WEB TECHNOLOGIES

LABORATORY MANUAL

B.TECH
(III YEAR – I SEM)
(2018-2019)

DEPARTMENT OF INFORMATION TECHNOLOGY

MALLA REDDY COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous Institution – UGC, Govt. of India)

Recognized under 2(f) and 12 (B) of UGC ACT 1956

(Affiliated to JNTUH, Hyderabad, Approved by AICTE - Accredited by NBA & NAAC – 'A' Grade - ISO 9001:2015 Certified)

Maisammaguda, Dhulapally (Post Via. Hakimpet), Secunderabad – 500100, Telangana State, India

DEPARTMENT OF INFORMATION TECHNOLOGY

VISION

- To improve the quality of technical education that provides efficient software engineers with an attitude to adapt challenging IT needs of local, national and international arena, through teaching and interaction with alumni and industry.

MISSION

- Department intends to meet the contemporary challenges in the field of IT and is playing a vital role in shaping the education of the 21st century by providing unique educational and research opportunities.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PEO1 – ANALYTICAL SKILLS

To facilitate the graduates with the ability to visualize, gather information, articulate, analyze, solve complex problems, and make decisions. These are essential to address the challenges of complex and computation intensive problems increasing their productivity.

PEO2 – TECHNICAL SKILLS

To facilitate the graduates with the technical skills that prepare them for immediate employment and pursue certification providing a deeper understanding of the technology in advanced areas of computer science and related fields, thus encouraging to pursue higher education and research based on their interest.

PEO3 – SOFT SKILLS

To facilitate the graduates with the soft skills that include fulfilling the mission, setting goals, showing self-confidence by communicating effectively, having a positive attitude, get involved in team-work, being a leader, managing their career and their life.

PEO4 – PROFESSIONAL ETHICS

To facilitate the graduates with the knowledge of professional and ethical responsibilities by paying attention to grooming, being conservative with style, following dress codes, safety codes, and adapting themselves to technological advancements.

PROGRAM SPECIFIC OUTCOMES (PSOs)

After the completion of the course, B. Tech Information Technology, the graduates will have the following Program Specific Outcomes:

1. **Fundamentals and critical knowledge of the Computer System:-** Able to Understand the working principles of the computer System and its components , Apply the knowledge to build, asses, and analyze the software and hardware aspects of it .

2. **The comprehensive and Applicative knowledge of Software Development:** Comprehensive skills of Programming Languages, Software process models, methodologies, and able to plan, develop, test, analyze, and manage the software and hardware intensive systems in heterogeneous platforms individually or working in teams.

3. **Applications of Computing Domain & Research:** Able to use the professional, managerial, interdisciplinary skill set, and domain specific tools in development processes, identify the research gaps, and provide innovative solutions to them.

PROGRAM OUTCOMES (POs)

Engineering Graduates should possess the following:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design / development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multi disciplinary environments.
12. **Life- long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

MALLA REDDY COLLEGE OF ENGINEERING & TECHNOLOGY

Maisammaguda, Dhulapally Post, Via Hakimpet, Secunderabad – 500100

DEPARTMENT OF INFORMATION TECHNOLOGY

GENERAL LABORATORY INSTRUCTIONS

1. Students are advised to come to the laboratory at least 5 minutes before (to the starting time), those who come after 5 minutes will not be allowed into the lab.
2. Plan your task properly much before to the commencement, come prepared to the lab with the synopsis / program / experiment details.
3. Student should enter into the laboratory with:

a. Laboratory observation notes with all the details (Problem statement, Aim, Algorithm, Procedure, Program, Expected Output, etc.,) filled in for the lab session.

b. Laboratory Record updated up to the last session experiments and other utensils (if any) needed in the lab.

c. Proper Dress code and Identity card.

4. Sign in the laboratory login register, write the TIME-IN, and occupy the computer system allotted to you by the faculty.
5. Execute your task in the laboratory, and record the results / output in the lab observation note book, and get certified by the concerned faculty.
6. All the students should be polite and cooperative with the laboratory staff, must maintain the discipline and decency in the laboratory.
7. Computer labs are established with sophisticated and high end branded systems, which should be utilized properly.
8. Students / Faculty must keep their mobile phones in SWITCHED OFF mode during the lab sessions. Misuse of the equipment, misbehaviors with the staff and systems etc., will attract severe punishment.
9. Students must take the permission of the faculty in case of any urgency to go out; if anybody found loitering outside the lab / class without permission during working hours will be treated seriously and punished appropriately.
10. Students should LOG OFF/ SHUT DOWN the computer system before he/she leaves the lab after completing the task (experiment) in all aspects. He/she must ensure the system / seat is kept properly.

HEAD OF THE DEPARTMENT **PRINCIPAL**

INDEX

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.

1) Save as **homepage.html**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE> Home Page </TITLE>
</HEAD>
<frameset rows="21%,79%">
<frame src="frame1.html" scrolling="no" noresize="noresize"/>
<frameset cols="20%,80%">
<frame src="frame2.html" scrolling="no" noresize="noresize">
<frame src="home.html" noresize="noresize" name="result">
</frameset>
</frameset>
</HTML>
```

Save as **frame1.html**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE> frame-1 </TITLE>
<style type="text/css">
th
{
background-color:rgb(59,156,150);
font-size:25px;
font-family:Microsoft Sans Serif;
letter-spacing:7px;
word-spacing:10px;
text-align:center;
color:white;
}
td
{
background-color:rgb(59,156,150);
font-size:20px;
height:25px;
text-align:center;
}
td
{
```

```
border:1px solid white;
}
a:visited{color:#003333}
a:hover{color:white}
a:active{color:red}
a
{
text-decoration:none;
color:#003333;
}
body
{
background-color:rgb(59,156,150);
}
</style>
</HEAD>
<BODY>
<table border="0" width="100%">
<tr>
<th colspan="5" valign="bottom">
<marquee style="font-size:16px;font-family:Castellar;">
WELCOME TO MRCET</marquee><img src="kottam.png"/></th>
</tr>
<tr>
<td><a href="home.html" target="result"><b>HOME</b></a></td>
<td><a href="login.html" target="result"><b>LOGIN</b></a></td>
<td><a href="registration.html"
target="result"><b>REGISTRATION</b></a></td> <td><a href="catalouge.html"
target="result"><b>CATALOUGE</b></a></td> <td><a href="cart.html"
target="result"><b>CART</b></a></td> </tr>
</BODY>
</HTML>
```

Save as **frame2.html**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE> frame-2 </TITLE>
<style type="text/css">
body
{
background-color:#FFFFFF;
}
a
{
text-decoration:none;
font-size:25px;
font-weight:bold;
color:rgb(59,156,150);
}
ul
{
list-style-image:url('bullet.png');
position:relative;
left:20px;
}
a:link{color:rgb(59,156,150)}
```

```
a:visited{color:#003333}
a:hover{color:#C0C0C0}
a:active{color:red}
</style>
</HEAD>
<BODY>
<ul>
<table border="0">
<tr><td>
<li><a href="cse.html" target="result"
>CSE</a><br/></li></td></tr> <tr><td>
<li><a href="it.html" target="result">IT</a><br/></li></td></tr>
<tr><td>
<li><a href="ece.html"
target="result">ECE</a><br/></li></td></tr> <tr><td>
<li><a href="mech.html" target="result">MECH</a><br/></li></td></tr>
</table>
</ul>
<!-- <p><marquee>New Updates</marquee></p-->
</BODY>
</HTML>
```

Save as **home.html**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title> home </title>
<style type="text/css">
p
{
color:rgb(59,156,150);
text-align:center;
}
</style>
</head>

<body>
<br/><hr/>
<p style="text-align:justify">Malla Reddy College of Engineering & Technology (MRCET) is a
constituent College of Malla Reddy Group of Institutions (MRGI) established in 2004, Approved
by AICTE, New Delhi and Affiliated to JNT University, Hyderabad. <br/> <br/>

The College offers Engineering courses (B.Tech.) in ECE, CSE, AE, and IT disciplines and Post
Graduate (M.Tech.) courses in ECE, CSE and AE specializations in addition to MBA and MCA
programmes. The College campus with its aboriculture and horticulture is situated away from the
hustle and bustle of the city provides a serene and tranquil atmosphere, which is conducive to both
teaching and learning processes. </p>
<hr/>
</body>
</html>
```

2) Save as **login.html**

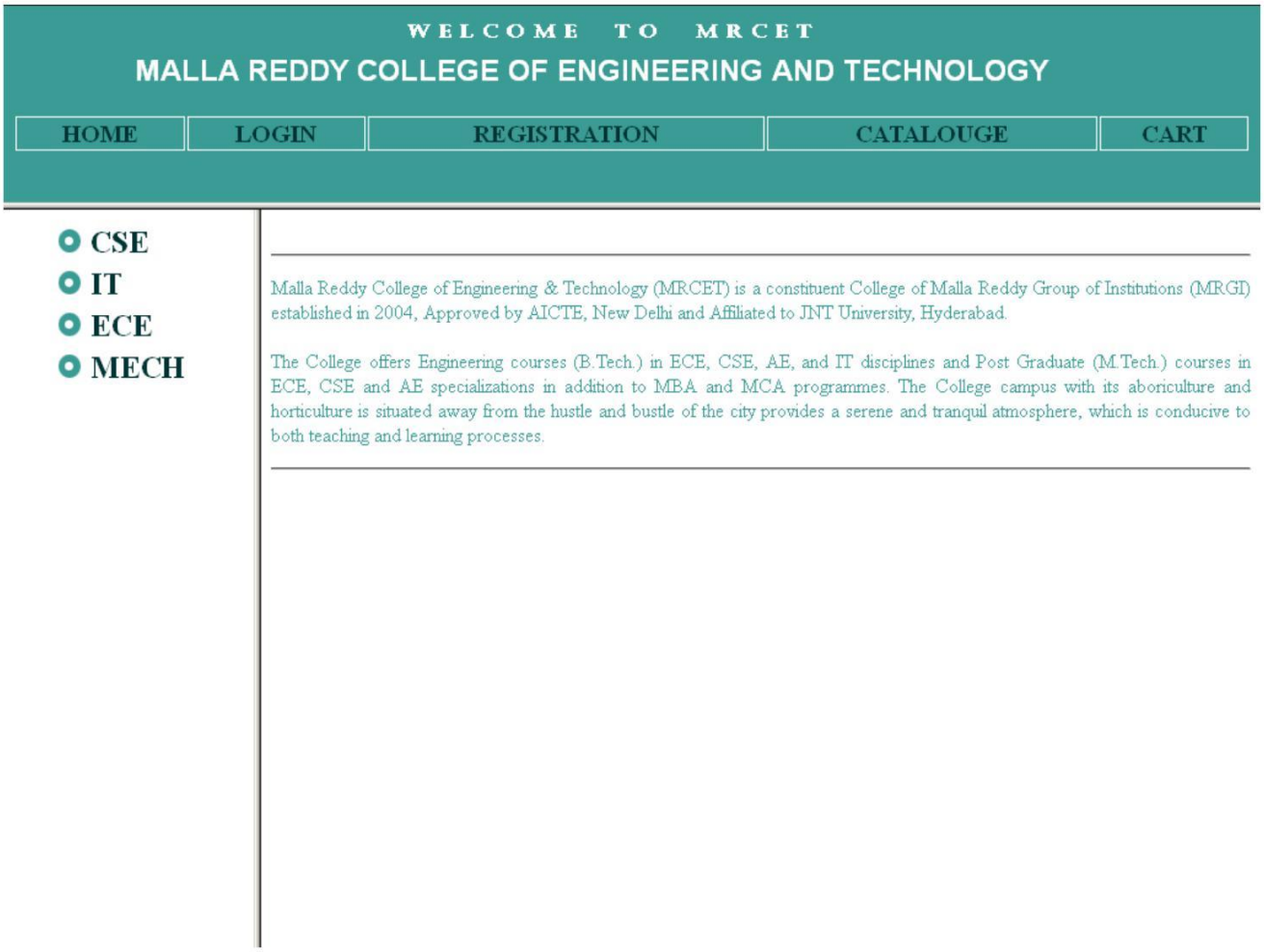
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title> login </title>
```

```
<style type="text/css">
div
{
left:20px;
}
td
{
color:rgb(59,156,150);
}
</style>
</head>
<body>
<form action="http://www.google.co.in" method="post">
<div>
<br/><br/>
<div align="center">
<table border="0">
<tr>
<td>Login:</td>
<td><input type="text" name="login_name"></td>
</tr>
<tr>
<td>Password:</td>
<td><input type="password" name="password"></td>
</tr>
<tr>
<td><input type="submit" name="submit"
value="Login"></td> <td><input type="reset"
name="reset"></td> </tr>
</table>
</div>
</body>
</html>
```

3) Save as **catalouge.html**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title> catalouge </title>
<style type="text/css">
table,tr,td,th
{
border:1px solid black;
color:rgb(59,156,150);
}
b{color:black}
.bs
{
width:120px;
height:30px;
}
</style>
</HEAD>
<BODY>
<table width="100%">
<tr>
<th width="25%">Snap Cover</th>
```

```
<th width="25%">Details</th>
<th width="25%">Cost</th>
<th width="25%">Select</th>
</tr>

<tr div align="center">
<td><img src="javascript.jpg"></div></td>
<td><p><b>Book:</b>Head First Java Script</p>
<p><b>Author:</b>Kathy Sierra and Bert
Bates</p> <p><b>Publication:</b>O'Reilly
Media</p></td> <td>$4.5</td>
<td><input class="bs" type="button" name="b1" value="Add to
cart"/></td> </tr>


<tr div align="center">
<td><img src="servlets.jpg"></td>
<td><p><b>Book:</b>Head First Servlets</p>
<p><b>Author:</b> Bryan Basham, Kathy Sierra and Bert Bates</p>
<p><b>Publication:</b>O'Reilly Media</p>
</td>
<td>$4.5</td>
<td><input class="bs" type="button" name="b2" value="Add to
cart"/></td> </tr>

<tr div align="center">
<td><img src="php&mysql.jpg"
alt="php&mysql"></div></td> <td><p><b>Book:</b>Head
First PHP & MYSQL</p> <p><b>Author:</b>Lynn Beighley
Michael Morrison </p> <p><b>Publication:</b>O'Reilly
Media</p></td> <td>$4.5</td>
<td><input class="bs" type="button" name="b3" value="Add to cart"/></td>
</tr>
<tr div align="center">
<td><img src="webdesign.jpg"></div></td>
<td><p><b>Book:</b>Head First
WebDesign</p> <p><b>Author:</b>Ethan
Watrall </p> <p><b>Publication:</b>O'Reilly
Media</p></td> <td>$4.5</td>
<td><input class="bs" type="button" name="b4" value="Add to
cart"/></td> </tr>
<tr div align="center">
<td><img src="csharp.jpg"></div></td>
<td><p><b>Book:</b>Head First CSharp</p>
<p><b>Author:</b>Andrew Stellman </p>
<p><b>Publication:</b>O'Reilly Media</p></td>
<td>$4.5</td>
<td><input class="bs" type="button" name="b5" value="Add to cart"/></td>
</tr>
</table>
</body>
</html>
```



Exercise

I) Develop static pages of any company using only HTML. The website should consists the following pages

1. Home page
2. Registration
3. User login
4. User Profile Page
5. Product catalog

Signature of the Faculty

WEEK-2

4) Save as **cart.html**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title> cart </title>
<style type="text/css">
table,tr,th,td
{
border:1px solid black;
color:rgb(59,156,150);
}
</style>
</head>
<body>
<table border="1" width="100%">
<tr><th>Book Name</th> <th>Price</th> <th>Quantity</th><th>Amount</th>
</tr>
<tr><td>Head First Java Script</td><td>$4.5</td>
<td>2</td><td>$9</td>
</tr>
<tr><td>Head First PHP & MYSQL</td><td>$4.5</td>
<td>1</td><td>$4.5</td>
</tr>
<tr>
<td colspan="3" align="right"> Total Amount</td>
<td>13.5</td>
</tr>
</table>
</body>
</html>
```

5) Save as **registration.html**

```
<html>
<head><title>Registration Page</title>
<style type="text/css">
b
{
color:rgb(59,156,150);
}
</style>
</head>
<body>
<h1 align="center">Registration</h1>
<form name="demo" onsubmit="return validateForm(this)" action="#">

<div align="center">
<table>
<tr>
<td><b>User Name:</b></td>
<td><input type="text" name="username" size="35"></td>
</tr>
<tr>
```

```
<td><b>Password:</b></td>
<td><input type="password" name="pwd" size="35"></td>
</tr>
<tr>
<td><b>E-mail:</b></td>
<td><input type="text" name="email" size="35"></td>
</tr>
<tr>
<td><b>Mobile number:</b></td>
<td><input type="text" name="phone" size="35"></td>
</tr>
<tr>
<td><b>Gender:</b></td>
<td><input type="radio" name="gender"/>Male</td>
</tr>
<tr>
<td></td>
<td><input type="radio" name="gender"/>Female</td>
</tr>
<tr>
<td><b>Date of Birth:</b></td>
<td>
<select>
<option selected>Date</option>
<option>1</option><option>2</option><option>3</option><option>4</option>
<option>5</option><option>6</option><option>7</option><option>8</option>
<option>9</option><option>10</option><option>11</option><option>12</option>
<option>13</option><option>14</option><option>15</option><option>16</option>
<option>17</option><option>18</option><option>19</option><option>20</option>
<option>21</option><option>22</option><option>23</option><option>24</option>
<option>25</option><option>26</option><option>27</option><option>28</option>
<option>29</option><option>30</option><option>31</option>
</select>
<select>
<option selected>Month</option>
<option>January</option><option>february</option><option>March</option>
<option>April</option><option>May</option><option>June</option><option>July</option>
<option>Augest</option><option>September</option><option>October</option>
<option>November</option><option>December</option>
</select>
<select>
<option selected>year</option>
<option>1988</option><option>1989</option>
<option>1990</option><option>1991</option>
<option>1992</option><option>1993</option>
<option>1994</option>
</select>
</td>
</tr>
<tr>
<td><b>Languages:</b></td>
<td><input type="checkbox" name="telugu"/>Telugu</td>
</tr>
<tr>
<td></td>
<td><input type="checkbox"
name="english"/>English</td> </tr>
```

```
<tr>
<td></td>
<td><input type="checkbox" name="hindi"/>Hindi</td>
</tr>
<tr>
<td><b>Address:<b></td>
<td><textarea cols="27" rows="5"></textarea></td>
</tr>
<tr>
<td> </td>
<td><input type="submit" name="Submit" value="Send"></td>
</tr>
</table>
</div>
</body>
</html>
```

Exercise

Design the following static web pages required for an online shopping store web site.
1) HOME PAGE:
The static home page must contain three frames.
Top frame: Logo and the company name and links to Home page, Login page, sign up page,
Left frame: At least four links for navigation, which will display the catalogue of respective links.(like Clothing & Accessories,Mobiles and Tablets, books,Computer & Accessories)
Right frame: The pages to the links in the left frame must be loaded here. Initially this page contains
description of the web site.

Signature of the Faculty

WEEK 3:

VALIDATION:

Write JavaScript to validate the following fields of the above registration page.
1. Name (Name should contains alphabets and the length should not be less than 6 (characters).
2. Password (Password should not be less than 6 characters length).
3. E-mail id (should not contain any invalid and must follow the standard pattern name@domain.com)
4. Phone number (Phone number should contain 10 digits only).

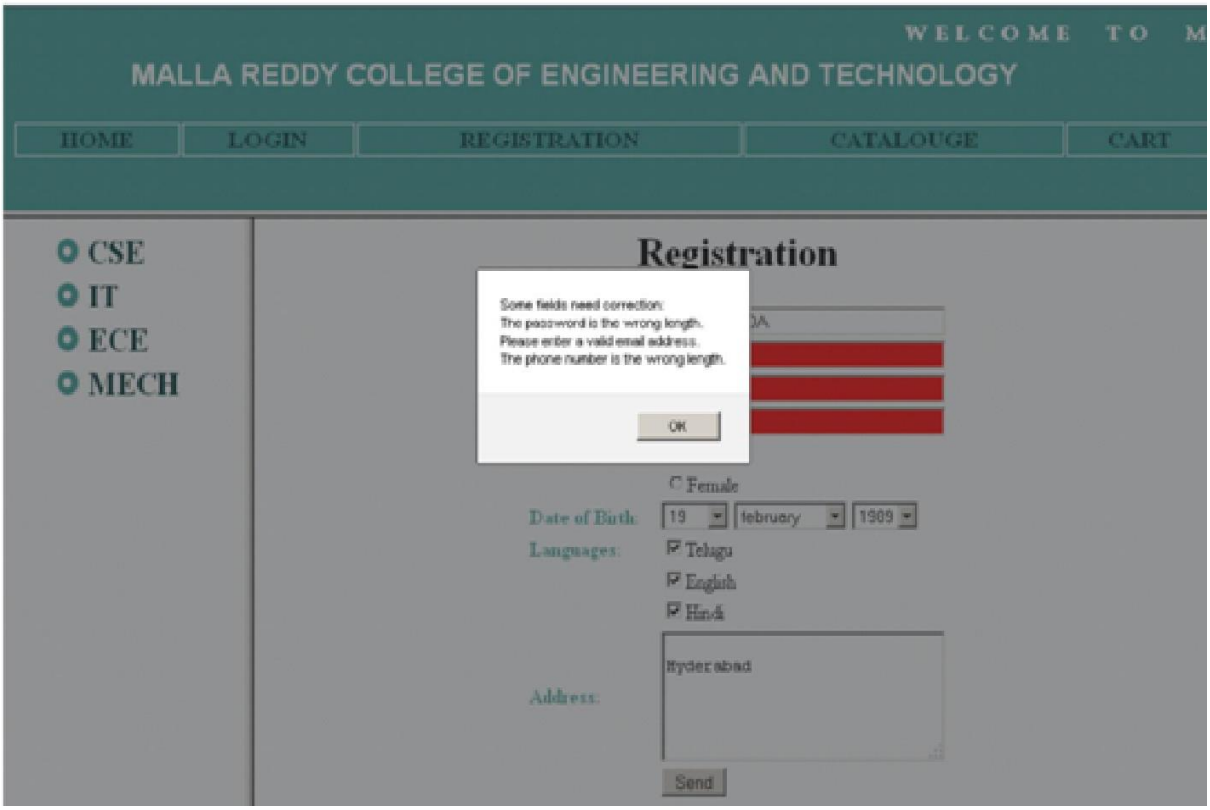
Save as **registration.html**

```
<html>
<head><title>Registration Page</title>
<style type="text/css">
b
{
color:rgb(59,156,150);
}
</style>
</head>
<body>
<h1 align="center">Registration</h1>
<form name="demo" onsubmit="return validateForm(this)" action="#">
<div align="center">
<table>
<tr>
<td><b>User Name:</b></td>
<td><input type="text" name="username" size="35"></td>
</tr>
<tr>
<td><b>Password:</b></td>
<td><input type="password" name="pwd" size="35"></td>
</tr>
<tr>
<td><b>E-mail:</b></td>
<td><input type="text" name="email" size="35"></td>
</tr>
<tr>
<td><b>Mobile number:</b></td>
<td><input type="text" name="phone" size="35"></td>
</tr>
<tr>
<td><b>Gender:</b></td>
<td><input type="radio" name="gender"/>Male</td>
</tr>
<tr>
<td></td>
<td><input type="radio" name="gender"/>Female</td>
</tr>
<tr>
<td><b>Date of Birth:</b></td>
<td>
```

```
<select>
<option selected>Date</option>
<option>1</option><option>2</option><option>3</option><option>4</option>
<option>5</option><option>6</option><option>7</option><option>8</option>
<option>9</option><option>10</option><option>11</option><option>12</option>
<option>13</option><option>14</option><option>15</option><option>16</option>
<option>17</option><option>18</option><option>19</option><option>20</option>
<option>21</option><option>22</option><option>23</option><option>24</option>
<option>25</option><option>26</option><option>27</option><option>28</option>
<option>29</option><option>30</option><option>31</option>
</select>
<select>
<option selected>Month</option>
<option>January</option><option>february</option><option>March</option>
<option>April</option><option>May</option><option>June</option><option>July</option>
<option>Augest</option><option>September</option><option>October</option>
<option>November</option><option>December</option>
</select>
<select>
<option selected>year</option>
<option>1988</option><option>1989</option>
<option>1990</option><option>1991</option>
<option>1992</option><option>1993</option>
<option>1994</option>
</select>
</td>
</tr>
<tr>
<td><b>Languages:</b></td>
<td><input type="checkbox" name="telugu"/>Telugu</td>
</tr>
<tr>
<td></td>
<td><input type="checkbox" name="english"/>English</td>
</tr>
<tr>
<td></td>
<td><input type="checkbox" name="hindi"/>Hindi</td>
</tr>
<tr>
<td><b>Address:<b></td>

<td><textarea cols="27" rows="5"></textarea></td>
</tr>
<tr>
<td> </td>
<td><input type="submit" name="Submit" value="Send"></td>
</tr>
</table>
</div>
</form>
<script language="javascript">
function validateForm(theform)
{
var reason = "";
reason += validateusername(theform.username);
```

```
reason += validatepassword(theform.pwd);
reason += validateemail(theform.email);
reason += validatephone(theform.phone);
if (reason != "")
{
alert("Some fields need correction:\n" + reason);
return false;
}
return true;
}
function validateusername(fld)
{
var error = "";
var illegalchars = new RegExp("^[a-zA-Z_.]+$");
if (fld.value == "")
{
fld.style.background = 'red';
error = "You didn't enter a username.\n";
}
else if ((fld.value.length < 6) || (fld.value.length > 18))
{
fld.style.background = 'red';
error = "The username is the wrong length.\n";
}
else if (!(illegalchars.test(fld.value)))
{
fld.style.background = 'red';
error = "The username contains illegal characters.\n";
}
else
{
fld.style.background = 'White';
}
return error;
}

function validatepassword(fld)
{
var error = "";
if (fld.value == "")
{
fld.style.background = 'red';
error = "You didn't enter a password.\n";
}
else if ((fld.value.length < 6) || (fld.value.length > 18))
{
fld.style.background = 'red';
error = "The password is the wrong length. \n";
}
else
{
fld.style.background = 'White';
}
return error;
}

function validateemail(fld)
{
```

```
var error="";
var sr = new RegExp("^[a-zA-Z0-9_.]+@[a-zA-Z]+\.[a-zA-Z]{2,3}$");
if (fld.value == "")
{
fld.style.background = 'red';
error = "You didn't enter an email address.\n";
}
else if (!(sr.test(fld.value)))
{
fld.style.background = 'red';
error = "Please enter a valid email address.\n";
}
else
{
fld.style.background = 'White';
}
return error;
}
function validatephone(fld)
{
var error = "";
if (fld.value == "")
{
error = "You didn't enter a phone number.\n";

fld.style.background = 'red';
}
else if (!(fld.value.length == 10))
{
error = "The phone number is the wrong
length.\n"; fld.style.background = 'red'; }

return error;
}
</script>
</body>
</html>
```



Exercise

1)Validate the Registration, User login, and payment by credit card pages using java script.

2)write a javascript which asks the user to enter two integers, obtains the numbers from the user and outputs html text that displays the larger number followed by the words"LARGER NUMBER"in an information message dialog.if the numbers are equal then output HTML text as"EQUAL NUMBRS".

Signature of the Faculty

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.

Save as **week4.html**

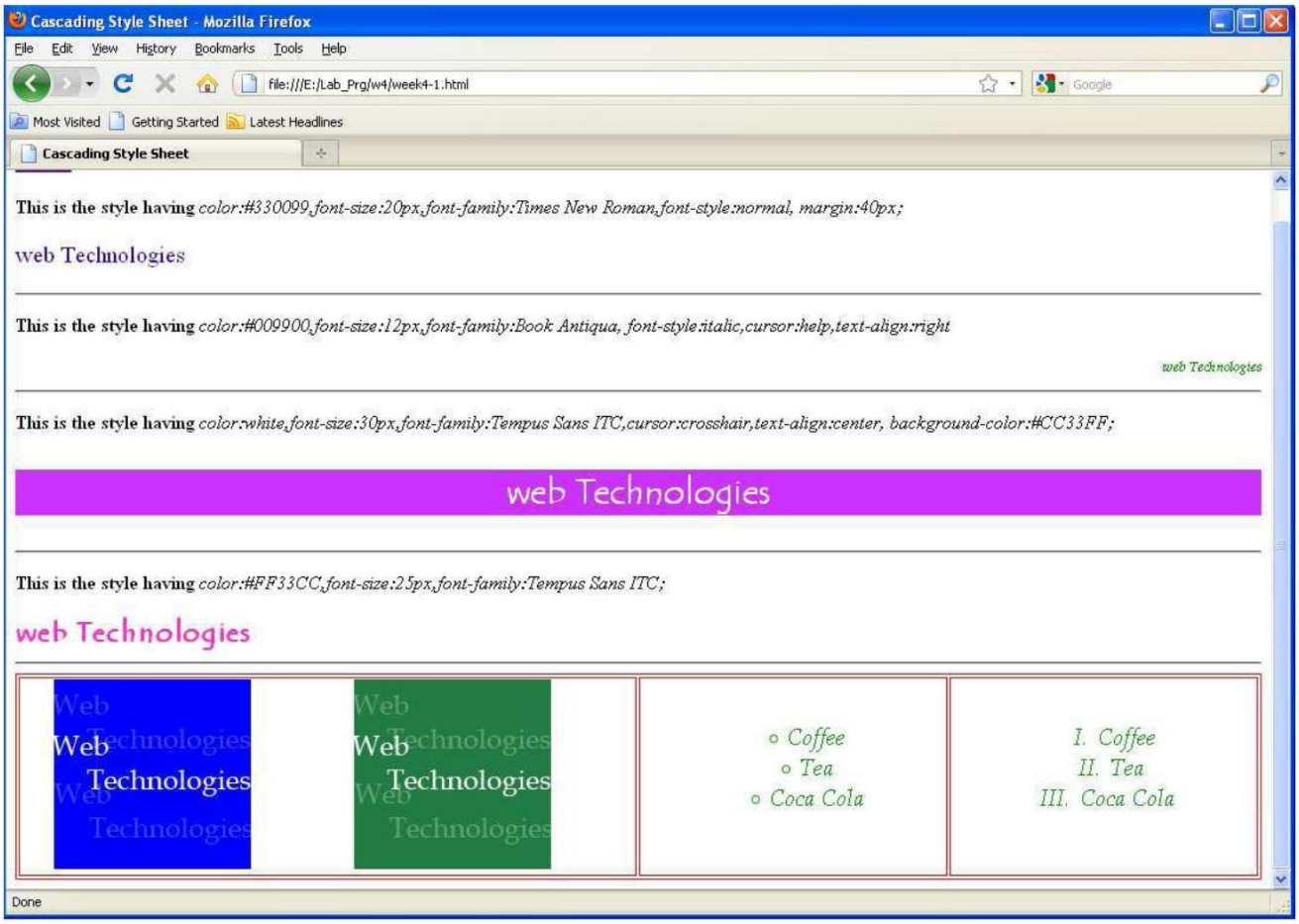
```
<html>
<head>
<title> week4 </title>
<style type="text/css">
a
{
text-decoration:none;
font-size:20px;
color:#006600;
}
</style>
</head>
<body>
<table border="0">
<tr><td>
<a href="week4-1.html">1) Use different font, styles</a>
</td></tr>
<tr><td>
<a href="week4-2.html">2) Set a background image for both the page and single elements on
the page</a>
</tr></td>
<tr><td>
<a href="week4-3.html">3) Control the repetition of the image with the background
property</a>
</tr></td>
<tr><td>
<a href="week4-4.html">4) Define styles for links as a:link a:visited a:active
a:hover</a>
</tr></td>
<tr><td>
<a href="week4-5.html">5) Work with layers </a>
</tr></td>
<tr><td>
<a href="week4-6.html">6) Add a customized cursor</a>
</tr></td>
</table>
</body>
</html>
```

Save as **week4-1.html**

```
<html>
<head>
<title> Cascading Style Sheet </title>
<style type="text/css">
```

```
.s1
{
color:#330099;
font-size:20px;
font-family:Times New Roman;
font-style:normal;
}
.s2
{
color:#009900;
font-size:12px;
font-family:Book Antiqua;
font-style:italic;
cursor:help;
text-align:right;
}
.s3
{
color:white;
font-size:30px;
font-family:Tempus Sans ITC;
cursor:crosshair;
text-align:center;
background-color:#CC33FF;
}
.s4
{
color:#FF33CC;
font-size:25px;
font-family:Tempus Sans ITC;
}
.img1
{
cursor:crosshair;
}
.img2
{
cursor:help;
}
ul.a {list-style-type:circle;}
ol.b {list-style-type:upper-roman;}


li
{
color:#009900;
font-size:20px;
font-family:Book Antiqua;
font-style:italic;
text-align:center;
}
table, td, th
{
border:1px solid red;
}
</style>
```

```
</head>
<body>
<h3><a href="week4.html">Home</a></h3>
<p><b>This is the style having</b> <i>color:#330099, font-size:20px, font-family:Times
New Roman, font-style:normal, margin:40px </i></p> <p class="s1">web
Technologies</p><hr/>
<p><b>This is the style having</b><i>color:#009900, font-size:12px, font-family:Book
Antiqua, font-style:italic, cursor:help, text-align:right</i></p> <p class="s2">web
Technologies</p><hr/>
<p><b>This is the style having</b><i>color:white,font-size:30px,font-family:Tempus Sans
ITC, cursor:crosshair, text-align:center, background-color:#CC33FF</i></p> <p
class="s3">web Technologies</p><hr/>
<p><b>This is the style having</b><i>color:#FF33CC,font-size:25px,font-family:Tempus
Sans ITC;</i>
</p><b class="s4">web Technologies</b><hr/>
<table border="0" width="100%">
<tr>
<td width="50%">
      
<img class="img1" src="blue.png"/>
            
      
<img class="img2" src="green.png"/>
</td>
<td width="25%">
<ul class="a">
<li>Coffee</li>
<li>Tea</li>
<li>Coca Cola</li>
</ul>
</td>
<td width="25%">
<ol class="b">

<li>Coffee</li>
<li>Tea</li>
<li>Coca Cola</li>
</ol>
</td>
</tr>
</table>
</body>
</html>
```



2) Set a background image for both the page and single elements on the

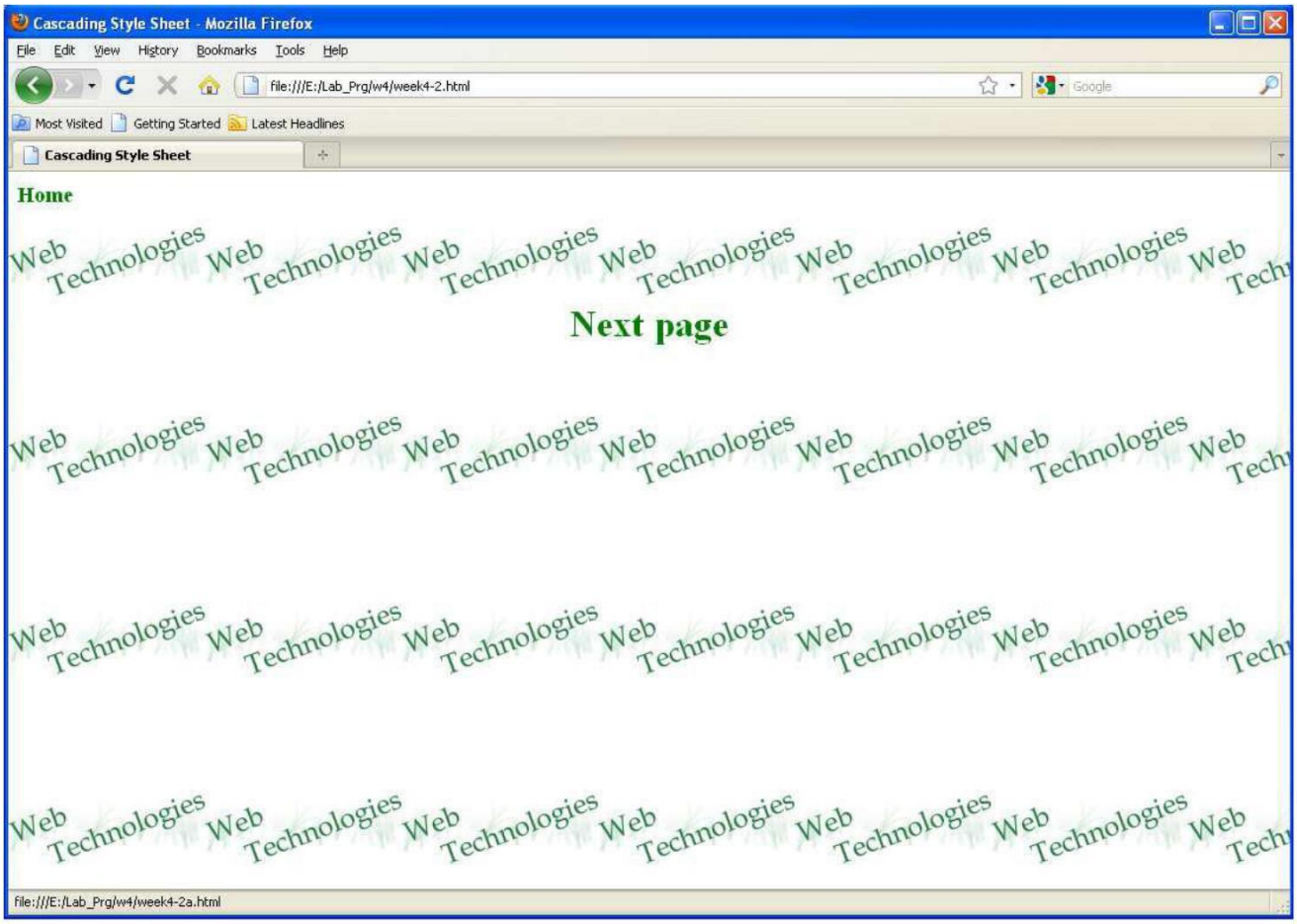
page. Save as **week4-2.html**

```
<html>
<head>
<title> Cascading Style Sheet </title>
<style type="text/css">
body
{
background-image:url(white1.png);
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>
<body>
<h3><a href="week4.html">Home</a></h3>
<a href="week4-2a.html"><center>Next page</center></a>
</body>
```

```
</html>
```

Save as **week4-2a.html**

```
<html>
<head>
<title> Cascading Style Sheet</title>
<style type="text/css">
body
{
background-image:url(white1.png);
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>
<body>
<a href="week4-2.html"><center>Home page</center></a>
</body>
</html>
```



3) Control the repetition of the image with the background-repeat property.
As background-repeat: repeat titles the image until the entire page is filled, just like an ordinary background image in plain HTML.

Save as **week4-3.html**

```
<html>
<head>
<title> Cascading Style Sheet </title>
<style type="text/css">
a
{
color:green;
text-decoration:none;
font-size:20px;
}
</style>
</head>
<body>

<h3><a href="week4.html">Home</a></h3>
<table border="0" width="100%">
<tr>
<td width="25%">

<a href="bgcolor.html">BackGround-Color</a>
</td>
</tr>
<tr>
<td width="25%">
<a href="bgimage.html">BackGround-Image</a>
</td>
</tr>
<tr>
<td width="25%">
<font color="red">BackGround-Repeat</font>
</td>
</tr>
<tr>
<td width="25%">
<a href="bgrepeatx.html">BackGround-Repeat-
X</a> </td>
<td width="25%">
<a href="bgrepeaty.html">BackGround-Repeat-
Y</a> </td>
<td width="25%">
<a href="bgnorepeaty.html">BackGround-NoRepeat</a>
</td>
</tr>
<tr>
<td width="25%">
<font color="red">BackGround-Attachment</font>
</td>
</tr>
<tr>
<td width="25%">
```

```
<a href="bgattachment-fixed.html">BackGround-Attachment-
Fixed</a> </td>
<td width="25%">
<a href="bgattachment-scroll.html">BackGround-Attachment-scroll</a>
</td>
</tr>
<tr>
<td width="25%">
<font color="red">BackGround-Position</font>
</td>
</tr>
<tr>
<td width="25%">

<a href="bgtopcenter.html">BackGround-Position Top
Center</a> </td>
<td width="25%">
<a href="bgptopright.html">BackGround-Position Top
Right</a> </td>
<td width="25%">
<a href="bgpxy.html">BackGround-Position Position-x(250) Position-y(300)</a>
</td>
</tr>
</table>
</body> </html>
```

Save as **bgcolor.html**

```
<html>
<head>
<title> bgcolor</title>
<style type="text/css">
body
{
background-color:green;
}
p
{
color:white;
font-size:20px;
text-align:center;
}
a
{
color:white;
text-decoration:none;
}
</style>
</head>
<body>
<a href="week4-3.html">Home</a>
<br/>
<p>This page having background color <i>green</i></p>
</body>
</html>
```

Save as **bgimage.html**

```
<html>
<head>
<title>bgimage</title>

<style type="text/css">
body
{
background-image:url(white2.png);
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>
<body>
<a href="week4-3.html">Home</a>
</body>
</html>
```

Save as bgrepeatx.html

```
<html>
<head>
<title> bgrepeatx </title>

<style type="text/css">
body
{
background-image:url(white2.png);
background-repeat:repeat-x;
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>
<body>
<a href="week4-3.html">Home</a>
</body>
</html>
```

Save as **bgrepeaty.html**

```
<html>
<head>
<title> bgrepeatx </title>
<style type="text/css">
body
{
background-image:url(white2.png);
background-repeat:repeat-y;
```

```
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>
<body>
<a href="week4-3.html">Home</a>
</body>
</html>
```

Save as **bgnorepeat.html**

```
<html>
<head>
<title> bgrepeatx </title>
<style type="text/css">
body
{
background-image:url(white2.png);
background-repeat:no-repeat;
}
a
{
color:green;

text-decoration:none;
}
</style>
</head>
<body>
<a href="week4-3.html">Home</a>
</body>
</html>
```

Save as **bgattachment-fixed.html**

```
<html>
<head>
<title> bgattachment</title>
<style type="text/css">
body
{
background-image:url(white2.png);
background-attachment:fixed;
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>
<body>
<a href="week4-3.html">Home</a>
<br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>
```

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

Save as **bgattachment-scroll.html**

```
<html>
<head>
<title> bgattachment</title>
<style type="text/css">
body
{
background-image:url(white2.png);
background-attachment:scroll;
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>

<body>
<a href="week4-3.html">Home</a>
<br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>
<br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>
<br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>
<br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>
<br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>
<br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>
</body>
</html>
```

Save as **bgtopcenter.html**

```
<html>
<head>
<title> bgtopcenter</title>
<style type="text/css">
body
{
background-image:url(white2.png);
background-repeat:no-repeat;
background-position:top center;
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>
```

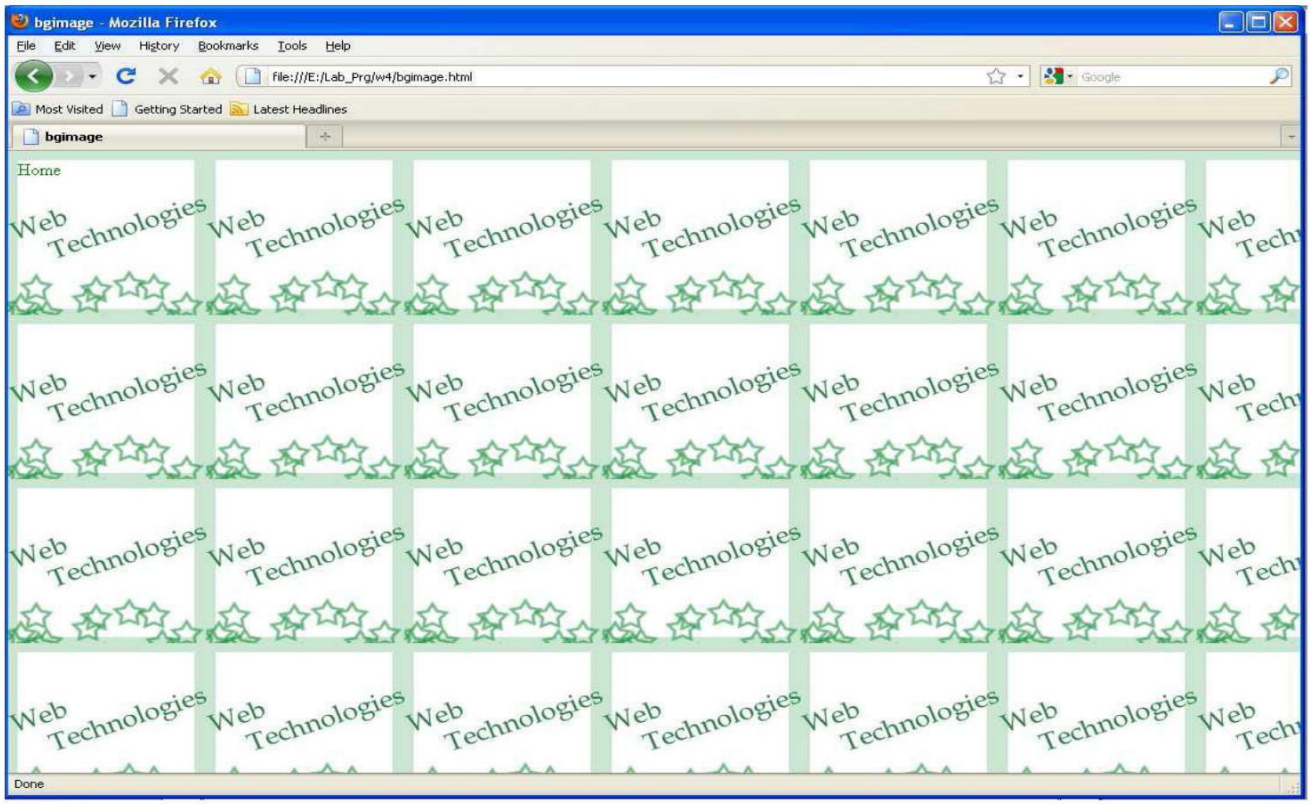
```
<body>
<a href="week4-3.html">Home</a>
</body>
</html>
```

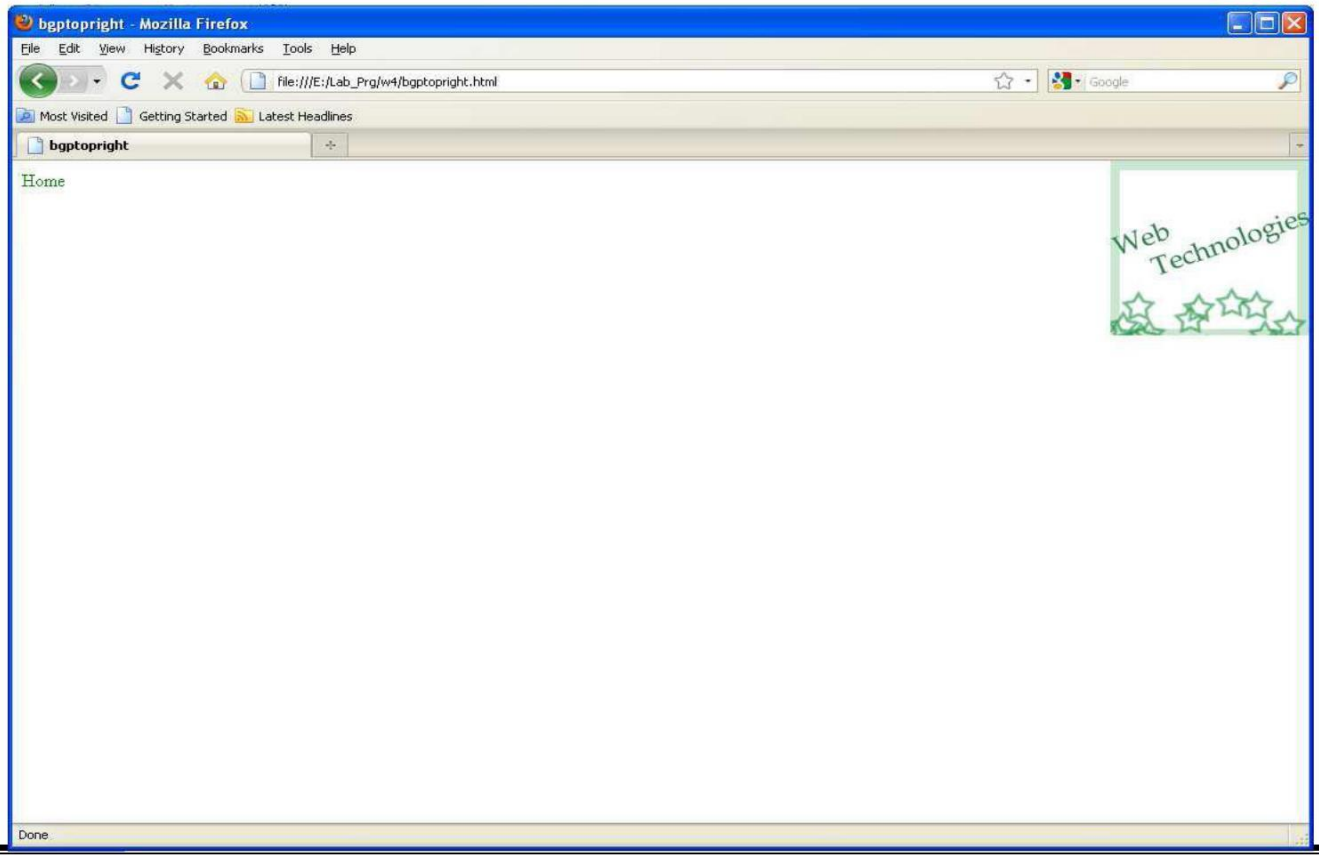
Save as **bgptopright.html**

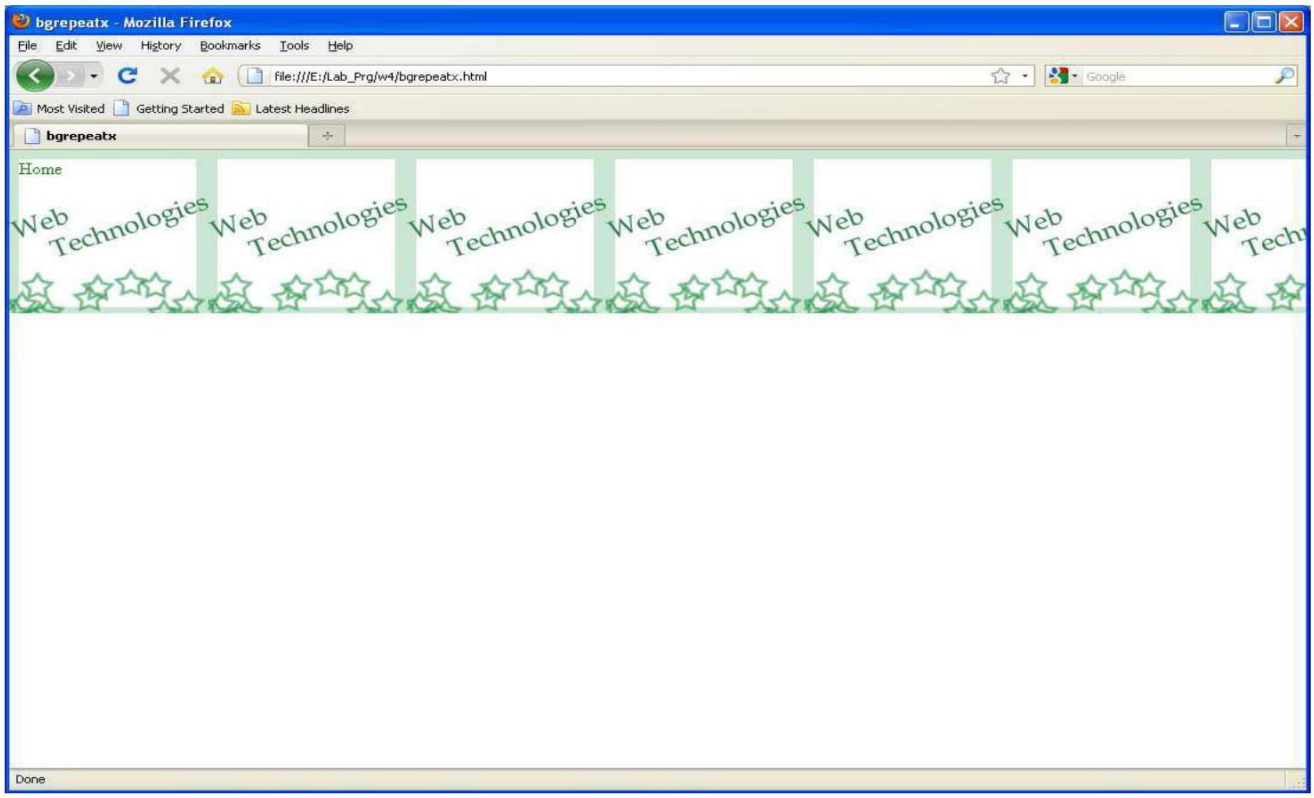
```
<html>
<head>
<title> bgptopright </title>
<style type="text/css">
body
{
background-image:url('white2.png');
background-repeat:no-repeat;
background-position:top right;
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>
<body>

<a href="week4-3.html">Home</a>
</body>
</html>
```

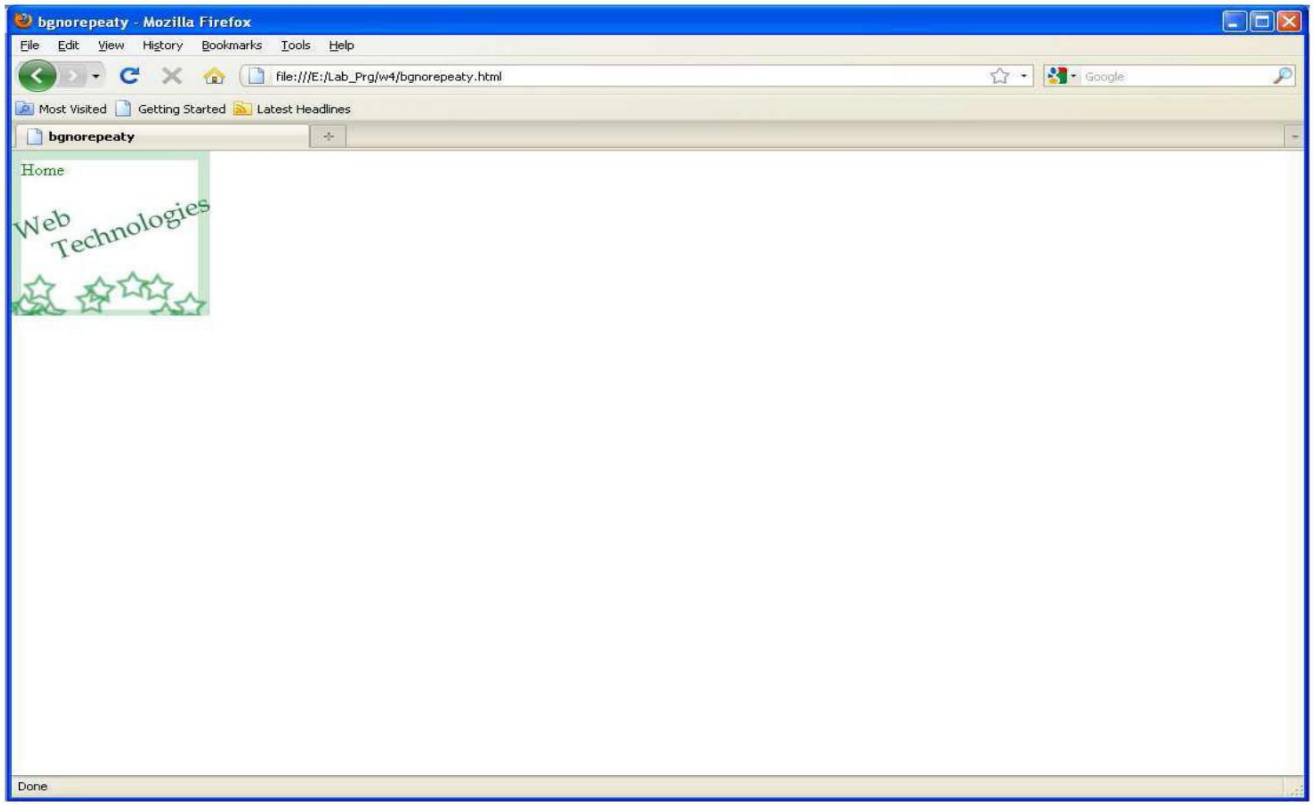
Save as **bgpxy.html**

```
<html>
<head>
<title> bgpbottomcenter </title>
<style type="text/css">
body
{
background-image:url(white2.png);
background-repeat:no-repeat;
background-position:250 300;
}
a
{
color:green;
text-decoration:none;
}
</style>
</head>
<body>
<a href="week4-3.html">Home</a>
</body>
</html>
```


bgimage - Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///E:/Lab_Prg/w4/bgimage.html
Most Visited Getting Started Latest Headlines
bgimage
Home
Web Technologies
Done


bgptopright - Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///E:/Lab_Prg/w4/bgptopright.html
Most Visited Getting Started Latest Headlines
bgptopright
Home
Web Technologies
Done

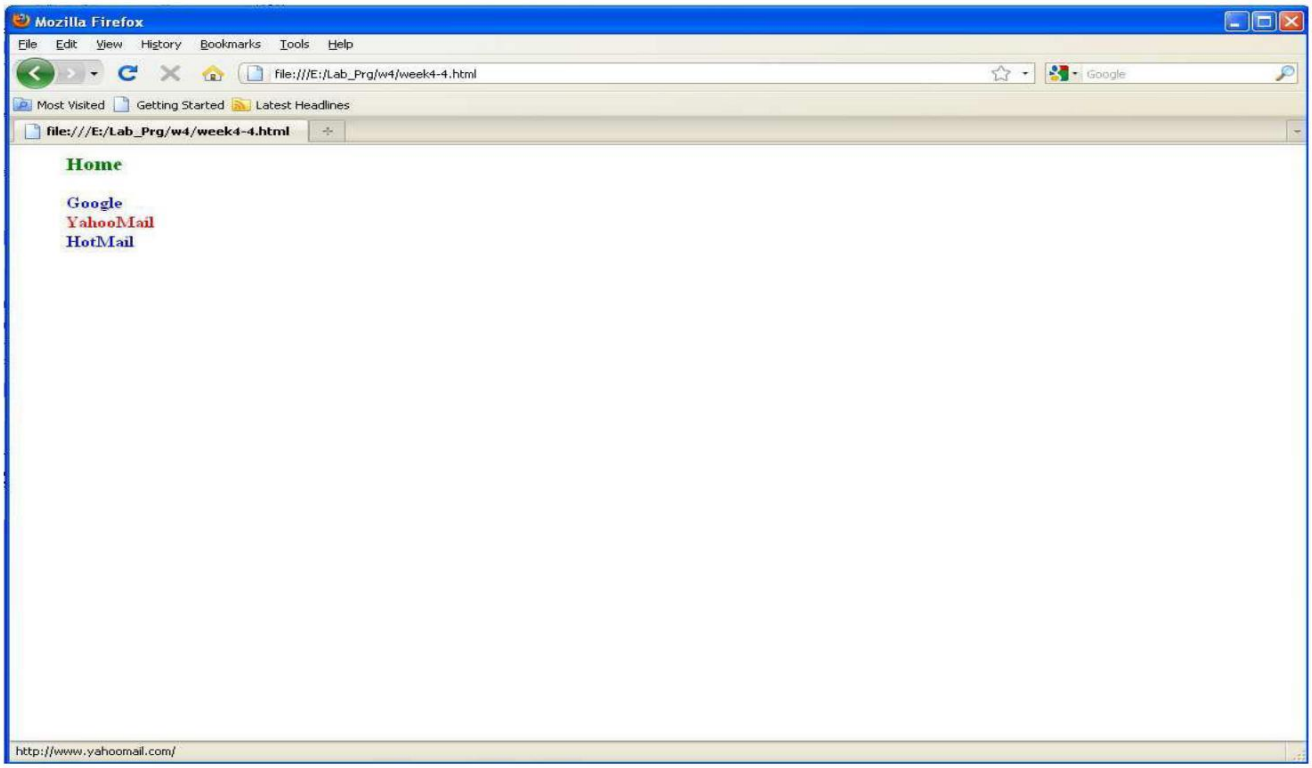

bgrepeatx - Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///E:/Lab_Prg/w4/bgrepeatx.html
Most Visited Getting Started Latest Headlines
bgrepeatx
Home
Web Technologies
Done


bgnorepeaty - Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///E:/Lab_Prg/w4/bgnorepeaty.html
Most Visited Getting Started Latest Headlines
bgnorepeaty
Home
Web Technologies
Done

4) Define styles for links as
a:link a:visited
a:active
a:hover

Save as **week4-4.html**

```
<html>
<head>
<style type="text/css">
a:link {color:blue} /* unvisited link */
a:visited {color:green} /* visited link */
a:hover {color:red} /* mouse over link */
a:active {color:yellow} /* selected link */
a
{
text-decoration:none;
margin:40px;
}
</style>
</head>
<body>
<h3><a href="week4.html">Home</a></h3>
<p><b>
<a href="http://www.google.co.in"
target="_blank">Google</a> <br/>
<a href="http://www.yahoomail.com"
target="_blank">YahooMail</a> <br/>
<a href="http://www.hotmail.com" target="_blank">HotMail</a>
</b></p>
</body>
</html>
```

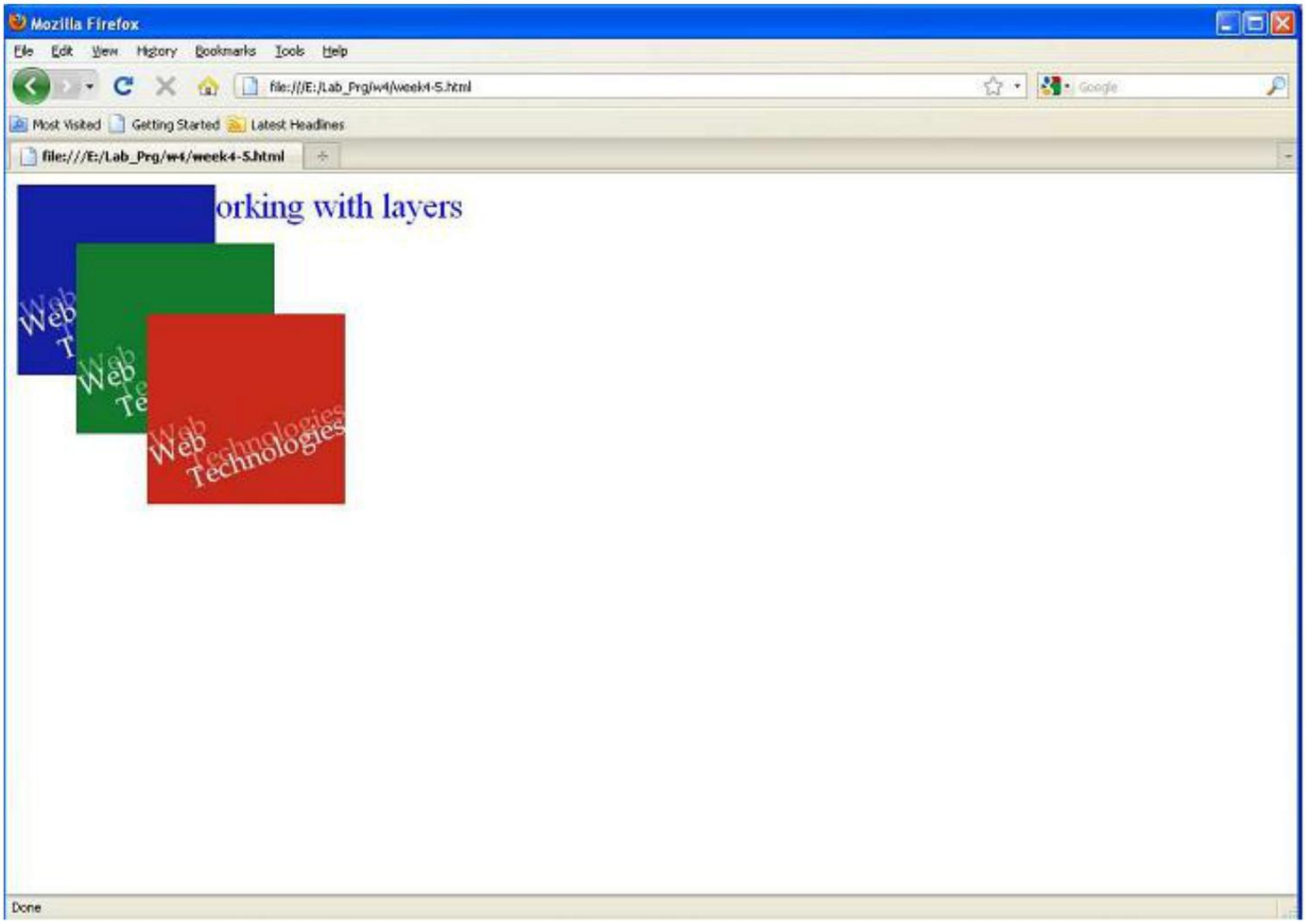


5) Work with layers:

Save as **week4-5.html**

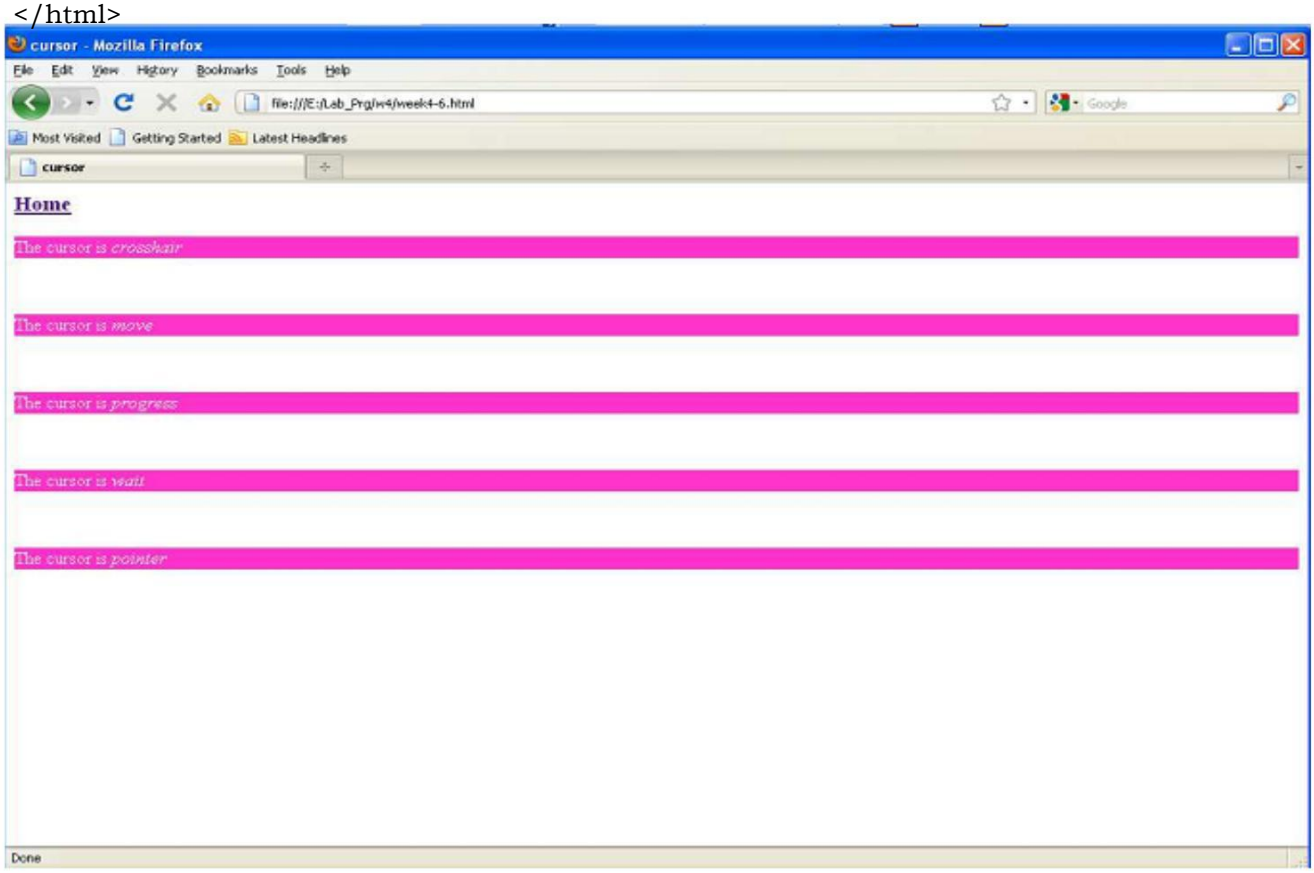
```
<html>
<head>
<style type="text/css">
.txt1
{
color:blue;
font-size:30px;
position:absolute;
left:150px;
top:10px;
z-index:0;
}
.img1
{
position:absolute;
left:10px;

top:10px;
z-index:1;
}

.img2
{
position:absolute;
left:60px;
top:60px;
z-index:2;
}
.img3
```

```
{
position:absolute;
left:120px;
top:120px;
z-index:3;
}
</style>
</head>
<body>
<p class="txt1">Working with layers</p>
<img class="img1" src="layer1.jpg"/>
<img class="img2" src="layer2.jpg"/>
<img class="img3" src="layer3.jpg"/>
</body>
</html>
```



6) Add a customized cursor
Save as **week4 .6.html**

```
<html>
<head>
<title> cursor </title>
<style type="text/css">
.c1{cursor:crosshair}
.c2{cursor:move}
.c3{cursor:progress}
.c4{cursor:wait}
.c5{cursor:pointer}
p{background-color:#FF33CC;color:white;}
</style>
</head>
<body>
<h3><a href="week4.html">Home</a></h3>
<p class="c1">The cursor is <i>crosshair</i></p>
<br/>
<p class="c2">The cursor is <i>move</i></p>

<br/>
<p class="c3">The cursor is <i>progress</i></p>
<br/>
<p class="c4">The cursor is <i>wait</i></p>
<br/>
<p class="c5">The cursor is <i>pointer</i></p>
</body>
</html>
```



Exercise

Design a web page using CSS which include the following:

1) Font Size
2)Font Weight
3)Font Stretch
4)Text Decoration
5)Text Transformation
6)Text alignment
7)Padding
8)Border style

Signature of the faculty

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.

Save as **catalog1.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE catalog SYSTEM "catalog1.dtd">
<catalog>
<book>
<title>Head First Java Script</title>
<author>Kathy Sierra and Bert Bates</author>
<isbn>1148574</isbn>
<publisher>O'Reilly Media</publisher>
<edition>Second</edition>
<price>375</price>
</book>
<book>
<title>Head First Servlets</title>
<author>Bryan Basham, Kathy Sierra and Bert
Bates</author> <isbn>58746321</isbn>
<publisher>O'Reilly Media</publisher>


<edition>Third</edition>
<price>475</price>
</book>
<book>
<title>Head First PHP &amp;amp; MYSQL</title>
<author>Lynn Beighley Michael Morrison </author>
<isbn>965844712</isbn>
<publisher>O'Reilly Media</publisher>
<edition>First</edition>
<price>365</price>
</book>
<book>
<title>Head First WebDesign</title>
<author>Ethan Watrall </author>
<isbn>764485142</isbn>
<publisher>O'Reilly Media</publisher>
<edition>First</edition>
<price>390</price>
</book>
</catalog>
```

Save as **catalog1.dtd**

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT catalog (book)*>
<!ELEMENT book (title,author,isbn,publisher,edition,price)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT author (#PCDATA)>
<!ELEMENT isbn (#PCDATA)>
<!ELEMENT publisher (#PCDATA)>
<!ELEMENT edition (#PCDATA)>
<!ELEMENT price (#PCDATA)>
```

XMLSchema

Save as **catalog2.html**

```
<?xml version="1.0" encoding="UTF-8"?>
<catalog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="catalog2.xsd">
<book>
<title>Head First Java Script</title>
<author>Kathy Sierra and Bert Bates</author>
<isbn>1148574</isbn>
<publisher>O'Reilly Media</publisher>
<edition>Second</edition>
<price>375</price>
</book>
<book>
<title>Head First Servlets</title>
<author>Bryan Basham, Kathy Sierra and Bert
Bates</author> <isbn>58746321</isbn>
<publisher>O'Reilly Media</publisher>
<edition>Third</edition>
<price>475</price>


</book>
<book>
<title>Head First PHP &amp;amp; MYSQL</title>
<author>Lynn Beighley Michael Morrison </author>
<isbn>965844712</isbn>

<publisher>O'Reilly Media</publisher>
<edition>First</edition>
<price>365</price>
</book>
<book>
<title>Head First WebDesign</title>
<author>Ethan Watrall </author>
<isbn>764485142</isbn>
<publisher>O'Reilly Media</publisher>
<edition>First</edition>
<price>390</price>
</book>
</catalog>
```

Save as **catalog2.xsd**

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="catalog"/>
<xs:complexType name="book">
<xs:sequence>
<xs:element name="title" type="xs:string"/>
<xs:element name="author" type="xs:string"/>
<xs:element name="isbn" type="xs:string"/>
<xs:element name="publisher" type="xs:string"/>
<xs:element name="edition" type="xs:string"/>
<xs:element name="price" type="xs:string"/>
</xs:sequence>
</xs:complexType>
<xs:element name="book"/>
</xs:schema>
```

eXtensible Stylesheet Language

Save as **catalog3.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="catalog3.xsl"?>
<!DOCTYPE catalog SYSTEM "catalog.dtd">
<catalog>
<book>
<title>Head First Java Script</title>
<author>Kathy Sierra and Bert Bates</author>
<isbn>1148574</isbn>
<publisher>O'Reilly Media</publisher>
<edition>Second</edition>
<price>375</price>
</book>
<book>
<title>Head First Servlets</title>
<author>Bryan Basham, Kathy Sierra and Bert
Bates</author> <isbn>58746321</isbn>
<publisher>O'Reilly Media</publisher>
<edition>Third</edition>
<price>475</price>
</book>
<book>
<title>Head First PHP &amp;amp; MYSQL</title>
<author>Lynn Beighley Michael Morrison </author>
<isbn>965844712</isbn>
<publisher>O'Reilly Media</publisher>
<edition>First</edition>
<price>365</price>
```

```
</book>
<book>
<title>Head First WebDesign</title>
<author>Ethan Watrall </author>
<isbn>764485142</isbn>
<publisher>O'Reilly Media</publisher>

<edition>First</edition>
<price>390</price>
</book>
</catalog>
```

Save as **catalog3.xsl**

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="catalog">
<html>
<head>
<title>converting in to xml</title>
</head>
<body>
<table border="1">
<tr>
<th>title</th><th>author</th><th>isbn</th>
<th>publisher</th><th>edition</th><th>price</th>
</tr>
<xsl:for-each select="book">
<tr>
<td><xsl:value-of select="title"/></td>
<td><xsl:value-of select="author"/></td>
<td><xsl:value-of select="isbn"/></td>
<td><xsl:value-of select="publisher"/></td>
<td><xsl:value-of select="edition"/></td>

<td><xsl:value-of select="price"/></td>
</tr>
</xsl:for-each>
</table>
</body>
</html>


</xsl:template>
</xsl:stylesheet>
```

Cascading Style Sheet

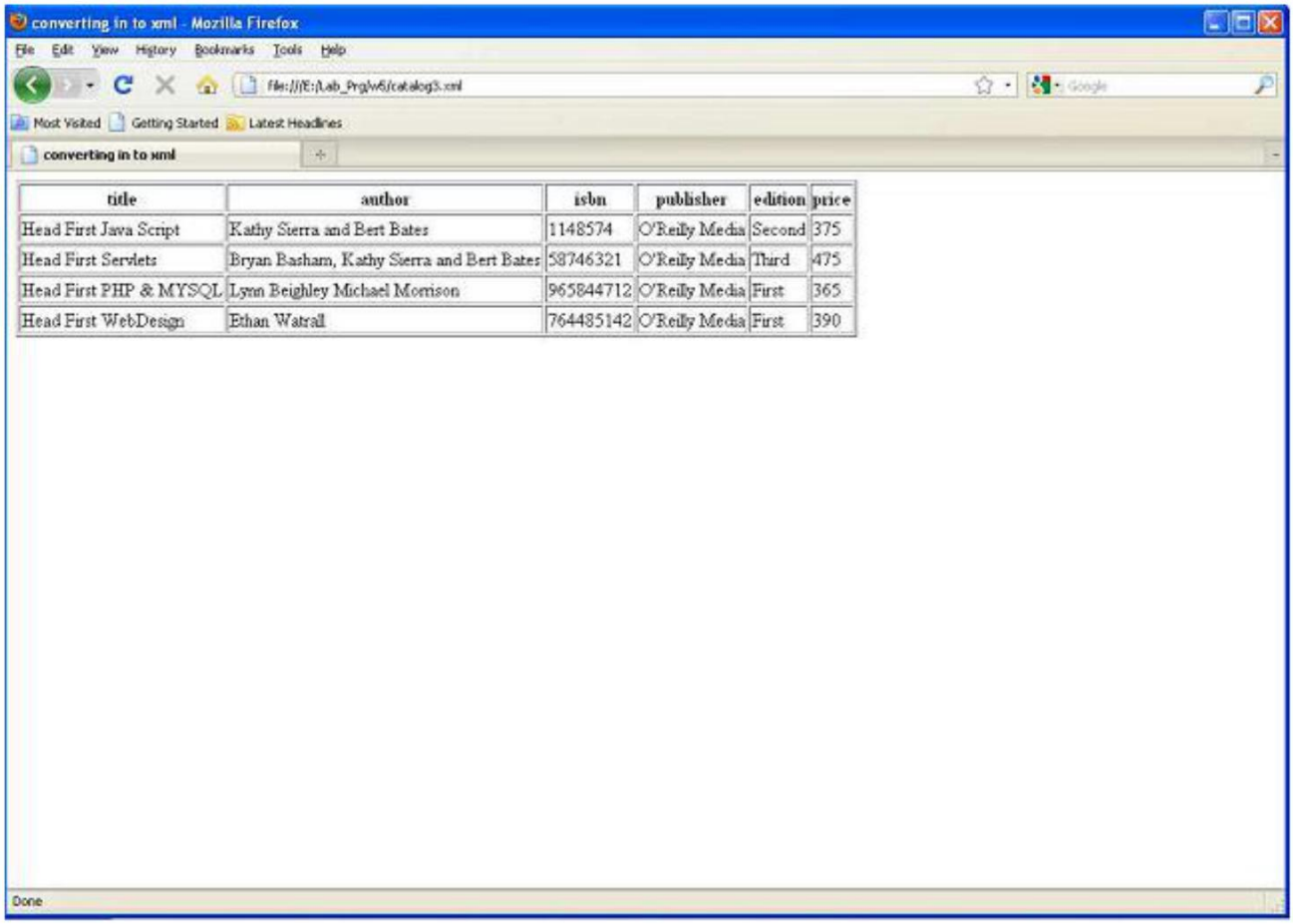
Save as **catalog4.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="catalog4.css"?>
<!DOCTYPE catalog SYSTEM "catalog.dtd">
<catalog>
<book>
<title>Head First Java Script</title>
```

```
<author>Kathy Sierra and Bert Bates</author>
<isbn>1148574</isbn>
<publisher>O'Reilly Media</publisher>
<edition>Second</edition>
<price>375</price>
</book>
<book>
<title>Head First Servlets</title>
<author>Bryan Basham, Kathy Sierra and Bert Bates</author>
<isbn>58746321</isbn>
<publisher>O'Reilly Media</publisher>
<edition>Third</edition>
<price>475</price>
</book>
<book>
<title>Head First PHP &amp;amp; MYSQL</title>
<author>Lynn Beighley Michael Morrison </author>
<isbn>965844712</isbn>
<publisher>O'Reilly Media</publisher>
<edition>First</edition>
<price>365</price>
</book>
<book>
<title>Head First WebDesign</title>
<author>Ethan Watrall </author>
<isbn>764485142</isbn>
<publisher>O'Reilly Media</publisher>
<edition>First</edition>
<price>390</price>

</book>
</catalog>
```

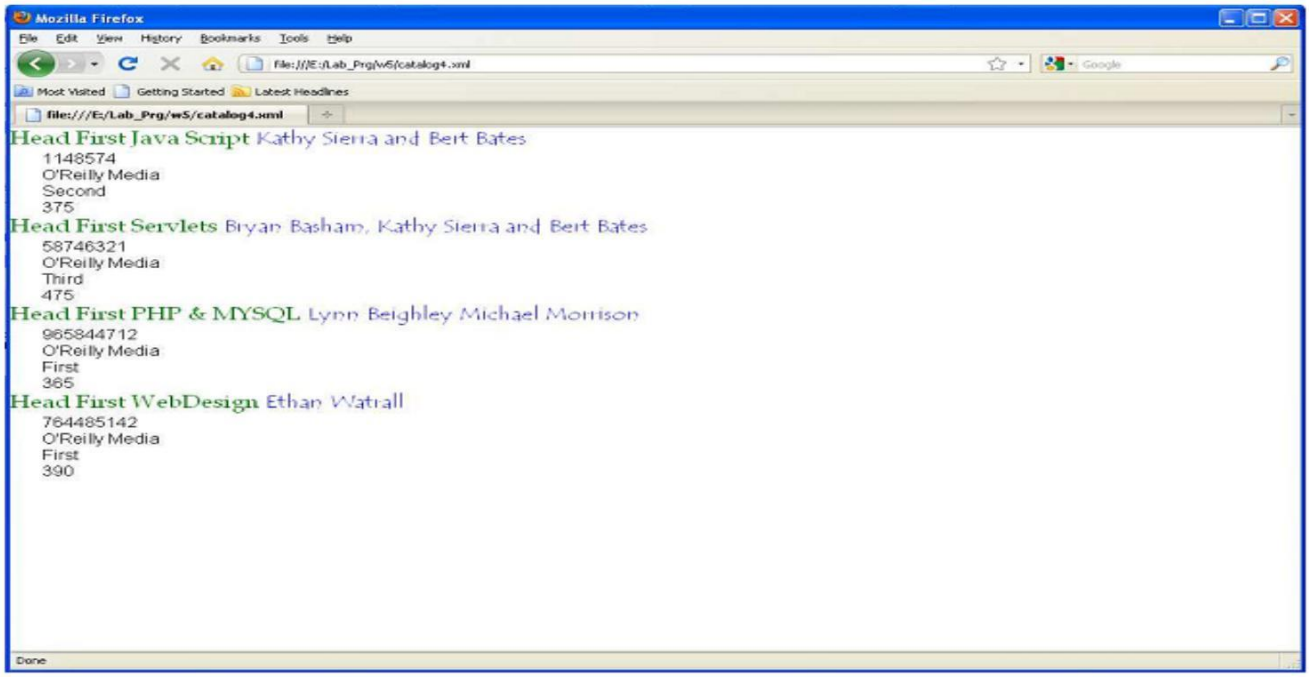
Save as **catalog4.css**

```
catalog
{
font-family:arial;
color:red;
font-size:16pt;
}
book
{
display:block;
font-family:times new roman;
color:blue;
font-size:14pt;
}
title
{
font-family:Book Antiqua;

color:green;
font-size:16pt;
}
author
{
```



title	author	isbn	publisher	edition	price
Head First Java Script	Kathy Sierra and Bert Bates	1148574	O'Reilly Media	Second	375
Head First Servlets	Bryan Basham, Kathy Sierra and Bert Bates	58746321	O'Reilly Media	Third	475
Head First PHP & MYSQL	Lynn Beighley Michael Morrison	965844712	O'Reilly Media	First	365
Head First WebDesign	Ethan Watrall	764485142	O'Reilly Media	First	390

```
font-family:Tempus Sans ITC;
color:blue;
font-size:16pt;
}
isbn,publisher,edition,price
{
display:block;
font-family:arial;
color:black;
font-size:12pt;
margin-left:20pt;
}
```



Exercise

1)Write an XML file which will display the Student information which includes the following:
1) Student name
2) College
3) Branch
4) Roll No
5) Percentage
6) Backlogs
Write a Document Type Definition (DTD) to validate the above XML file.

2) Create a DTD for a catalog of four stroke motorbikes, where each motor bike has the following child elements-make,model,year,engine,chasis number accessories.The engine element has the child elements engine number, number of cylinders,type of fuel.The accessories element has the attributes like disc brake, auto-start and radio,each of which is required and has the possible values yes and no. Entities must be declared for the names of the popular motorbike makes.

Signature of the Faculty

WEEK- 6

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 ".

Save as **colors.java**

```
import java.io.*;
import java.awt.*;
import java.awt.event.*;
public class colors extends Canvas
{
public boolean square;
public Color color;
public colors()
{
addMouseListener(new MouseAdapter()
{
public void mouseClicked(MouseEvent me)
{
change();

}
});
square=true;
setSize(200,200);
change();
}
public void change()
{
color = random();
repaint();
}
public Color getColor()
{
return color;
}
public void setColor(Color color)
{
this.color=color;
}
```

```
private Color random()
{
int red = (int)(255*Math.random());

int green = (int)(255*Math.random());
int blue = (int)(255*Math.random());
return new Color(red,green,blue);
}
public boolean isSquare()
{
return square;
}


public void setSquare(boolean square)
{
this.square=square;
repaint();
}
public void paint(Graphics g)
{
if(square)
{
g.setColor(color);
//g.drawRect(10,10,160,160);
g.fillRect(10,10,180,180);
}
else
{
g.setColor(color);
//g.drawOval(10,10,160,160);
g.fillOval(10,10,180,180);
}
}
}
```

Save as **colors.mft**

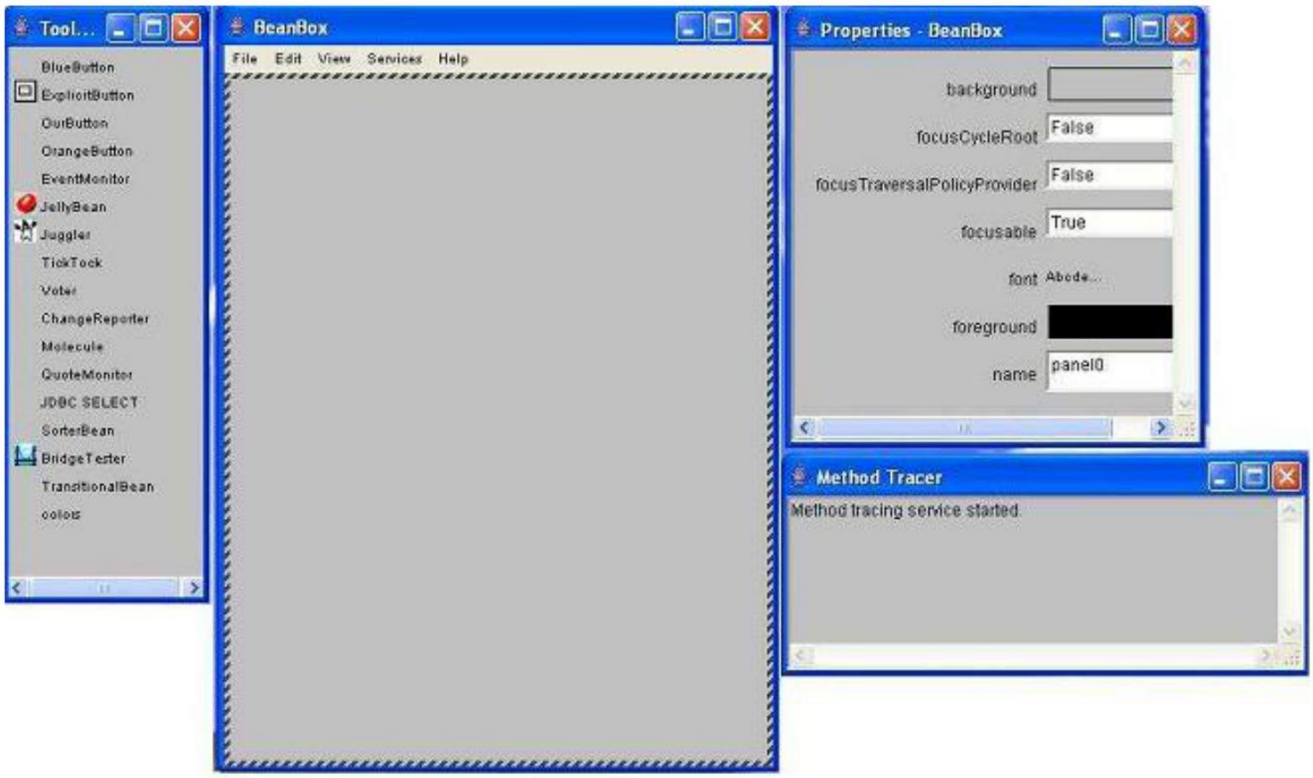
Name: colors.class
Java-Bean: True

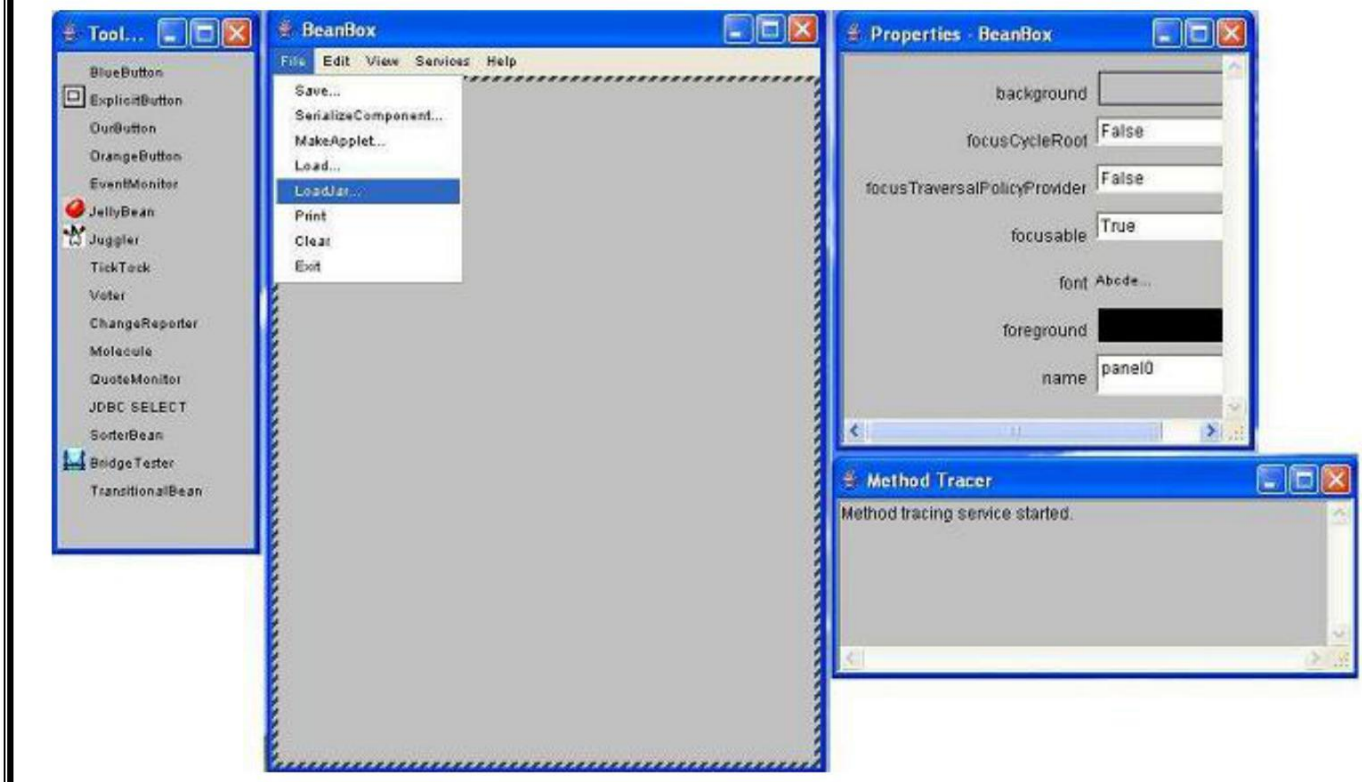
Create a jar file
jar cfmv colors.jar colors.mft colors.class colors$1.class
Now jar file has been created.

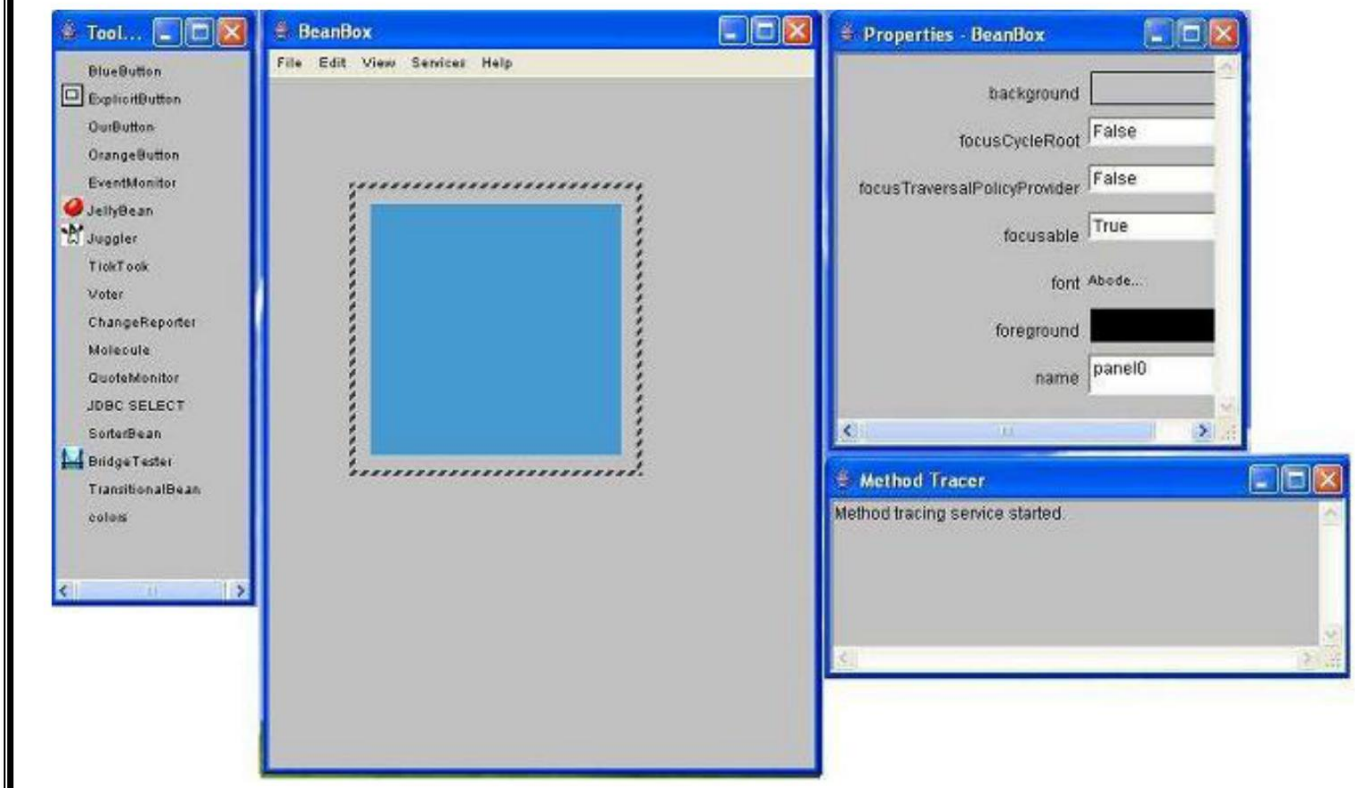
Go to
C:\beans\beanbox>run *[press enter]*
Opens four dialogue boxes **toolbox,**
beanbox,
properties window,
method tracer

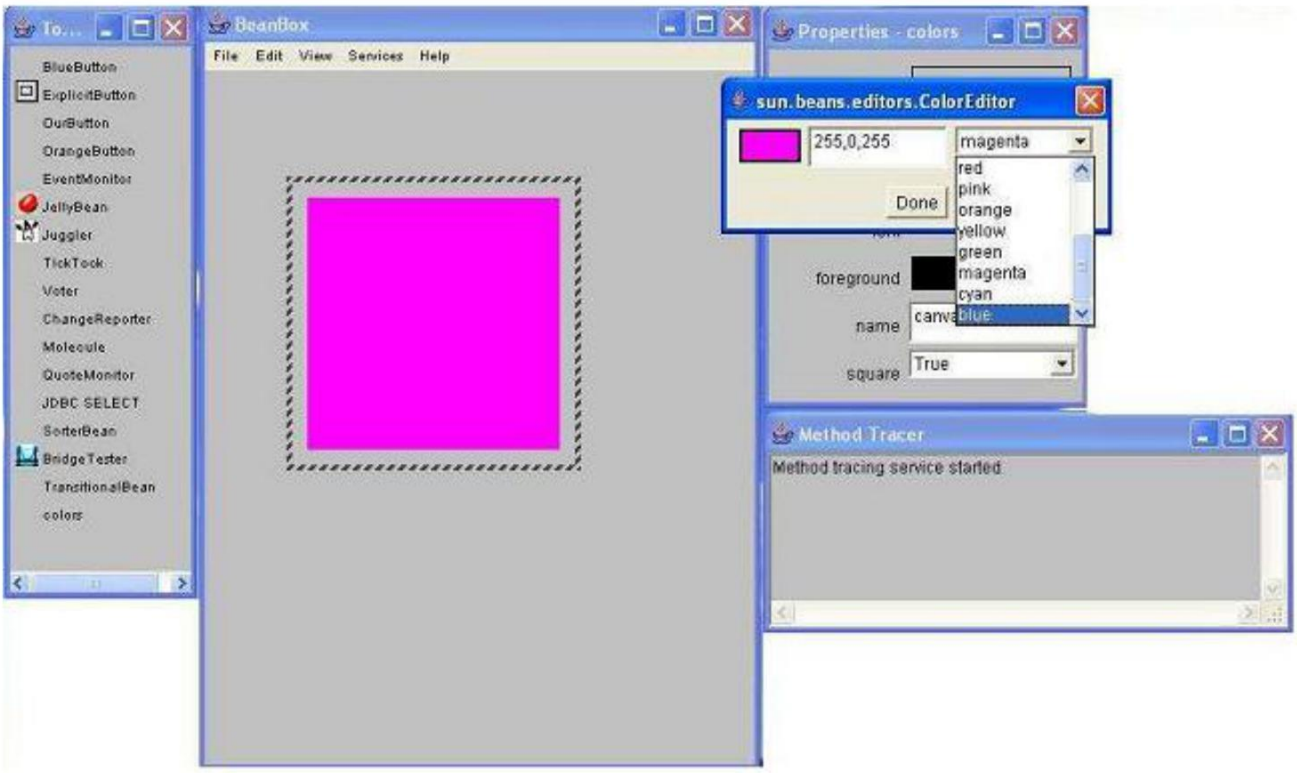
Goto file->Loadjar *[load the jar colors.jar]*
In the **toolbox** you can observe colors bean, select the colors the bean and place on the beanbox Click the shape of the area, you can observe the color changing dynamically for every mouse click

In the property window change the property square [True / False] observe the shape changes to square when its is True and shape changes to circle when it is False




Tool...
BlueButton
ExplicitButton
OurButton
OrangeButton
EventMonitor
JellyBean
Juggler
TickTock
Voter
ChangeReporter
Molecule
QuoteMonitor
JDBC SELECT
SorterBean
BridgeTester
TransitionalBean
BeanBox
File Edit View Services Help
Save...
SerializeComponent...
MakeApplet...
Load...
LoadJar...
Print
Clear
Exit
Properties - BeanBox
background
focusCycleRoot False
focusTraversalPolicyProvider False
focusable True
font Abcde...
foreground
name panel0
Method Tracer
Method tracing service started.


Tool...
BlueButton
ExplicitButton
OurButton
OrangeButton
EventMonitor
JellyBean
Juggler
TickTock
Voter
ChangeReporter
Molecule
QuoteMonitor
JDBC SELECT
SorterBean
BridgeTester
TransitionalBean
colors
BeanBox
File Edit View Services Help
Properties - BeanBox
background
focusCycleRoot False
focusTraversalPolicyProvider False
focusable True
font Abcde...
foreground
name panel0
Method Tracer
Method tracing service started.



Exercises

1) To start and stop the Animation(on Juggler Bean) by creating Start and Stop buttons: use BDK Tool.

2) Create two Beans Traffic Light(implemented as a label with only three background colors-red, green, yellow) and Automobile(Implemented as a Text Box which states its state/movement). The state of the Automobile should depend on the following Light Transition table.

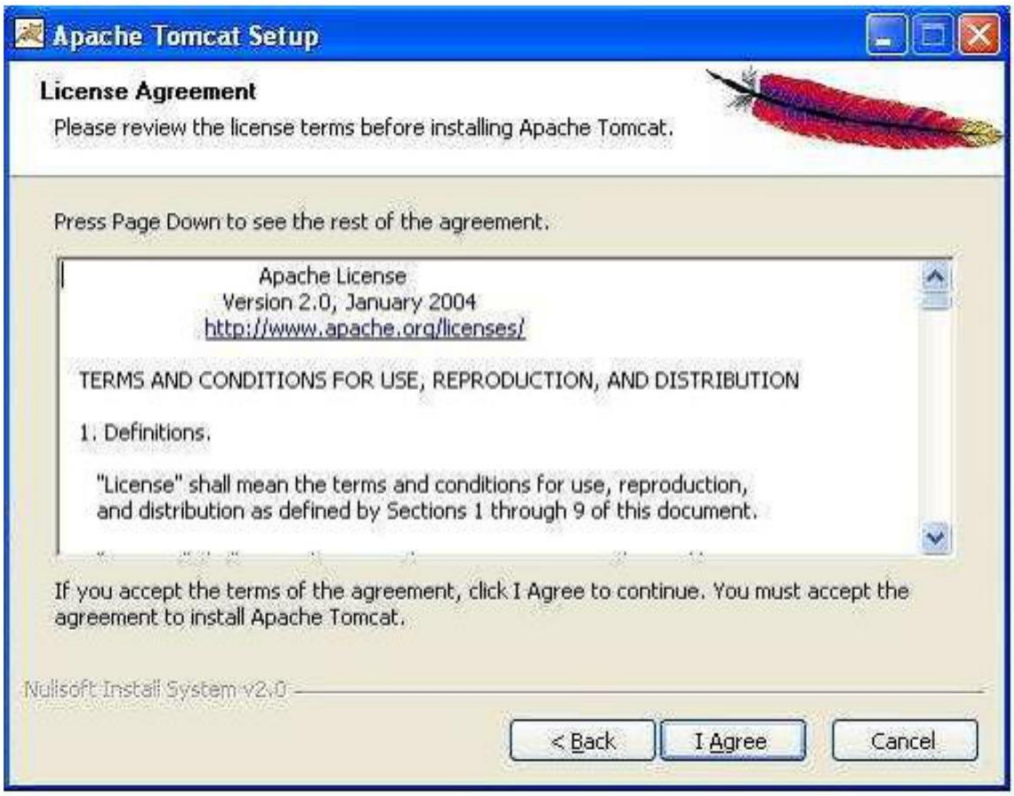
Signature of the faculty

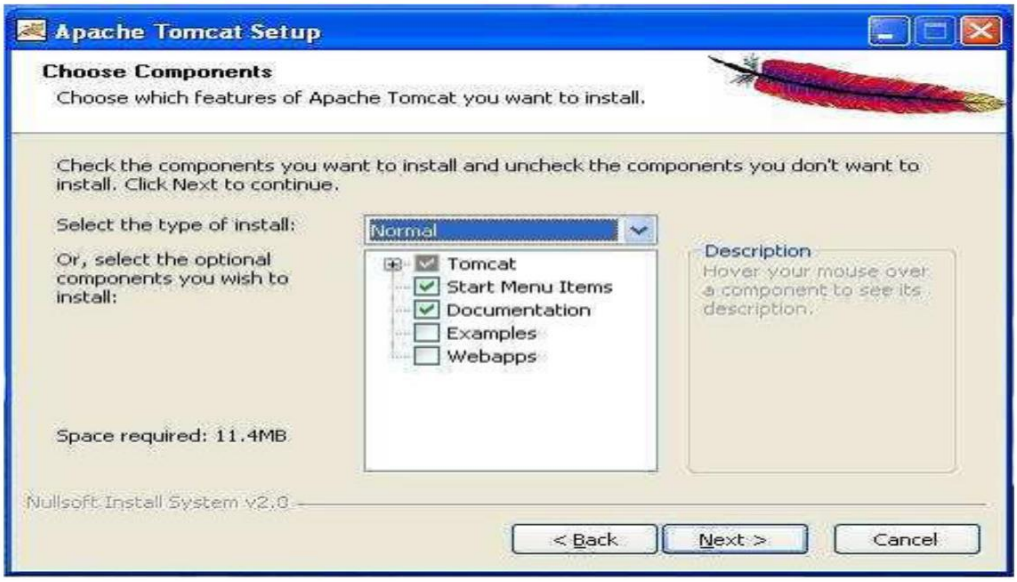
\ \

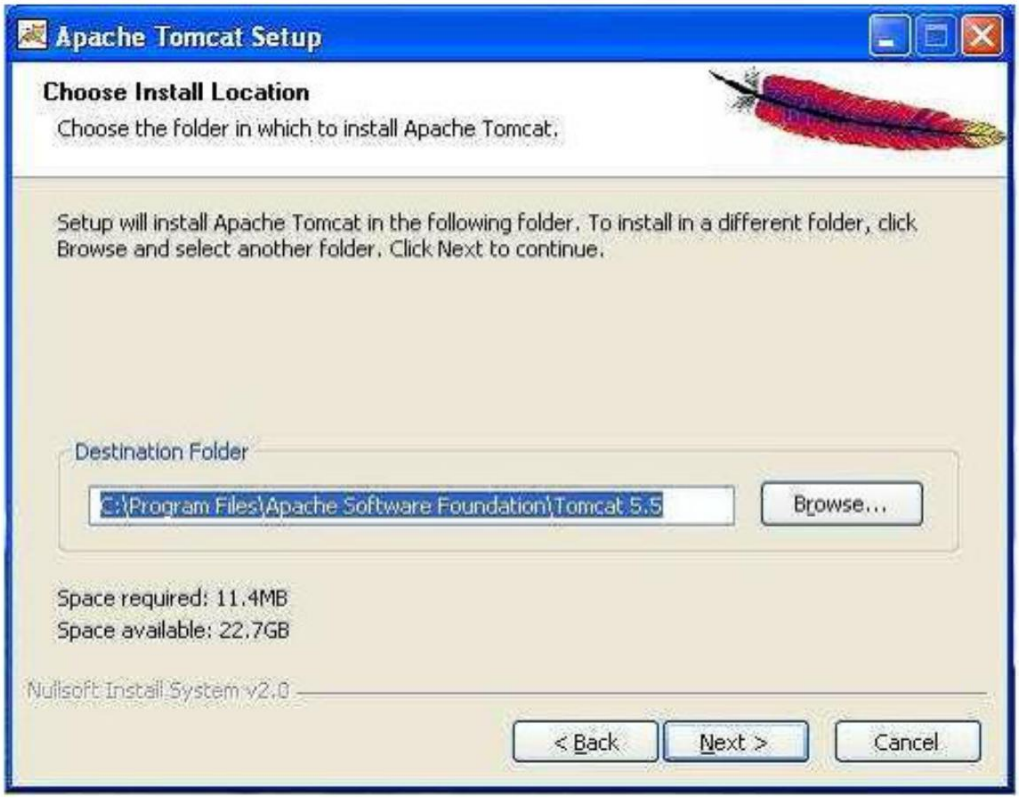
WEEK-7

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)

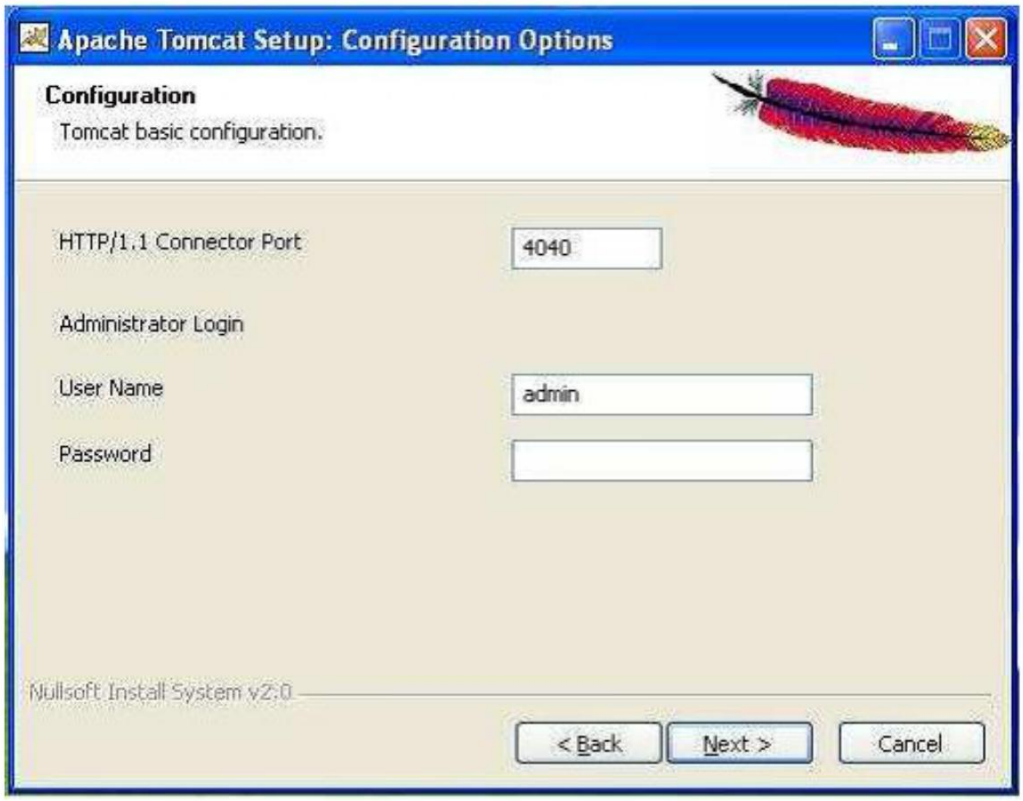
Installation of Tomcat

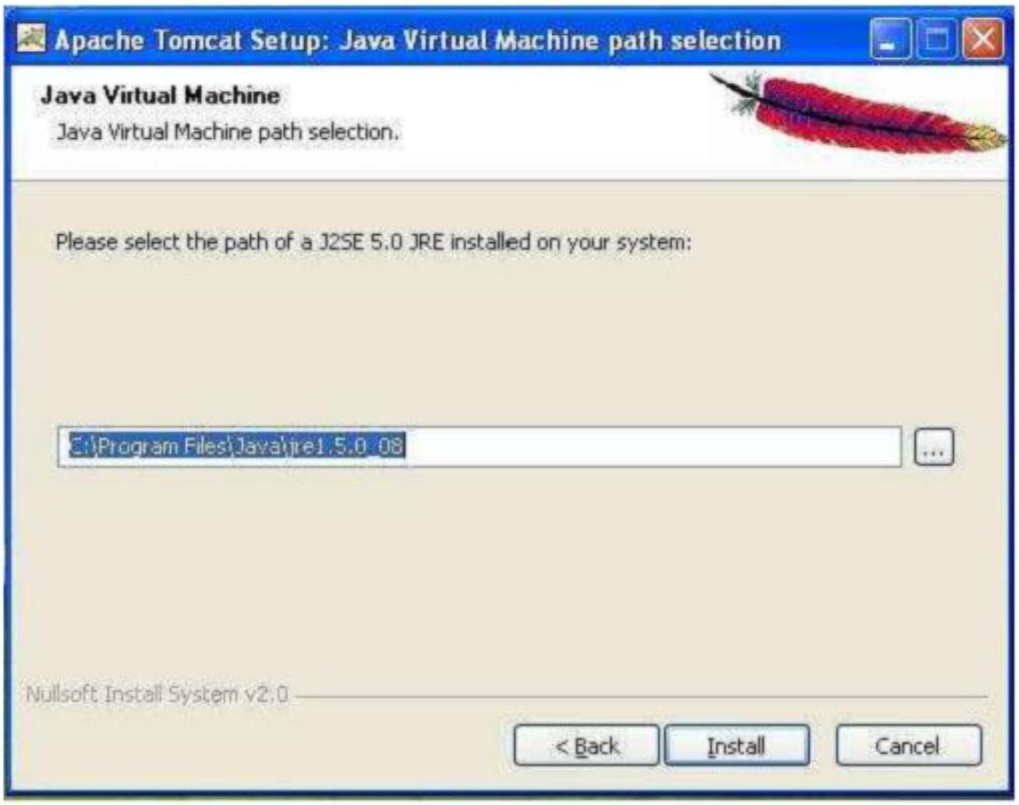

Apache Tomcat Setup
License Agreement
Please review the license terms before installing Apache Tomcat.
Press Page Down to see the rest of the agreement.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.
"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Apache Tomcat.
Nullsoft Install System v2.0
< Back
I Agree
Cancel

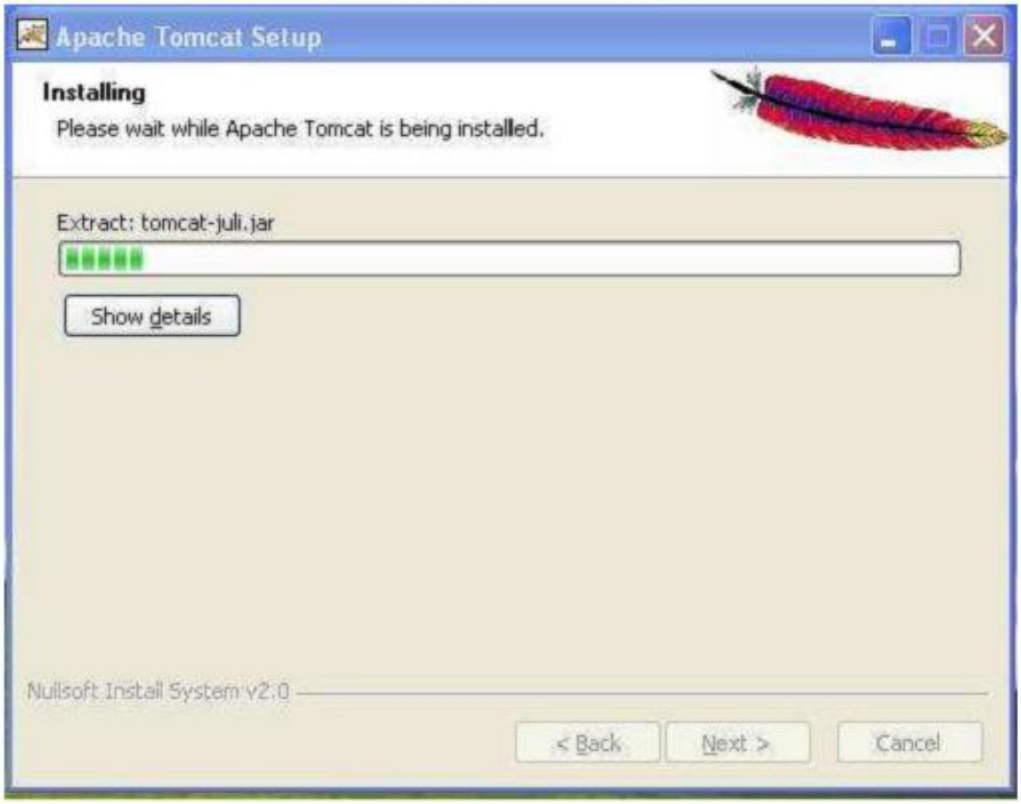

Apache Tomcat Setup
Choose Components
Choose which features of Apache Tomcat you want to install.
Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.
Select the type of install:
Normal
Or, select the optional components you wish to install:
Tomcat
Start Menu Items
Documentation
Examples
Webapps
Description
Hover your mouse over a component to see its description.
Space required: 11.4MB
Nullsoft Install System v2.0
< Back
Next >
Cancel

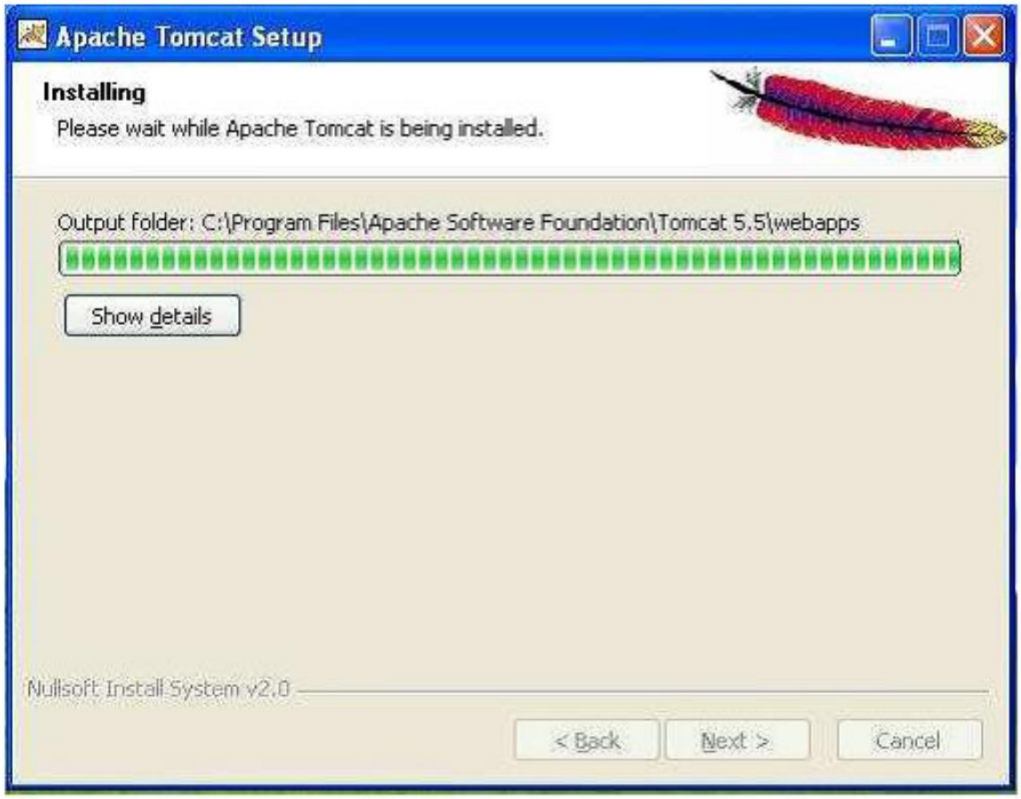


Setting the port number to 4040

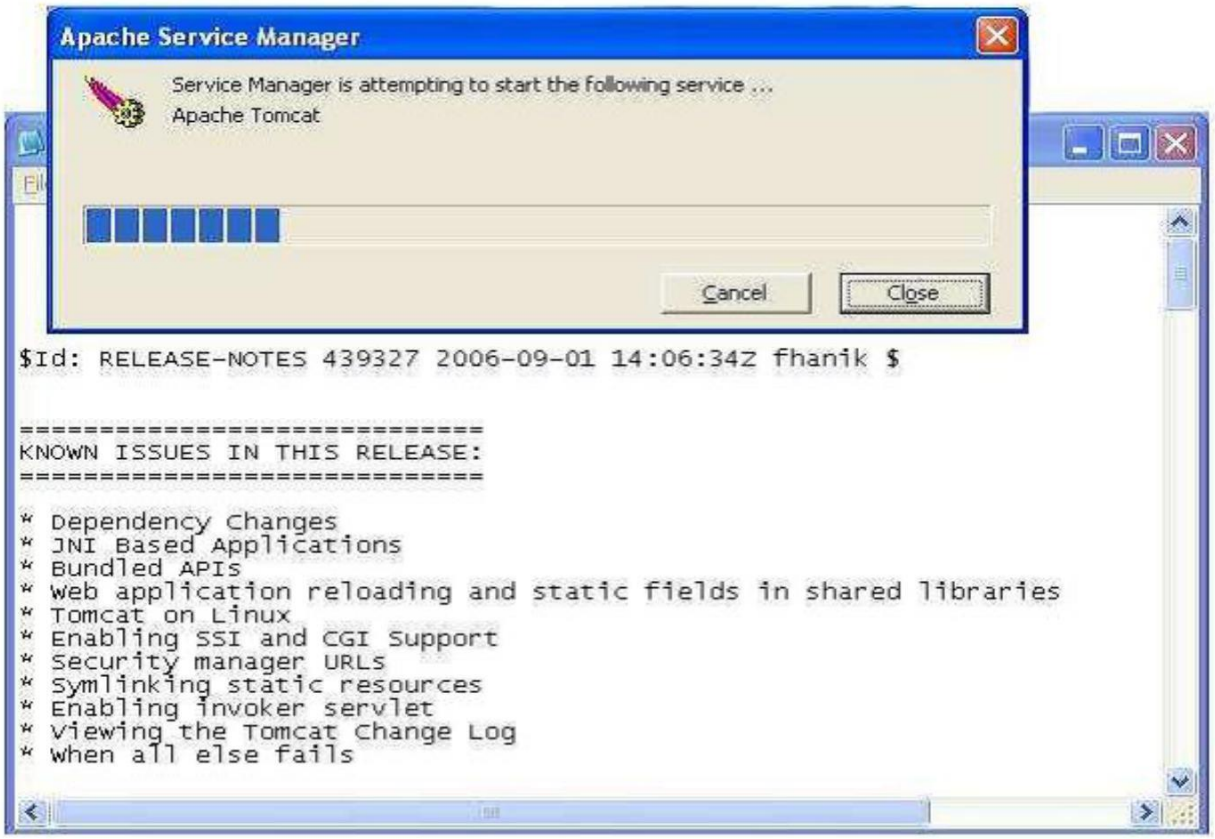


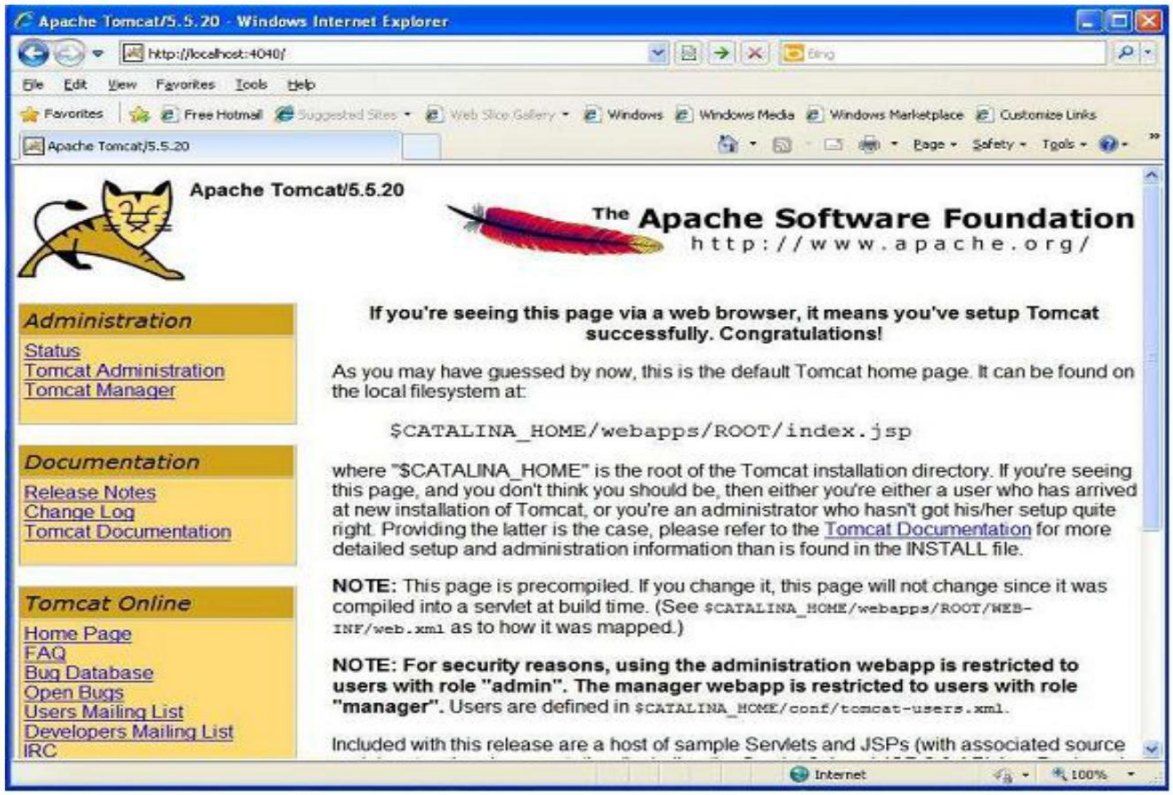

Apache Tomcat Setup: Java Virtual Machine path selection
Java Virtual Machine
Java Virtual Machine path selection.
Please select the path of a J2SE 5.0 JRE installed on your system:
C:\Program Files\Java\jre1.5.0_08
...
Nullsoft Install System v2.0
< Back
Install
Cancel


Apache Tomcat Setup
Installing
Please wait while Apache Tomcat is being installed.
Extract: tomcat-juli.jar
Show details
Nullsoft Install System v2.0
< Back
Next >
Cancel


Apache Tomcat Setup
Installing
Please wait while Apache Tomcat is being installed.
Output folder: C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps
Show details
Nullsoft Install System v2.0
< Back
Next >
Cancel

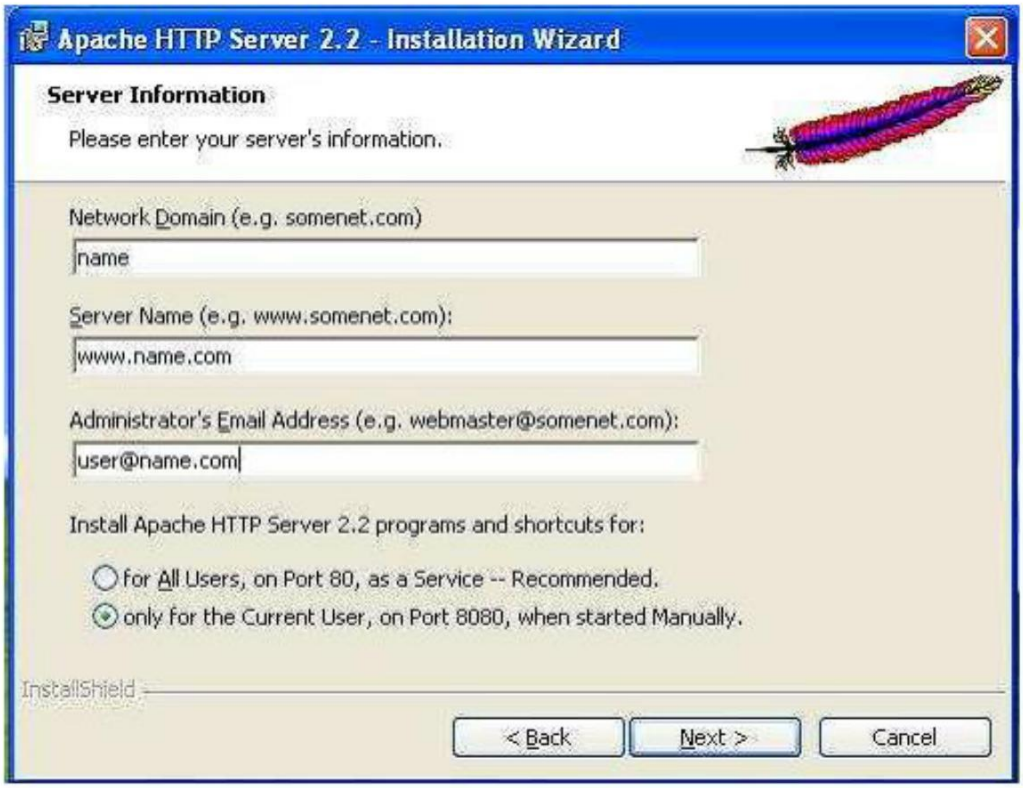

Apache Tomcat Setup
The Apache Jakarta Project
http://jakarta.apache.org/
Completing the Apache Tomcat Setup Wizard
Apache Tomcat has been installed on your computer.
Click Finish to close this wizard.
Run Apache Tomcat
Show Readme
< Back
Finish
Cancel

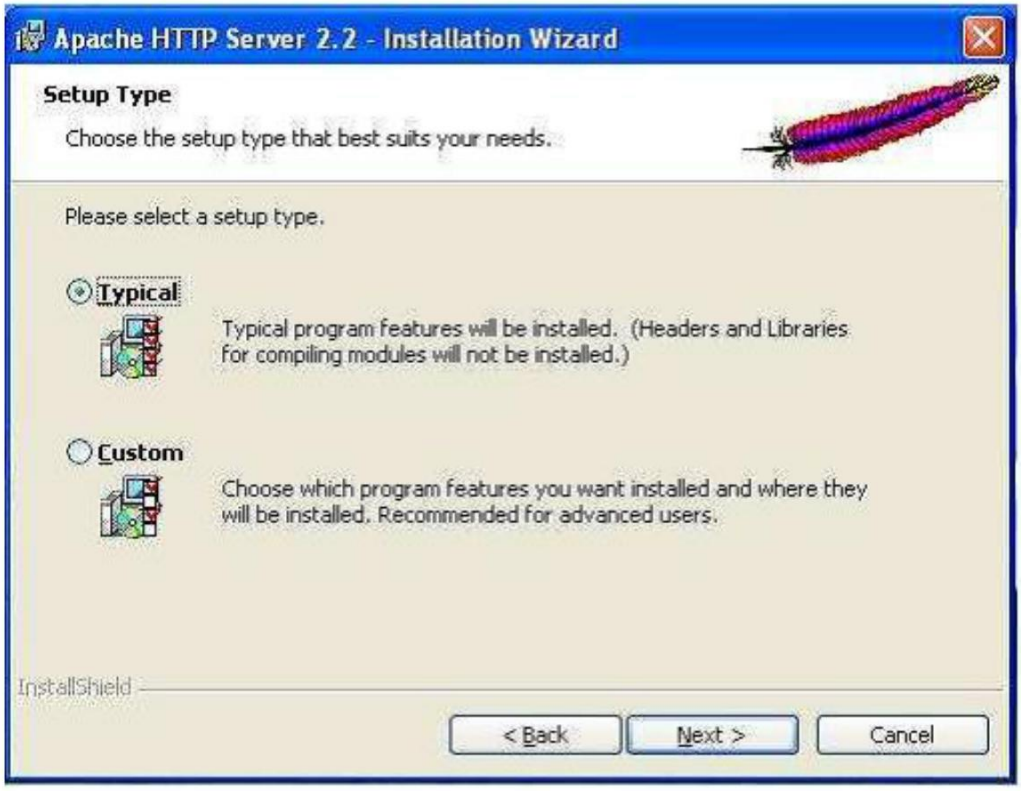

Apache Service Manager
Service Manager is attempting to start the following service ...
Apache Tomcat
Cancel
Close
$Id: RELEASE-NOTES 439327 2006-09-01 14:06:34Z fhanik $
==========
KNOWN ISSUES IN THIS RELEASE:
==========
* Dependency Changes
* JNI Based Applications
* Bundled APIs
* Web application reloading and static fields in shared libraries
* Tomcat on Linux
* Enabling SSI and CGI Support
* Security manager URLs
* Symlinking static resources
* Enabling invoker servlet
* Viewing the Tomcat Change Log
* When all else fails

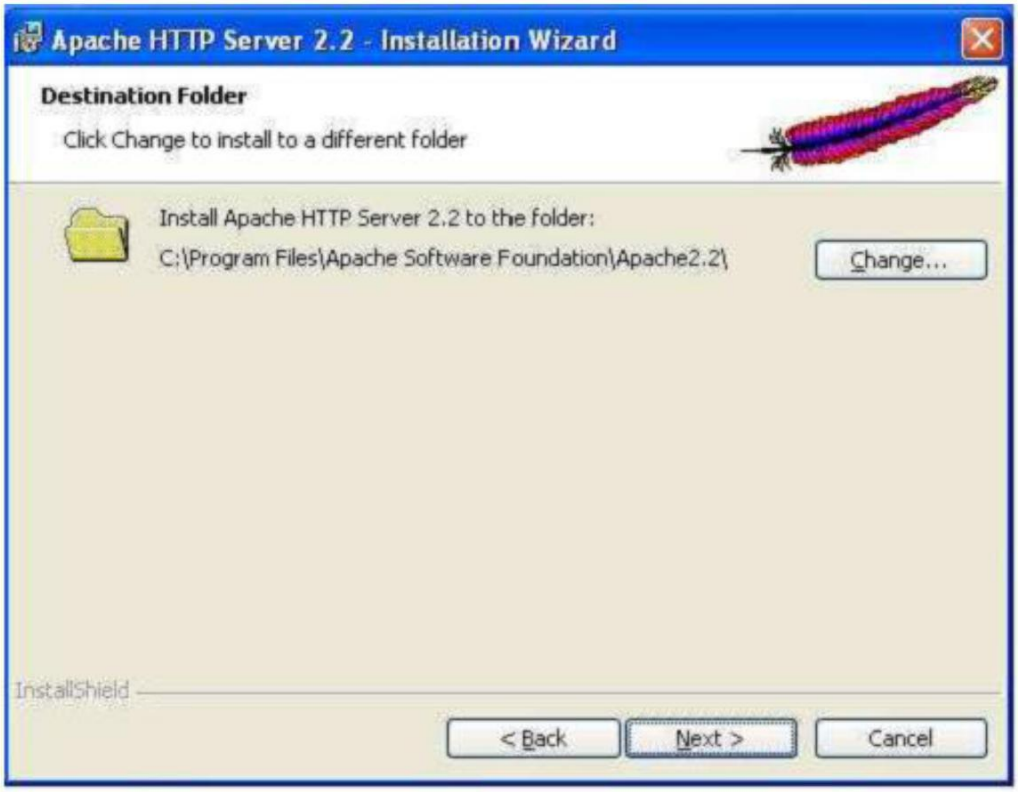

Apache Tomcat/5.5.20 - Windows Internet Explorer
http://localhost:4040/
Apache Tomcat/5.5.20
The Apache Software Foundation
http://www.apache.org/
Administration
Status
Tomcat Administration
Tomcat Manager
Documentation
Release Notes
Change Log
Tomcat Documentation
Tomcat Online
Home Page
FAQ
Bug Database
Open Bugs
Users Mailing List
Developers Mailing List
IRC
If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!
As you may have guessed by now, this is the default Tomcat home page. It can be found on the local filesystem at:
$CATALINA_HOME/webapps/ROOT/index.jsp
where "$CATALINA_HOME" is the root of the Tomcat installation directory. If you're seeing this page, and you don't think you should be, then either you're either a user who has arrived at new installation of Tomcat, or you're an administrator who hasn't got his/her setup quite right. Providing the latter is the case, please refer to the Tomcat Documentation for more detailed setup and administration information than is found in the INSTALL file.
NOTE: This page is precompiled. If you change it, this page will not change since it was compiled into a servlet at build time. (See $CATALINA_HOME/webapps/ROOT/WEB-INF/web.xml as to how it was mapped.)
NOTE: For security reasons, using the administration webapp is restricted to users with role "admin". The manager webapp is restricted to users with role "manager". Users are defined in $CATALINA_HOME/conf/tomcat-users.xml.
Included with this release are a host of sample Servlets and JSPs (with associated source

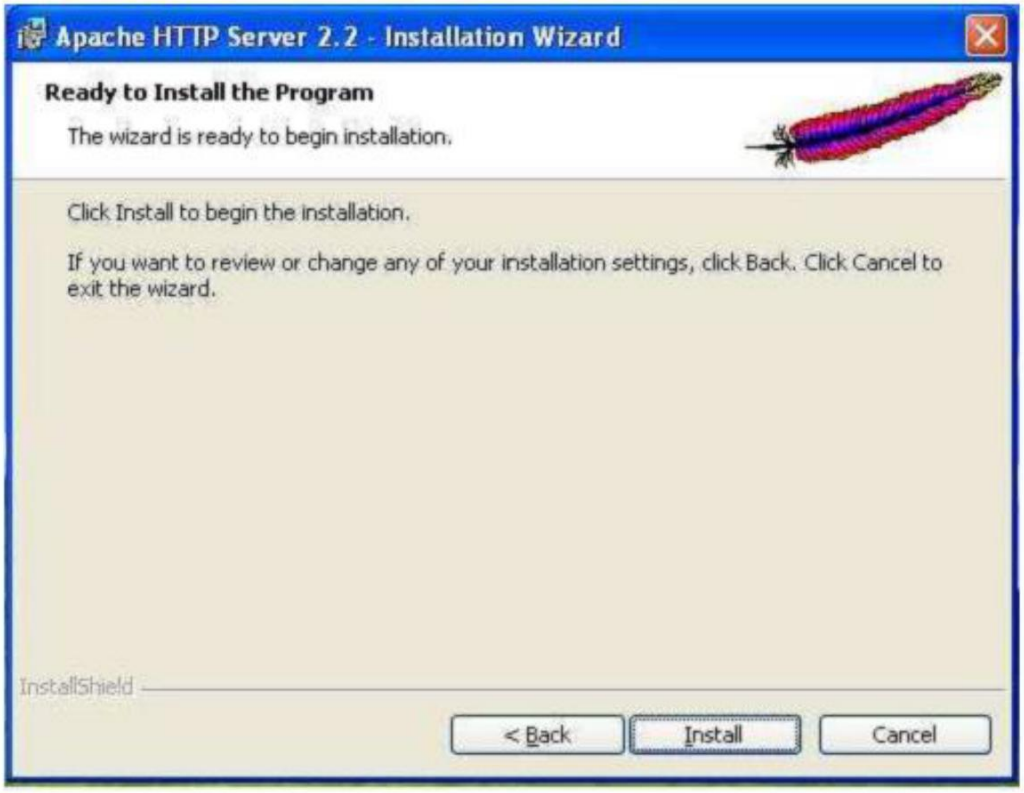
Installation of Apache web server


Apache HTTP Server 2.2 - Installation Wizard
Read This First
Read this Before Running Apache on Windows.
Apache HTTP Server
What is it?
The Apache HTTP Server is a powerful and flexible HTTP/1.1 compliant web server. Originally designed as a replacement for the NCSA HTTP Server, it has grown to be the most popular web server on the Internet. As a project of the Apache Software Foundation, the developers aim to collaboratively develop and maintain a robust, commercial-grade, standards-based server with freely available source code.
The Latest Version
Details of the latest version can be found on the Apache HTTP server project page under:
http://httpd.apache.org/
InstallShield
< Back
Next >
Cancel

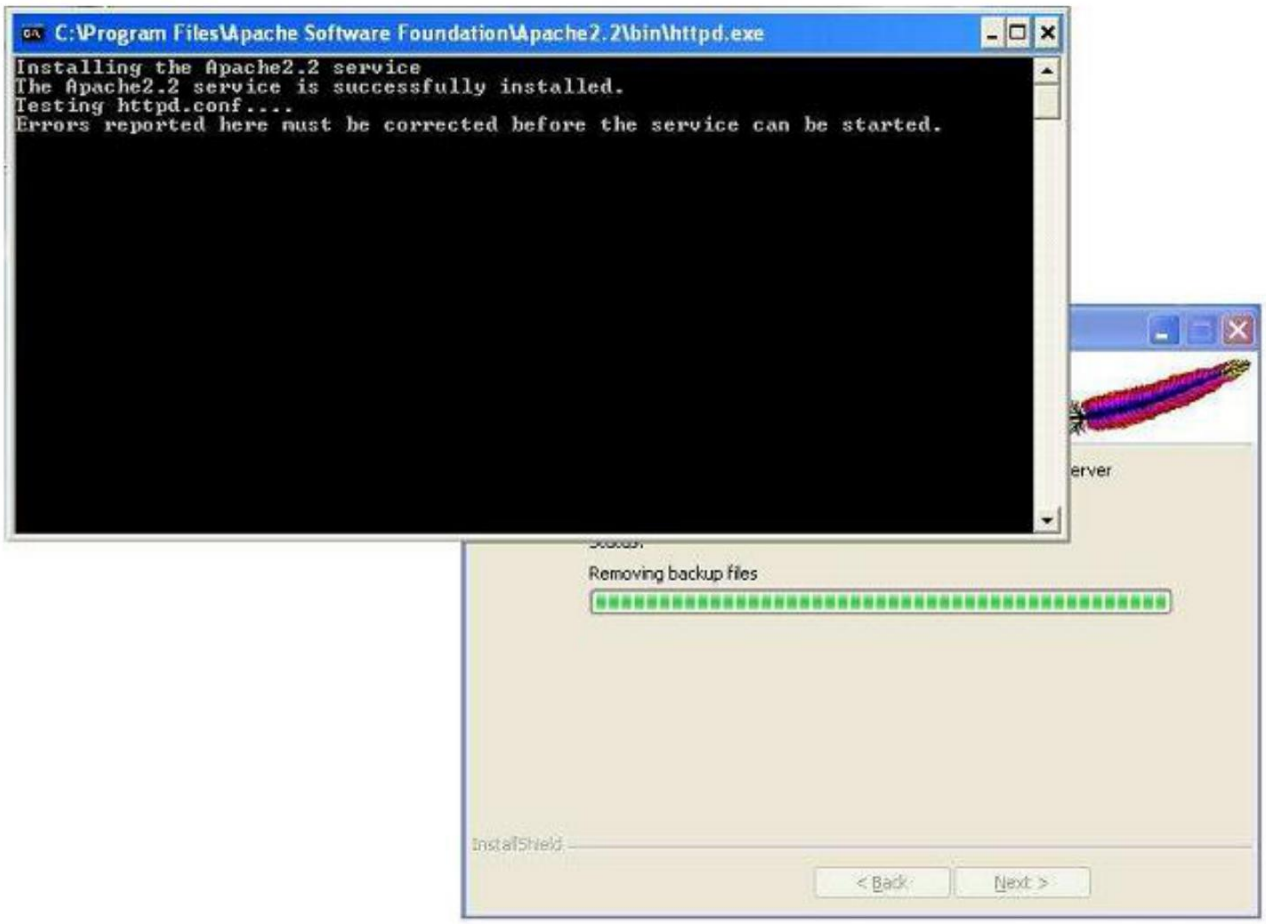

Apache HTTP Server 2.2 - Installation Wizard
Server Information
Please enter your server's information.
Network Domain (e.g. somenet.com)
name
Server Name (e.g. www.somenet.com):
www.name.com
Administrator's Email Address (e.g. webmaster@somenet.com):
user@name.com
Install Apache HTTP Server 2.2 programs and shortcuts for:
for All Users, on Port 80, as a Service -- Recommended.
only for the Current User, on Port 8080, when started Manually.
InstallShield
< Back
Next >
Cancel


Apache HTTP Server 2.2 - Installation Wizard
Setup Type
Choose the setup type that best suits your needs.
Please select a setup type.
Typical
Typical program features will be installed. (Headers and Libraries for compiling modules will not be installed.)
Custom
Choose which program features you want installed and where they will be installed. Recommended for advanced users.
InstallShield
< Back
Next >
Cancel

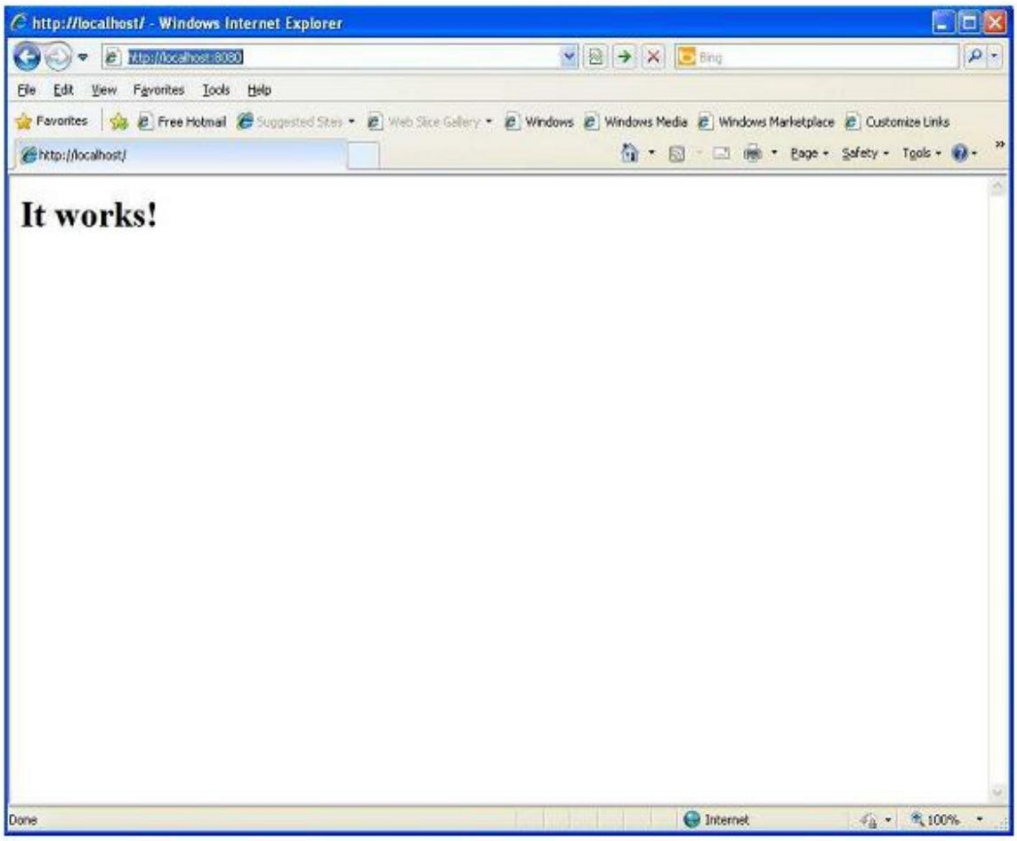

Apache HTTP Server 2.2 - Installation Wizard
Destination Folder
Click Change to install to a different folder
Install Apache HTTP Server 2.2 to the folder:
C:\Program Files\Apache Software Foundation\Apache2.2\
Change...
InstallShield
< Back
Next >
Cancel


Apache HTTP Server 2.2 - Installation Wizard
Ready to Install the Program
The wizard is ready to begin installation.
Click Install to begin the installation.
If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.
InstallShield
< Back
Install
Cancel


Apache HTTP Server 2.2 - Installation Wizard
Installing Apache HTTP Server 2.2.14
The program features you selected are being installed.
Please wait while the Installation Wizard installs Apache HTTP Server 2.2.14. This may take several minutes.
Status:
InstallShield
< Back
Next >
Cancel


C:\Program Files\Apache Software Foundation\Apache2.2\bin\httpd.exe
Installing the Apache2.2 service
The Apache2.2 service is successfully installed.
Testing httpd.conf....
Errors reported here must be corrected before the service can be started.
erver
Removing backup files
InstallShield
< Back
Next >


Apache HTTP Server 2.2 - Installation Wizard
Installation Wizard Completed
The Installation Wizard has successfully installed Apache HTTP Server 2.2.14. Click Finish to exit the wizard.
< Back
Finish
Cancel


http://localhost/ - Windows Internet Explorer
File Edit View Favorites Tools Help
Favorites
Free Hotmail
Windows
Windows Media
Windows Marketplace
Customize Links
Page
Safety
Tools
It works!
Done
Internet
100%

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)

In Tomcat
The code written in week1, week2 and week3 just copy the complete files to the C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\book

Web.xml

```
<web-app>
<display-name>Books</display-name>
<welcome-file-list>
<welcome-file>Homepage.html</welcome-file>
</welcome-file-list>
</web-app>
```

Now start the tomcat and run using the given below url (the port number we are using 8080 instead of 4040)
http://localhost:8080/book/Homepage.html

In Apache
The code written in week1, week2 and week3 just copy the complete files to the C:\Program Files\Apache Software Foundation\Apache2.2\htdocs

Now start the apache and run using the given below url (the port number we are using 80 instead of 8080)
http://localhost:80/Homepage.html

Signature of the Faculty

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(username)
else you should display " You are not an authenticated user ".
(b) Use init-parameters to do this. Store the user-names and passwords in the web.xml and access them in the servlet by using the getInitParameters() method.

Save as **cookie1.java**

[Save in path C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\login2\WEB-INF\classes**]**

```
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class cookie1 extends HttpServlet
{
public void init(ServletConfig config) throws ServletException
{
super.init(config);
}
public void doPost(HttpServletRequest req, HttpServletResponse res) throws
IOException,ServletException
{
res.setContentType("text/html");
PrintWriter out = res.getWriter();
Cookie mycookie = new Cookie("null","null");
Enumeration keys;
String key,value;
keys=req.getParameterNames();
while(keys.hasMoreElements())
{
key = (String)keys.nextElement();
value = req.getParameter(key);
mycookie = new Cookie(value,key);

res.addCookie(mycookie);
}
out.println("\nThe cookie is added!");
}
}
```

Save as **getcookie.java**

[Save in path C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\login2\WEB-INF\classes**]**

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class getcookie extends HttpServlet
{
public void doGet(HttpServletRequest req, HttpServletResponse res) throws
IOException,ServletException
{
res.setContentType("text/html");
PrintWriter out = res.getWriter();
Cookie[] mycookie = req.getCookies();
String[] name = new String[10];
String[] value = new String[10];
int i=0;
int n=mycookie.length;
out.println("The number of cookies:"+n);
for(i=0;i<n;i++)
{
name[i]=mycookie[i].getName();
value[i]=mycookie[i].getValue();
}
for(i=0;i<n-1;i=i+2)
{
try
{
if(name[i].equals("pwd1"))
{
if(name[i+1].equals("user1"))
{
out.println("WelCome"+name[i+1]);
}
else
{
out.println("Unauthorised User");
}

}
else if(name[i].equals("pwd2"))
{
if(name[i+1].equals("user2"))
{
out.println("WelCome"+name[i+1]);
}
else
{
out.println("Unauthorised User");
}
}
else if(name[i].equals("pwd3"))
{
if(name[i+1].equals("user3"))
{
out.println("WelCome"+name[i+1]);
}
```

```
else
{
out.println("Unauthorised User");
}
}
else if(name[i].equals("pwd4"))
{
if(name[i+1].equals("user4"))
{
out.println("WelCome"+name[i+1]);
}
else
{
out.println("Unauthorised User");
}
}
}
catch(Exception e)
{
out.println("Invalid username/password");
}
}
}
}
```

Save as **web.xml**

[Save in path C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\login2\WEB-INF**]**

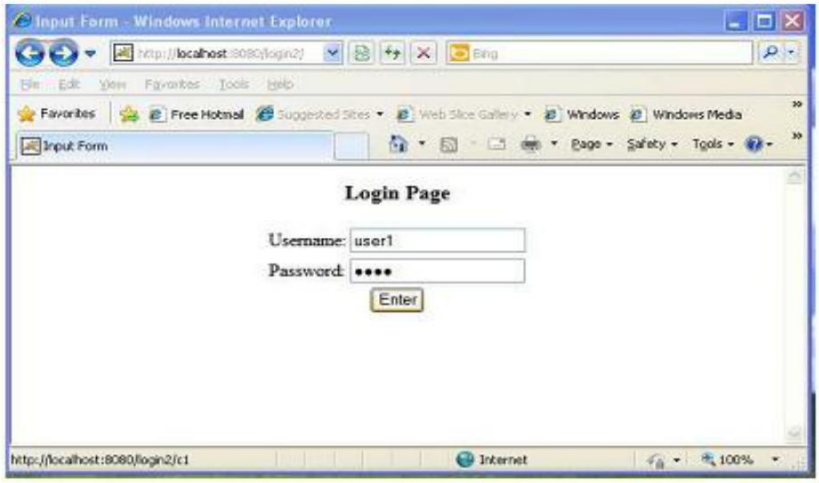
```
<web-app>
<welcome-file-list>
<welcome-file>login.html</welcome-file>
</welcome-file-list>
<servlet>
<servlet-name>servlet1</servlet-name>
<servlet-class>cookie1</servlet-class>
</servlet>
<servlet>
<servlet-name>servlet2</servlet-name>
<servlet-class>getcookie</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>servlet1</servlet-name>
<url-pattern>/c1</url-pattern>
</servlet-mapping>
<servlet-mapping>
<servlet-name>servlet2</servlet-name>
<url-pattern>/gc</url-pattern>
</servlet-mapping>
</web-app>
```

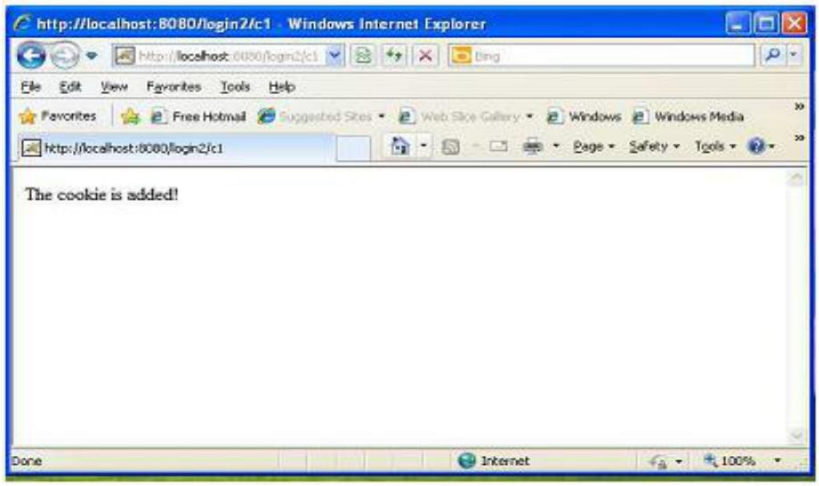
Save as **login.html**

[Save as C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\login2**]**

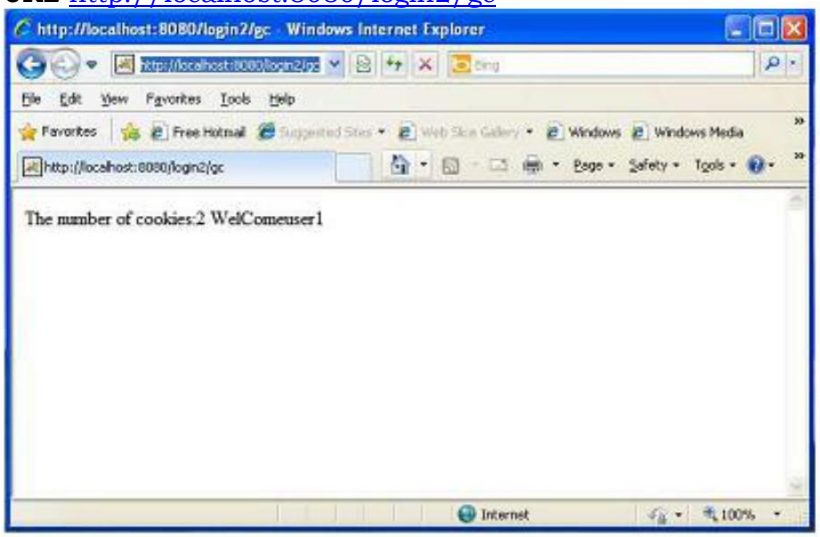
```
<html>
<head><title>Input Form</title></head>
<body>
<center>
<h3>Login Page</h3>
<form name="login" method="post" action="http://localhost:8080/login2/c1">
<table>
<tr>
<td>Username:</td>
<td><input type="text" name="username"></td>
</tr>
<tr>
<td>Password:</td>
<td><input type="password" name="password"></td>
</tr>
</table>
<input type="submit" value="Enter">
</form>
</center>
</body>
</html>
```

URL http://localhost:8080/login2
Username: user1
Password: pwd1

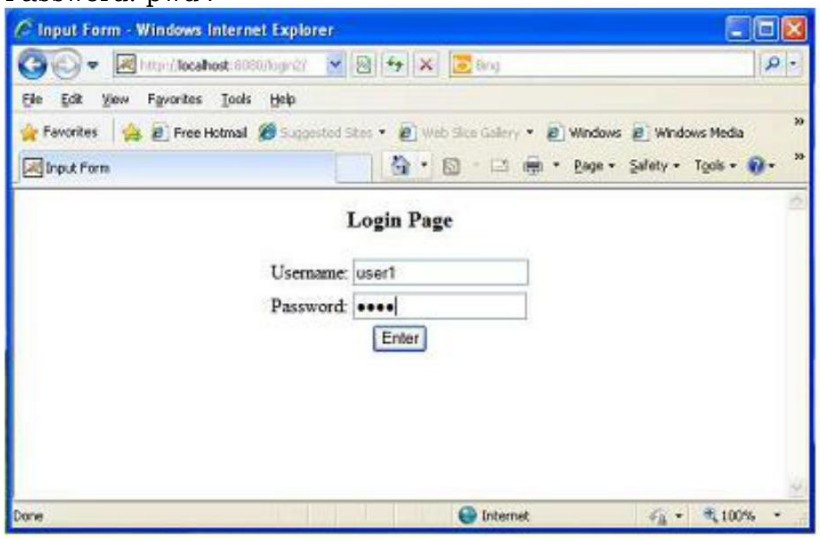




URL http://localhost:8080/login2/gc



http://localhost:8080/login2
Username: user1
Password: pwd4



URL http://localhost:8080/login2/gc

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

Save as **initparam.java**

[Save in path C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\login\WEBINF\classes**]**

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class initparam extends HttpServlet
{
public void doGet(HttpServletRequest req, HttpServletResponse res) throws
IOException,ServletException
{
res.setContentType("text/html");
PrintWriter out = res.getWriter();
String username1=getServletConfig().getInitParameter("username1");
String password1=getServletConfig().getInitParameter("password1");
String username2=getServletConfig().getInitParameter("username2");
String password2=getServletConfig().getInitParameter("password2");
String username3=getServletConfig().getInitParameter("username3");
String password3=getServletConfig().getInitParameter("password3");
String username4=getServletConfig().getInitParameter("username4");
String password4=getServletConfig().getInitParameter("password4");

String un= req.getParameter("username"); String p=
req.getParameter("password");
if(((username1.equals(un))&&(password1.equals(p)))||
((username2.equals(un))&&(password2.equals(p))||
((username3.equals(un))&&(password3.equals(p)))||
((username4.equals(un))& &(password4.equals(p)))) {

out.println("<b>WelCome to "+un+"</b>");
}
else
{
out.println("<b>Unathorised User</b>");
}
}
}
```

Save as **Web.xml**

[Save in path C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\login\WEB-INF**]**

```
<web-app>
<welcome-file-list>
<welcome-file>login.html</welcome-file>
</welcome-file-list>
<servlet>
<servlet-name>servlet10</servlet-name>
<servlet-class>initparam</servlet-class>
<init-param>
<param-name>username1</param-name>
```

```
<param-value>user1</param-value>
</init-param>
<init-param>
<param-name>password1</param-name>
<param-value>pwd1</param-value>
</init-param>
<init-param>
<param-name>username2</param-name>
<param-value>user2</param-value>
</init-param>
<init-param>
<param-name>password2</param-name>
<param-value>pwd2</param-value>
</init-param>
<init-param>
<param-name>username3</param-name>

<param-value>user3</param-value>
</init-param>
<init-param>
<param-name>password3</param-name>
<param-value>pwd3</param-value>
</init-param>
<init-param>
<param-name>username4</param-name>
<param-value>user4</param-value>
</init-param>
<init-param>
<param-name>password4</param-name>
<param-value>pwd4</param-value>
</init-param>
</servlet>
<servlet-mapping>

<servlet-name>servlet10</servlet-name>
<url-pattern>/initparam</url-pattern>
</servlet-mapping>
</web-app>
```

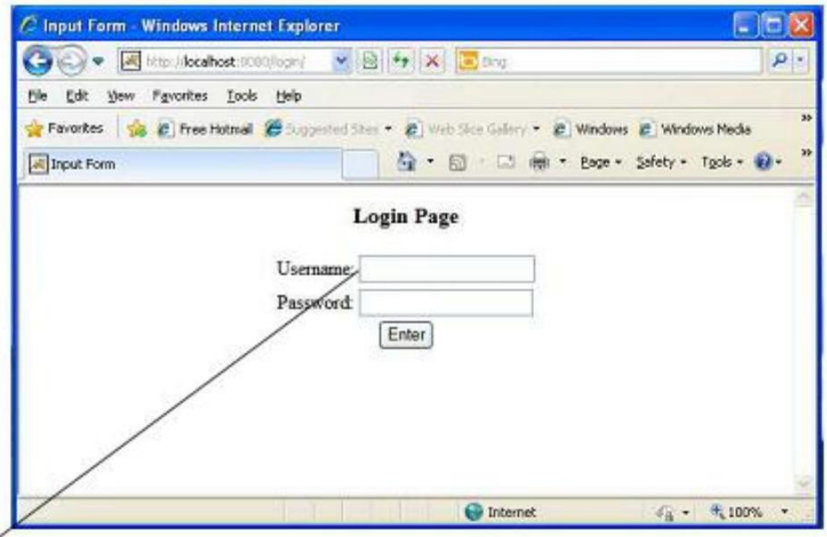
Save as **login.html**

[Save in path C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\login**]**

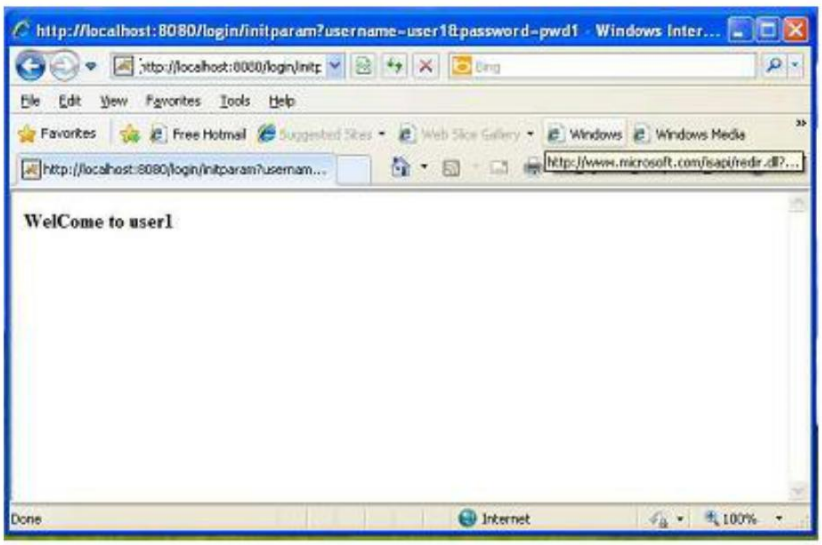
```
<html>
<head>
<title>Input Form</title>
</head>
<body>
<center>
<h3>Login Page</h3>
<form name="login" method="get" action="http://localhost:8080/login/initparam">
<table>
<tr>
<td>Username:</td>
<td><input type="text" name="username"></td>
</tr>
<tr>
```

```
<td>Password:</td>
<td><input type="password" name="password"></td>
</tr>
</table>
<input type="submit" value="Enter">
</form>
</center>
</body>
</html>
```

URL http://localhost:8080/login

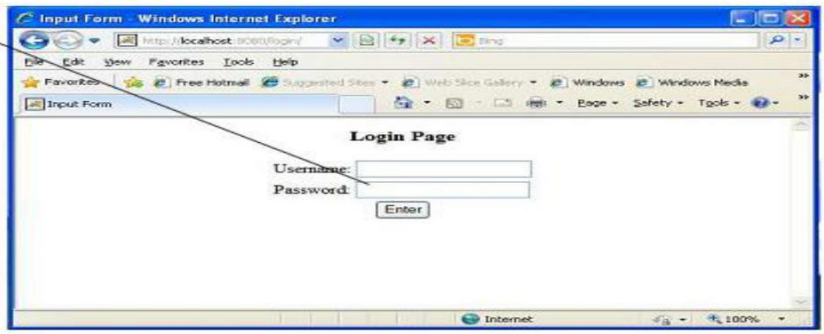


Username: user1
Password: pwd1



URL http://localhost:8080/login2

Username: user1
Password: pwd4



Result

Exercise

1.Create a HTML form with three input fields first name, last name and e-mail. Pass the values to servlet. In the servlet,verify all input fields are not null and display them back to the client.

Signature of the Faculty

WEEK-9

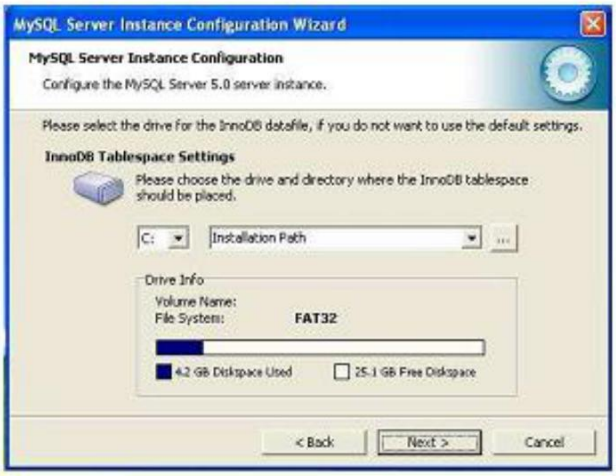
a) Install a database (Mysql or Oracle).
Create a table which should contain at least the following fields: name, password, email-id, phone number (these should hold the data from the registration form).
Practice 'JDBC' connectivity.
b) Write a java program/servlet/JSP to connect to that database and 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).Installing the database Mysql.


MySQL Server Instance Configuration Wizard
Welcome to the MySQL Server Instance Configuration Wizard 1.0.12.0
The Configuration Wizard will allow you to configure the MySQL Server 5.0 server instance. To Continue, click Next.
MySQL
Next >
Cancel


MySQL Server Instance Configuration Wizard
MySQL Server Instance Configuration
Configure the MySQL Server 5.0 server instance.
Please select a configuration type.
Detailed Configuration
Choose this configuration type to create the optimal server setup for this machine.
Standard Configuration
Use this only on machines that do not already have a MySQL server installation. This will use a general purpose configuration for the server that can be tuned manually.
< Back
Next >
Cancel

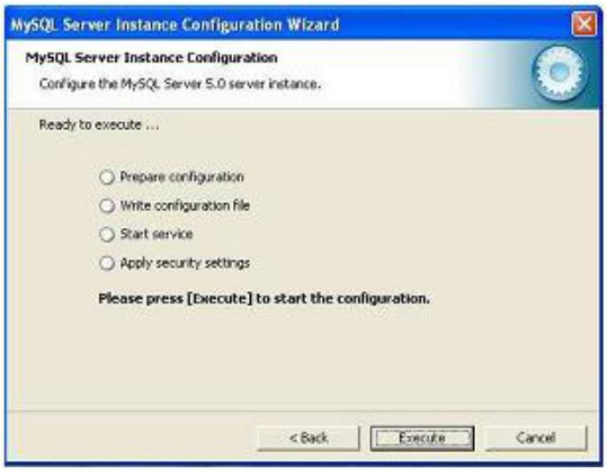

MySQL Server Instance Configuration Wizard
MySQL Server Instance Configuration
Configure the MySQL Server 5.0 server instance.
Please select a server type. This will influence memory, disk and CPU usage.
Developer Machine
This is a development machine, and many other applications will be run on it. MySQL Server should only use a minimal amount of memory.
Server Machine
Several server applications will be running on this machine. Choose this option for web/application servers. MySQL will have medium memory usage.
Dedicated MySQL Server Machine
This machine is dedicated to run the MySQL Database Server. No other servers, such as a web or mail server, will be run. MySQL will utilize up to all available memory.
< Back
Next >
Cancel


MySQL Server Instance Configuration Wizard
MySQL Server Instance Configuration
Configure the MySQL Server 5.0 server instance.
Please select the database usage.
Multifunctional Database
General purpose databases. This will optimize the server for the use of the fast transactional InnoDB storage engine and the high speed MyISAM storage engine.
Transactional Database Only
Optimized for application servers and transactional web applications. This will make InnoDB the main storage engine. Note that the MyISAM engine can still be used.
Non-Transactional Database Only
Suited for simple web applications, monitoring or logging applications as well as analysis programs. Only the non-transactional MyISAM storage engine will be activated.
< Back
Next >
Cancel

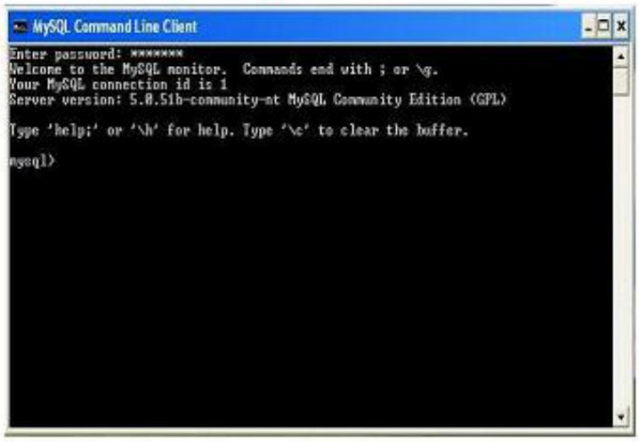

MySQL Server Instance Configuration Wizard
MySQL Server Instance Configuration
Configure the MySQL Server 5.0 server instance.
Please select the drive for the InnoDB datafile, if you do not want to use the default settings.
InnoDB Tablespace Settings
Please choose the drive and directory where the InnoDB tablespace should be placed.
C:
Installation Path
Drive Info
Volume Name:
File System: FAT32
4.2 GB Diskspace Used
25.1 GB Free Diskspace
< Back
Next >
Cancel


MySQL Server Instance Configuration Wizard
MySQL Server Instance Configuration
Configure the MySQL Server 5.0 server instance.
Please set the approximate number of concurrent connections to the server.
Decision Support (DSS)/OLAP
Select this option for database applications that will not require a high number of concurrent connections. A number of 20 connections will be assumed.
Online Transaction Processing (OLTP)
Choose this option for highly concurrent applications that may have at any one time up to 500 active connections such as heavily loaded web servers.
Manual Setting
Please enter the approximate number of concurrent connections.
Concurrent connections: 15
< Back
Next >
Cancel

New root password: manager
Confirm password: manager



Goto **Start->All Programs->MySql->MySql Server5.0->MySql Command Line Client**



Set the path for the Mysql and JSP (Set the Environment Variables)

user variable:
Variable name: classpath
Variable value: C:\Program Files\Apache Software Foundation\Tomcat 5.5\common\lib\servletapi.
jar;.;C:\Program Files\Apache Software Foundation\Tomcat 5.5\common\lib\jsp-api;.;C:\Program Files\Apache Software Foundation\Tomcat 5.5\common\lib\mysql-connector-java-5.0.8-bin.jar -
Download the file "**mysql-connector-java-5.0.8-bin.jar"** and copy the file in the path
"C:\Program Files\Apache Software Foundation\Tomcat 5.5\common\lib

```
mysql> create database students;
Query OK, 1 row affected (0.00 sec)
mysql> use students;
Database changed
mysql> create table CSE(sname varchar(10),password varchar(10),email
varchar(10),mobile varchar(10));
Query OK, 0 rows affected (0.06 sec)
mysql> select * from CSE;
Empty set (0.00 sec)
```

Save as **StudentForm.jsp**

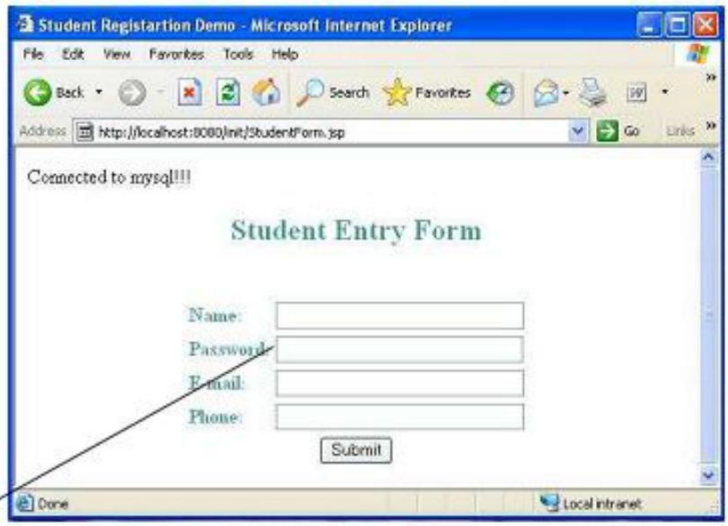
```
<%@ page language="java" import="java.sql.*" %>
<%@ page import="java.io.*" %>
<%
Connection conn=null;
ResultSet rs=null;
Statement stmt=null;
Class.forName("com.mysql.jdbc.Driver").newInstance();

conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/students","root","manager");
out.write("Connected to mysql!!!");
stmt=conn.createStatement();
if(request.getParameter("action") != null)
{
String sname=request.getParameter("sname");
String password=request.getParameter("password");
String email=request.getParameter("email");
```

```
String mobile=request.getParameter("mobile");
stmt.executeUpdate("insert into CSE values('"+sname+"','"+password+"','"+email+"','"+mobile+"');");
rs=stmt.executeQuery("select * from CSE");
%>
<html>
<head>
<style type="text/css">
h2,b
{
color:rgb(59,156,150);
}
</style>
</head>
<body>
<center>
<h2>Student List</h2>
<table border="1" cellspacing="0" cellpadding="0">
<tr>
<td><b>  S.No  </b></td>
<td><b>  Name  </b></td>
<!-- <td><b>Password</b></td> -->
<td><b>  Email  </b></td>
<td><b>  Mobile  </b></td>
</tr>
<%
int num=1;
while(rs.next())
{
%>
<tr>
<td><%=num%></td>
<td><%=rs.getString("sname")%></td>
<!-- <td><%=rs.getString("password")%></td> --
> <td><%=rs.getString("email")%></td>
<td><%=rs.getString("mobile")%></td> </tr>

<%
num++;

}
rs.close();
stmt.close();
conn.close();
%>
</table>
</center>
</body>
</html>
<%
}
else
{
%>

<html>
<head>
<title>Student Registartion Demo</title>
```

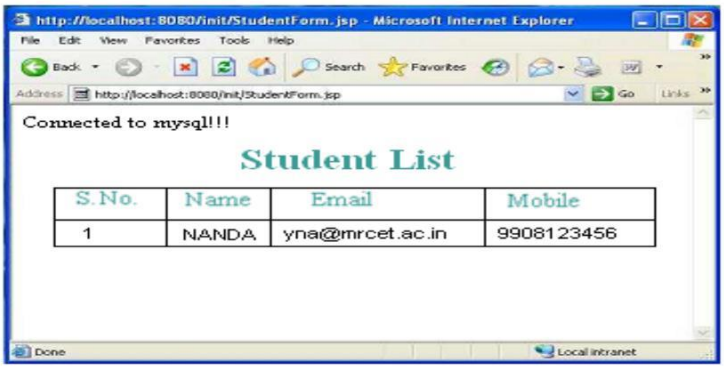
```
<script language="javascript">
function validation(Form_obj)
{
if(Form_obj.sname.value.length==0)
{
alert("Please,fill up the remaining information!!");
Form_obj.sname.focus();
return false;
}
if(Form_obj.password.value.length==0)
{
alert("Please,fill up the remaining information!!");
Form_obj.password.focus();
return false;
}
if(Form_obj.email.value.length==0)
{
alert("Please,fill up the remaining information!!");
Form_obj.email.focus();
return false;
}
if(Form_obj.mobile.value.length==0)
{
alert("Please,fill up the remaining information!!");
Form_obj.mobile.focus();
return false;
}
return true;
}
</script>
<style type="text/css">
dddb
{
color:rgb(59,156,150);
}
</style></head>
</body>
<center>
<form action="StudentForm.jsp" method="post" name="entry" onSubmit="return
validation(this)"> <input type="hidden" value="list" name="action">
<table border="0" cellpadding="0" cellspacing="0">
<tr>
<td>
<table>
<tr>
<td colspan="2" align="center">
<h2><b>Student Entry Form</b></h2></td>
</tr>
<tr>
<td colspan="2"> </td>
</tr>
<tr>
<td><b>Name:</b></td>
<td><input name="sname" type= "text" size="30"></td>
</tr>
<tr>
<td><b>Password:</b></td>
```

```
<td><input name="password" type= "password" size="30"></td>
</tr>
<tr>
<td><b>E-mail:</b></td>
<td><input name="email" type="text" size="30"></td>
</tr>
<td><b>Phone:</b></td>
<td><input name="mobile" type="text" size="30"></td>
</tr>
<tr>
<td colspan="2" align="center">
<input type="submit" value="Submit"></td>
</tr>
</table>
</td>
</tr>
</table>
</form>
</center>
</body>
</html>

<%
}
%>
```

Save the **StudentForm.jsp** file in the path
C:\Program Files\Apache Software Foundation\Tomcat 5.5\webapps\jsp
Now start the tomcat services and URL **http://localhost:8080/jsp/StudentForm.jsp**
Enter the student details: name, password, email, mobile



Name: yna
Password: pwd
Email: yna@mrcet.ac.in
Phone: 9908123456



Press **Submit** button the data entered in the fields will be added to the database and shown as in the given below

Check whether data added in the database
Goto **Start->All Programs->MySql->MySql Server5.0->MySql Command Line Client**
mysql> select * from CSE;

```
+---------- +------------- +----------------------- +---------------------  +
| sname | password | email                    | mobile       |
+---------- +------------- +------------------------------------------------ +
| yna       | pwd       | yna@mrcet.ac.in | 9908123456 |
+---------- +------------- +----------------------- +---------------------  +
```

1 row in set (0.00 sec)

To see the student details:

Save as **StudentDetails.jsp**

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

Connection conn=null;
ResultSet rs=null;
Statement stmt=null;
Class.forName("com.mysql.jdbc.Driver").newInstance();
conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/students","root","manager");
out.write("Connected to mysql!!!");
stmt=conn.createStatement();
rs=stmt.executeQuery("select * from CSE");
%>
<html><head>
<style type="text/css">
h2,b
{
color:rgb(59,156,150);
}
</style></head>
<body>
<center>
<h2>Student List</h2>
<table border="1" cellspacing="0" cellpadding="0">
<tr>
<td><b>  S.No  </b></td>
<td><b>  Name  </b></td>
```

```
<td><b>  Email  </b></td>
<td><b>  Mobile  </b></td>
</tr>
<%
int num=1;
while(rs.next())
{
%>
<tr>
<td><%=num%></td>
<td><%=rs.getString("sname")%></td>
<td><%=rs.getString("email")%></td>
<td><%=rs.getString("mobile")%></td>
</tr>
<%
num++;
}
rs.close();
stmt.close();
conn.close();
%>
</table>
</center>
</body>
</html>
```

URL: **http://localhost:8080/jsp/StudentDetails.jsp**

Exercises

1)Write a program to transfer some amount from source account to destination account .

2)Write a program to list all the students available from the database.

Signature of the Faculty

Signature of the