



CMR College of Engineering & Technology

Kandlakoya (V), Medchal Road, Hyderabad - 501 401. Andhra Pradesh. INDIA

Phone No: 08418 - 200699. Fax No: 08418 - 200240.

E-Mail: principal@cmrcet.org, www.cmrcet.org



LAB MANUAL

Dept. of Computer Science & Engineering

Subject WEB TECHNOLOGIES AND COMPILER DESIGN LAB

Academic Year 2014-2015

Name of the Faculty K.YELLASWAMY

Department CSE

Branch & Year III-CSE-A&C (2014-2015)

COURSE OBJECTIVES:

- To create a fully functional web site with mvc architecture.
- To Develop an online Book store using we can sell books(Ex.amazon.com)
- To provide an understanding of the language translation peculiarities by designing a complete translator for a mini language.

4.1 SYLLABUS

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY

HYDERABAD

III Year B.Tech. CSE -II Sem

T	P	C
	0	3
		2

WEB TECHNOLOGIES AND COMPILER DESIGN LAB(56610)

Objective :

To create a fully functional website with mvc architecture. To Develop an online Book store using we can sell books (Ex amazon .com).

Hardware and Software required :

- A working computer system with either Windows or Linux
- A web browser either IE or firefox
- Tomcat web server and Apache web server
- A database either Mysql or Oracle
- JVM(Java virtual machine) must be installed on your system
- BDK(Bean development kit) must be also be installed

Week-1:

Design the following static web pages required for an online book store web site.

1) HOME PAGE:

The static home page must contain three frames.

Top frame : Logo and the college name and links to Home page, Login page, Registration page, Catalogue page and Cart page (the description of these pages will be given below).

Left frame : At least four links for navigation, which will display the catalogue of respective links.

For e.g.: When you click the link “CSE” the catalogue for CSE Books should be displayed in the Right frame.

Right frame: The *pages to the links in the left frame must be loaded here*. Initially this page contains

description of the web site

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Description of the Web Site			

Fig 1.1

2) LOGIN PAGE:

This page looks like below:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE	<p>Login : <input type="text"/></p> <p>Password: <input type="text"/></p> <p><input type="submit" value="Submit"/> <input type="submit" value="Reset"/></p>			
ECE				
EEE				
CIVIL				

3) CATALOGUE PAGE:

The catalogue page should contain the details of all the books available in the web site in a table.

The details should contain the following:

- Snap shot of Cover Page.
- Author Name.
- Publisher.
- Price.
- Add to cart button

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart

CSE		Book : XML Bible Author : Winston Publication : Wiely	\$ 40.5	
ECE		Book : AI Author : S.Russel Publication : Princeton hall	\$ 63	
EEE		Book : Java 2 Author : Watson Publication : BPB publications	\$ 35.5	
CIVIL		Book : HTML in 24 hours Author : Sam Peter Publication : Sam publication	\$ 50	

Note: Week 2 contains the remaining pages and their description.

Week-2:

4) CART PAGE:

The cart page contains the details about the books which are added to the cart.

The cart page should look like this:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE	Book name	Price	Quantity	Amount
ECE				
EEE	Java 2	\$35.5	2	\$70
CIVIL	XML bible	\$40.5	1	\$40.5
			Total amount -	\$130.5

5) REGISTRATION PAGE:

Create a "registration form" with the following fields

- 1) Name (Text field)
- 2) Password (password field)
- 3) E-mail id (text field)
- 4) Phone number (text field)
- 5) Sex (radio button)
- 6) Date of birth (3 select boxes)
- 7) Languages known (check boxes – English, Telugu, Hindi, Tamil)
- 8) Address (text area)

WEEK 3:

VALIDATION:

Write *JavaScript* to validate the following fields of the above registration page.

- Name (Name should contains alphabets and the length should not be less than 6 characters).
 - Password (Password should not be less than 6 characters length).
 - E-mail id (should not contain any invalid and must follow the standard pattern [name@domain.com](#))
4. Phone number (Phone number should contain 10 digits only).

Note : You can also validate the login page with these parameters

Week-4:

Design a web page using CSS (Cascading Style Sheets) which includes the following:

1) Use different font, styles:

In the style definition you define how each selector should work (font, color etc.).

Then, in the body of your pages, you refer to these selectors to activate the styles.

For example:

```
<HTML>  
<HEAD>  
<style type="text/css">
```

```
B.headline { color:red; font-size:22px; font-family:arial; text-decoration:underline }
</style>
```

```
</HEAD>
```

```
<BODY>
<b>This is normal bold</b><br>
Selector { cursor:value }
```

For example:

```
<html>
<head>
<style type="text/css">
.xlink { cursor:crosshair}
.hlink { cursor:help }
</style>
</head>

<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>

<b class="headline">This is headline style bold</b>
</BODY>

</HTML>
```

- 2) Set a background image for both the page and single elements on the page.
You can define the background image for the page like this:

```
BODY { background-image:url(myimage.gif); }
```




3) Control the repetition of the image with the background-repeat property.

As background-repeat: repeat

Tiles the image until the entire page is filled, just like an ordinary background image in plain HTML.

4) Define styles for links as

A:link
A:visited
A:active
A:hover

Example:

```
<style type="text/css">  
A:link {text-decoration: none}  
A:visited {text-decoration: none}  
A:active {text-decoration: none}  
A:hover {text-decoration: underline; color: red;}  
</style>
```

5) Work with layers:

For example:

LAYER 1 ON TOP:

```
<div style="position:relative; font-size:50px; z-index:2;">LAYER 1</div>
```

```
<div style="position:relative; top:-50; left:5; color:red; font-size:80px;
```

```
z-index:1">LAYER 2</div>
```

LAYER 2 ON TOP:

```
<div style="position:relative; font-size:50px; z-index:3;">LAYER 1</div>
```

```
<div style="position:relative; top:-50; left:5; color:red; font-size:80px;
```

```
z-index:4">LAYER 2</div>
```

6) Add a customized cursor:

Selector {cursor:value}

For example:

```
<html>
<head>
<style type="text/css">
.xlink { cursor:crosshair}
.hlink { cursor:help}
</style>
</head>

<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>
```

Week-5:

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 in bold. Use your own colors for remaining columns.

Use XML schemas XSL and CSS for the above purpose.

Note: Give at least for 4 books. It should be valid syntactically.

Hint: You can use some xml editors like XML-spy

Week-6:

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. The color should also be changed if we change the color in the “property window “.

Week-7:

- 1) Install TOMCAT web server and APACHE.

While installation assign port number 4040 to TOMCAT and 8080 to APACHE.

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:4040/rama/books.html>(for tomcat)

<http://localhost:8080/books.html> (for Apache)

Week-8:

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.

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 “.

Use init-parameters to do this. Store the user-names and passwords in the webinf.xml and access them in the servlet by using the getInitParameters() method.

Week-9:

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).

Week-10:

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 (similar to week8 instead of cookies).

Week-11:

Create tables in the database which contain the details of items (books in our case like Book name , Price, Quantity, Amount)) of each category. Modify your catalogue page (week 2)in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using JDBC.

Week-12:

HTTP is a stateless protocol. Session is required to maintain the state.

The user may add some items to cart from the catalog page. He can check the cart page for the

selected items. He may visit the catalogue again and select some more items. Here our interest is the selected items should be added to the old cart rather than a new cart.

Multiple users can do the same thing at a time(i.e., from different systems in the LAN using the ip-address instead of localhost). This can be achieved through the use of sessions. Every user will have his own session which will be created after his successful login to the website. When the user logs out his session should get invalidated (by using the method `session.invalidate()`).

Modify your catalogue and cart JSP pages to achieve the above mentioned functionality using sessions.

Compiler Design

1. Design a Lexical analyzer for the mini language. The lexical analyzer should ignore redundant spaces, tabs and newlines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value.
2. Implement the lexical analyzer using JLex, flex or lex or other lexical analyzer generating tools.
3. Design Predictive parser for the given language
4. Design LALR bottom up parser for the mini language.
5. Convert the BNF rules into Yacc form and write code to generate abstract syntax tree.
6. Write program to generate machine code from the abstract syntax tree generated by the parser.

Web Technologies
LAB PROGRAMS

WEEK1:

AIM:

Design the following static web pages requires for an online store website.

1. Homepage:

Frame.html:

```
<!doctype html>
<html lang="en">
<head>
<title>week1</title>
</head>
<frameset rows="10%,10%,*" cols="10%,*">
<frame src="logo.html" name="logo">
<frame src="site.html" name="site">
<frame src="home.html" name="home">
<frameset cols="40%,40%,40%,40%">
<frame src="login.html" name="login">

<frame src="registration.html" name="registration">
```

```
<frame src="catalogue.html" name="catalogue">
```

```
<frame src="cart.html" name="cart">
```

```
</frameset>
```

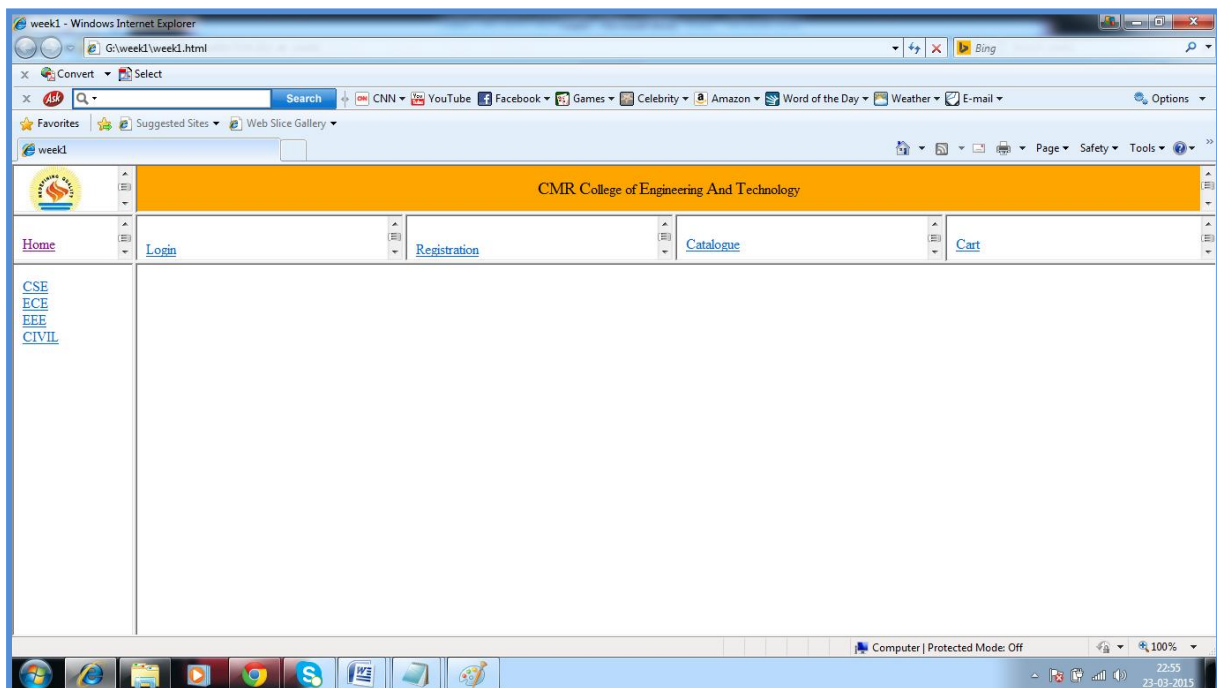
```
<frame src="list.html" name="">
```

```
<frame src="" name="x">
```

```
</frameset>
```

```
</html>
```

Output:



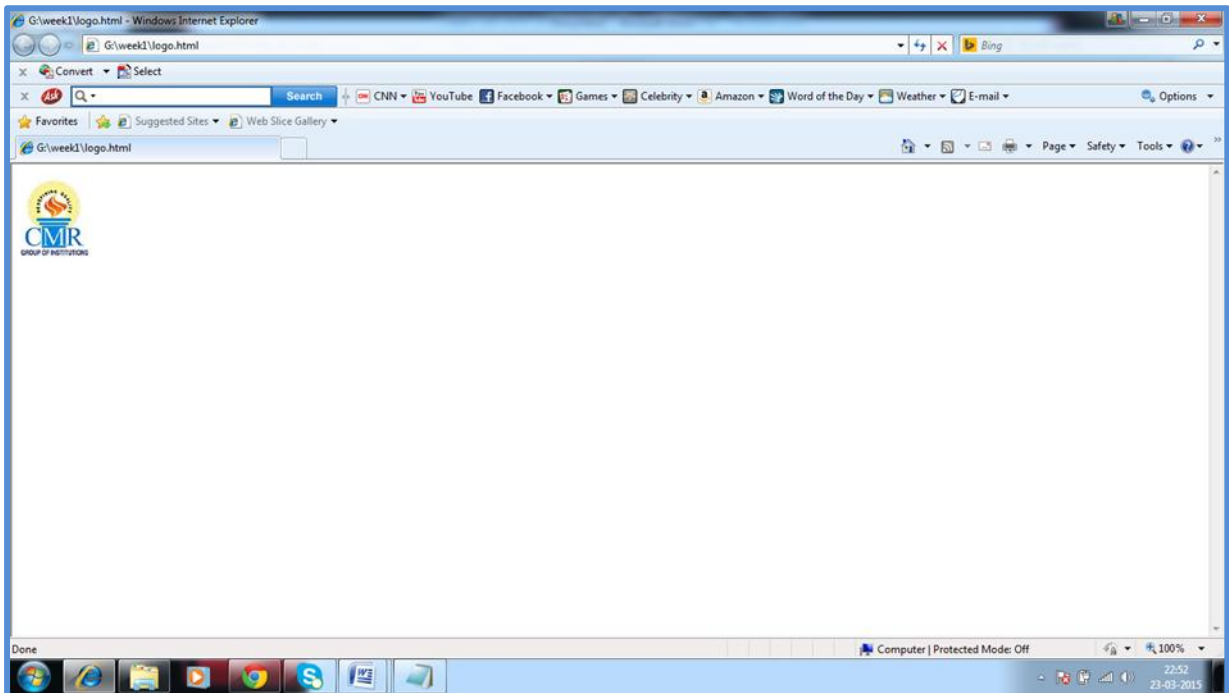
Logo.html:

```
<html>
```


<body>

</body>

</html>



Site.html:

<html>

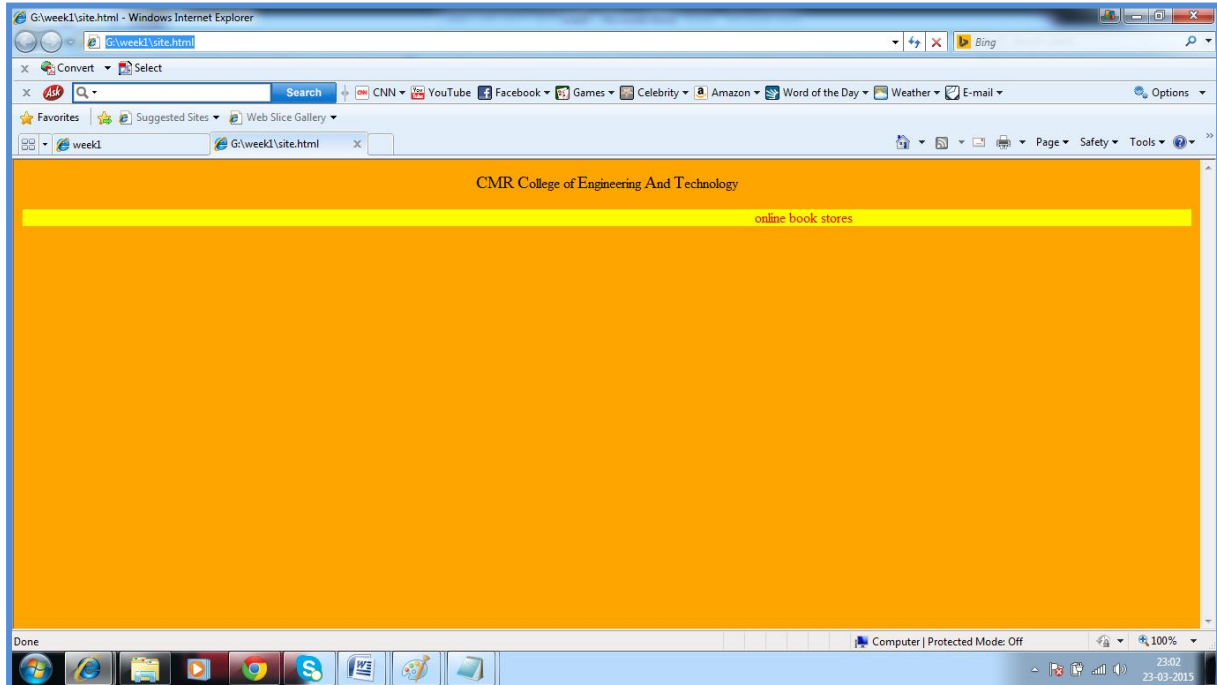
<style>

p

```
{
color:red;
}
body
{
background-color:orange;
}
</style>
<body>
<center>
<span style="font-size:20px;">CMR</span>
<span style="font-size:20px;">C</span>ollege of
<span style="font-size:20px;">E</span>ngineering
<span style="font-size:20px;">A</span>nd
<span style="font-size:20px;">T</span>echnology
</center>
<p><marquee behavior="alternate"
bgcolor="yellow">online book stores</marquee></p>
</body>
```

</html>

Output:



Homepage.html:

<head>

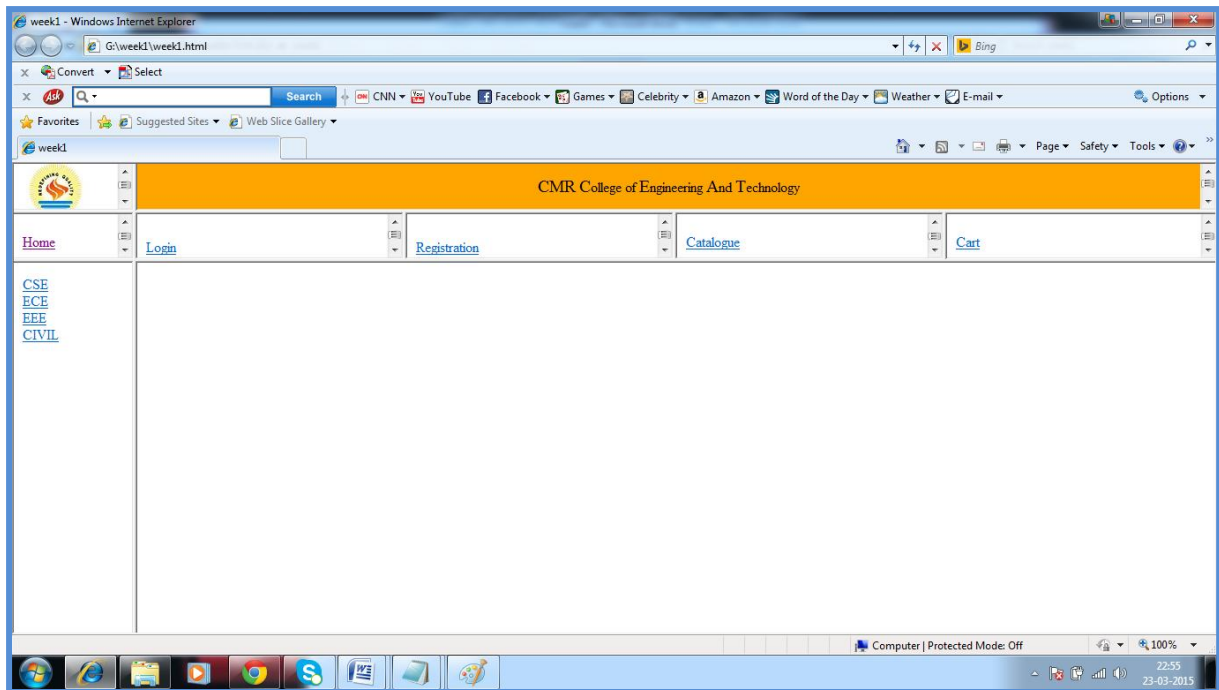
<body>

Home

</body>

</head>

Output:



2. Login Page:

Login.html:

```
<html>
```

```
<body>
```

```
<br>
```

```
<a href="login1.html" target="x">Login</a>
```

```
</body>
```

```
</html>
```

Login1.html:

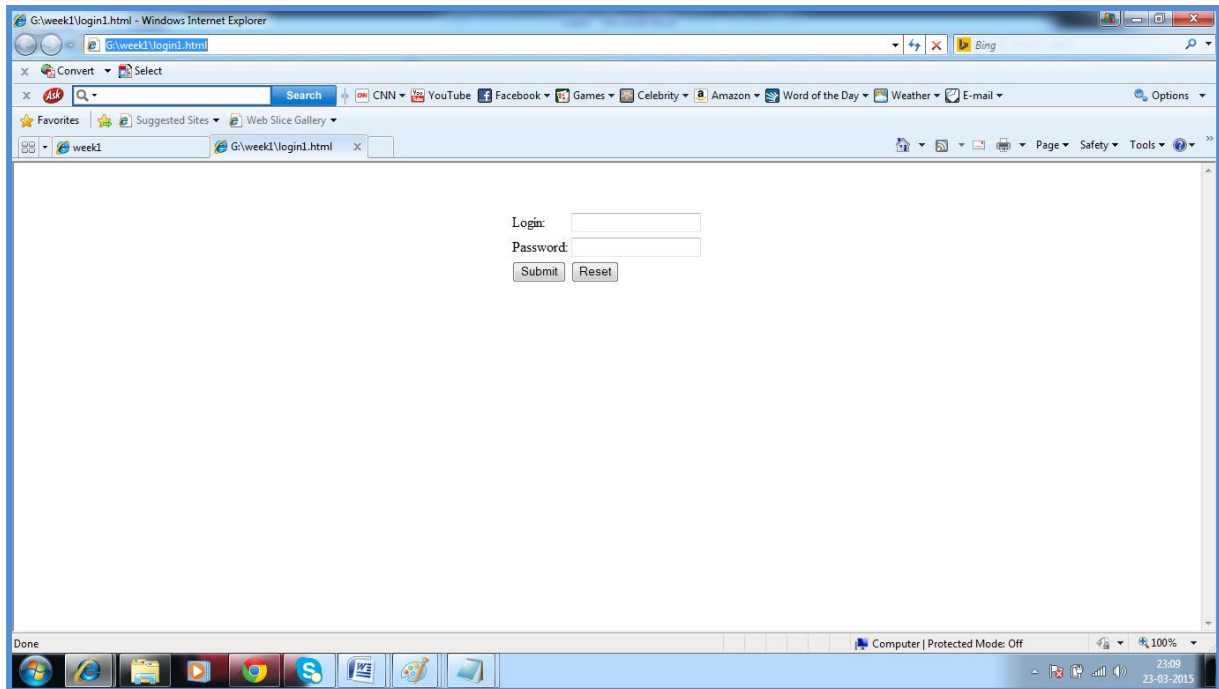
```
<html>
<body>
<center>
<br>
<br>
<table border="0">
<tr>
<td>Login:</td>
<td><input type="text" name="in"/></td>
</tr>
<tr>
<td>Password:</td>
<td><input type="text" name="pwd"/></td>
</tr>
<tr>
<td><button>Submit</button></td>
<td><button>Reset</button></td>
</tr>
</table>
```

</center>

</body>

</html>

Output:



3. CATALOGUE PAGE:

Catalogue.html:

<html>

<body>

Catalogue

</body>

```
</html>
```

Catalogue1.html:

```
<html>
```

```
<body>
```

```
<table border="1" rules="cols" width="900"  
height="500" cellspacing="0" cellpadding="0">
```

```
<tr>
```

```
<td></td>
```

```
<td>
```

Book: XML Bible

```
<br>
```

Author:Winston

```
<br>
```

Publication:Wiely

```
</td>
```

```
<td>
```

\$ 40.5

```
</td>
```

<td>

<button>

Add to cart

</button>

</td>

</tr>

<tr>

<td></td>

<td>

Book: AI

Author:S.Russel

Publication:

Princeton hall

</td>

<td>

\$ 63

</td>

<td>

<button>

Add to cart

</button>

</td>

</tr>

<tr>

<td></td>

<td>Book: Java 2

Author: Watson

Publication: BPB

publications

</td>

<td>

\$ 35.5

</td>

<td>

<button>

Add to cart

</button>

</td>

</tr>

<tr>

<td></td>

<td>

Book: HTML in

24 hours

Author:Sam Peter

Publication:

Sam publication

</td>

<td>

\$ 50

</td>

<td>

<button>

Add to cart

</button>

</td>

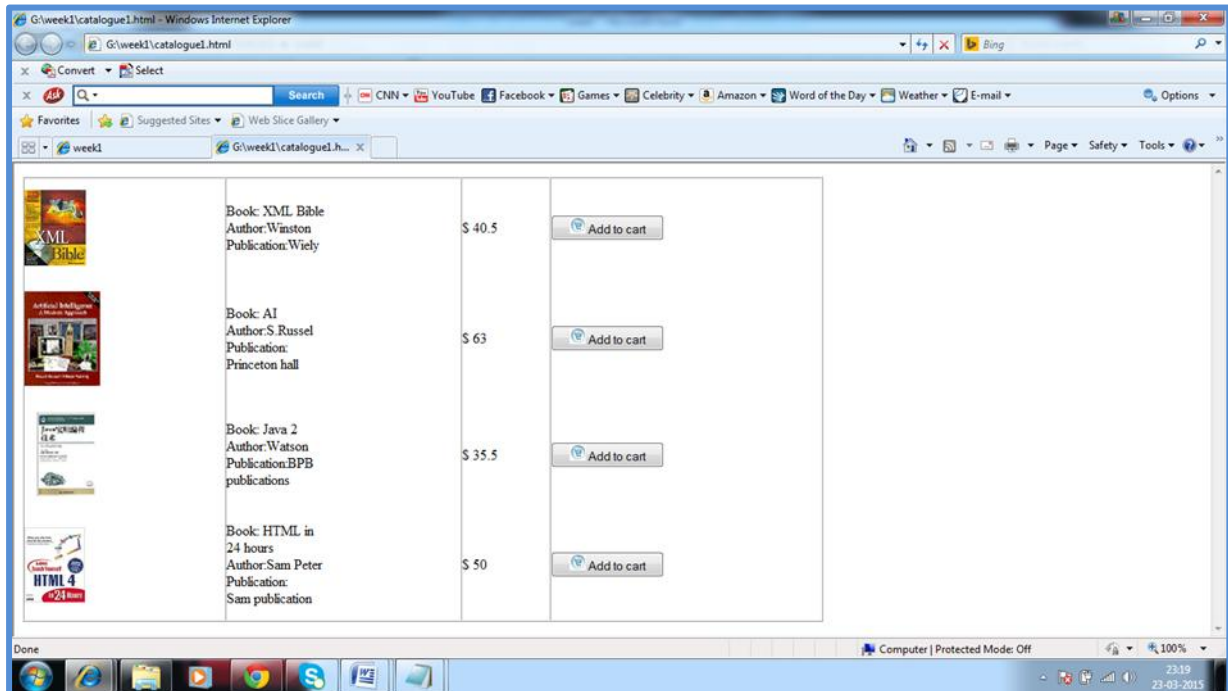
</tr>

</table>

</body>

</html>

Output:



List.html:

```
<html>
```

```
<body>
```

```
<a href="cse.html" target="x">CSE</a>
```

```
<br>
```

```
<a href="ece.html" target="x">ECE</a>
```

```
<br>
```

```
<a href="eee.html" target="x">EEE</a>
```


CIVIL

</body>

</html>

Cse.html:

<html>

<body>

This is the catalogue of dept. of CSE

</body>

</html>

Ece.html:

<html>

<body>

This is the catalogue of dept. of ECE

</body>

</html>

Eee.html:

```
<html>
```

```
<body>
```

This is the catalogue of dept. of EEE

```
</body>
```

```
</html>
```

Civil.html:

```
<html>
```

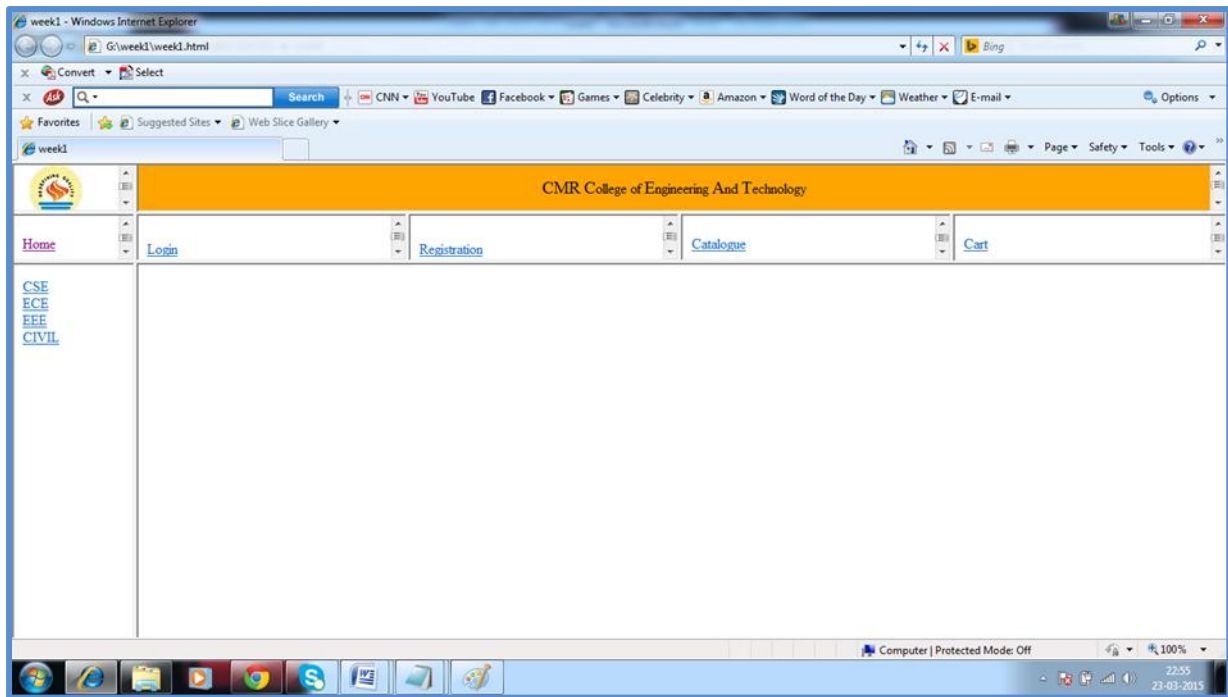
```
<body>
```

This is the catalogue of dept. of CIVIL

```
</body>
```

```
</html>
```

Output:



WEEK2:

4. Cart Page:

Cartlink.html:

```
<!doctype html>
```

```
<html>
```

```
<head>
```

```
<title>login</title>
```

```
</head>
```

```
<body>
```

```
<a href="cart.html"
target="x"><center><h1>Cart</center>
</h1></a>
</body>
</html>
```

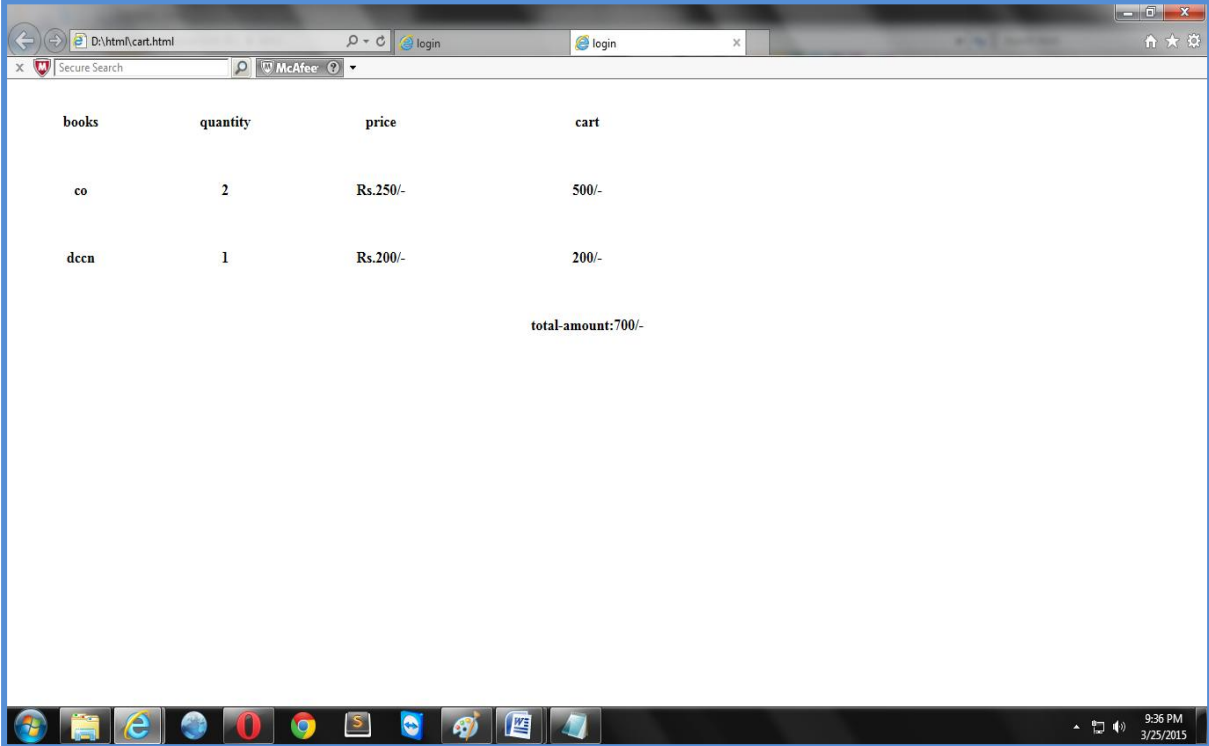
Cart.html:

```
<!doctype html>
<html>
<head>
</head>
<body>
<table width="800" height="50" cellpadding="25" >
<tr>
<th>books</th>
<th>quantity</th>
<th>price</th>
<th>cart</th>
</tr>
<tr>
```



```
<th>co</th>
<th>2</th>
<th>Rs.250/-</th>
<th>500/-</th>
</tr>
</tr>
<tr>
<th>dccn</th>
<th>1</th>
<th>Rs.200/-</th>
<th>200/-</th>
</tr>
<tr>
<th></th>
<td></td>
<th></th>
<th>total-amount:700/-</th>
</tr></table>
</body></html>
```

Output:



The screenshot shows a web browser window with the address bar displaying "D:\html\cart.html". The page content is as follows:

books	quantity	price	cart
co	2	Rs.250/-	500/-
dccn	1	Rs.200/-	200/-

total-amount:700/-

5. Registration page:

Registrationlink.html:

```
<!doctype html>
```

```
<html>
```

```
<head>
```

```
<title>Registration hyperlink</title>
```

```
</head>
```

```
<body>
```

```
<a href="Registration.html"
target="x"><h1><center>registration</center></h1></a
>
</body>
</html>
```

Registration.html:

```
<!doctype html>
<html>
<head>
<title>regtion</title>
</head>
<body>
<center><h1>REGISTRATION FORM</h1></center>
<h3>
<center>
<table>
<tr>
<td>name</td>
<td><input type="text" name="" value=""></td>
```

```
</tr>
```

```
<tr>
```

```
<td>password</td>
```

```
<td><input type="password" name="" value=""></td>
```

```
</tr>
```

```
<tr>
```

```
<td>emailid</td>
```

```
<td><input type="text" name="" value=""></td>
```

```
</tr>
```

```
<tr>
```

```
<td>phno</td>
```

```
<td><input type="text" name="" value=""></td>
```

```
</tr>
```

```
<tr>
```

```
<td>gendor</td>
```

```
<td><input type="radio" name="m"  
value="">male</td>
```

```
<td><input type="radio" name="m"  
value="">female</td>
```

```
</tr>
```

```
</table>
```

```
<br>
```

```
<table>
```

date of birth

```
<tr>
```

```
<select name="month">
```

```
<option value="na">month</option>
```

```
<option value="1">jan</option>
```

```
<option value="2">feb</option>
```

```
<option value="3">mar</option>
```

```
<option value="4">apr</option>
```

```
<option value="5">may</option>
```

```
<option value="6">june</option>
```

```
<option value="7">july</option>
```

```
<option value="8">aug</option>
```

```
<option value="9">sep</option>
```

```
<option value="10">oct</option>
```

```
<option value="11">nov</option>
```

```
<option value="12">dec</option>
```

```
</select>
```

```
<select name="month">
```

```
<option value="na">date</option>
```

```
<option value="1">1</option>
```

```
<option value="2">2</option>
```

```
<option value="3">3</option>
```

```
<option value="4">4</option>
```

```
<option value="5">5</option>
```

```
<option value="6">6</option>
```

```
<option value="7">7</option>
```

```
<option value="8">8</option>
```

```
<option value="9">9</option>
```

```
<option value="10">10</option>
```

```
<option value="11">11</option>
```

```
<option value="12">12</option>
```

```
<option value="13">13</option>
```

```
<option value="14">14</option>
```

```
<option value="15">15</option>
```

```
<option value="16">16</option>
```

```
<option value="17">17</option>
```

```
<option value="18">18</option>
```

```
<option value="19">19</option>
```

```
<option value="20">20</option>
```

```
<option value="21">21</option>
```

```
<option value="22">22</option>
```

```
<option value="23">23</option>
```

```
<option value="24">24</option>
```

```
<option value="25">25</option>
```

```
<option value="26">26</option>
```

```
<option value="27">27</option>
```

```
<option value="28">28</option>
```

```
<option value="29">29</option>
```

```
<option value="30">30</option>
```

```
<option value="31">31</option>
```

```
</select>
```

```
<select name="year">
```

```
<option value="na">year</option>
```

```
<option value="1">1990</option>
```

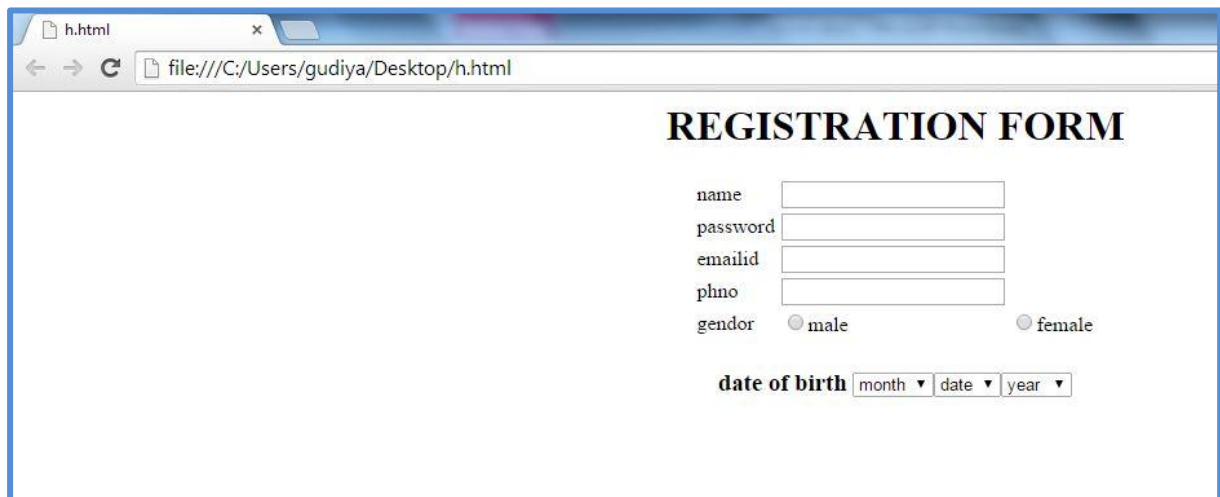
```
</select>
```

```
</table>
```

```
</body>
```

```
</html>
```

Output:



The screenshot shows a web browser window with a single tab titled 'h.html'. The address bar shows the file path 'file:///C:/Users/gudiya/Desktop/h.html'. The main content area displays a registration form with the following elements:

- REGISTRATION FORM** (Section Header)
- name:
- password:
- emailid:
- phno:
- gender: male female
- date of birth: month date year

WEEK3:

AIM:

Write a JavaScript program to validate the fields in the registration page.

FormValidation.html:

```
<!DOCTYPE html>
```

```
<html lang="en"><head>
```



```
<meta charset="utf-8">
<title>JavaScript Form Validation using a sample
registration form</title>
<meta name="keywords" content="example, JavaScript
Form Validation, Sample registration form" />
<meta name="description" content="This document is
an example of JavaScript Form Validation using a
sample registration form. " />
<link rel='stylesheet' href='js-form-validation.css'
type='text/css' />
<script src="sample-registration-form-
validation.js"></script>
</head>
<body
onLoad="document.registration.userid.focus();">
<h1>Registration Form</h1>
<p>Use tab keys to move from one input field to the
next.</p>
<form name='registration' onSubmit="return
formValidation();">
<ul>
```

```
<li><label for="userid">User id:</label></li>
```

```
<li><input type="text" name="userid" size="12" /></li>
```

```
<li><label for="passid">Password:</label></li>
```

```
<li><input type="password" name="passid" size="12" /></li>
```

```
<li><label for="username">Name:</label></li>
```

```
<li><input type="text" name="username" size="50" /></li>
```

```
<li><label for="address">Address:</label></li>
```

```
<li><input type="text" name="address" size="50" /></li>
```

```
<li><label for="country">Country:</label></li>
```

```
<li><select name="country">
```

```
<option selected="" value="Default">(Please select a country)</option>
```

```
<option value="AF">Australia</option>
```

```
<option value="AL">Canada</option>
```

```
<option value="DZ">India</option>
```

```
<option value="AS">Russia</option>
```

```
<option value="AD">USA</option>
</select></li>
<li><label for="zip">ZIP Code:</label></li>
<li><input type="text" name="zip" /></li>
<li><label for="email">Email:</label></li>
<li><input type="text" name="email" size="50" /></li>
<li><label id="gender">Sex:</label></li>
<li><input type="radio" name="sex" value="Male"
/><span>Male</span></li>
<li><input type="radio" name="sex" value="Female"
/><span>Female</span></li>
<li><label>Language:</label></li>
<li><input type="checkbox" name="en" value="en"
checked /><span>English</span></li>
<li><input type="checkbox" name="nonen"
value="noen" /><span>Non English</span></li>
<li><label for="desc">About:</label></li>
<li><textarea name="desc" id="desc"></textarea></li>
<li><input type="submit" name="submit"
value="Submit" /></li>
```

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

FormValidation.js:

```
function formValidation()
{
var uid = document.registration.userid;
var passid = document.registration.passid;
var uname = document.registration.username;
var uadd = document.registration.address;
var ucountry = document.registration.country;
var uzip = document.registration.zip;
var uemail = document.registration.email;
var umsex = document.registration.msex;
var ufsex = document.registration.fsex;
if(userid_validation(uid,5,12))
{
if(passid_validation(passid,7,12))
```

```
{
  if(allLetter(uname))
  {
    if(alphanumeric(uadd))
    {
      if(countryselect(ucountry))
      {
        if(allnumeric(uzip))
        {
          if(ValidateEmail(uemail))
          {
            if(validsex(umsex,ufsex))
            {
              }
            }
          }
        }
      }
    }
  }
}
```

```
}  
  
}  
  
return false;  
  
} function userid_validation(uid,mx,my)  
{  
var uid_len = uid.value.length;  
if (uid_len == 0 || uid_len >= my || uid_len < mx)  
{  
alert("User Id should not be empty / length be between  
"+mx+" to "+my);  
uid.focus();  
return false;  
}  
return true;  
}  
  
function passid_validation(passid,mx,my)  
{  
var passid_len = passid.value.length;
```

```
if (passid_len == 0 || passid_len >= my || passid_len <
mx)
{
alert("Password should not be empty / length be
between "+mx+" to "+my);
passid.focus();
return false;
}
return true;
}
function allLetter(uname)
{
var letters = /^[A-Za-z]+$/;
if(uname.value.match(letters))
{
return true;
}
else
{
alert('Username must have alphabet characters only');
```

```
uname.focus();  
return false;  
}  
}  
function alphanumeric(uadd)  
{  
var letters = /^[0-9a-zA-Z]+$/;  
if(uadd.value.match(letters))  
{  
return true;  
}  
else  
{  
alert('User address must have alphanumeric characters  
only');  
uadd.focus();  
return false;  
}  
}
```



```
function countrysselect(ucountry)
{
if(ucountry.value == "Default")
{
alert('Select your country from the list');
ucountry.focus();
return false;
}
else
{
return true;
}
}

function allnumeric(uzip)
{
var numbers = /^[0-9]+$/;
if(uzip.value.match(numbers))
{
return true;
}
```

```
}  
else  
{  
alert('ZIP code must have numeric characters only');  
uzip.focus();  
return false;  
}  
}  
function ValidateEmail(uemail)  
{  
var mailformat = /^\w+([\.-]?\w+)*@\w+([\.-]  
]? \w+)*(\.\w{2,3})+$/;  
if(uemail.value.match(mailformat))  
{  
return true;  
}  
else  
{  
alert("You have entered an invalid email address!");  
}
```

```
return false;
}
} function validsex(umsex,ufsex)
{
x=0;

if(umsex.checked)
{
x++;
}
if(ufsex.checked)
{
x++;
}

if (x==2)
{
alert('Both Male/Female are checked');
ufsex.checked=false
```

```
umsex.checked=false
```

```
umsex.focus();
```

```
return false;
```

```
}
```

```
if(x==0)
```

```
{
```

```
alert('Select Male/Female');
```

```
umsex.focus();
```

```
return false;
```

```
}
```

```
else
```

```
{
```

```
alert('Form Successfully Submitted');
```

```
window.location.reload()
```

```
return true;
```

```
}
```

```
}
```

FormValidation.css:

```
h1 {
```

```
margin-left: 70px;
```

```
}
```

```
form li {
```

```
list-style: none;
```

```
margin-bottom: 5px;
```

```
}
```

```
form ul li label{
```

```
float: left;
```

```
clear: left;
```

```
width: 100px;
```

```
text-align: right;
```

```
margin-right: 10px;
```

```
font-family: Verdana, Arial, Helvetica, sans-serif;
```

```
font-size: 14px;
```

```
}
```

```
form ul li input, select, span {  
float: left;  
margin-bottom: 10px;  
}
```

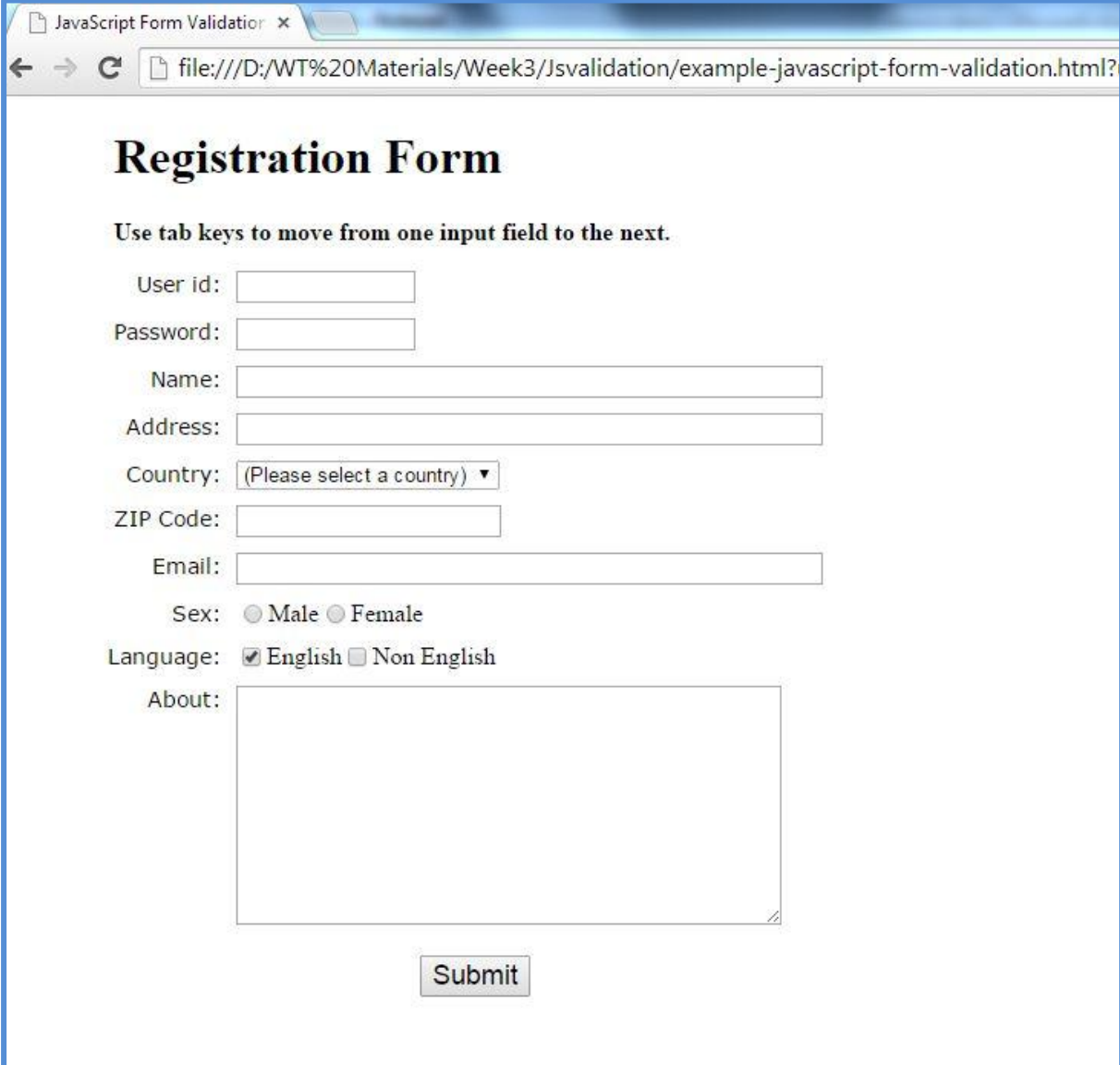
```
form textarea {  
float: left;  
width: 350px;  
height: 150px;  
}
```

```
[type="submit"] {  
clear: left;  
margin: 20px 0 0 230px;  
font-size: 18px  
}
```

```
p {  
margin-left: 70px;
```

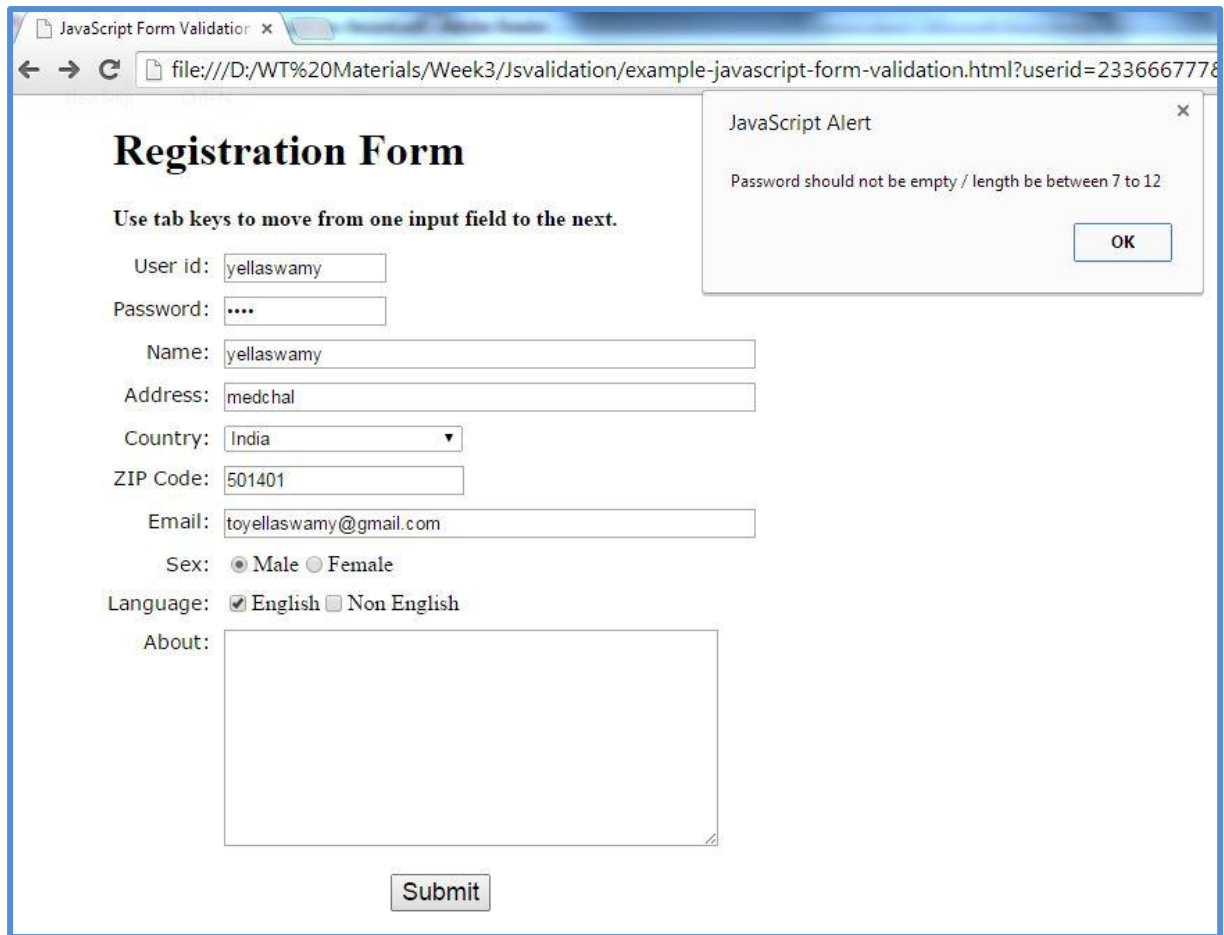
```
font-weight: bold;  
}
```

Output:



The screenshot shows a web browser window with the title "JavaScript Form Validation" and the address bar containing the file path: "file:///D:/WT%20Materials/Week3/Jsvalidation/example-javascript-form-validation.html?". The main content of the page is a registration form with the following fields and options:

- Registration Form**
- Use tab keys to move from one input field to the next.
- User id:
- Password:
- Name:
- Address:
- Country:
- ZIP Code:
- Email:
- Sex: Male Female
- Language: English Non English
- About:
-



WEEK4:

AIM:

Design a web page using CSS which includes the following.

i. Font Styles:

Fonts.html:

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

```
<html>
```



```
<head>
<title>Font properties</title>
<style type="text/css">
p.big
{
font-size:14pt;
font-style:italic;
font-family:'Century';
color:red;
}
p.small
{
font: 10pt bold 'Courier New';
color:violet;
}
h2
{
font-family:'Times New Roman';
font-size:14pt;
```

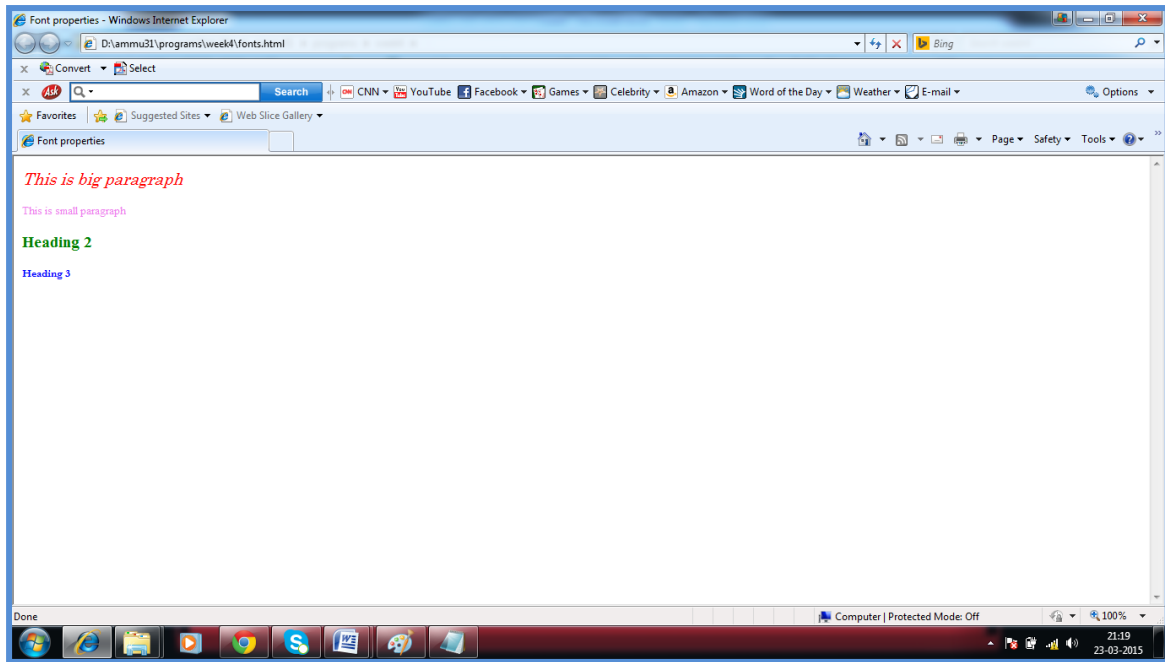
```
font-weight:bold;
color:green;
}
h3
{
font-family: 'Garamond' ;
font-size: 10pt;
color:blue;
}
</style>
</head>
<body>
<p class="big">
This is big paragraph
</p>
<p class="small">
This is small paragraph
</p>
<h2>Heading 2</h2>
```

<h3>Heading 3</h3>

</body>

</html>

Output:



ii. Background image:

Bckgrnd.html:

<html>

<head>

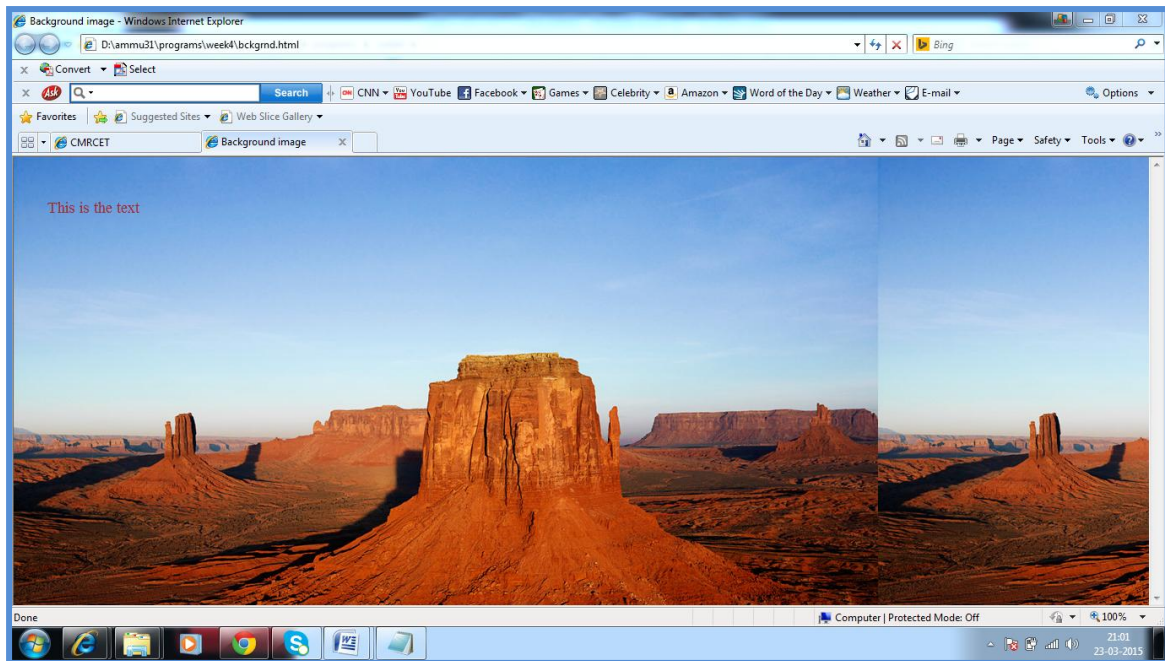
<title>Background image</title>

<style type="text/css">

body

```
{  
background-image:url(pics/img1.jpg);  
}  
  
p  
{  
margin-left:30px;  
margin-right:30px;  
margin-top:50px;  
font-size:14pt;  
color:brown;  
}  
  
</style>  
</head>  
<body>  
<p>This is the text</p>  
</body>  
</html>
```

Output:



iii. Background repeat:

repeat.html:

```
<html>
```

```
<title>Background image</title>
```

```
<head>
```

```
<style type="text/css">
```

```
body
```

```
{
```

```
background-image:url(pics/download.jpg);
```

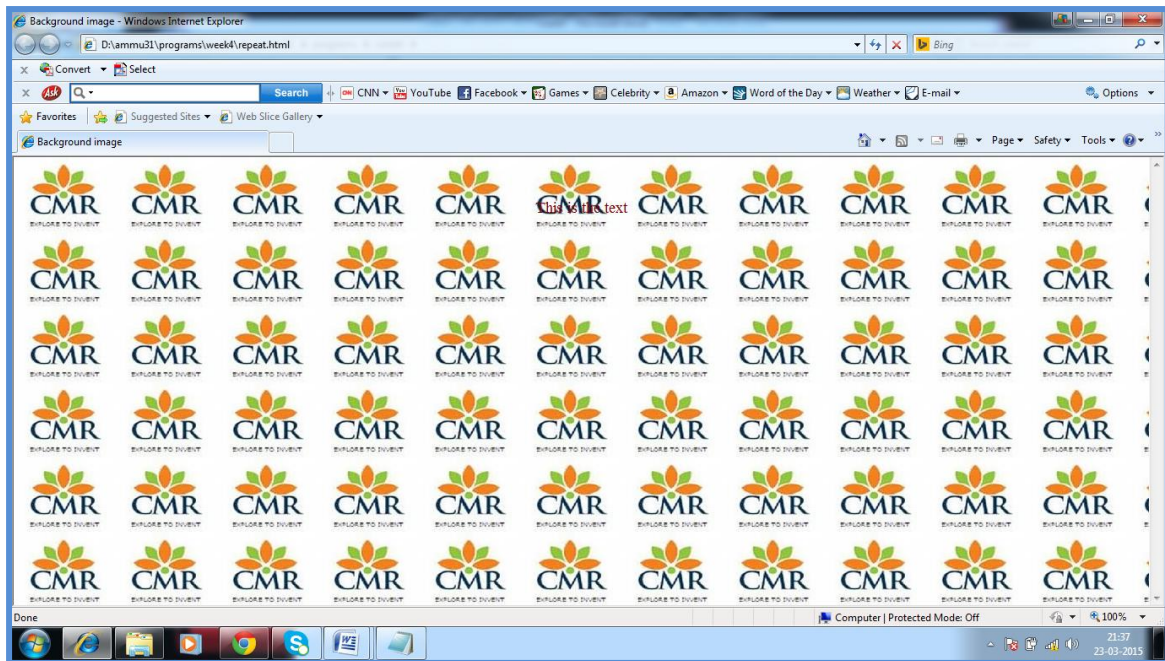
```
background-repeat:repeat x;
```

```
}
```

```
p
```

```
{
margin-left:30px;
margin-right:30px;
margin-top:50px;
font-size:14pt;
color:maroon;
}
</style>
</head>
<body>
<center>
<p>This is the text</p>
</center>
</body>
</html>
```

Output:



iv. Links:

Links.html:

```
<html>
```

```
<head>
```

```
<title>styles for links</title>
```

```
<style type="text/css">
```

```
A:link
```

```
{
```

```
text-decoration:none;
```

```
color:blue;
```

```
}
```

A:visited

{

text-decoration:none;

color:purple;

}

A:active

{

text-decoration:none;

color:red;

}

A:hover

{

text-decoration:underline;

color:yellow;

}

</style>

</head>

<body>

<center>

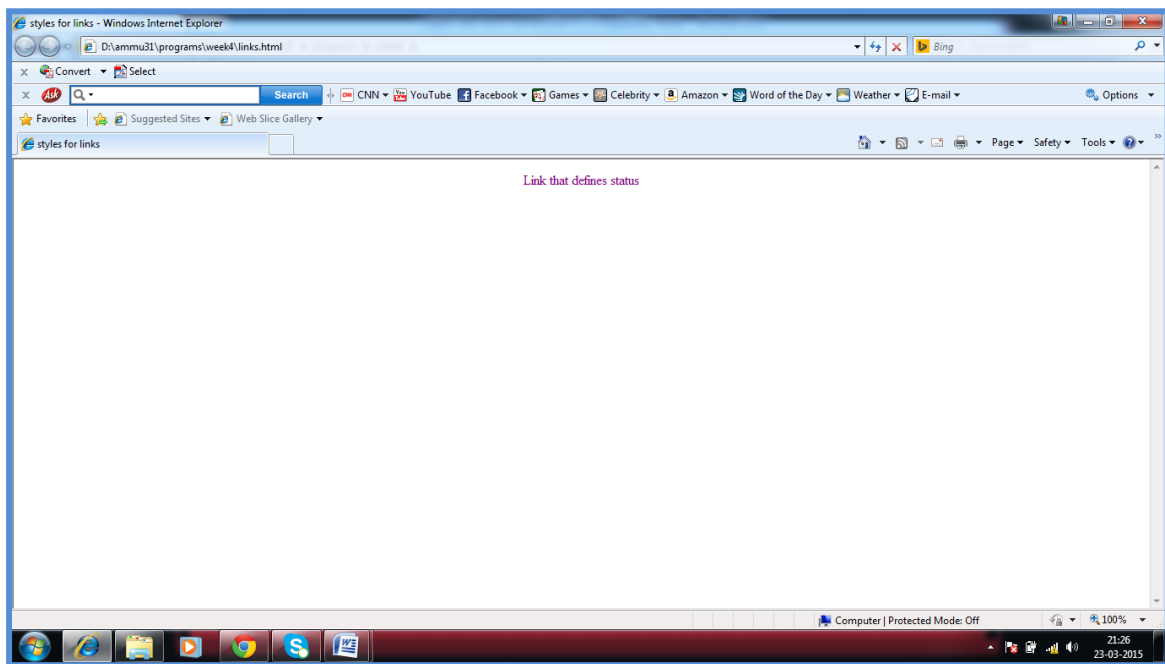
`Link that defines status`

`</center>`

`</body>`

`</html>`

Output:



v. Layers:

Layers.html:

`<html>`

`<head>`

`<title>html css</title>`

`<style type="text/css">`

```
body
{
background-image:url(pics/img1.jpg);
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<center>
```

LAYER 1 ON TOP:

```
<div style="position:relative; font-size:50px; z-
index:2;">LAYER 1</div>
```

```
<div style="position:relative; top:-50; left:5; color:red;
font-size:80px; z-index:1;">LAYER 2</div>
```

```
<div style="position:relative; font-size:50px; z-
index:3;">LAYER 1</div>
```

```
<div style="position:relative; top:-50; left:5; color:red;
font-size:80px; z-index:4;">LAYER 2</div>
```

```
</center>
```

```
</body>
```

```
</html>
```

Output:



vi. Cursors:

Cursors.html:

```
<html>
```

```
<head>
```

```
<title>HTML CSS</title>
```

```
<style type="text/css">
```

```
p
```

```
{
```

```
color:purple;
```

```
font-size:14pt;
```

```
font-style:italic bold;
font-family:'Brush Script MT';
text-decoration:underline;
text-shadow:2px 8px red;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>This is a cursor:</p>
```

```
<a style="cursor:crosshair;font-size:25px;">crosshair</a><br>
```

```
<a style="cursor:default;font-size:25px;">default</a><br>
```

```
<a style="cursor:grab;font-size:25px;">grab</a><br>
```

```
<a style="cursor:zoom-in;font-size:25px;">zoomin</a><br>
```

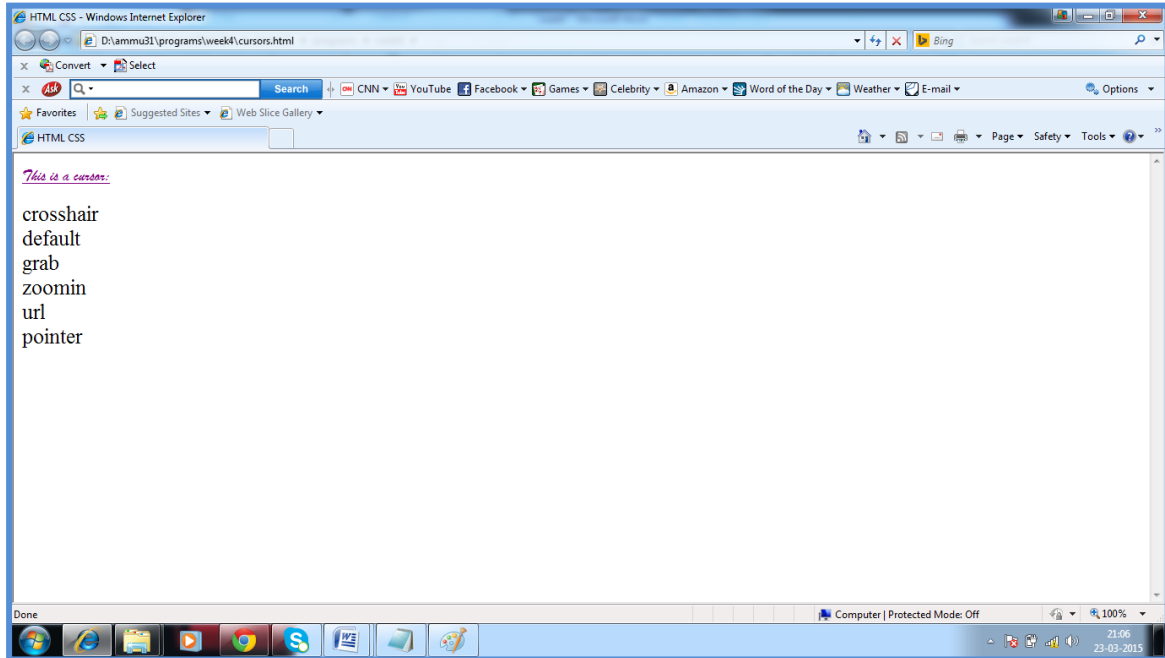
```
<a style="cursor:url;font-size:25px;">url</a><br>
```

```
<a style="cursor:pointer;font-size:25px;">pointer</a><br>
```

```
</body>
```

</html>

Output:



WEEK5:

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

Catalog.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="library.css"?>
<Catalog
xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
xsi:noNamespaceSchemaLocation="Catalog.xsd">
  <Book>
    <Title>Title1</Title>
    <Author>Author1</Author>
    <Edition>Edition 1</Edition>
    <ISBN>0-7645-4760-7</ISBN>
    <Price>Rs.450</Price>
  </Book>
  <Book>
    <Title>Title2</Title>
    <Author>Author2</Author>
    <Edition>Edition2</Edition>
    <ISBN>0-7645-4760-7</ISBN>
```

```
<Price>Rs.450</Price>
</Book>
<Book>
<Title>Title3</Title>
<Author>Author3</Author>
<Edition>Edition3</Edition>
<ISBN>0-7645-4760-7</ISBN>
<Price>Rs.450</Price>
</Book>
</Catalog>
```

Catalog.xsd:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
xmlns:xs="http://w3.org/2001/XMLSchema">
<xs:element name="Catalog">
<xs:complexType>
<xs:sequence>
<xs:element ref="Book" minOccurs="1"
maxOccurs="unbounded"/>
```

```
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Book">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Title" minOccurs="1"
maxOccurs="1"/>
      <xs:element ref="Author" minOccurs="1"
maxOccurs="1"/>
      <xs:element ref="Publication" minOccurs="1"
maxOccurs="1"/>
      <xs:element ref="Element" minOccurs="1"
maxOccurs="1"/>
      <xs:element ref="ISBN" minOccurs="1"
maxOccurs="1"/>
      <xs:element ref="Price" minOccurs="1"
maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```


Library.css:

Catalog

```
{  
font-family:arial;  
color:red;  
font-size:16pt;  
}
```

Book

```
{  
display:block;  
font-family:Timesnewroman;  
color:blue;  
table-layout:auto;  
font-size:14pt;  
}
```

Title

```
{  
font-family:arial;
```

color:green;

font-size:12pt;

}

Author

{

font-family:arial;

color:magenta;

}

Publication,Edition,ISBN,Price

{

display:block;

font-family:arial;

color:black;

font-size:10pt;

margin-left:20pt;

}

MyCatalog-xsl.xsl:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
<html>
<body>
<center>
<h2>Books Information</h2>
<table border="1">
<tr bgcolor="gray">
<th>TITLE</th>
<th>AUTHOR</th>
<th>PUBLICATION</th>
<th>ISBN</th>
<th>EDITION</th>
<th>PRICE</th>
</tr>
<xsl:for-each select="Catalog/Book">
<tr>
```

```
<td bgcolor="green">
```

```
<xsl:value-of select="Title"/>
```

```
</td>
```

```
<td bgcolor="red">
```

```
<b><xsl:value-of  
select="translate(Author,'abcdefghijklmnopqrstuvwxy'  
, 'ABCDEFGHIJKLMNOPQRSTUVWXYZ')"/>
```

```
</b>
```

```
</td>
```

```
<td bgcolor="cyan">
```

```
<xsl:value-of select="Publication"/>
```

```
</td>
```

```
<td bgcolor="yellow">
```

```
<xsl:value-of select="ISBN"/>
```

```
</td>
```

```
<td bgcolor="silver">
```

```
<xsl:value-of select="Edition"/>
```

```
</td>
```

```
<td bgcolor="magenta">
```

```
<xsl:value-of select="Price"/>
```

</td>

</tr>

</xsl:for-each>

</table>

</center>

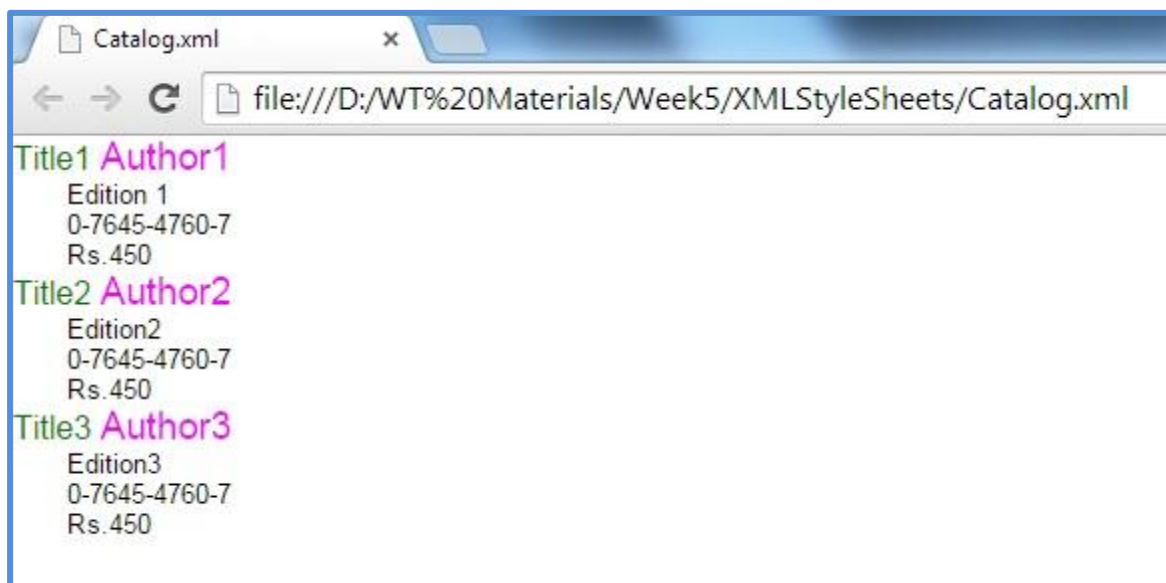
</body>

</html>

</xsl:template>

</xsl:stylesheet>

Output:



WEEK6:

Visual Beans:

VisualBeanEx.java:

```
package com.cmrcet.cse.yellaswamy.beans;
```

```
import java.awt.*;
```

```
import java.awt.event.*;
```

```
public class VisualBeanEx extends Canvas
```

```
{
```

```
    transient private Color color;
```

```
    private boolean rectangular;
```

```
    //constructor
```

```
    public VisualBeanEx()
```

```
    {
```

```
        addMouseListener(new MouseAdapter()
```

```
        {
```

```
            public void mousePressed(MouseEvent me)
```

```
            {
```

```
                change();
```

```
            }
```

```
});  
rectangular=false;  
setSize(100,100);  
change();  
}  
public boolean getRectangular()  
{  
return rectangular;  
}  
public void setRectangular(boolean bool)  
{  
this.rectangular=bool;  
repaint();  
}  
public void change()  
{  
color=randomColor();  
repaint();  
}
```

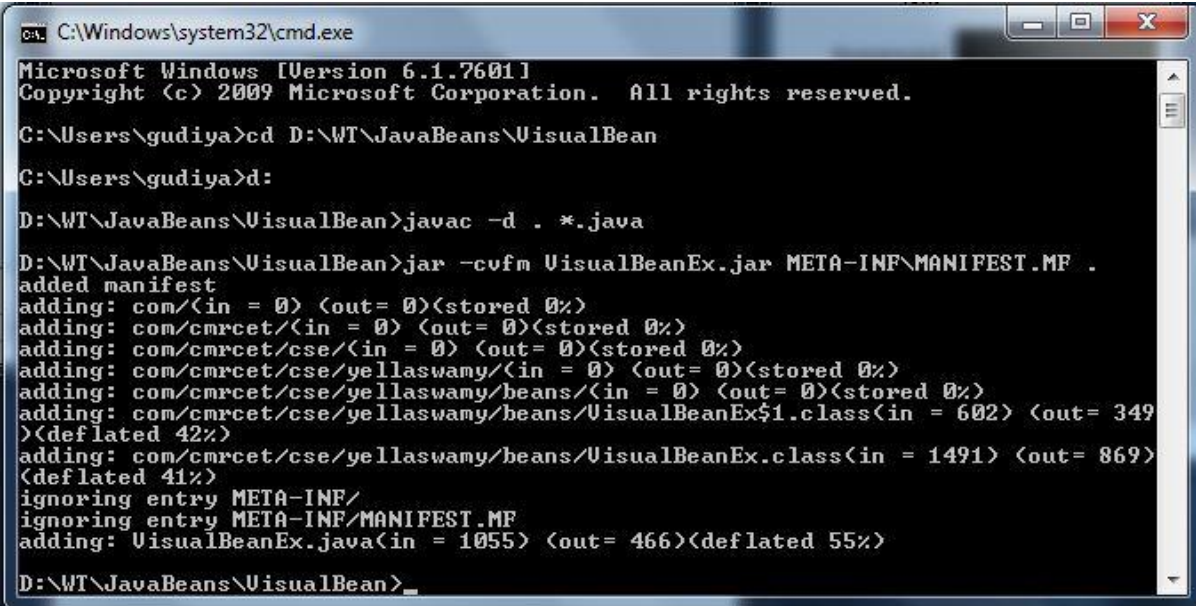
```
private Color randomColor()
{
int r=(int)(255*Math.random());
int g=(int)(255*Math.random());
int b=(int)(255*Math.random());
return new Color(r,g,b);
}

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);
    }
}
};
```

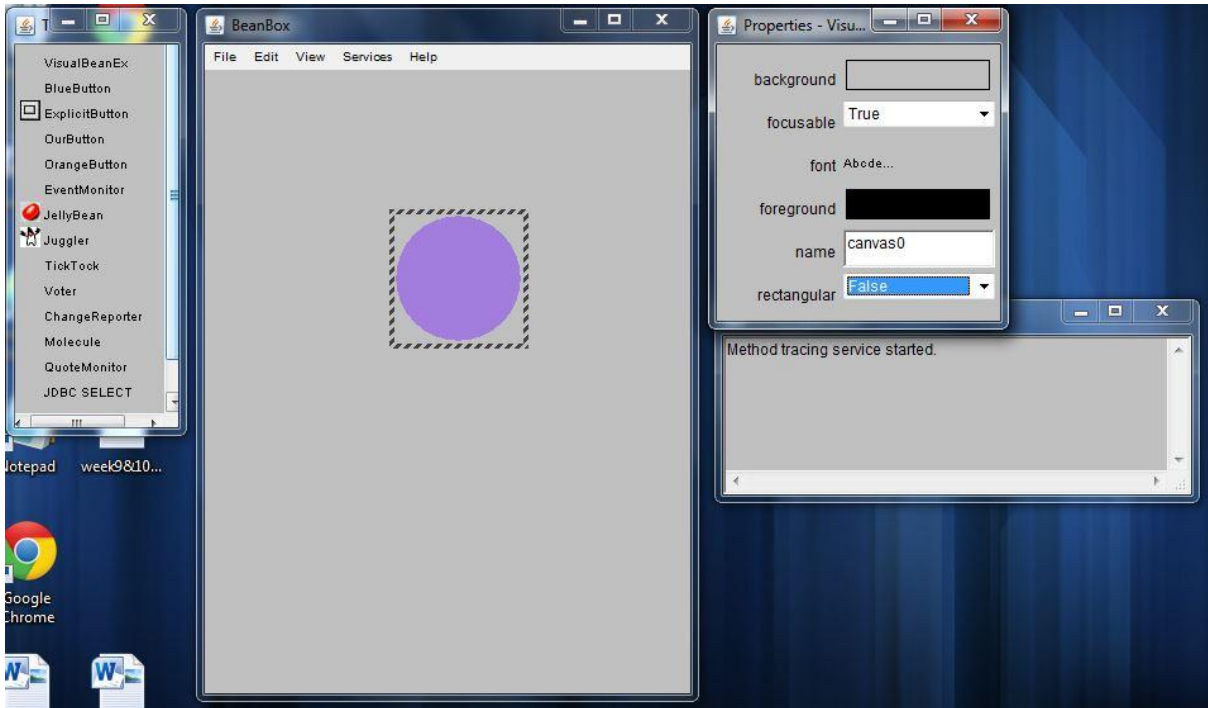
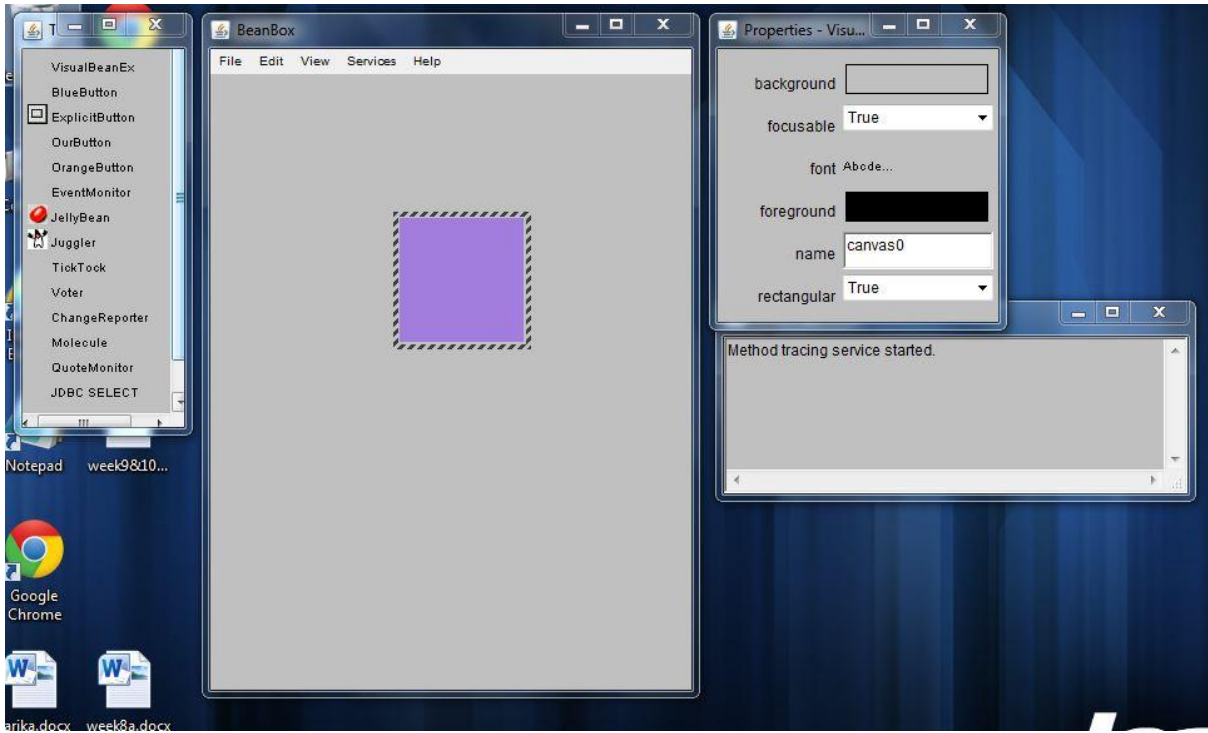
Output:



```
ca. C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\gudiya>cd D:\WT\JavaBeans\VisualBean
C:\Users\gudiya>d:
D:\WT\JavaBeans\VisualBean>javac -d . *.java
D:\WT\JavaBeans\VisualBean>jar -cvfm VisualBeanEx.jar META-INF\MANIFEST.MF .
added manifest
adding: com/<in = 0> <out= 0><stored 0%>
adding: com/cmrcet/<in = 0> <out= 0><stored 0%>
adding: com/cmrcet/cse/<in = 0> <out= 0><stored 0%>
adding: com/cmrcet/cse/yellaswamy/<in = 0> <out= 0><stored 0%>
adding: com/cmrcet/cse/yellaswamy/beans/<in = 0> <out= 0><stored 0%>
adding: com/cmrcet/cse/yellaswamy/beans/VisualBeanEx$1.class<in = 602> <out= 349>
<deflated 42%>
adding: com/cmrcet/cse/yellaswamy/beans/VisualBeanEx.class<in = 1491> <out= 869>
<deflated 41%>
ignoring entry META-INF/
ignoring entry META-INF/MANIFEST.MF
adding: VisualBeanEx.java<in = 1055> <out= 466><deflated 55%>

D:\WT\JavaBeans\VisualBean>_
```



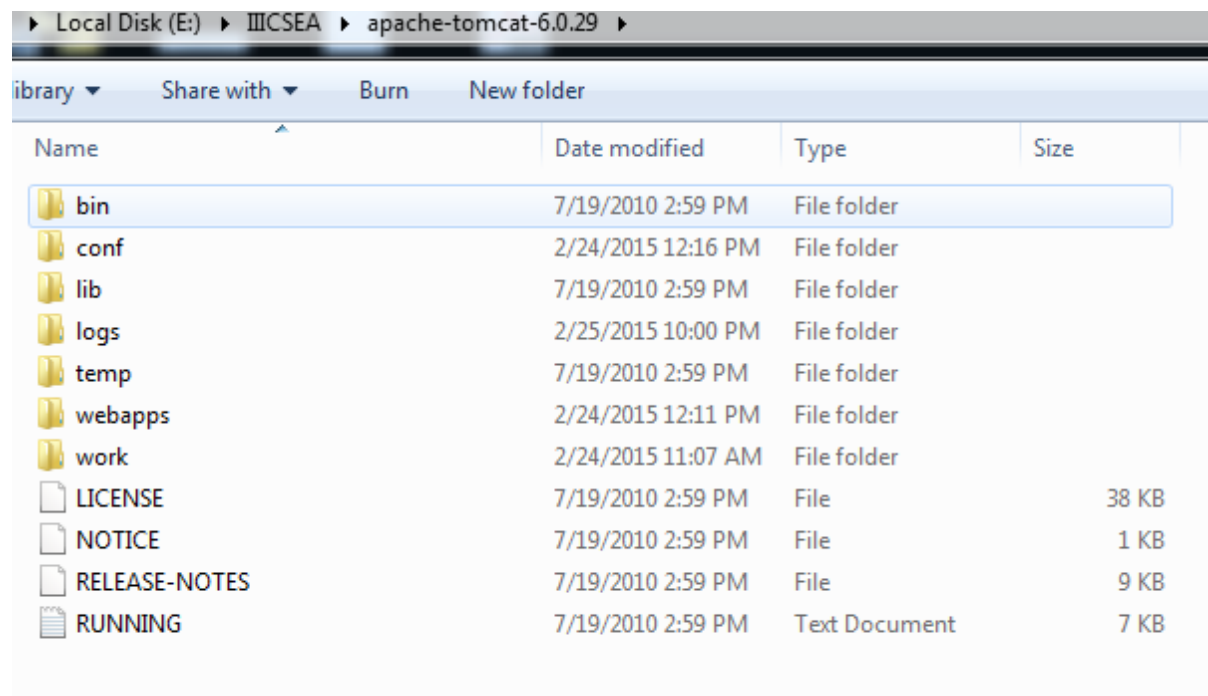
WEEK7:

Tomcat Installation:

Step1:

Download the zipped tomcat version from the link

<https://tomcat.apache.org/>.



The screenshot shows a Windows File Explorer window with the address bar displaying 'Local Disk (E:) > III CSEA > apache-tomcat-6.0.29'. The window title is 'Library' and it has menu options for 'Share with', 'Burn', and 'New folder'. The main area displays a list of files and folders with columns for Name, Date modified, Type, and Size.

Name	Date modified	Type	Size
bin	7/19/2010 2:59 PM	File folder	
conf	2/24/2015 12:16 PM	File folder	
lib	7/19/2010 2:59 PM	File folder	
logs	2/25/2015 10:00 PM	File folder	
temp	7/19/2010 2:59 PM	File folder	
webapps	2/24/2015 12:11 PM	File folder	
work	2/24/2015 11:07 AM	File folder	
LICENSE	7/19/2010 2:59 PM	File	38 KB
NOTICE	7/19/2010 2:59 PM	File	1 KB
RELEASE-NOTES	7/19/2010 2:59 PM	File	9 KB
RUNNING	7/19/2010 2:59 PM	Text Document	7 KB

Step2:

Extract the folder and open bin->startup.batch file.

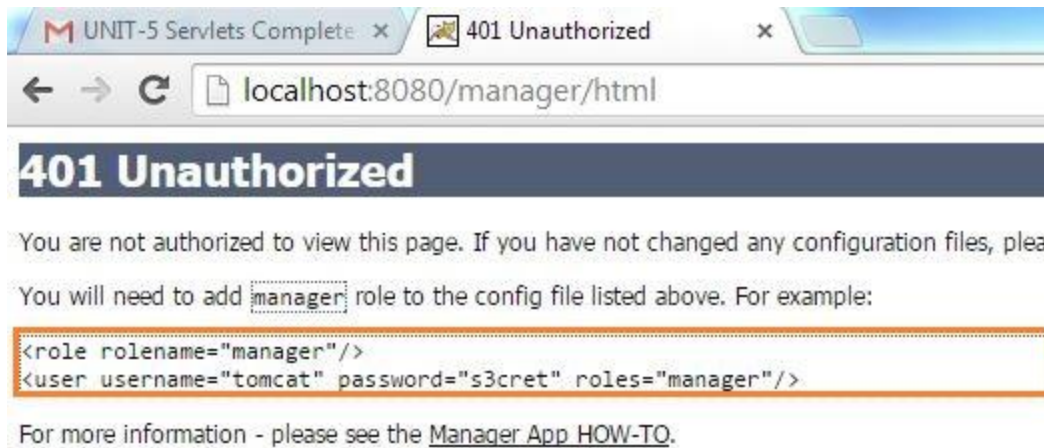
```
Tomcat
Mar 26, 2015 10:03:44 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory httpsession
Mar 26, 2015 10:03:45 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory initparameter
Mar 26, 2015 10:03:45 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory lifecycle
Mar 26, 2015 10:03:45 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory logincookie
Mar 26, 2015 10:03:45 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory param
Mar 26, 2015 10:03:45 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory ROOT
Mar 26, 2015 10:03:45 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory url
Mar 26, 2015 10:03:45 AM org.apache.coyote.http11.Http11AprProtocol start
INFO: Starting Coyote HTTP/1.1 on http-8080
Mar 26, 2015 10:03:45 AM org.apache.coyote.ajp.AjpAprProtocol start
INFO: Starting Coyote AJP/1.3 on ajp-8009
Mar 26, 2015 10:03:45 AM org.apache.catalina.startup.Catalina start
INFO: Server startup in 2602 ms
Mar 26, 2015 10:04:05 AM org.apache.catalina.startup.HostConfig checkResources
INFO: Undeploying context [/initparameter]
Mar 26, 2015 10:04:16 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory initparameter
```

Step3:

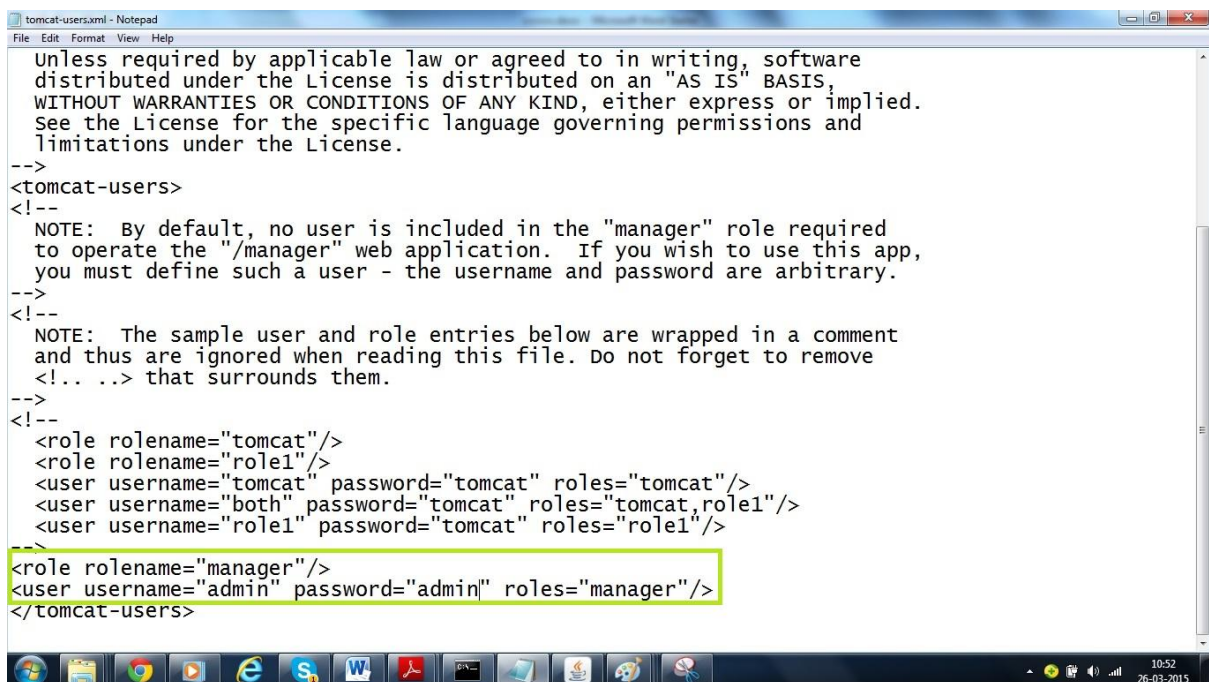
To set the username and password open

<http://localhost:8080> in the web browser and click on the Tomcat Manager.

Then an alert named Authentication Required appears, click on cancel.



Copy the code highlighted above and paste it in conf-> tomcat-users.xml file.



Step4:

Inorder to change the port number open conf/server.xml in Tomcat Installation Folder.


```
Documentation at /docs/config/service.html
-->
<Service name="Catalina">
  <!--The connectors can use a shared executor, you can define one or more named thread
  pools-->
  <!--
  <Executor name="tomcatThreadPool" namePrefix="catalina-exec-"
    maxThreads="150" minSpareThreads="4"/>
  -->

  <!-- A "Connector" represents an endpoint by which requests are received
  and responses are returned. Documentation at :
  Java HTTP Connector: /docs/config/http.html (blocking & non-blocking)
  Java AJP Connector: /docs/config/ajp.html
  APR (HTTP/AJP) Connector: /docs/apr.html
  Define a non-SSL HTTP/1.1 Connector on port 8080
  -->
  <Connector port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443" />
  <!-- A "Connector" using the shared thread pool-->
  <!--
  <Connector executor="tomcatThreadPool"
    port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443" />
```

As the default port number is 8080 change it to any required port number.

WEEK8

a. Cookies:

index.html:

<html>

<head>

<meta charset="ISO-8859-1">

<title>Servlet Login Example</title>

</head>

<body>

```
<h1>Welcome to Login App by Cookie</h1>
```

```
<a href="login.html">Login</a>|
```

```
<a href="LogoutServlet">Logout</a>|
```

```
<a href="ProfileServlet">Profile</a>
```

```
</body>
```

```
</html>
```

Link.html:

```
<html>
```

```
<body>
```

```
<a href="login.html">Login</a> |
```

```
<a href="LogoutServlet">Logout</a> |
```

```
<a href="ProfileServlet">Profile</a>
```

```
<hr>
```

```
</body>
```

```
</html>
```

Login.html:

```
<html>
```

```
<body>
```

```
<form action="LoginServlet" method="post">
```

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

Password:<input type="password"
name="password">

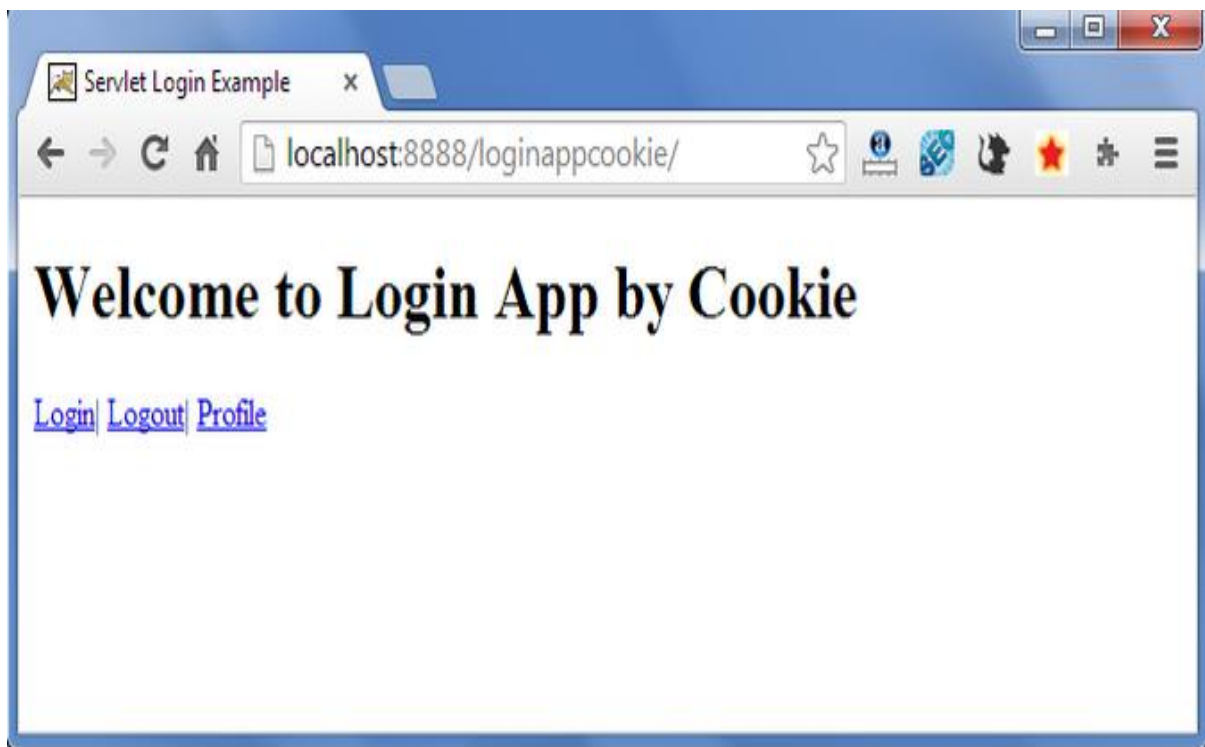
<input type="submit" value="login">

</form>

</body>

</html>

Output:



LoginServlet.java:

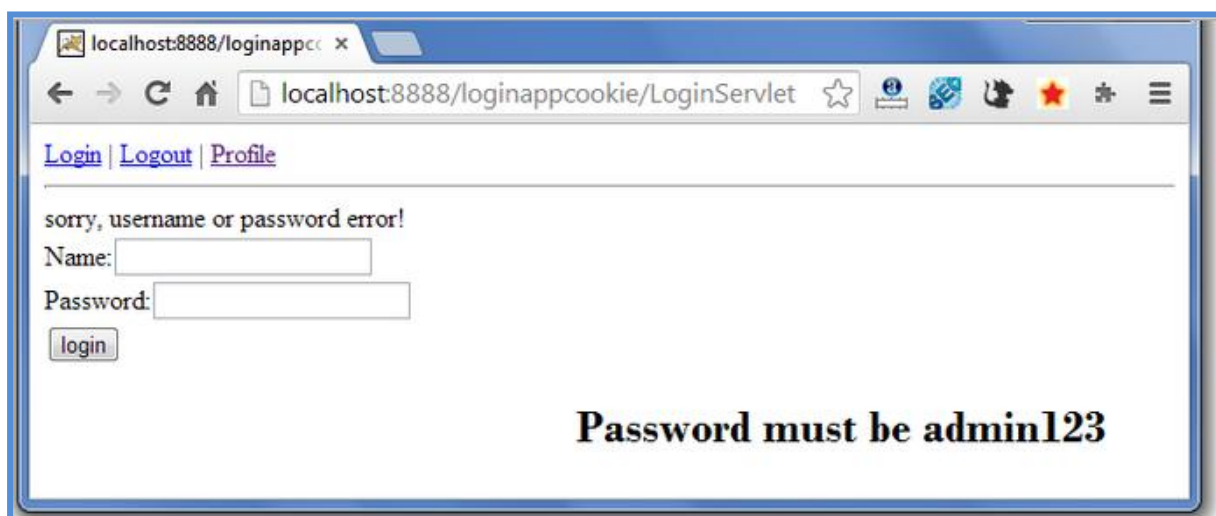
package com.cmrcet;

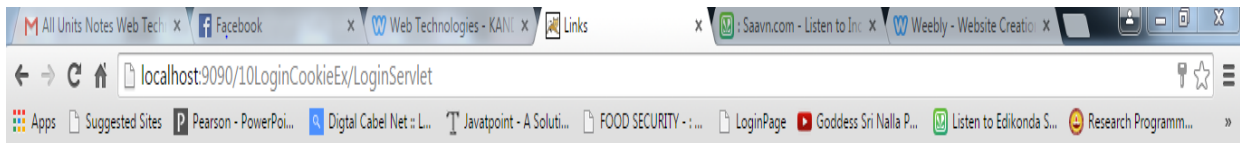

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class LoginServlet extends HttpServlet {
protected void doPost(HttpServletRequest request,
HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html");
PrintWriter out=response.getWriter();
request.getRequestDispatcher("link.html").include(request, response);
String name=request.getParameter("name");
String password=request.getParameter("password");
if(password.equals("admin123")){
out.print("You are successfully logged in!");
out.print("<br>Welcome, "+name);
```

```
Cookie ck=new Cookie("name",name);
response.addCookie(ck);
}else{
out.print("sorry, username or password error!");
request.getRequestDispatcher("login.html").include(request, response);
}

        out.close();
    }
}
```

Output:





LogoutServlet.java:

```
package com.cmrcet;

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class LogoutServlet extends HttpServlet {

    protected void doGet(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html");
```

```
PrintWriter out=response.getWriter();
request.getRequestDispatcher("link.html").include(requ
est, response);

Cookie ck=new Cookie("name","");

ck.setMaxAge(0);

response.addCookie(ck);

out.print("you are successfully logged out!");

}

}
```

Output:



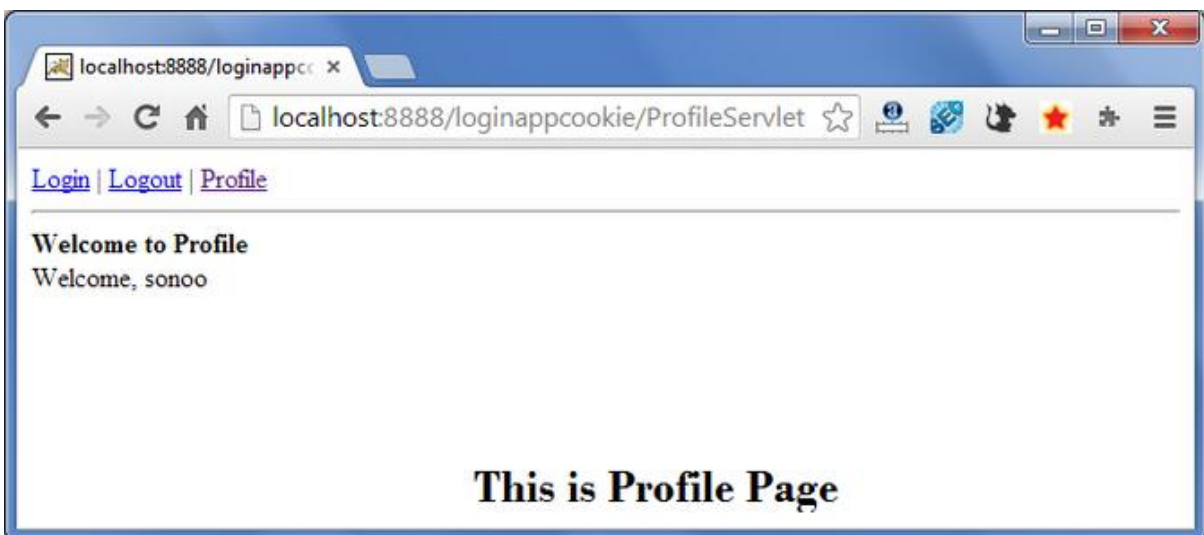
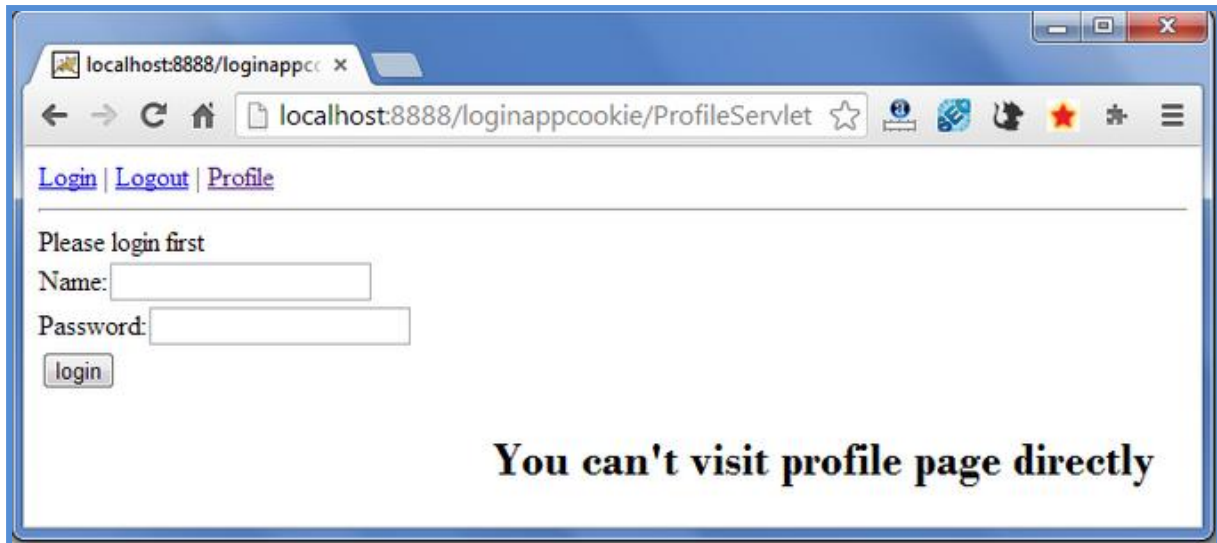
ProfileServlet.java:

```
package com.javatpoint;
```

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class LogoutServlet extends HttpServlet {
protected void doGet(HttpServletRequest request,
HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html");
PrintWriter out=response.getWriter();
request.getRequestDispatcher("link.html").include(request, response);
Cookie ck=new Cookie("name","");
ck.setMaxAge(0);
response.addCookie(ck);
out.print("you are successfully logged out!");
}
```

}

Output:



b. Init parameters:

index.html:

```
<html>
```

```
<body>
```

```
<form action="Demo">
```

```
<input type="submit" value="Click here">
```

```
</form>
```

```
</body>
```

```
</html>
```

FirstServlet.java:

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

```
import javax.servlet.*;
```

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

```
public class FirstServlet extends HttpServlet {
```

```
public void doGet(HttpServletRequest request,
```

```
HttpServletResponse response) throws
```

```
ServletException,IOException{
```

```
PrintWriter pw = response.getWriter();
```

```
pw.println(getServletConfig().getInitParameter("email"  
));}
```

```
}
```

Web.xml:

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

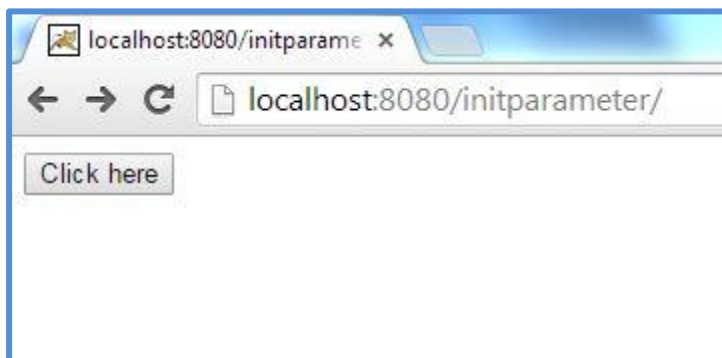
public class FirstServlet extends HttpServlet {

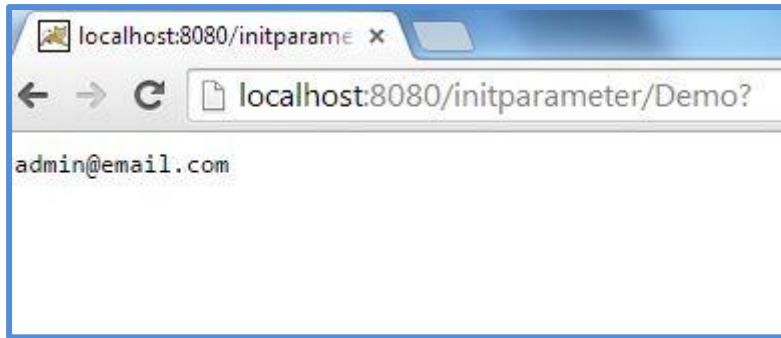
    public void doGet(HttpServletRequest request,
        HttpServletResponse response) throws
        ServletException,IOException{

        PrintWriter pw = response.getWriter();

        pw.println(getServletConfig().getInitParameter("email"
        ));
    }
}
```

Output:





WEEK9:

a. Servlets:

- i. Login servlet with JDBC
- ii. Registration servlet with jdbc

Home.html:

```
<!--Home.html-->
```

```
<html><body>
```

```
<center><h1>CMR COLLGE OF ENGINEERING &  
TECHNOLOGY.</h1></center>
```

```
<table border="1" width="100%" height="100%">
```

```
<tr>
```

```
<td width="15%" valign="top" align="center">
```

```
<br/><a href="Login.html">Login</a><br/>
```

```
<br/><a href="Register.html">Register</a><br/>
```

```
</td>
```

```
<td valign="top" align="center"><br/>
```

```
Welcome to CMRCET
```

```
</td>
```

```
</tr>
```

```
</table>
```

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

Login.html:

```
<!--Login.html-->
```

```
<html><body>
```

```
<center><h1>CMR College of Engineering &  
Technology</h1></center>
```

```
<form action="login">
```

```
<table border="1" width="100%" height="100%">
```

```
<tr>
```

```
<td width="15%" valign="top" align="center">
```

```
<br/><a href="Login.html">Login</a><br/>
```

```
<br/><a href="Register.html">Register</a><br/>
```

```
</td>
```

```
<td valign="top" align="center"><br/>
```

```
<form action="login"><table>
<tr>
<td colspan="2" align="center"><b>Login
Page</b></td>
</tr>
<tr>
<td colspan="2" align="center"><b>&nbsp;</td>
</tr>
<tr>
<td>User Name</td>
<td><input type="text" name="uname"/></td>
</tr>
<tr>
<td>Password</td>
<td><input type="password" name="pass"/></td>
</tr>
<tr>
<td>&nbsp;</td>
<td>&nbsp;</td>
```

```
</tr>
```

```
<tr>
```

```
<td colspan="2" align="center"><input type="submit"  
value="LogIN"/></td>
```

```
</tr>
```

```
</table></form>
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
</form>
```

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

Register.html:

```
<!--Register.html-->
```

```
<html><body>
```

```
<center><h1>CMR COLLEGE OF ENGINEERING &  
TECHNOLOGY</h1></center>
```

```
<form action="register" method="post">
```

```
<table border="1" width="100%" height="100%">
```

```
<tr>
```

```
<td width="15%" valign="top" align="center">
```



```
</tr>
```

```
<tr>
```

```
<td>Re-Password</td>
```

```
<td><input type="password" name="repass"/></td>
```

```
</tr>
```

```
<tr>
```

```
<td>Address</td>
```

```
<td><input type="text" name="addr"/></td>
```

```
</tr>
```

```
<tr>
```

```
<td>Phone Number</td>
```

```
<td><input type="text" name="phno"/></td>
```

```
</tr>
```

```
<tr>
```

```
<td>Email ID</td>
```

```
<td><input type="text" name="email"/></td>
```

```
</tr>
```

```
<tr>
```

```
<td>&nbsp;</td>
```

```
<td>&nbsp;</td>
</tr>
<tr>
<td colspan="2" align="center"><input type="submit"
value="Register"/></td>
</tr>
</table></form>
</td>
</tr>
</table>
</form>
</body></html>
```

Web.xml:

```
<?xml version="1.0"?>
<!DOCTYPE web-app
PUBLIC "-//Sun Microsystems, Inc.//DTD Web
Application 2.3//EN" "http://java.sun.com/dtd/web-
app_2_3.dtd">
<web-app>
<context-param>
```

```
<param-name>driverClassName</param-name>
<param-value>com.mysql.jdbc.Driver</param-value>
</context-param>
<context-param>
<param-name>url</param-name>
<param-value>jdbc:mysql://localhost:3306/lavzdb
</param-value>
</context-param>
<servlet>
<servlet-name>ls</servlet-name>
<servlet-
class>com.yellaswamy.servlets.LoginServlet</servlet-
class>
<init-param>
<param-name>dbuser</param-name>
<param-value>root</param-value>
</init-param>
<init-param>
<param-name>dbpass</param-name>
<param-value>root</param-value>
```


</init-param>

<init-param>

<param-name>sqlstatement</param-name>

<param-value>select * from suserdetails where
uname=? and pass=?</param-value>

</init-param>

</servlet>

<servlet>

<servlet-name>rs</servlet-name>

<servlet-
class>com.yellaswamy.servlets.RegistrationServlet

</servlet-class>

<init-param>

<param-name>dbuser</param-name>

<param-value>root</param-value>

</init-param>

<init-param>

<param-name>dbpass</param-name>

<param-value>root</param-value>

</init-param>

```
<init-param>
<param-name>sqlstatement</param-name>
<param-value>insert into suserdetails values(?,?,?,?,:)
</param-value>
</init-param>
</servlet>
<servlet-mapping>
<servlet-name>ls</servlet-name>
<url-pattern>/login</url-pattern>
</servlet-mapping>
<servlet-mapping>
<servlet-name>rs</servlet-name>
<url-pattern>/register</url-pattern>
</servlet-mapping>
<welcome-file-list>
<welcome-file>Home.html</welcome-file>
</welcome-file-list>
</web-app>
```

LoginServlet.java:

```
//LoginServlet

package com.yellaswamy.servlets;

import javax.servlet.*;

import java.io.*;

import java.sql.*;

public class LoginServlet extends GenericServlet {

private Connection con;

private PreparedStatement ps=null;

public void init() throws ServletException {

System.out.println("In init");

try {

        ServletContext ctxt=getServletContext();

        /Getting the Driver class name from context parameter

String

driverClassName=ctxt.getInitParameter("driverClassNa

me");

Class.forName(driverClassName);

        //Getting the JDBC URL from context parameter

String url=ctxt.getInitParameter("url");
```

```
//Getting the DB Username, password & sqlstatement
from servlet init parameter

String dbuser=getInitParameter("dbuser");
String dbpass=getInitParameter("dbpass");
String sqlst=getInitParameter("sqlstatement");
con=DriverManager.getConnection(url, dbuser,dbpass);
ps=con.prepareStatement(sqlst);
}try
catch(Exception e){
e.printStackTrace();
throw new ServletException("Initialization failed,
Unable to get DB connection");
}catch
}init

public void service (ServletRequest req,
ServletResponse res) throws ServletException,
IOException {
System.out.println("In service");
res.setContentType("text/html");
```

```
PrintWriter out=res.getWriter();
try {
String uname=req.getParameter("uname");
String pass= req.getParameter("pass");
if
(uname==null||uname.equals("")||pass==null||pass.equals("")) {
out.println("<html><body><center>");
out.println("<li><i>User Name and Password cannot be empty</i></li><br/>");
out.println("<li><i>We cannot log you into your account at this time. Please try again later</i>");
out.println("</center></body></html>");
return;
} //if

ps.setString(1,uname);
ps.setString(2,pass);
ResultSet rs=ps.executeQuery();
if (rs.next()){
```

```
out.println("<html><body>");
out.println("<center><h1>CMR COLLEGE OF
ENGINEERING & Technology</h1></center>");
out.println("<table border='1' width='100%'
height='100%'>");
out.println("<tr>");
out.println("<td width='15%' valign='top'
align='center'>");
out.println("<br/><a
href='Login.html'>Login</a><br/>");
out.println("<br/><a
href='Register.html'>Register</a>");
out.println("</td>");
out.println("<td valign='top' align='center'><br/>");
out.println("<h3>Welcome, "+uname+"</h3><br/>");
out.println("<h2>Enjoy browsing the Site</h2>");
out.println("</td></tr></table>");
out.println("</body></html>");
}else{
out.println("<html><body><center>");
```

```
out.println("Given username and password are  
incorrect<br/>");  
  
out.println("<li><i>We cannot log you into your  
account at this time. Please try again later</i>");  
  
out.println("</center></body></html>");  
    }  
  
} //try  
  
catch(Exception e){  
  
out.println("<html><body><center>");  
  
out.println("<h2>Unable to the process the request try  
after some time</h2>");  
  
out.println("</center></body></html>");  
  
} //catch  
  
} //service  
  
public void destroy () {  
  
System.out.println("In destroy");  
  
try {con.close();}  
  
catch(Exception e){ }  
  
} //destroy  
  
} //class
```

RegistrationServlet.java:

```
//LoginServlet

package com.yellaswamy.servlets;

import javax.servlet.*;

import java.io.*;

import java.sql.*;

/**
 * @author yellaswamy
 */

public class RegistrationServlet extends GenericServlet
{

private Connection con;

private PreparedStatement ps=null;

public void init() throws ServletException {

System.out.println("In init");

try {

ServletContext ctxt=getServletContext();

//Getting the Driver class name from context parameter
```



```
String
driverClassName=context.getInitParameter("driverClassNa
me");

Class.forName(driverClassName);

//Getting the JDBC URL from context parameter
String url=context.getInitParameter("url");

//Getting the DB Username, password & sqlstatement
from servlet init parameter

String dbuser=context.getInitParameter("dbuser");
String dbpass=context.getInitParameter("dbpass");
String sqlst=context.getInitParameter("sqlstatement");

con=DriverManager.getConnection(url, dbuser,
dbpass);

ps=con.prepareStatement(sqlst);

}try

catch(Exception e){

e.printStackTrace();

throw new ServletException("Initialization failed,
Unable to get DB connection");

}catch
```

```
}//init

public void service (ServletRequest req,
ServletResponse res) throws ServletException,
IOException {

System.out.println("In service");

res.setContentType("text/html");

PrintWriter out=res.getWriter();

try {

String uname=req.getParameter("uname");

String pass= req.getParameter("pass");

String repass= req.getParameter("repass");

if (uname==null||uname.equals(""))

||pass==null||pass.equals("")

||!pass.equals(repass)) {

out.println("<html><body><center>");

out.println("<li><i>Given details are not valid to
register</i></li><br/>");

out.println("<li><i>Please try again later</i>");

out.println("</center></body></html>");
```

```
return;

} //if

String addr=req.getParameter("addr");

String phno=req.getParameter("phno");

String email=req.getParameter("email");

ps.setString(1,uname);

ps.setString(2,pass);

ps.setString(3,addr);

ps.setString(4,phno);

ps.setString(5,email);

int count=ps.executeUpdate();

if
(count==1||count==Statement.SUCCESS_NO_INFO){

out.println("<html><body>");

out.println("<center><h1>CMRCET</h1></center>");

out.println("<table border=\"1\" width=\"100%\"
height=\"100%\">");

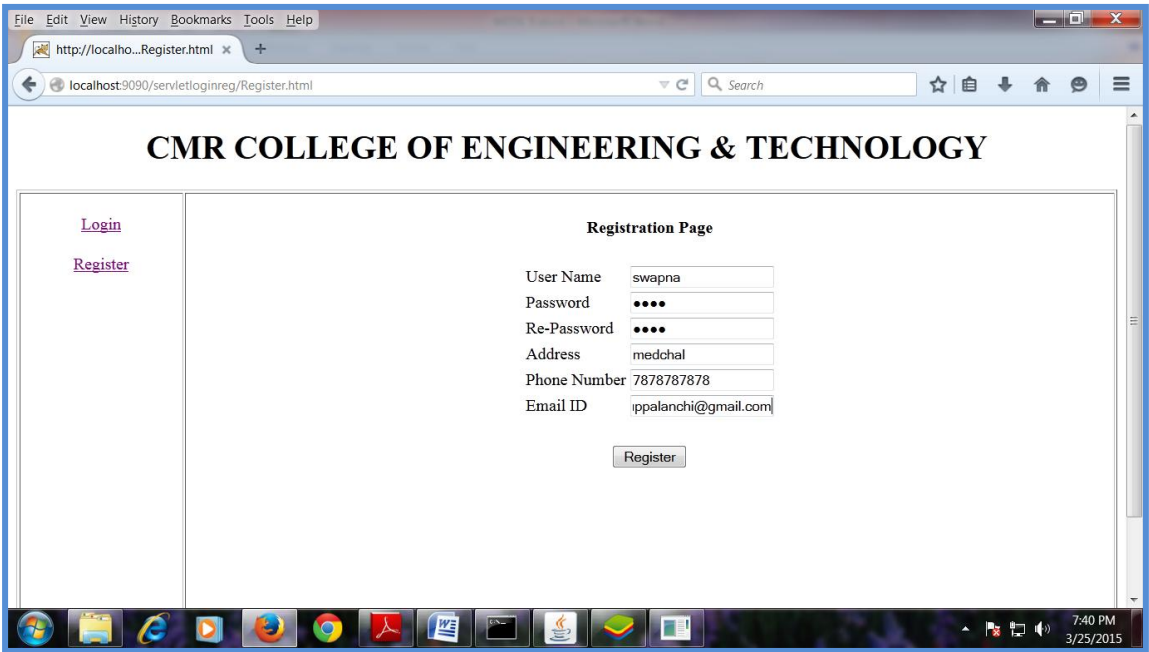
out.println("<tr>");

out.println("<td width=\"15%\" valign=\"top\"
align=\"center\">");
```

```
out.println("<br/><a  
href=\"Login.html\">Login</a><br/>");  
  
out.println("<br/><a  
href=\"Register.html\">Register</a><br/>");  
  
out.println("</td>");  
  
out.println("<td valign=\"top\  
align=\"center\"><br/>");  
  
out.println("<h3>Welcome, "+uname+"</h3><br/>");  
out.println("<h2>Enjoy browsing the Site</h2>");  
out.println("</td></tr></table>");  
out.println("</body></html>");  
  
}  
  
else{  
  
out.println("<html><body><center>");  
out.println("Given details are incorrect<br/>");  
out.println("<li><i>Please try again later</i>");  
out.println("</center></body></html>");  
        }  
  
} //try  
  
catch(Exception e){
```



```
C:\Windows\system32\cmd.exe
D:\tomcatapps\servletloginreg>tree/f
Folder PATH listing
Volume serial number is 00650072 5A0F:F19E
D:
| Home.html
| Login.html
| Register.html
| WEB-INF
| | web.xml
| | classes
| | | LoginServlet.java
| | | RegistrationServlet.java
| | | com
| | | | yellaswamy
| | | | | servlets
| | | | | LoginServlet.class
| | | | | RegistrationServlet.class
| | lib
| | | mysql-connector-java-5.1.8.jar
```



```
C:\Program Files\MySQL\MySQL Server 5.1\bin\mysql.exe
mysql> show tables;
+-----+
| Tables_in_lavzdb |
+-----+
| login            |
| mytable1         |
| mytable2         |
| mytable3         |
| mytable4         |
| students         |
| suserdetails     |
| userdetails      |
+-----+
8 rows in set (0.83 sec)

mysql> select * from suserdetails;
+-----+-----+-----+-----+-----+
| uname | pass | address | phno | email |
+-----+-----+-----+-----+-----+
| lavanya | pandu | medchal | 8989898989 | babydoll@my.org |
| swapna | sony | medchal | 7878787878 | uppalanchi@gmail.com |
+-----+-----+-----+-----+-----+
2 rows in set (0.29 sec)

mysql>
```

b. Basic servlet programs:

i. Request Parameters:

Post.html:

<html>

<title>post parameters</title>

<body>

<form action="First" method="POST">

First Name: <input type="text" name="firstName"><p>

Last Name: <input type="text" name="lastName"><p>

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

<input type="submit" value="Display">

</form>

</body>

</html>

Web.xml:

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

<!DOCTYPE web-app PUBLIC "-//Sun Microsystems,
Inc.//DTD Web Application 2.3//EN"

"http://java.sun.com/dtd/web-app_2_3.dtd">

<web-app >

<servlet>

<servlet-name>DisplayServlet</servlet-name>

<servlet-

class>com.cmrcet.yellaswamy.labprograms.DisplaySer
vlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>DisplayServlet</servlet-name>

<url-pattern>/First</url-pattern>

</servlet-mapping>

</web-app>

DisplayServlet.java:

```
package com.cmrcet.yellaswamy.labprograms;

import java.io.*;

import javax.servlet.*;

import javax.servlet.http.*;

public class DisplayServlet extends HttpServlet

{

public void doPost(HttpServletRequest req,
HttpServletResponse resp) throws
ServletException,IOException

    {

resp.setContentType("text/html");

PrintWriter out = resp.getWriter();

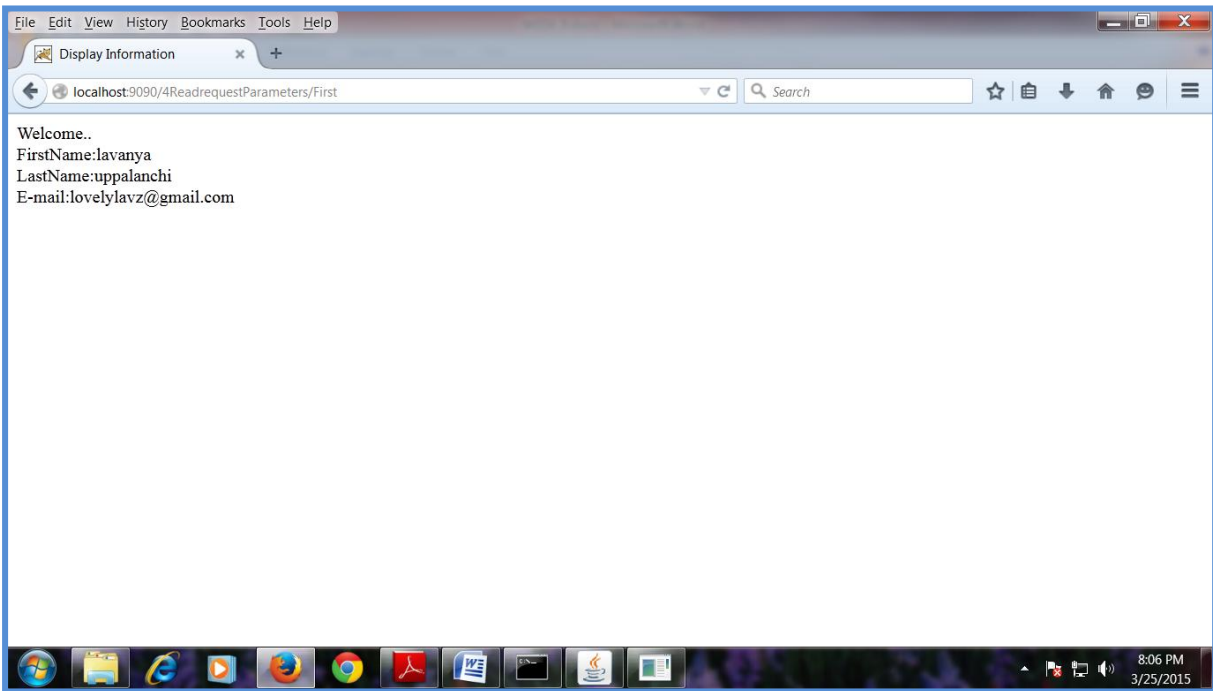
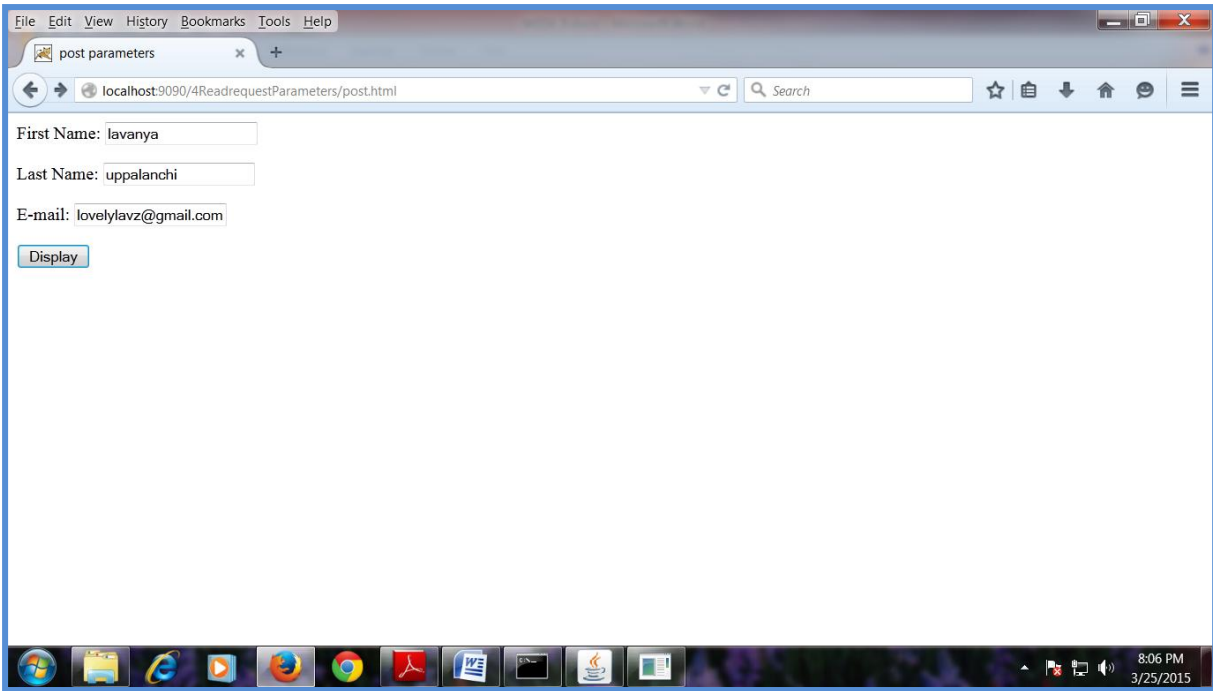
out.println("<html><head><title>Display
Information");

out.println("</title></head><body>");

out.println("Welcome..");

out.println("<br/>");

out.println("FirstName:" +req.getParameter("firstName"
));
```

ii. Life cycle servlet

life.html:

```
<html>
<body>
<form action="lifecycle">
<input type="submit" value="Invoke Lifecycle
Servlet"/>
</form>
</body>
</html>
```

LifeCycle.java:

```
package com.cmrcet.lifecycleex;
import javax.servlet.*;
import java.io.*;
public class LifeCycle implements Servlet
{
private ServletConfig myconfig;
public void init(ServletConfig sc)
```

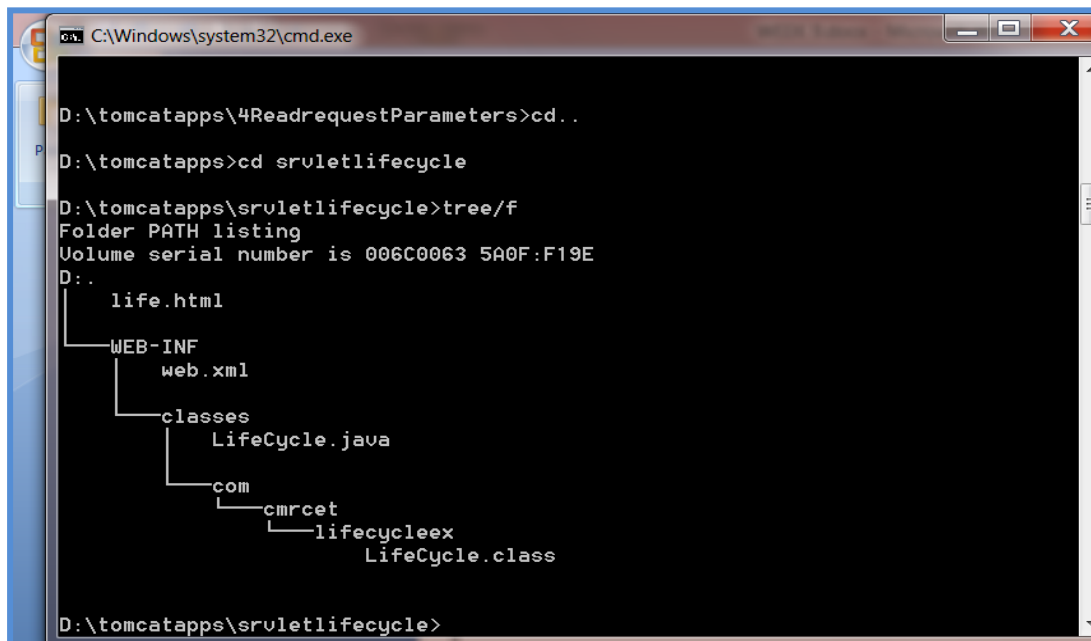
```
{  
this.myconfig=sc;  
System.out.println("In init() method");  
}  
  
public void service(ServletRequest  
myreq,ServletResponse myres)throws  
ServletException,IOException  
{  
System.out.println("in service() method");  
PrintWriter out=myres.getWriter();  
out.println("Hi iam frm tomcat server");  
}  
  
public void destroy()  
{  
System.out.println("in destroy() method");  
}  
  
public String getServletInfo()  
{  
return "Developed by CMRCET";  
}
```

```
public ServletConfig getServletConfig()
{
return myconfig;
}
}
```

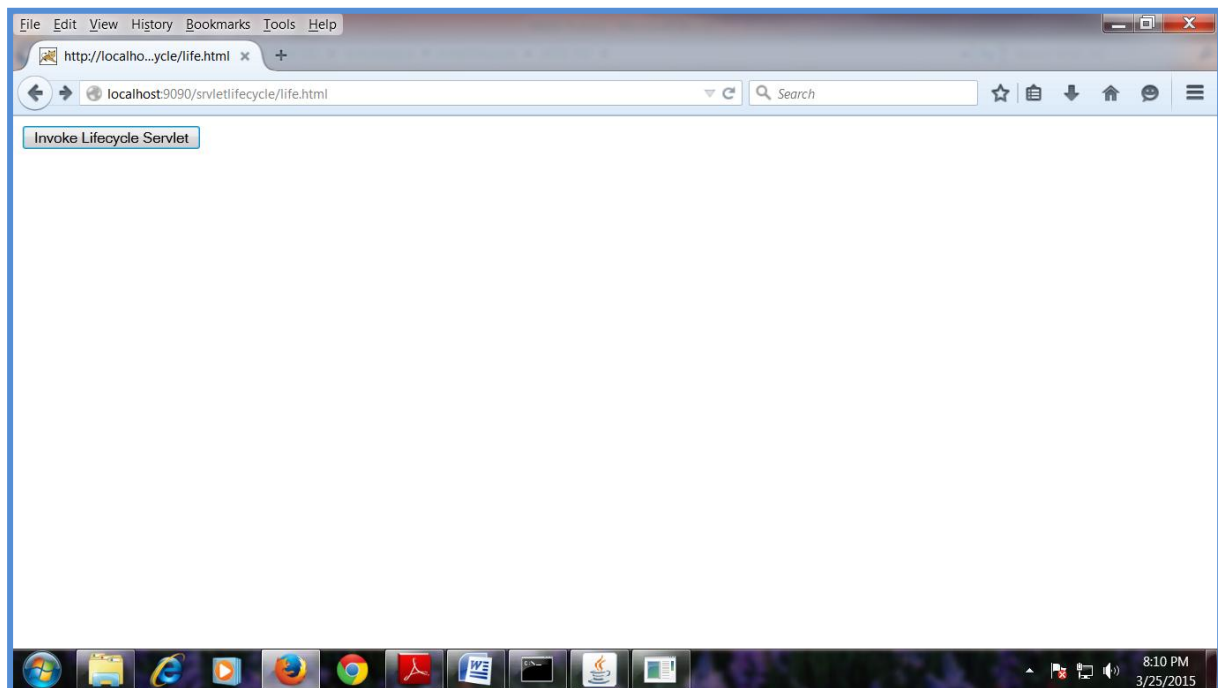
Web.xml:

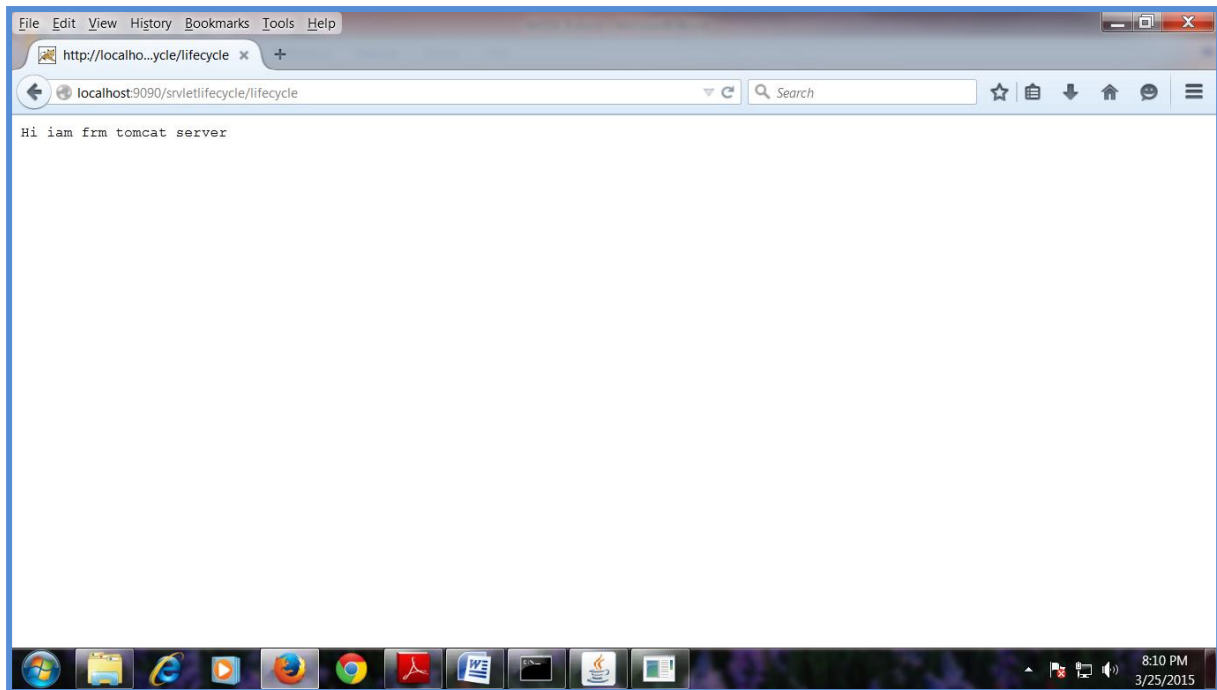
```
<web-app>
<servlet>
<servlet-name>LifeCycle</servlet-name>
<servlet-
class>com.cmrcet.lifecycleex.LifeCycle</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>LifeCycle</servlet-name>
<url-pattern>/lifecycle</url-pattern>
</servlet-mapping>
</web-app>
```

Output:



```
C:\Windows\system32\cmd.exe
D:\tomcatapps\4ReadrequestParameters>cd..
D:\tomcatapps>cd srveletlifecycle
D:\tomcatapps\srvletlifecycle>tree/f
Folder PATH listing
Volume serial number is 006C0063 5A0F:F19E
D:
|
|_ life.html
|_ WEB-INF
|   |_ web.xml
|   |_ classes
|       |_ Lifecycle.java
|       |_ com
|           |_ cmrcet
|               |_ lifecycleex
|                   |_ Lifecycle.class
D:\tomcatapps\srvletlifecycle>
```





c. Use Jsp to develop following programs

i. Login page with jdbc

ii. Registration Page with JDBC

Home.html:

```
<html >
```

```
<body>
```

```
<br /><br /><br /><br /><br />
```

```
<h1 align="center"><U>ONLINE BOOK  
STORAGE</U></h1><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 /><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>
```

```
<h1><center>LOGIN</center></h1>
```

```
<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>
```



```
</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 />
```



```
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
AGAINID<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("</body></html>");
```

```
con.close();
```

```
%>
```

Reg.jsp:

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

```
<% @page import="java.io.*"%>
```

```
<% response.setContentType("text/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("sun.jdbc.odbc.JdbcOdbcDriver");
```

```
Connection
```

```
con=DriverManager.getConnection("jdbc:odbc:regjspds  
n","","");
```

```
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.getStrin
g(2)))
{ flag=1;
}
} if(flag==1)
{ out.println("<br><br>SORRY INVALID ID
ALREADY EXISTS 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+"','"+no+"','"+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();  
  
%>
```

d. Basic Jsp Programs

i. Using Scripting Elements

ScriptingTags.jsp:

```
<% @ page import = "java.util.*" language = "java" %>  
  
<html><body>  
  
<center><h1>Using Scripting Elements</h1></center>  
  
<% ! int count,a,b;  
  
int fun(int a)  
  
{  
  
    return 10*a;
```



```
}%>
<% a=1;
count++;
for (int i=0;i<5;i++)
{
out.println("Value of i in iteration
no."+i+":&nbsp;&nbsp;&nbsp;<b>"+i+"</b><br/>");
}
b=fun(a);
out.println("Value returned by
fun():&nbsp;&nbsp;&nbsp;<b>"+b+"</b><br/>");
%>
```

This page is requested by **<%=count%>**
number of times on date

<%=new Date()%>.

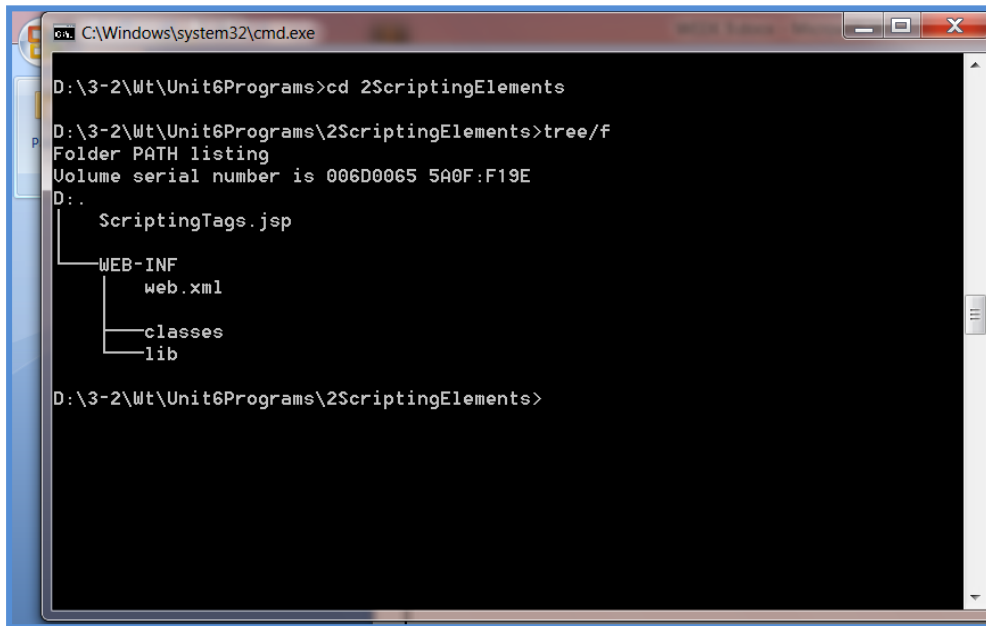
```
</body></html>
```

Web.xml:

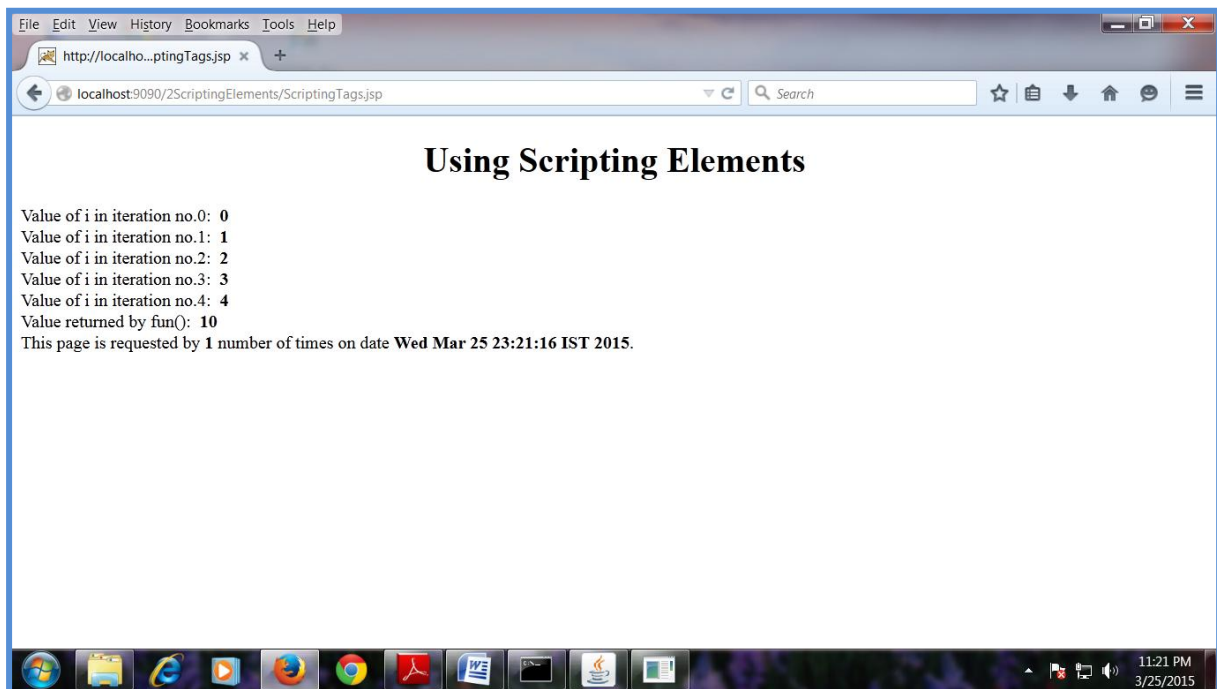
```
<web-app>
```

```
</web-app>
```

Output:



```
C:\Windows\system32\cmd.exe
D:\3-2\Wt\Unit6Programs>cd 2ScriptingElements
D:\3-2\Wt\Unit6Programs\2ScriptingElements>tree/f
Folder PATH listing
Volume serial number is 006D0065 5A0F:F19E
D:
  ScriptingTags.jsp
  WEB-INF
    web.xml
    classes
    lib
D:\3-2\Wt\Unit6Programs\2ScriptingElements>
```



ii. Jsp program for java beans using action tags

GetName.html:

```
<HTML>
```

```
<BODY>
```

```
<FORM METHOD=POST  
ACTION="SaveName.jsp">
```

```
What's your name? <INPUT TYPE=TEXT  
NAME=username SIZE=20><BR>
```

```
What's your e-mail address? <INPUT TYPE=TEXT  
NAME=email SIZE=20><BR>
```

```
What's your age? <INPUT TYPE=TEXT NAME=age  
SIZE=4>
```

```
<P><INPUT TYPE=SUBMIT>
```

```
</FORM>
```

```
</BODY>
```

```
</HTML>
```

NextPage.jsp

```
<jsp:useBean id="user" class="user.UserData"  
scope="session"/>
```

```
<HTML>
```

```
<BODY>
```

```
You entered<BR>
```

```
Name: <%= user.getUsername() %><BR>
```

```
Email: <%= user.getEmail() %><BR>
```

```
Age: <%= user.getAge() %><BR>
```

```
</BODY>
```

```
</HTML>
```

SaveName.jsp:

```
<jsp:useBean id="user" class="user.UserData"  
scope="session"/>
```

```
<jsp:setProperty name="user" property="*/>
```

```
<HTML>
```

```
<BODY>
```

```
<A HREF="NextPage.jsp">Continue</A>
```

```
</BODY>
```

```
</HTML>
```

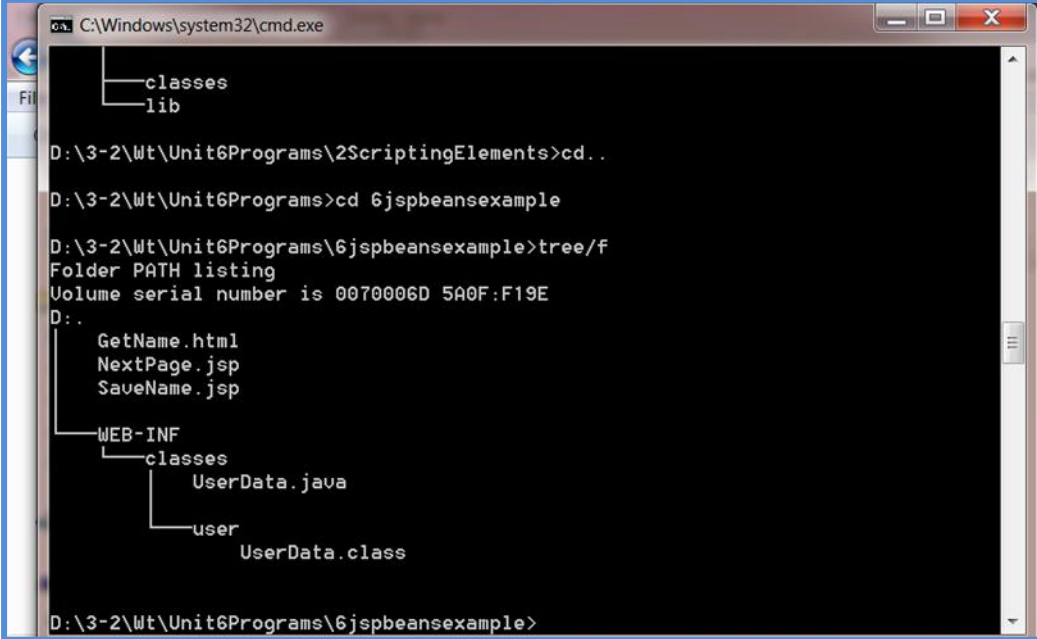
UserData.java:

```
package user;
```

```
public class UserData {  
    String username;  
    String email;  
    int age;  
    public void setUsername( String value )  
    {  
        username = value;  
    }  
    public void setEmail( String value )  
    {  
        email = value;  
    }  
    public void setAge( int value )  
    {  
        age = value;  
    }  
    public String getUsername() { return username; }  
    public String getEmail() { return email; }  
    public int getAge() { return age; }  
}
```

}

Output:

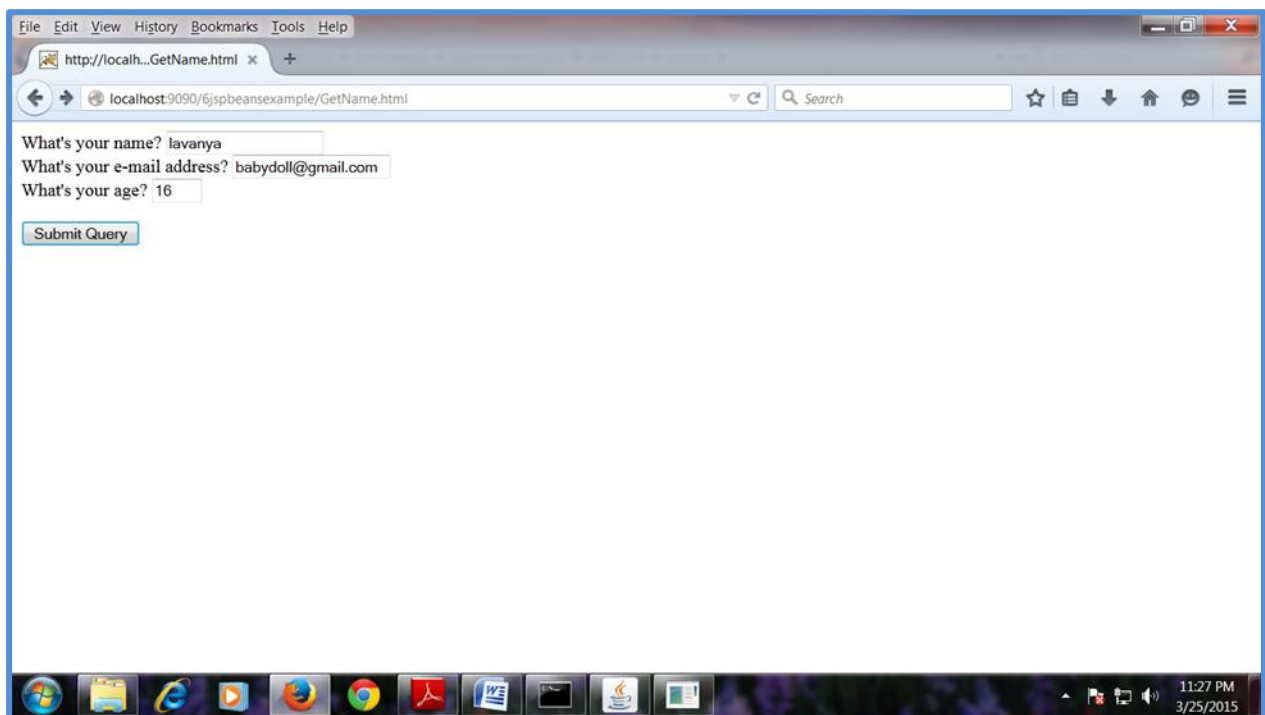


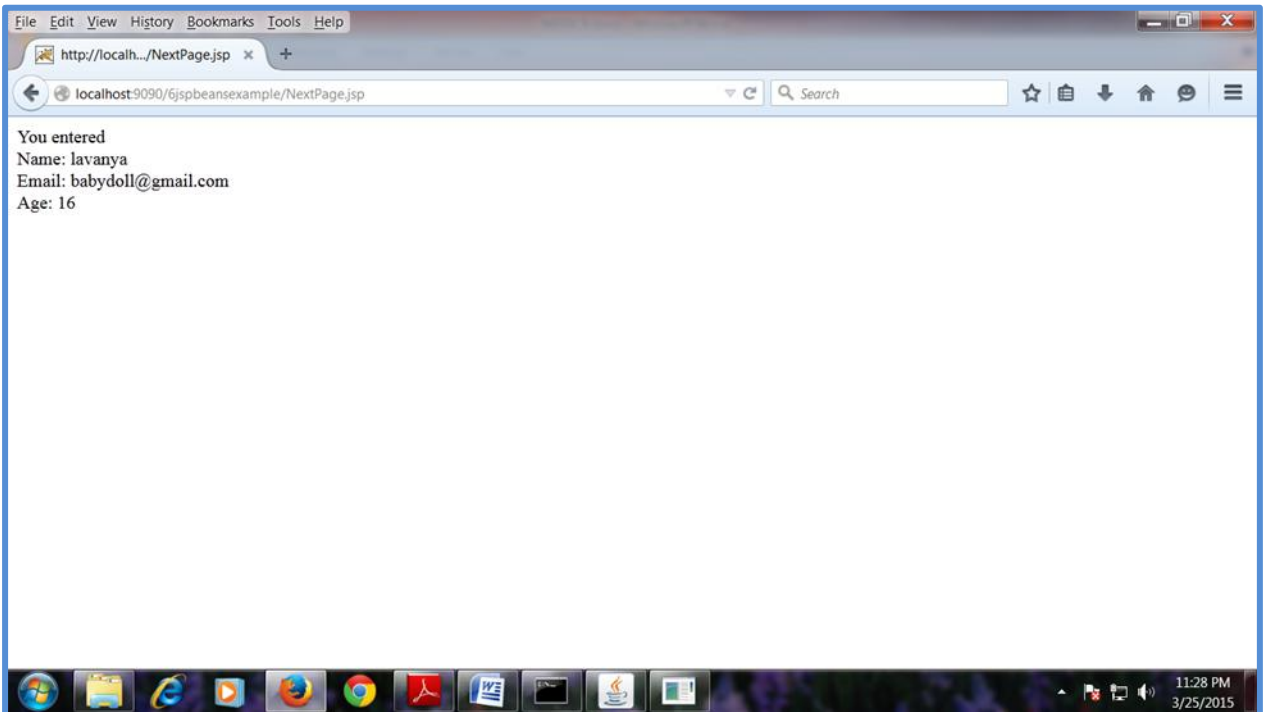
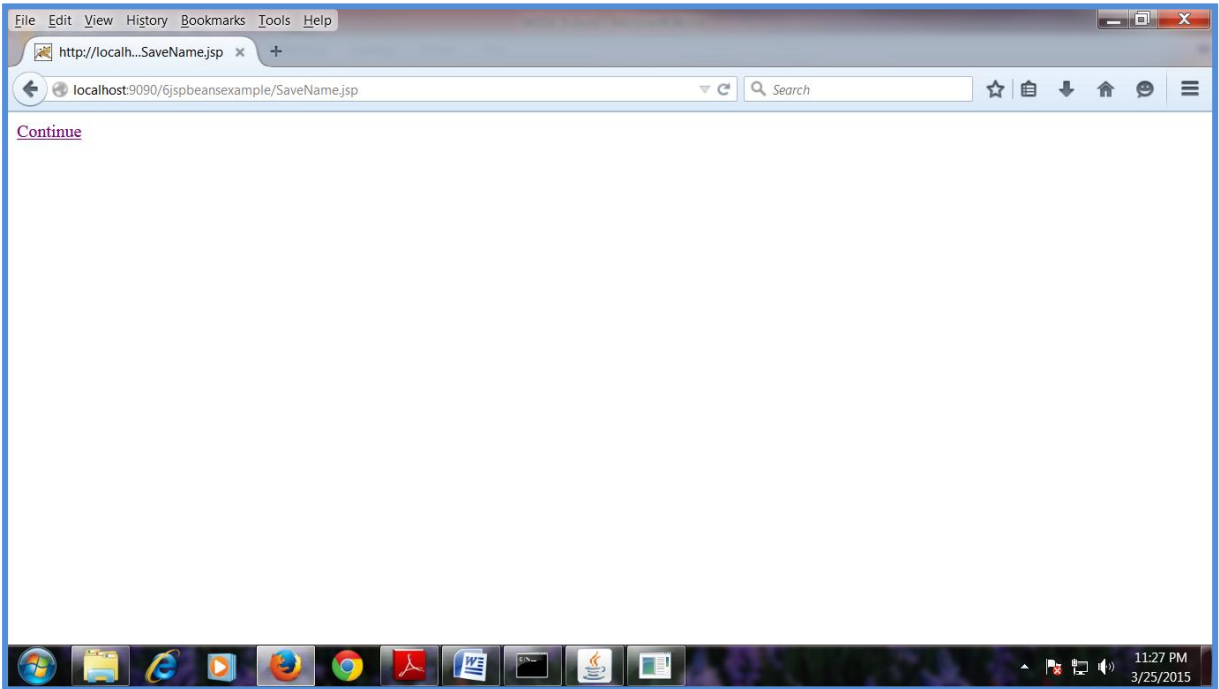
```
C:\Windows\system32\cmd.exe

classes
lib

D:\3-2\Wt\Unit6Programs>cd ..
D:\3-2\Wt\Unit6Programs>cd 6jspbeansexample
D:\3-2\Wt\Unit6Programs\6jspbeansexample>tree /f
Folder PATH listing
Volume serial number is 0070006D 5A0F:F19E
D: .
  GetName.html
  NextPage.jsp
  SaveName.jsp
  WEB-INF
    classes
      UserData.java
    user
      UserData.class

D:\3-2\Wt\Unit6Programs\6jspbeansexample>
```





Week10:

a.Database operations

i.Creating database table using type 4 driver

CreateTable.java:

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

class CreateTable
{
    public static void main(String args[]) throws
    SQLException,ClassNotFoundException
    {
        String driverClass="com.mysql.jdbc.Driver";

        String
        connectionURL="jdbc:mysql://localhost:3306/lavzdb";

        String username="root";

        String password="root";

        String query="create table student555(col1
        varchar(20),col2 numeric,col3 numeric(10))";
```



```
Class.forName(driverClass);
```

```
Connection
```

```
conn=DriverManager.getConnection(connectionURL,u  
sename,password);
```

```
Statement st=conn.createStatement();
```

```
int count=st.executeUpdate(query);
```

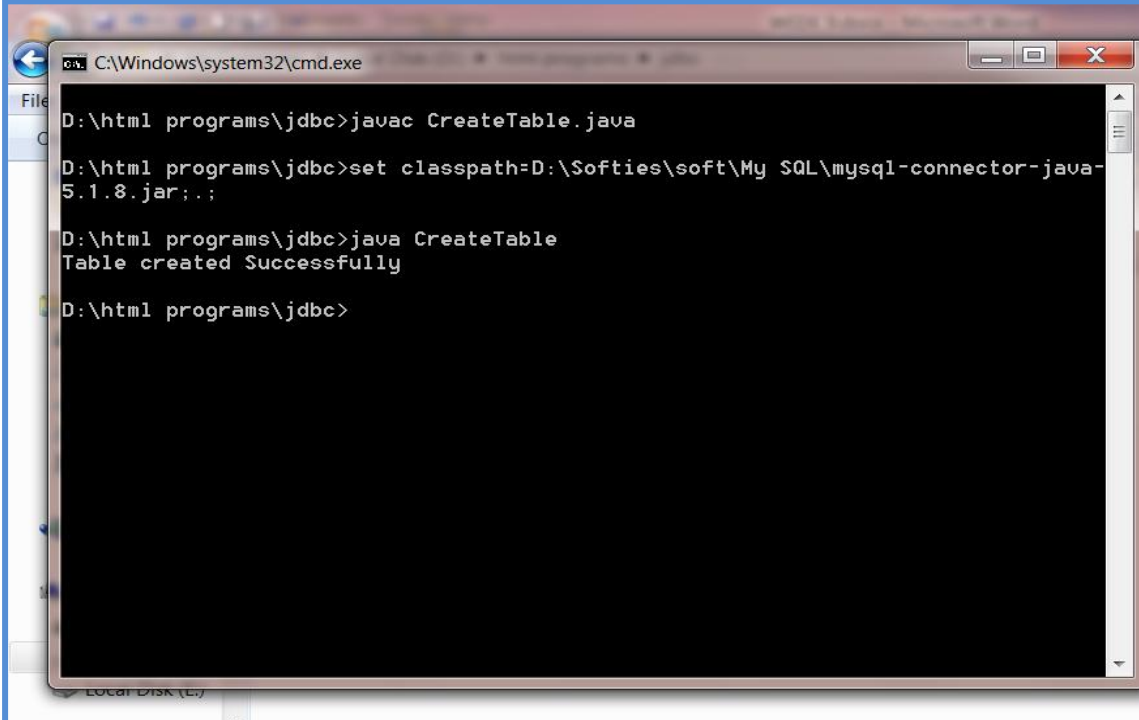
```
System.out.println("Table created Successfully");
```

```
conn.close();
```

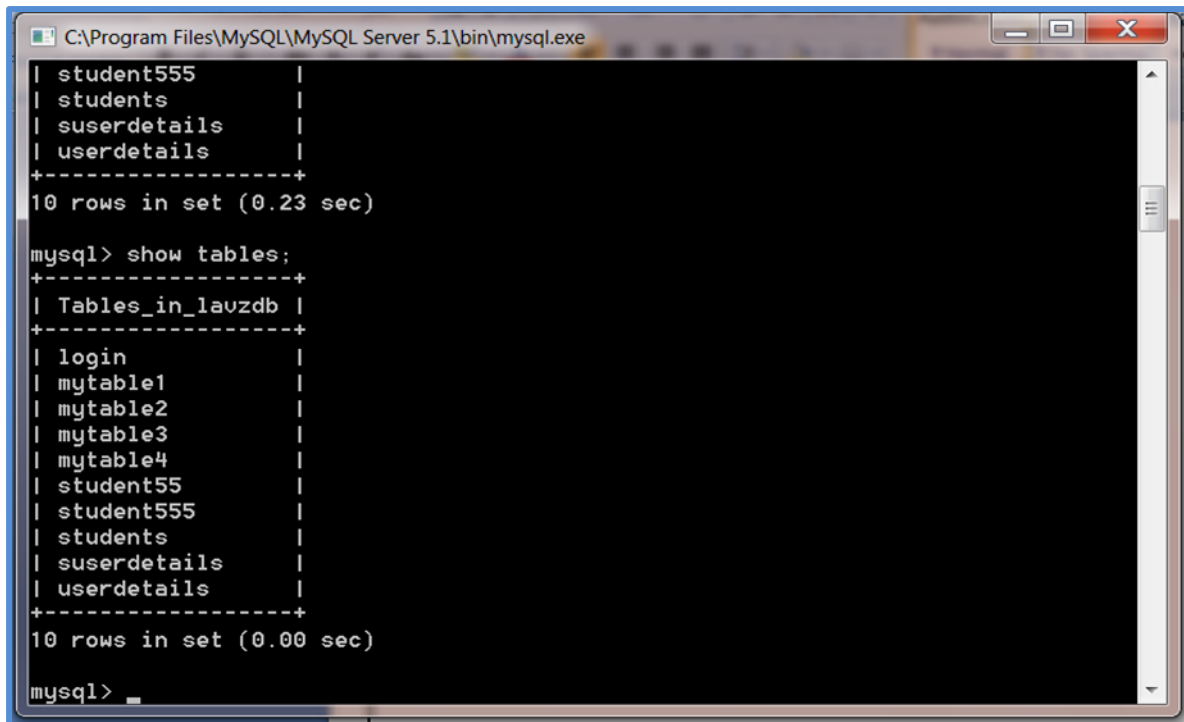
```
}
```

```
}
```

Output:



```
C:\Windows\system32\cmd.exe  
D:\html programs\jdbc>javac CreateTable.java  
D:\html programs\jdbc>set classpath=D:\Softies\soft\My SQL\mysql-connector-java-  
5.1.8.jar;. ;  
D:\html programs\jdbc>java CreateTable  
Table created Successfully  
D:\html programs\jdbc>
```



```
C:\Program Files\MySQL\MySQL Server 5.1\bin\mysql.exe
| student555 |
| students  |
| suserdetails |
| userdetails |
+-----+
10 rows in set (0.23 sec)

mysql> show tables;
+-----+
| Tables_in_laazdb |
+-----+
| login            |
| mytable1         |
| mytable2         |
| mytable3         |
| mytable4         |
| student555      |
| student555      |
| students         |
| suserdetails     |
| userdetails     |
+-----+
10 rows in set (0.00 sec)

mysql>
```

ii. Inserting a row into database using type 4 driver

Prep.java:

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

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

```
public class Prep
```

```
{
```

```
public static void main(String args[]) throws  
SQLException, ClassNotFoundException
```

```
{
```

```
String driverClassName="com.mysql.jdbc.Driver";
```

```
String url="jdbc:mysql://localhost:3306/lavzdb";
String username="root";
String password="root";
Class.forName(driverClassName);
Connection
conn=DriverManager.getConnection(url,username,pass
word);
String query="insert into student55 values(?,?,?)";
//step1:Get Prepared Statement
PreparedStatement ps=conn.prepareStatement(query);
//step2:set parameters
ps.setString(1,"shashank1");
ps.setInt(2,1216);
ps.setDouble(3,123.56);
//Step3:Execute the Query
int i=ps.executeUpdate();
System.out.println("Record inserted count="+i);
//To excecute the query once again
ps.setString(1,"Kamamma1");
ps.setInt(2,1217);
```

```
ps.setDouble(3,125.65);
```

```
i=ps.executeUpdate();
```

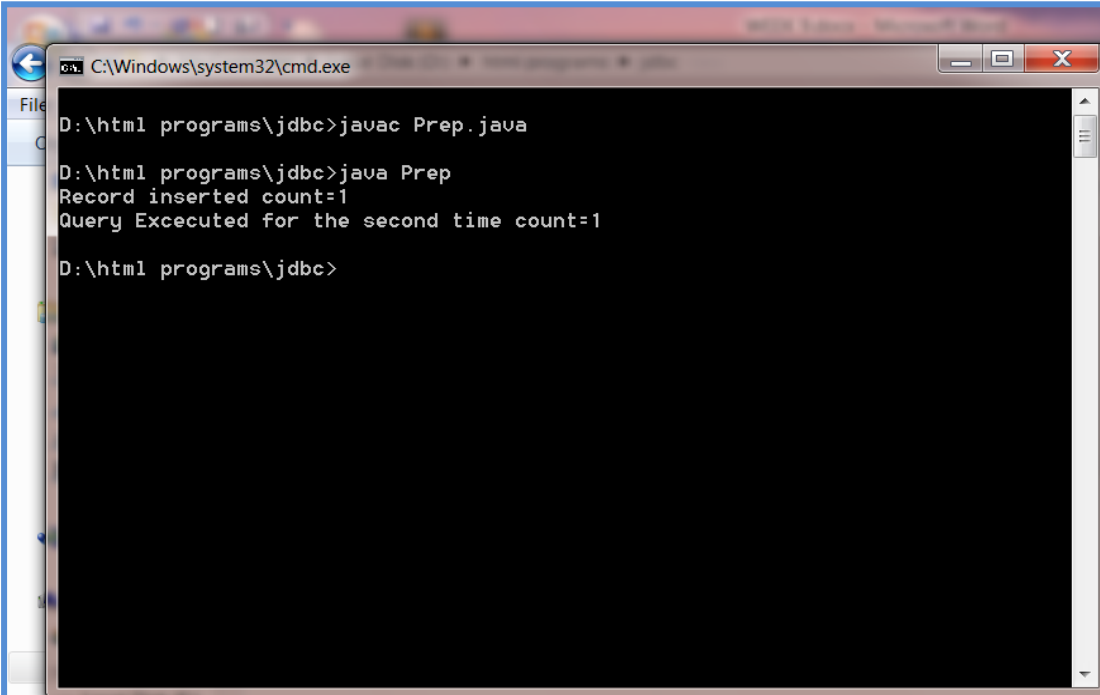
```
System.out.println("Query Excecuted for the second  
time count="+i);
```

```
conn.close();
```

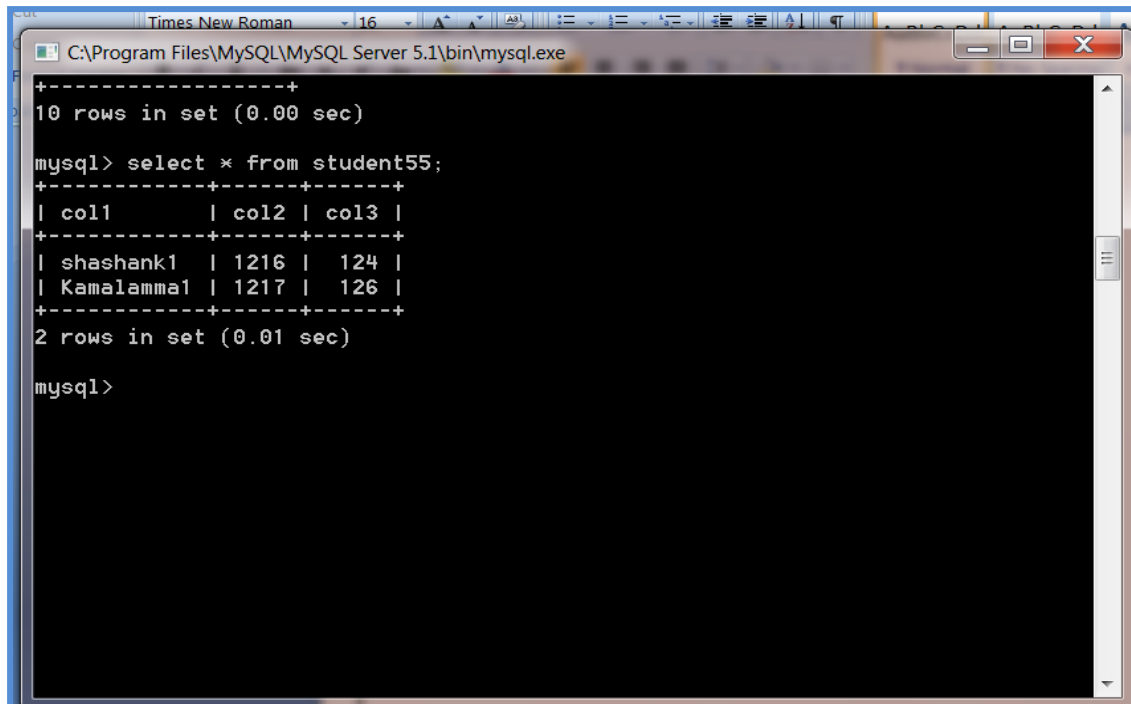
```
}
```

```
}
```

Output:



```
C:\Windows\system32\cmd.exe
D:\html programs\jdbc>javac Prep.java
D:\html programs\jdbc>java Prep
Record inserted count=1
Query Excecuted for the second time count=1
D:\html programs\jdbc>
```



The screenshot shows a MySQL command-line window with the following text:

```
C:\Program Files\MySQL\MySQL Server 5.1\bin\mysql.exe
+-----+
10 rows in set (0.00 sec)

mysql> select * from student55;
+-----+-----+-----+
| col1      | col2 | col3 |
+-----+-----+-----+
| shashank1 | 1216 | 124  |
| Kamamma1  | 1217 | 126  |
+-----+-----+-----+
2 rows in set (0.01 sec)

mysql>
```

b. Declaring variables and methods in Jsp

varmeth.jsp:

```
<% @ page language="java" contentType="text/html"
%>
```

```
<% String msg="Hello"; %>
```

```
<%! public String MyFunction(String msg)
```

```
{
```

```
return msg;
```

```
}
```

```
%>
```

```
<html>
```

```
<head>
```

```
<title>Use of Method</title>
```

```
</head>
```

```
<body>
```

```
<%
```

```
out.println("Before function call:"+msg); %>
```

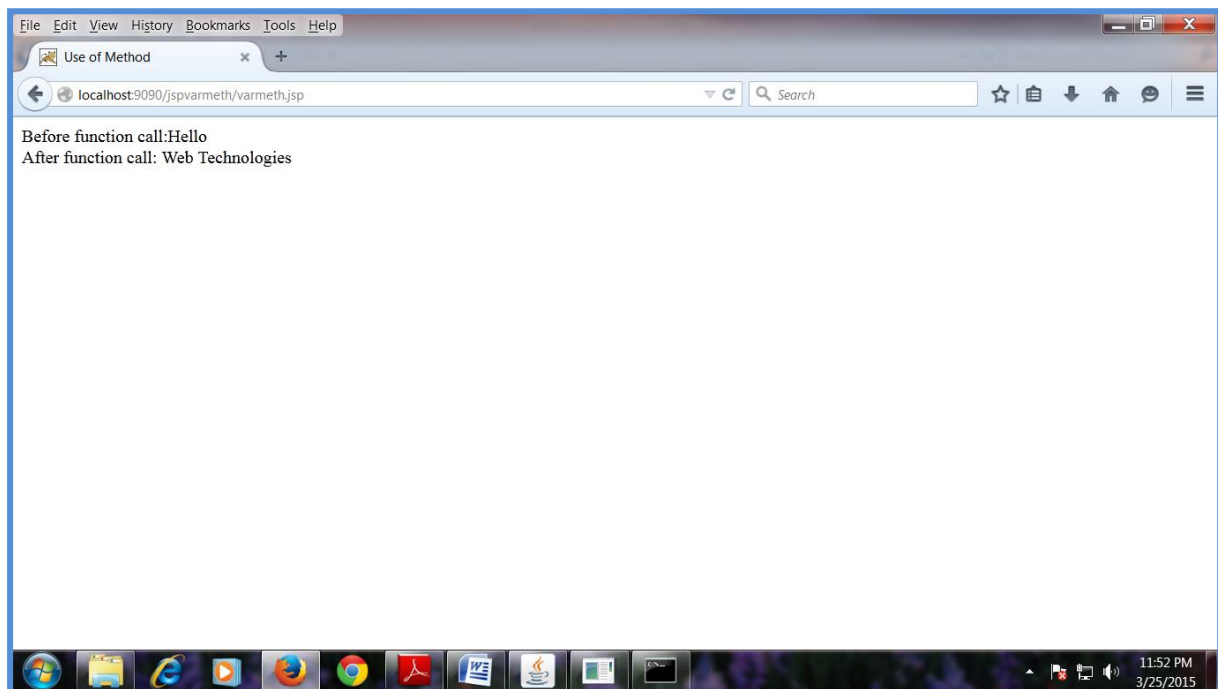
```
<br>
```

```
After function call: <%= MyFunction("Web  
Technologies") %>
```

```
</body>
```

```
</html>
```

Output:



c.Implicit objects in jsp

Home.html:

```
<html>
```

```
<body>
```

```
<form action="request.jsp">
```

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

```
<input type="submit" value="Invoke jsp"/>
```

```
</form>
```

```
</body>
```

```
</html>
```

Pagecontext.jsp

```
<html>
```

```
<head>
```

```
<title>PageContext</title>
```

```
</head>
```

```
<body>
```

```
<%
```

```
if("yes".equals(request.getParameter("other")))
```

```
{
```

```
pageContext.forward("other");
```

```
}
```

```
%>
```

```
</body>
```

```
</html>
```

```
request.jsp:
```

```
<html>
```

```
<head>
```

```
<title>ImplicitObjects</title>
```

```
</head>
```

```
<body>
```

```
Hello,<b><%=request.getParameter("name")%></b><br/><br/>
```

```
your request deatails are<br/><br/>
```

```
<table border="1">
```

```
<tr>
```

```
<th>Name</th><th>Value</th>
```

```
</tr>
```

```
<tr>
```



```
<td>Request method</td>
<td><%=request.getMethod()%></td>
</tr>
<tr>
<td>Request uri</td>
<td><%=request.getRequestURI()%></td>
</tr>
<tr>
<td>Request protocol</td>
<td><%=request.getProtocol()%></td>
</tr>
<tr>
<td>Browser</td>
<td><%=request.getHeader("user-agent")%></td>
</tr>
</table>
<% if(session.getAttribute("sessionVar")==null)
{
session.setAttribute("sessionVar",new Integer(0));
```

```
}
```

```
%>
```

```
<table>
```

```
<tr>
```

```
<th align="left">would u like to see use of remaining  
implicit objects?</th>
```

```
</tr>
```

```
<tr>
```

```
<form action="pageContext.jsp" method="post"  
name="form1">
```

```
<td><input type="radio" name="other"  
value="yes">yes</td>
```

```
<td><input type="radio" name="other"  
value="no">no</td>
```

```
</tr>
```

```
<tr>
```

```
<td><input type="submit" value="submit"/></td>
```

```
</form>
```

```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

Other.jsp:

```
<html>
```

```
<body>
```

```
<% !int count;
```

```
public void jspinit()
```

```
{
```

```
ServletConfig sc=getServletConfig();
```

```
count=Integer.parseInt(sc.getInitParameter("count"));
```

```
System.out.println("in jspInit");
```

```
}
```

```
%>
```

```
count value without using config implicit object:<b>
```

```
<%=count%></b>
```

```
<br/>
```

```
<%
```

```
this.log("log message");
```

```
((HttpServlet)page).log("another message");
ServletContext ct=config.getServletContext();
out.println("value of sessionVar
is:"+session.getAttribute("sessionVar")+"<br/>");
out.println("Server name and version using config
implicit object:"+ct.getServerInfo()+"<br/>");
out.println("vlaue of context parameter param1 get
using application implicit
object:"+application.getInitParameter("param1")+"<br/
>");
out.println("count value retrieved using config implicit
object:"+config.getInitParameter("count"));
%>
</body>
</html>
```

Web.xml:

```
<web-app>
<context-param>
<param-name>param1</param-name>
<param-value>param1</param-value>
</context-param>
```

<servlet>

<servlet-name>myjsp</servlet-name>

<jsp-file>/WEB-INF/jsppages/other.jsp</jsp-file>

<init-param>

<param-name>count</param-name>

<param-value>10</param-value>

</init-param>

</servlet>

<servlet-mapping>

<servlet-name>myjsp</servlet-name>

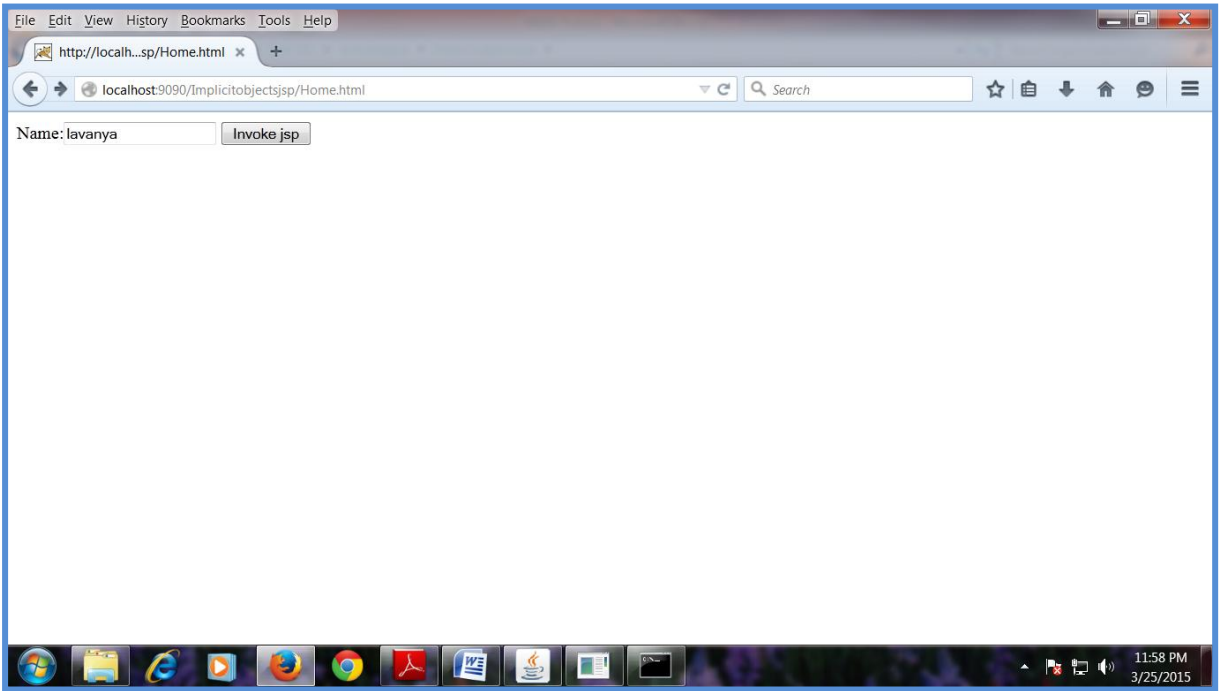
<url-pattern>/other</url-pattern>

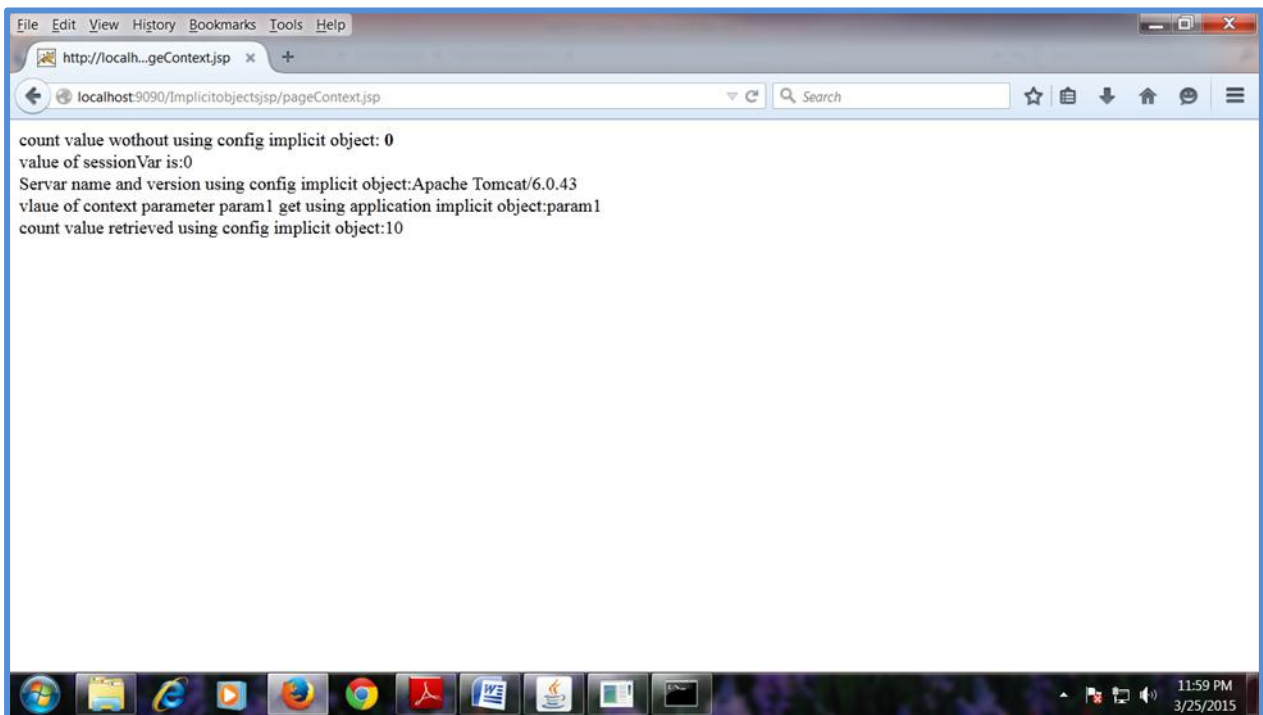
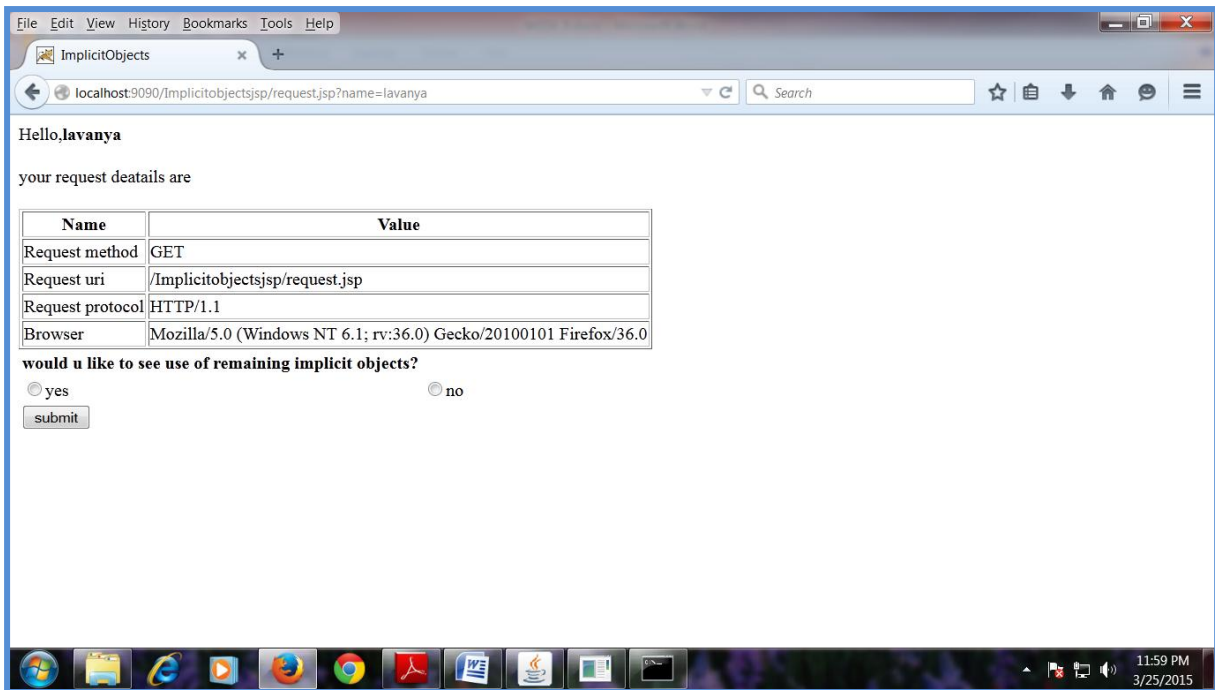
</servlet-mapping>

</web-app>

Output:

```
C:\Windows\system32\cmd.exe
D:\html programs\jdbc>cd..
D:\html programs>cd..
D:\>cd tomcatapps
D:\tomcatapps>cd Implicitobjectsjsp
D:\tomcatapps\Implicitobjectsjsp>tree/f
Folder PATH listing
Volume serial number is 00740063 5A0F:F19E
D:.
  Home.html
  pageContext.jsp
  request.jsp
  WEB-INF
    web.xml
  classes
  jsppages
  other.jsp
D:\tomcatapps\Implicitobjectsjsp>
```





WEEK11:

AIM:

Write a program to implement various Session tracking techniques using

a. Url Rewriting:

Index.html:

```
<form action="servlet1">
```

```
Name:<input type="text" name="userName"/><br/>
```

```
<input type="submit" value="go"/>
```

```
</form>
```

FirstServlet.java:

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

```
import javax.servlet.*;
```

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

```
public class FirstServlet extends HttpServlet {
```

```
public void doGet(HttpServletRequest request,  
HttpServletResponse response){
```

```
try{
```

```
response.setContentType("text/html");
```



```
PrintWriter out = response.getWriter();
String n=request.getParameter("userName");
out.print("Welcome "+n);
//appending the username in the query string
out.print("<ahref='servlet2?uname="+n+"'>visit</a>");
out.close();
}catch(Exception e){System.out.println(e);}
}
}
```

SecondServlet.java:

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

public class SecondServlet extends HttpServlet {

public void doGet(HttpServletRequest request,
HttpServletResponse response){

try{

response.setContentType("text/html");
```

```
PrintWriter out = response.getWriter();  
  
//getting value from the query string  
String n=request.getParameter("uname");  
out.print("Hello "+n);  
out.close();  
  
}catch(Exception e){System.out.println(e);}  
  
}
```

Web.xml:

```
<web-app>  
  
<servlet>  
  
<servlet-name>s1</servlet-name>  
  
<servlet-class>FirstServlet</servlet-class>  
  
</servlet>  
  
<servlet-mapping>  
  
<servlet-name>s1</servlet-name>  
  
<url-pattern>/servlet1</url-pattern>  
  
</servlet-mapping>  
  
<servlet>
```

```
<servlet-name>s2</servlet-name>
```

```
<servlet-class>SecondServlet</servlet-class>
```

```
</servlet>
```

```
<servlet-mapping>
```

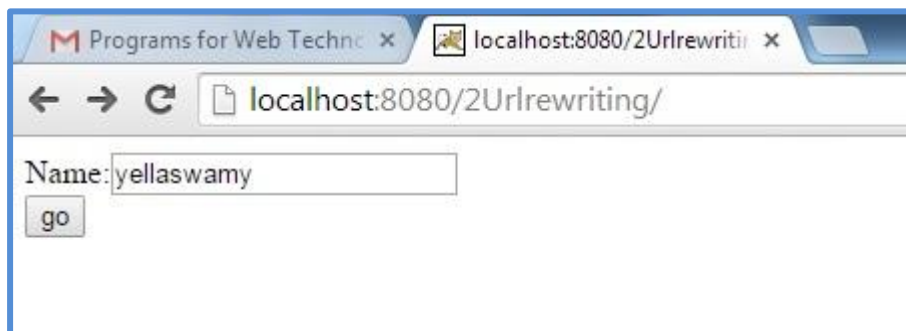
```
<servlet-name>s2</servlet-name>
```

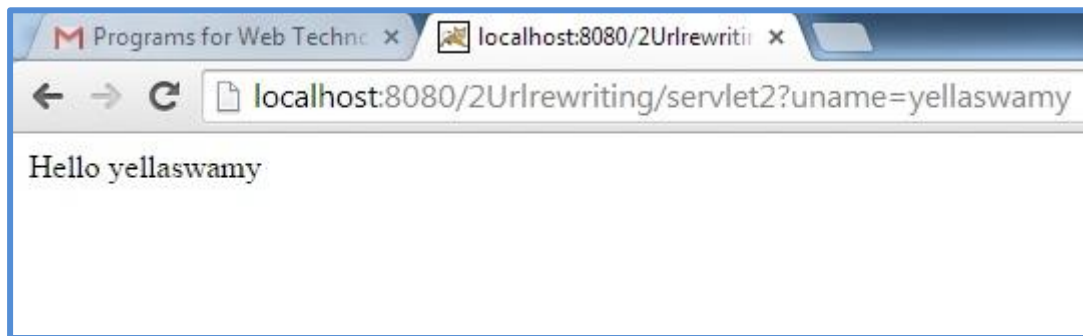
```
<url-pattern>/servlet2</url-pattern>
```

```
</servlet-mapping>
```

```
</web-app>
```

Output:





b. Hidden Form Fields:

Index.html:

```
<form action="go" method="get">
```

```
Name:<input type="text" name="uname"/><br/>
```

```
<input type="submit" value="go"/>
```

```
</form>
```

FirstServlet1.java:

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

```
import javax.servlet.*;
```

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

```
public class FirstServlet1 extends HttpServlet {
```

```
public void doGet(HttpServletRequest request,
HttpServletRequest response){
try{
response.setContentType("text/html");
PrintWriter out = response.getWriter();
String n=request.getParameter("uname");
out.print("Welcome "+n);
//creating form that have invisible textfield
out.print("<form action='welcome'>");
/*out.print("<input type='hidden' name='uname'
value='"+n+"'>");*/
out.print("<input type='hidden' name='pass'
value='CMRCET_CSEA'>");
out.print("<input type='submit' value='go'>");
out.print("</form>");
out.close();
}catch(Exception e){System.out.println(e);}
}
}
```

SecondServlet.java:

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class SecondServlet extends HttpServlet {
public void doGet(HttpServletRequest request,
HttpServletResponse response){
try{
response.setContentType("text/html");
PrintWriter out = response.getWriter();
//Getting the value from the hidden field
//String n=request.getParameter("uname");
String pass1=request.getParameter("pass");
out.print("Password from Servlet1 "+pass1);
out.close();
}catch(Exception e){System.out.println(e);}
}
}
```

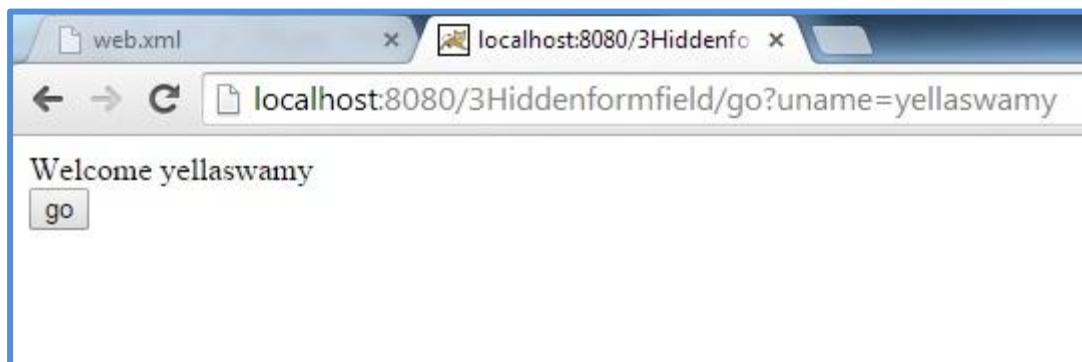
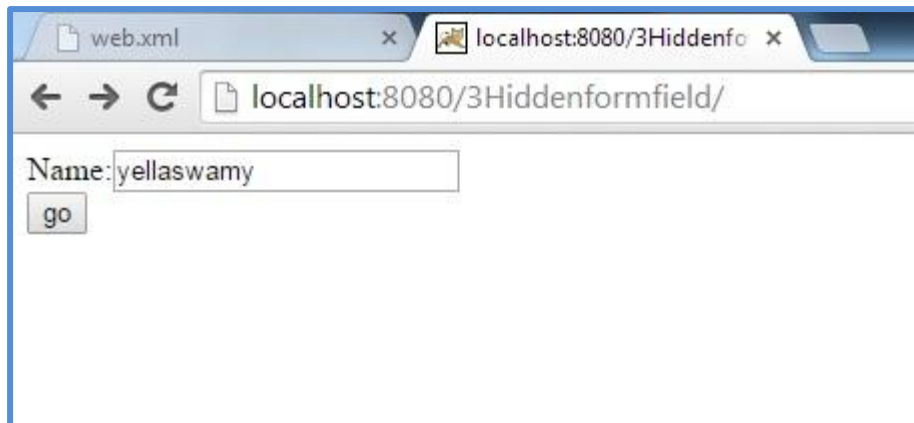
Web.xml:

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

```
<web-app version="2.5"
xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">
<servlet>
<description>This is the description of my J2EE
component</description>
<display-name>This is the display name of my J2EE
component</display-name>
<servlet-name>Servlet1</servlet-name>
<servlet-class>FirstServlet1</servlet-class>
</servlet>
<servlet>
<description>This is the description of my J2EE
component</description>
<display-name>This is the display name of my J2EE
component</display-name>
<servlet-name>SecondServlet</servlet-name><servlet-
class>SecondServlet</servlet-class>
```

```
</servlet>
<servlet-mapping>
<servlet-name>Servlet1</servlet-name>
<url-pattern>/go</url-pattern>
</servlet-mapping>
<servlet-mapping>
<servlet-name>SecondServlet</servlet-name>
<url-pattern>/welcome</url-pattern>
</servlet-mapping>
<welcome-file-list>
<welcome-file>index.html</welcome-file>
</welcome-file-list>
</web-app>
```

Output:



WEEK12:

AIM:

Write a program to implement various Session tracking techniques using

a. **Http Session:**

Index.html:

```
<form action="servlet1">
```

```
Name:<input type="text" name="userName"/><br/>
```

```
<input type="submit" value="go"/>
```

```
</form>
```

FirstServlet.java:

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

```
import javax.servlet.*;
```

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

```
public class FirstServlet extends HttpServlet {
```

```
public void doGet(HttpServletRequest request,  
HttpServletResponse response){
```

```
try{
```

```
response.setContentType("text/html");
PrintWriter out = response.getWriter();
String n=request.getParameter("userName");
out.print("Welcome "+n);
HttpSession session=request.getSession();
session.setAttribute("uname",n);
out.print("<a href='servlet2'>visit</a>");
out.close();
} catch(Exception e){ System.out.println(e);}
}
}
```

SecondSevlet.java:

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

```
response.setContentType("text/html");  
PrintWriter out = response.getWriter();  
HttpSession session=request.getSession(false);  
String n=(String)session.getAttribute("uname");  
out.print("Hello "+n);  
out.close();  
}catch(Exception e){System.out.println(e);}  
}  
}
```

Web.xml:

```
<web-app>  
<servlet>  
<servlet-name>s1</servlet-name>  
<servlet-class>FirstServlet</servlet-class>  
</servlet>  
<servlet-mapping>  
<servlet-name>s1</servlet-name>  
<url-pattern>/servlet1</url-pattern>  
</servlet-mapping>
```

```
<servlet>
```

```
<servlet-name>s2</servlet-name>
```

```
<servlet-class>SecondServlet</servlet-class>
```

```
</servlet>
```

```
<servlet-mapping>
```

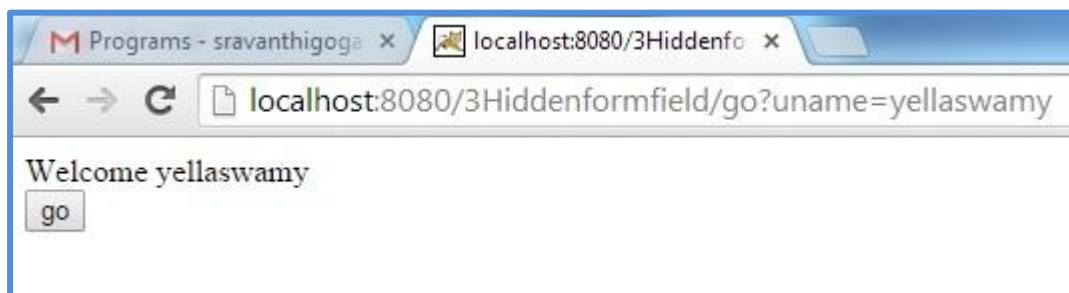
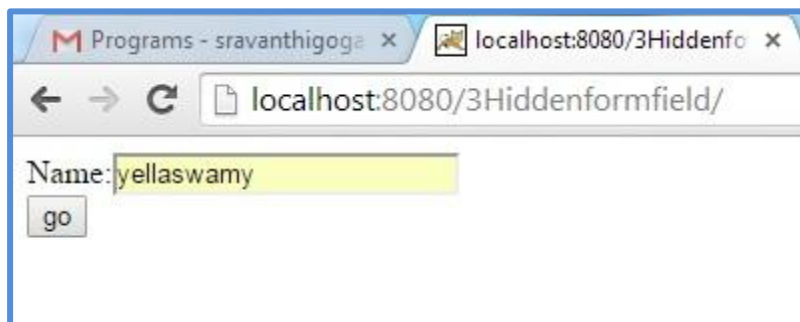
```
<servlet-name>s2</servlet-name>
```

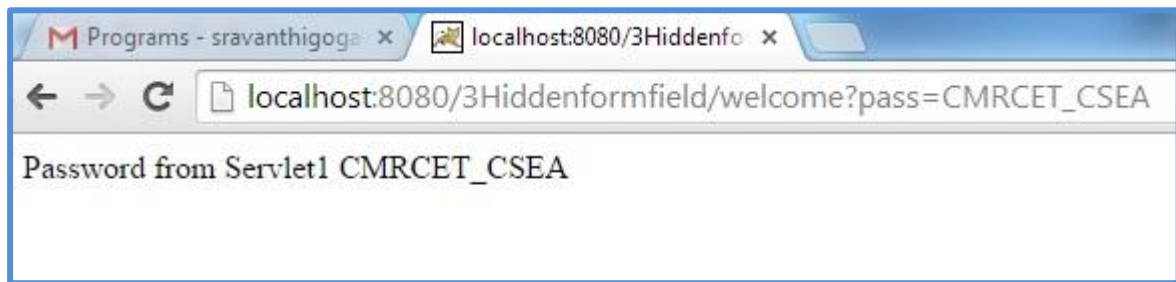
```
<url-pattern>/servlet2</url-pattern>
```

```
</servlet-mapping>
```

```
</web-app>
```

Output:





b. Cookies:

Index.html:

```
<form action="servlet1" method="post">
```

```
Name:<input type="text" name="userName"/><br/>
```

```
<input type="submit" value="go"/>
```

```
</form>
```

FirstServlet.java:

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

```
import javax.servlet.*;
```

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

```
public class FirstServlet extends HttpServlet {
```

```
public void doPost(HttpServletRequest request,  
HttpServletResponse response){
```

```
try{
```

```
response.setContentType("text/html");  
    PrintWriter out = response.getWriter();  
    String n=request.getParameter("userName");  
    out.print("Welcome "+n);  
    Cookie ck=new Cookie("uname",n);//creating cookie  
    object  
    response.addCookie(ck);//adding cookie in the  
    response  
    //creating submit button  
    out.print("<form action='servlet2'>");  
    out.print("<input type='submit' value='go'>");  
    out.print("</form>");  
    out.close();  
    }catch(Exception e){System.out.println(e);}  
    }  
    }
```

SecondServlet.java:

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class SecondServlet extends HttpServlet {
public void doPost(HttpServletRequest request,
HttpServletResponse response){
try{
response.setContentType("text/html");
PrintWriter out = response.getWriter();
Cookie ck[]=request.getCookies();
out.print("Hello "+ck[0].getValue());
out.close();
}catch(Exception e){System.out.println(e);}
}
}
```

Web.xml:

```
<web-app>
<servlet>
<servlet-name>s1</servlet-name>
```



```
<servlet-class>FirstServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>s1</servlet-name>
<url-pattern>/servlet1</url-pattern>
</servlet-mapping>
<servlet>
<servlet-name>s2</servlet-name>
<servlet-class>SecondServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>s2</servlet-name>
<url-pattern>/servlet2</url-pattern>
</servlet-mapping>
</web-app>
```

Output:

