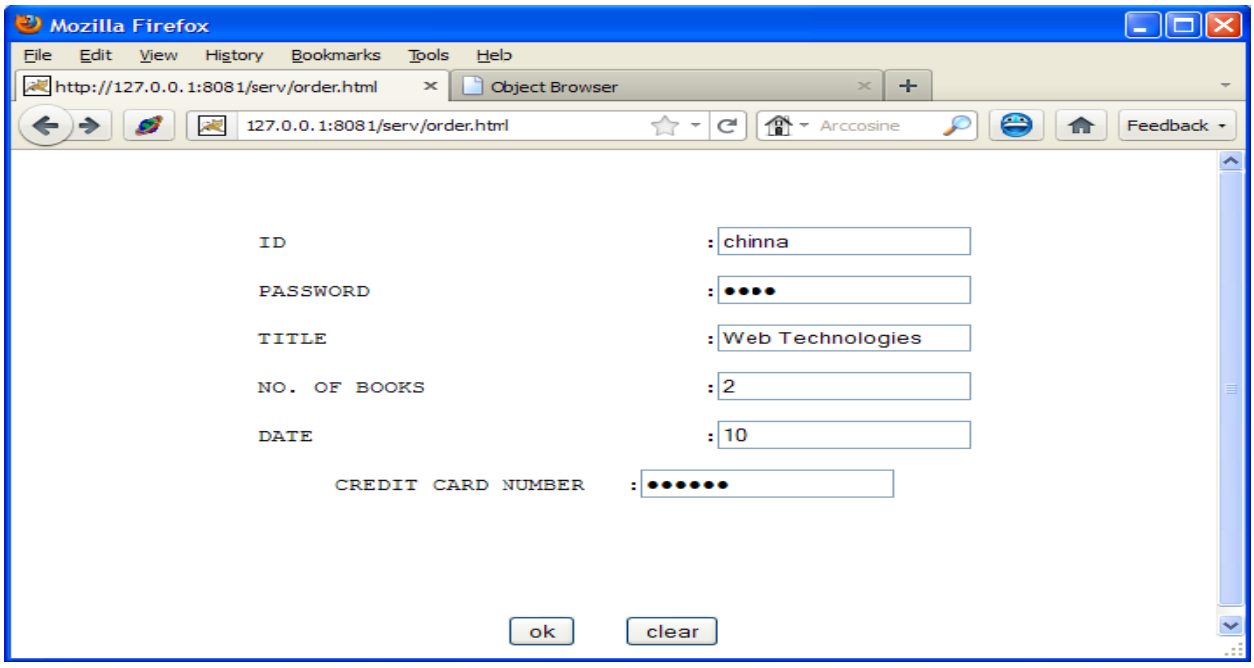
Catalog page:



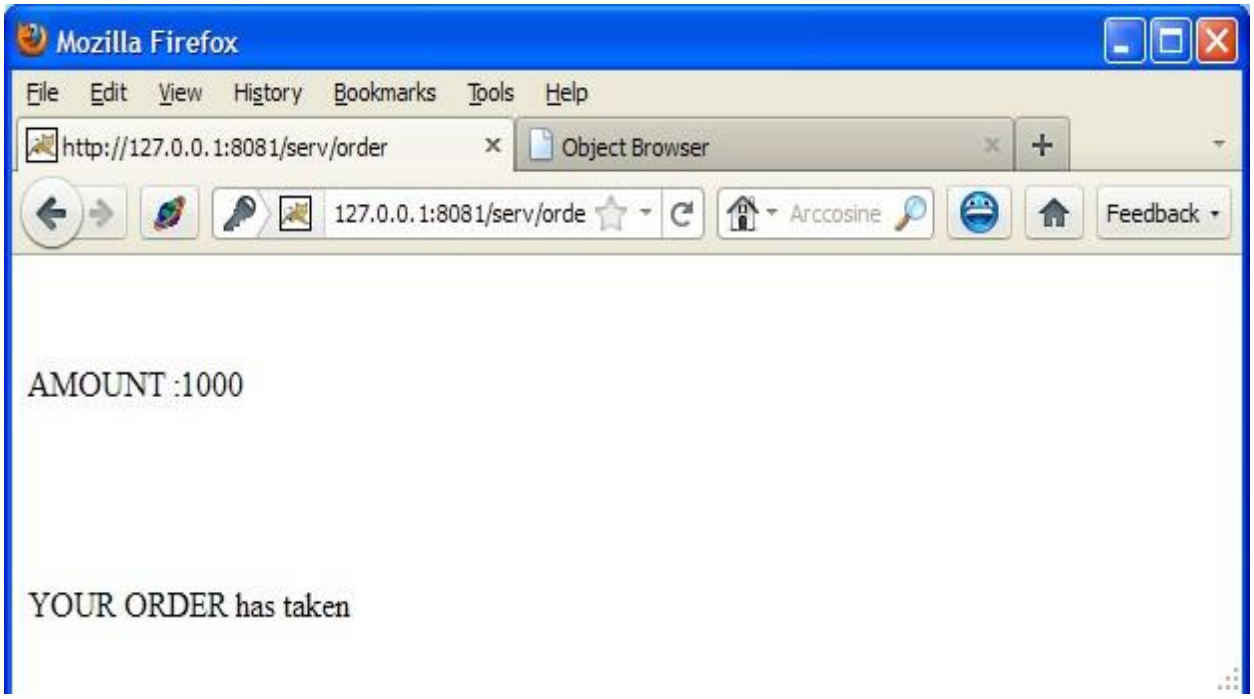
Catalog Servlet page:



Order page:



Order Servlet page:



WEEK-13:

Write a JSP which does the following job:

Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database.

Main.html:

```
<html>
<body>
<br /><br /><br /><br />
<h1 align="center"><U>ONLINE BOOK STORAGE</U></h1><br /><br /><br />
<h2 align="center"><pre>
<b>Welcome to online book
storage.Press LOGIN if you are
having id otherwise press
REGISTRATION
</b></pre></h2>
<br /><br /><pre>
<div align="center"><a href="login.html">LOGIN</a> <a
href="reg.html">REGISTRATION</a></div></pre>
</body>
</html>
```

Login.html:

```
<html>
<body><br /><br /><br />
<form name="myform" method="post" action="login.jsp">
<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()" />
&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;<input type="reset" value="clear" />
</div>
</form>
</body>
```

```
</html>
```

Reg.html:

```
<html>
<body><br /><br />
<form name="myform" method="post" action="reg.jsp">
<table align="center" >

  <tr>
    <td>NAME</td>
    <td>:<input type="text" name="name" /></td>
  </tr>
  <tr>
    <td>ADDRESS</td>
    <td>:<input type="text" name="addr" /></td>
  </tr>
  <tr>
    <td>CONTACT NUMBER</td>
    <td>:<input type="text" name="phno" /></td>
  </tr>
  <tr>
    <td>LOGINID</td>
    <td>:<input type="text" name="id" /></td>
  </tr>
  <tr>
    <td>PASSWORD</td>
    <td>:<input type="password" name="pwd" /></td>
  </tr>
</table>
<br /><br />
<div align="center">
<input type="submit" value="ok" onclick="validate()" />
      <input type="reset" value="clear" />
</div>
</form>
</body>
</html>
```

Reg.jsp:

```
<%@page import="java.sql.*"%>
  <%@page import="java.io.*"%>
   <%
                    response.setContentType("tex
               t/html");
               out.println("<html><body>");
               String
               name=request.getParameter("name");
               String
               addr=request.getParameter("addr");
               String
               phno=request.getParameter("phno");
               String id1=request.getParameter("id");
               String
               pwd1=request.getParameter("pwd");
               int no=Integer.parseInt(phno);
               Class.forName("oracle.jdbc.driver.OracleDriver");
Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","system","tige
r");
               Statement
               stmt=con.createStatement(); String
               sqlstmt="select id,pwd from login";
               ResultSet
               rs=stmt.executeQuery(sqlstmt); int
               flag=0;
               while(rs.next())
                    { if(id1.equals(rs.getString(1))&&pwd1.equals(rs.getString(2)))
                         {       flag=1;
                         }
                    }
                    if(flag==1)
{ out.println("<br><br>SORRY INVALID ID ALREADY EXITS TRY AGAIN WITH NEW
ID<br><br>");
out.println("<a href=\"reg.html\">press REGISTER to RETRY</a>");
                    }
                    else
                    { Statement
stmt1=con.createStatement(); stmt1.executeUpdate("insert
into login
values('"+name+"','"+addr+"',"+phno+",'"+id1+"','"+pwd1+"');");
out.println("<br><br>YOUR DETAILS ARE ENTERED<br><br>");
```

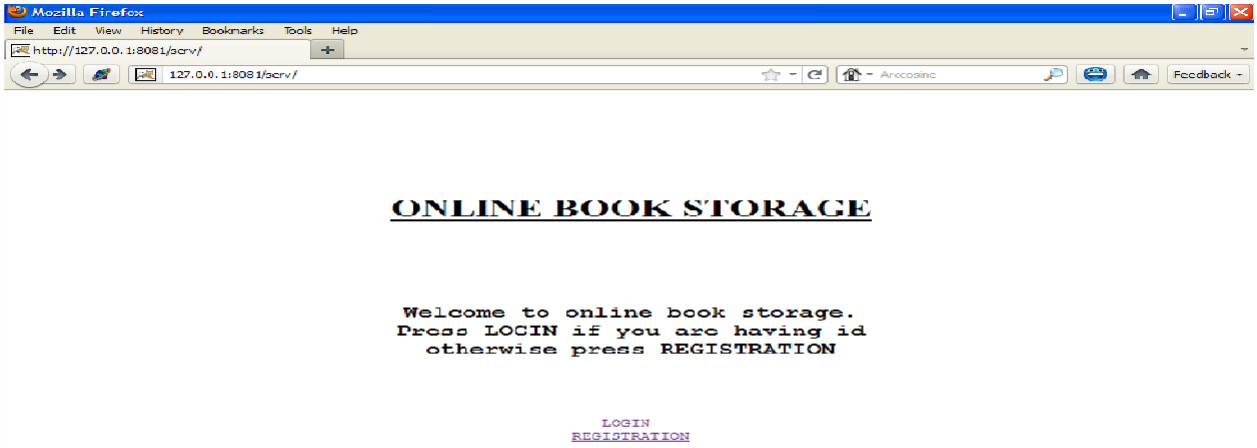
```
out.println("<a href=\"login.html\">press LOGIN to login</a>");
                        }
                        out.println("</body></html>"
                );con.close();
    %>
```

Login.jsp:

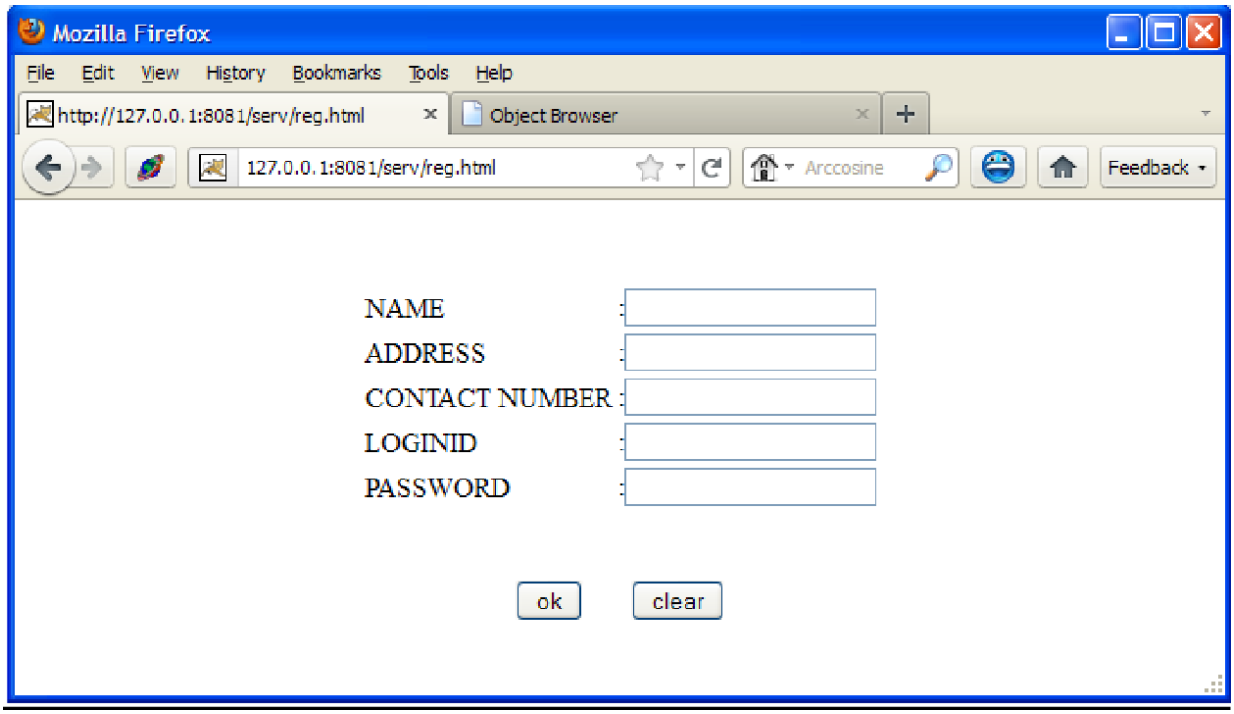
```
<%@page import="java.sql.*"%>
    <%@page import="java.io.*"%>
     <html>
          <body>
               <% String
               id=request.getParameter("id");String
               pwd=request.getParameter("pwd");
               Class.forName("oracle.jdbc.driver.OracleDriver");
Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","system","tige
r");
               Statement
               stmt=con.createStatement(); String
               sqlstmt="select id,pwd from login";
               ResultSet
               rs=stmt.executeQuery(sqlstmt); int
               flag=0;
              while(rs.next())
                    {
                         if(id.equals(rs.getString(1))&&pwd.equals(rs.getString(2)))
                         {
                              flag=1;
                         }
                    }
                    if(flag==0)
                    {
                    out.println("<br><br>SORRY INVALID ID TRY AGAIN ID<br><br>");
                    out.println("<a href=\"login.html\">press LOGIN to RETRY</a>");
                    }
                    else
                    {       out.println("<br><br>VALID LOGIN ID<br><br>");
                            out.println("WELCOME <br>To<br>”+id);
                    }
                    con.close();
                     %>
                    </body>
                    </html>
```

OUTPUT:

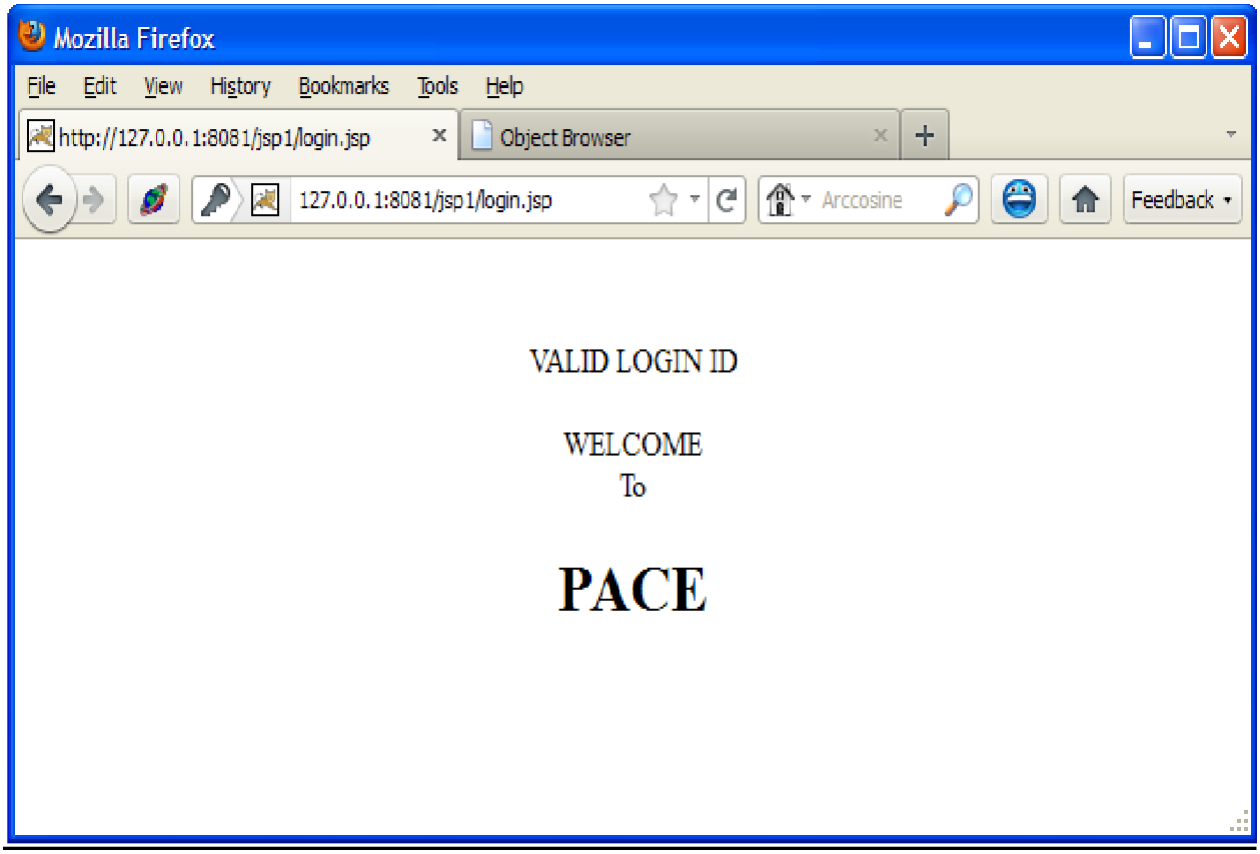
Main.html:



Registration page:



Login page:


Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://127.0.0.1:8081/jsp1/login.jsp
Object Browser
127.0.0.1:8081/jsp1/login.jsp
Arccosine
Feedback
VALID LOGIN ID
WELCOME
To
PACE