

ST. JOSEPH UNIVERSITY IN TANZANIA

St. Joseph College of Engineering and Technology
Internet Programming Laboratory
055CS68

By Robert Innocent Karamagi



List of Experiments

1. Creation of HTML pages with frames, links, tables and other tags.
2. Usage of internal and external CSS along with HTML pages.
3. Client Side Programming
 - a. Java script for displaying date and comparing two dates.
 - b. Form Validation including text field, radio buttons, check boxes, list box and other controls.
4. Usage of ASP/JSP objects response, Request, Application, Session, Server, ADO etc.
 - a. Writing online applications such as shopping, railway/air/bus ticket reservation system with the set of ASP/JSP pages.
 - b. Using sessions and cookies as part of the web application
5. Writing Servlet Program using HTTP Servlet.
6. Any online application with database access.
7. Creation of XML document for a specific domain
8. Writing DTD or XML schema for the domain specific XML document
9. Parsing an XML document using DOM and SAX Parsers
10. Sample web application development in the open source environment

CREATION OF HTML PAGES WITH FRAMES, LINKS, TABLES AND OTHER TAGS

AIM

The aim of the experiment is to implement the creation of Hypertext Markup Language pages with frames, links, tables and other tags.

APPARATUS REQUIRED

Notepad

Internet Browser

THEORY

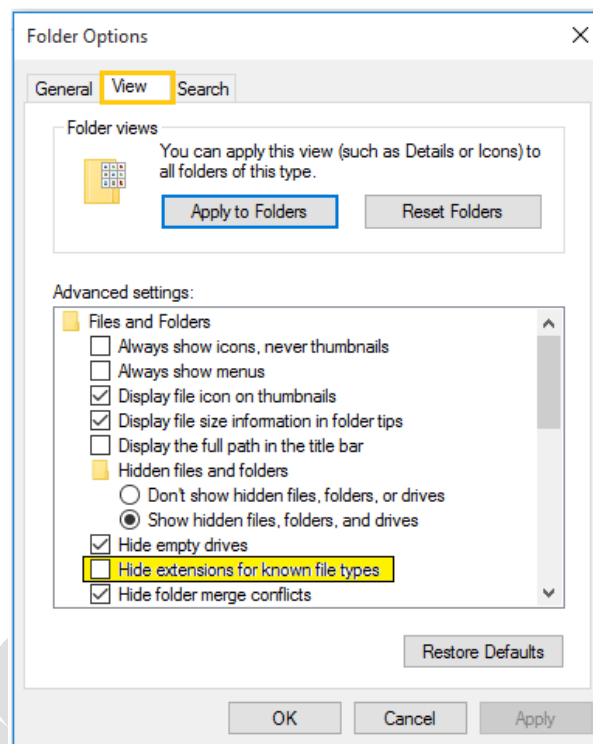
Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages.

HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.

HTML elements are delineated by tags, written using angle brackets. Tags such as `<img />` and `<input />` directly introduce content into the page. Other tags such as `<p>...</p>` surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags but use them to interpret the content of the page.

PROCEDURE

1. Enter the Control Panel and open or search for “Folder Options” (Windows 7) / “File Explorer Options” (Windows 10). Once opened, click and the view tab and under the Advanced Settings options make sure that the check box labelled “Hide extensions for known file types” is unchecked”.

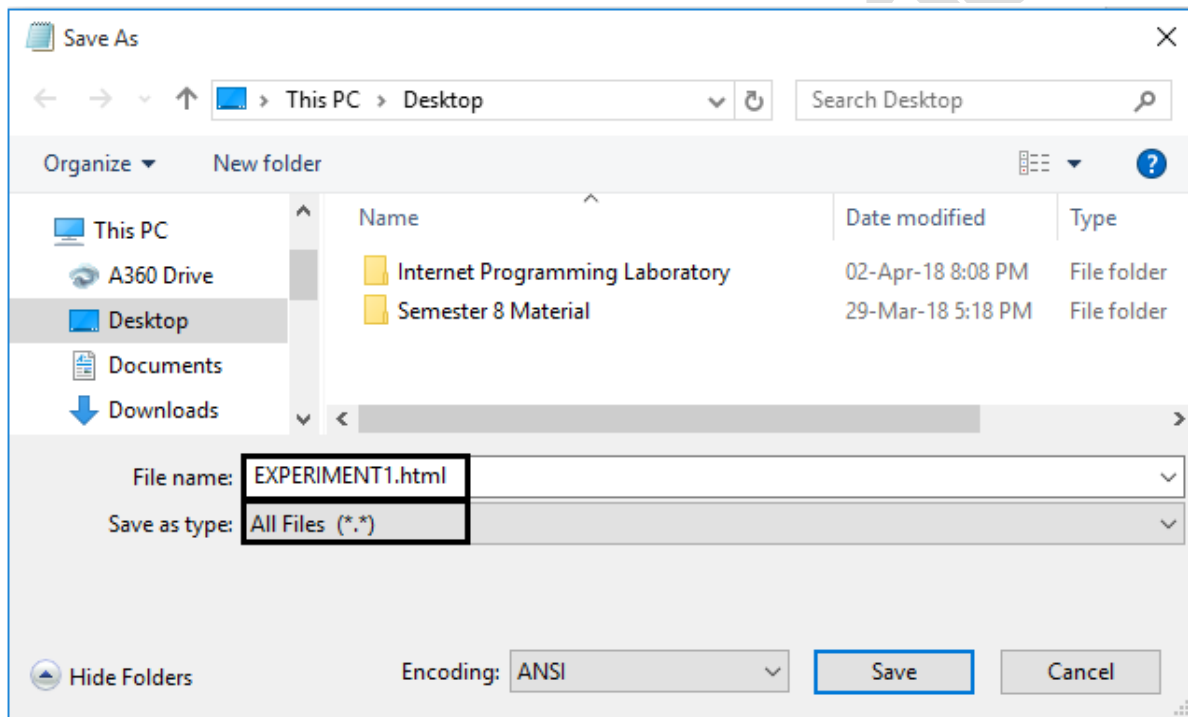


2. Open the notepad; Copy the codes in the “Program” section into the notepad and save each file with the .html extension as shown.

```
Untitled - Notepad
File Edit Format View Help

<head><title>Robert Innocent Karamagi</title>
</head>
<frameset rows="25%,*">
<frame src="frame1.html">
<frameset cols="25%,*">
<frame src="frame2.html" name="f2">
<frame src="frame3.html" name="f3">
</frameset>
</html>

Ln 10, Col 1
```



3. Make sure all the html files and .jpg picture are saved into the same folder as shown below. The file name of the picture should correspond to the code in the "picture.html" file.

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

 about.html	01-Apr-18 9:47 PM	Firefox HTML Document	1 KB
 contact.html	26-Mar-17 5:49 PM	Firefox HTML Document	1 KB
 EXPERIMENT1.html	26-Mar-17 5:50 PM	Firefox HTML Document	1 KB
 expname.html	08-May-17 4:31 PM	Firefox HTML Document	1 KB
 frame1.html	08-Aug-17 8:22 PM	Firefox HTML Document	1 KB
 frame2.html	28-Jul-17 10:53 AM	Firefox HTML Document	1 KB
 frame3.html	26-Mar-17 5:49 PM	Firefox HTML Document	1 KB
 modules.html	25-Mar-17 7:39 PM	Firefox HTML Document	1 KB
 picture.html	01-Apr-18 10:00 PM	Firefox HTML Document	1 KB
 robert.jpg	25-Mar-17 5:19 PM	PicosmosShows (jpg)	66 KB

4. Open the file named “EXPERIMENT1.html” and click on the side links to verify the output.

PROGRAM

about.html

```
<html>
<head><title>About</title>
</head>
<body bgcolor="red">
<font color=white size="5" align=center>
<p>
Robert Innocent Karamagi was born on 20th June 1992. He is a 6th semester student
in the Department of Computer Science and Engineering at St. Joseph College of
Engineering and Technology.
</p>
</font>
</body>
</html>
```

contact.html

```
<html>
<head><title>Contact</title>
</head>
<body bgcolor="springgreen">
<p align=center>
<font color=black size="5">
Name: Robert Innocent Karamagi<br>
Location: Mbezi, Luguruni<br>
City : Dar es salaam<br>
Country : Tanzania<br>
Contact No: 0762 087 332<br>
Email: robert_karamagi@yahoo.com<br>
</font>
</p>
</body>
</html>
```

EXPERIMENT.html

```
<html>
<head><title>Robert Innocent Karamagi</title>
</head>

<frameset rows="25%,*">
<frame src="frame1.html">
<frameset cols="25%,*">
<frame src="frame2.html" name="f2">
<frame src="frame3.html" name="f3">
</frameset>
```

```
</html>
```

expname.html

```
<html>
<head><title>Experiment</title>
</head>
<body bgcolor="cyan">
<p align="center">
<font color=black size=5>
Creation of Hyper Text Mark-up Language pages with frames, links, tables and other tags
</font>
</p>
</body>
</html>
```

frame1.html

```
<html>
<head><title>Frame 1</title>
</head>
<body bgcolor="green">
<h1 style=font size ="15pt">
<marquee >
Robert Innocent Karamagi </marquee>
</h1>
</body>
</html>
```

frame2.html

```
<html>
<head><title>Frame 2</title>
<body bgcolor="yellowgreen">
<h1>Click to Learn</h1>
<a href="frame3.html" target=f3>Home</a><br>
<a href="about.html" target=f3>About Robert</a><br>
<a href="modules.html" target=f3>Modules</a><br>
<a href="expname.html" target=f3>Experiment</a><br>
<a href="contact.html" target=f3>Contact Details</a><br>
<a href="picture.html" target=f3>Picture</a><br>
</body>
</html>
```

frame3.html

```
<html>
<head><title>Home</title>
</head>
<body bgcolor="lightgreen">
<h2> <center>Learn about Robert</center></h2>
</body>
</html>
```

modules.html

```
<html>
<head><title>Modules</title>
</head>
<body bgcolor="yellow">
```

<div align="center">

<table border=2>

<tr>

<th>Module code</th>

<th>Module name</th>

</tr>

<tr>

<td>055CS61</td>

<td>Computer Networks</td>

</tr>

<tr>

<td>055CS62</td>

<td>Digital Signal Processing</td>

</tr>

<tr>

<td>055CS63</td>

<td>Software Engineering </td>

</tr>

<tr>

<td>055CS64</td>

<td>Network Protocols, Management & Security</td>

</tr>

<tr>

<td>055CS65</td>

<td>Web Technology</td>

</tr>

<tr>

<td>055CS66</td>

<td>Principles of Compiler Design</td>

</tr>

<tr>

```
<td>055CS67</td>
<td>Network Programming Laboratory</td>
</tr>
<tr>
<td>055CS68</td>
<td>Internet Programming Laboratory</td>
</tr>
<tr>
<td>055PJ69</td>
<td>Mini Project</td>
</tr>
</table>
</div>
</body>
</font>
</html>
```

picture.html

```
<html>

<head><title>Picture</title>
</head>

<body bgcolor="magenta">
<p align=center>
<font color=green size=5>
Robert Innocent Karamagi</font><br>
</p>

<center align="middle">
```

```

<br><br>
Enter your name: <input type="text"/> <!-- This is for a text input --><br>
Comments: <br>
<textarea rows="4" cols="50"> </textarea><br> <!-- This is for a text area -->
Upload a file: <input type="file" /> <!-- This is for browsing for files --><br>
Like: <input type="checkbox" /> <!-- This is for checkboxes --><br>
</center>

</body>
</html>

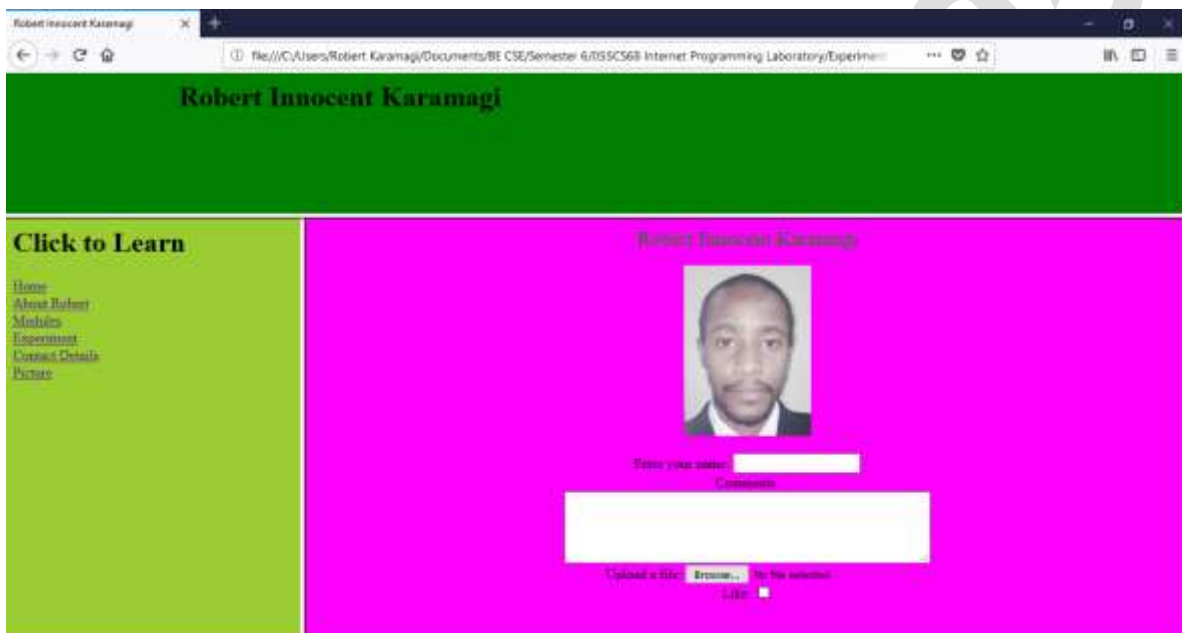
```

SAMPLE OUTPUT









RESULT

Thus, the experiment to implement the creation of Hypertext Markup Language pages with frames, links, tables and other tags was executed successfully and the output is verified.

USAGE OF INTERNAL AND EXTERNAL CSS ALONG WITH HTML PAGES

AIM

The aim of the experiment is to implement the usage of Internal and External Cascading Style Sheets along with HTML pages.

APPARATUS REQUIRED

Notepad

Internet Browser

THEORY

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

An internal stylesheet holds the CSS code for the webpage in the head section of the particular file. This makes it easy to apply styles like classes or id's in order to reuse the code. The downside of using an internal stylesheet is that changes to the internal stylesheet only effect the page the code is inserted into.

The External Stylesheet is a .css file that you link your website to. This makes it so that whatever you change in the .css sheet, will affect every page in your website. This prevents you from having to make many code changes in each page. This is for "global" site changes.

The Inline style is specific to the tag itself. The inline style uses the HTML "style" attribute to style a specific tag. This is not recommended, as every CSS change has to be made in every tag that has the inline style applied to it. The Inline style is good for one an individual CSS change that you do not use repeatedly through the site.

PROCEDURE

INTERNAL CSS

5. Open the notepad; Copy the codes in the “Program” section into the notepad and save each file with the .html extension as shown below.
6. Make sure all the html files and .jpg picture are saved into the same folder as shown below. The .jpg file name should correspond to the code in the “slide.html” file.

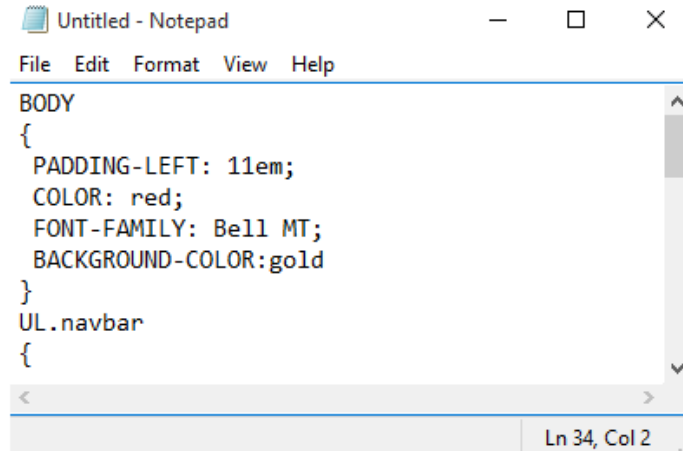
``

	about.html	27-Mar-17 12:36 PM	Firefox HTML Document	1 KB
	contact.html	07-May-17 9:46 PM	Firefox HTML Document	1 KB
	css.jpg	27-Mar-17 12:29 PM	PicosmosShows (jpg)	73 KB
	experiment.html	08-May-17 12:33 ...	Firefox HTML Document	1 KB
	internalcss.html	07-May-17 8:53 PM	Firefox HTML Document	2 KB
	slide.html	07-May-17 8:24 PM	Firefox HTML Document	1 KB
	student.html	07-May-17 10:01 P...	Firefox HTML Document	1 KB

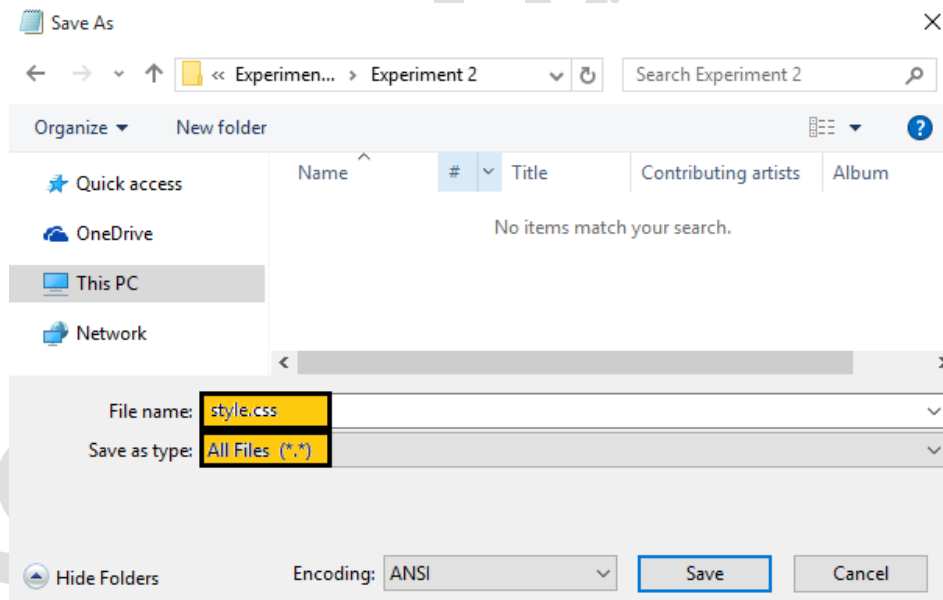
7. Open the file named “internalcss.html” and verify the output.

EXTERNAL CSS

1. Open the notepad and copy the code in the “style.css” program into it.
2. Save the file with the .css extension as shown below.

A screenshot of a Notepad window titled "Untitled - Notepad". The menu bar includes "File", "Edit", "Format", "View", and "Help". The text area contains CSS code for a body and a navbar. The status bar at the bottom right indicates "Ln 34, Col 2".

```
BODY
{
  PADDING-LEFT: 11em;
  COLOR: red;
  FONT-FAMILY: Bell MT;
  BACKGROUND-COLOR: gold
}
UL.navbar
{
```



3. Copy the codes in the “Program” section into the notepad and save each file with the .html extension as shown below.

4. Make sure all the html files, .jpg picture and .css cascading style sheet are saved into the same folder as shown below.

	about.html	27-Mar-17 12:36 PM	Firefox HTML Document	1 KB
	contact.html	07-May-17 9:46 PM	Firefox HTML Document	1 KB
	css.jpg	27-Mar-17 12:29 PM	PicosmosShows (jpg)	73 KB
	experiment.html	08-May-17 12:33 ...	Firefox HTML Document	1 KB
	externalcss.html	07-May-17 9:18 PM	Firefox HTML Document	1 KB
	slide.html	07-May-17 8:24 PM	Firefox HTML Document	1 KB
	student.html	07-May-17 10:01 P...	Firefox HTML Document	1 KB
	style.css	08-May-17 7:02 AM	Cascading Style Sheet Document	1 KB

5. Open the file named “externalcss.html” and verify the output.

PROGRAM

about.html

```
<html>
<head><title>About</title>
</head>
<body bgcolor="red">
<font color=white size="5" align=center>
<p>
```

Cascading Style sheet is the language for describing the presentation of Web pages, including colours, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers. CSS is independent of HTML and can be used with any XML-based mark-up language.

```
</p>
</font>
</body>
</html>
```

contact.html

```
<html>
<head><title>Contact</title>
</head>
<body bgcolor="springgreen">
<p align=center>
<font color=black size="5">
Name: Robert Innocent Karamagi<br>
Location: Mbezi, Luguruni<br>
City : Dar es Salaam<br>
Country : Tanzania<br>
Contact No: 0762 087 332<br>
Email: robert_karamagi@yahoo.com<br>
</font>
</p>
</body>
</html>
```

experiment.html

```
<html>
<head><title>Experiment</title>
</head>
<body bgcolor="cyan">
<p align="center">
<font color=black size=5>
Usage of Internal and External Cascading Style Sheets along with
Hyper Text Mark-up Language pages.
</font>
</p>
```

```
</body>
```

```
</html>
```

internalcss.html

```
<html>
```

```
<head><title>Internal Cascading Style sheet</title>
```

```
<style>
```

```
BODY
```

```
{
```

```
PADDING-LEFT: 11em;
```

```
COLOR: darkblue;
```

```
FONT-FAMILY: Georgia;
```

```
BACKGROUND-COLOR: cyan
```

```
}
```

```
UL.navbar
```

```
{
```

```
PADDING-RIGHT: 0px;
```

```
PADDING-LEFT: 0px;
```

```
LEFT: 1em;
```

```
PADDING-BOTTOM: 0px;
```

```
MARGIN: 0px;
```

```
WIDTH: 9em;
```

```
PADDING-TOP: 0px;
```

```
LIST-STYLE-TYPE: none;
```

```
POSITION: absolute;
```

```
TOP: 2em
```

```
}
```

```
H1
```

```
{
```

```
FONT-FAMILY: Helvetica;
```

```

}
UL.navbar LI
{
  BORDER-RIGHT: 1em solid;
  PADDING-RIGHT: 0px;
  PADDING-LEFT: 0px;
  BACKGROUND: lightblue;
  PADDING-BOTTOM: 0.3em;
  MARGIN: 0.5em 0px;
  PADDING-TOP: 0.3em
}
</style>
</head>
<body>
<!--site navigation menu-->
<ul class="navbar">
<li><a href="about.html">About</a>
<li><a href="contact.html">Contact</a>
<li><a href="experiment.html">Experiment</a>
<li><a href="student.html">Student Details</a>
<li><a href="slide.html">Slide</a>
</ul>
<!--Main content-->
<h1>Internal Cascading Style sheet</h1>
<p>An Internal Style Sheet is a section on an HTML page that contains style definitions.
Internal style sheets are defined by using the "style" tag within the "head" area of the document.
<br>
</body>
</html>

```

externalcss.html

```
<html>
<head>
<title>External Cascading Style Sheet</title>
<link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
<!--site navigation menu-->
<ul class="navbar">
<li><a href="about.html">About</a>
<li><a href="contact.html">Contact</a>
<li><a href="experiment.html">Experiment</a>
<li><a href="student.html">Student Details</a>
<li><a href="slide.html">Slide</a>
</ul>
<!--Main content-->
<h1>External Cascading Style sheet</h1>
<p>An External Cascading Style Sheet is a file containing only CSS syntax
(no document content or elements) and should carry a Multi-purpose Internet
Mail Extensions (MIME) type of "text/css." The style information is not
explicitly tied directly to the document's elements, so Selector syntax is used
to specify what styles attach to which portions of the document tree. <BR>
</body>
</html>
```

slide.html

```
<html>
<head><title>Slide</title>
</head>
```

```
<body bgcolor="magenta">
<p align=center>
<font color=green size=5>
Cascading Style Sheet</font>
</p>
<center align="middle">

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

student.html

```
<html>
<head><title>Student Details</title>
</head>
<body bgcolor="yellow">
<div align="center">
<table border=2>
<tr>
<th>Student Name</th>
<th>Registration Number</th>
</tr>
<tr>
<td>Robert Innocent Karamagi</td>
<td>151730551333</td>
</tr>
</table>
</div>
</body>
</html>
```

style.css

BODY

```
{  
  PADDING-LEFT: 11em;  
  COLOR: red;  
  FONT-FAMILY: Bell MT;  
  BACKGROUND-COLOR:gold  
}
```

UL.navbar

```
{  
  PADDING-RIGHT: 0px;  
  PADDING-LEFT: 0px;  
  LEFT: 1em;  
  PADDING-BOTTOM: 0px;  
  MARGIN: 0px;  
  WIDTH: 9em;  
  PADDING-TOP: 0px;  
  LIST-STYLE-TYPE: none;  
  POSITION: absolute;  
  TOP: 2em  
}
```

H1

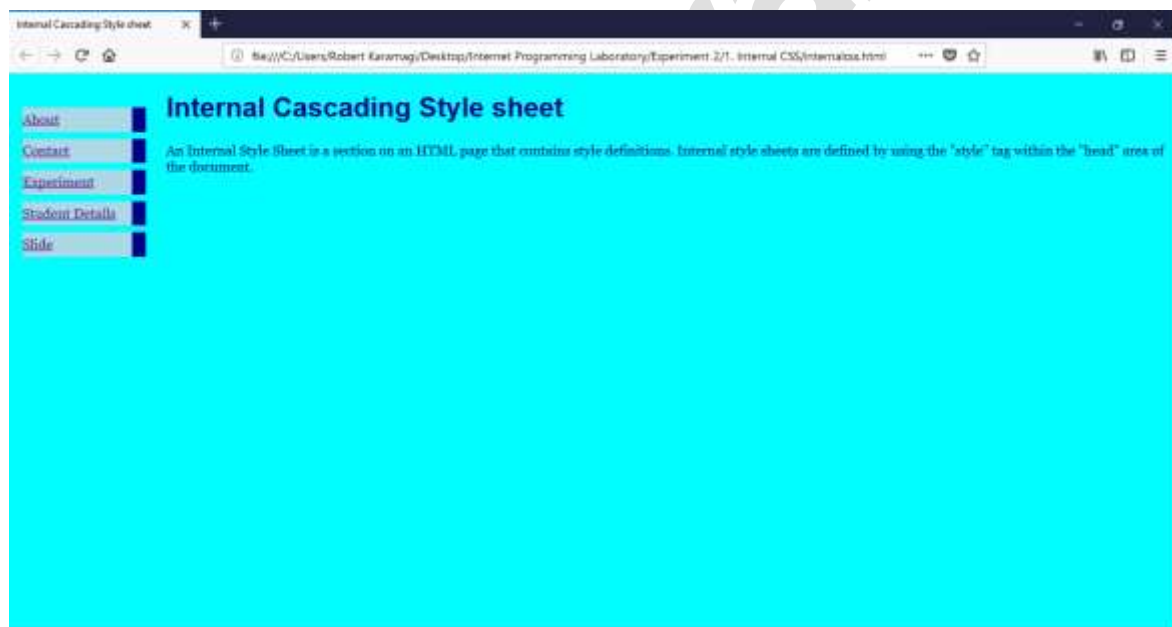
```
{  
  FONT-FAMILY: Arial;  
}
```

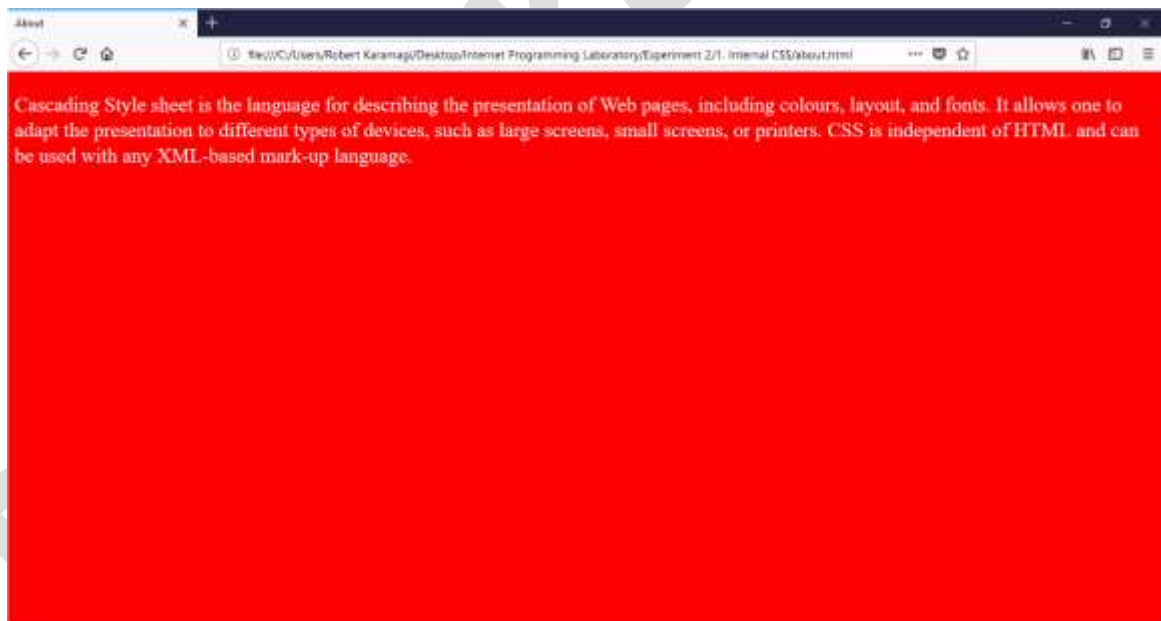
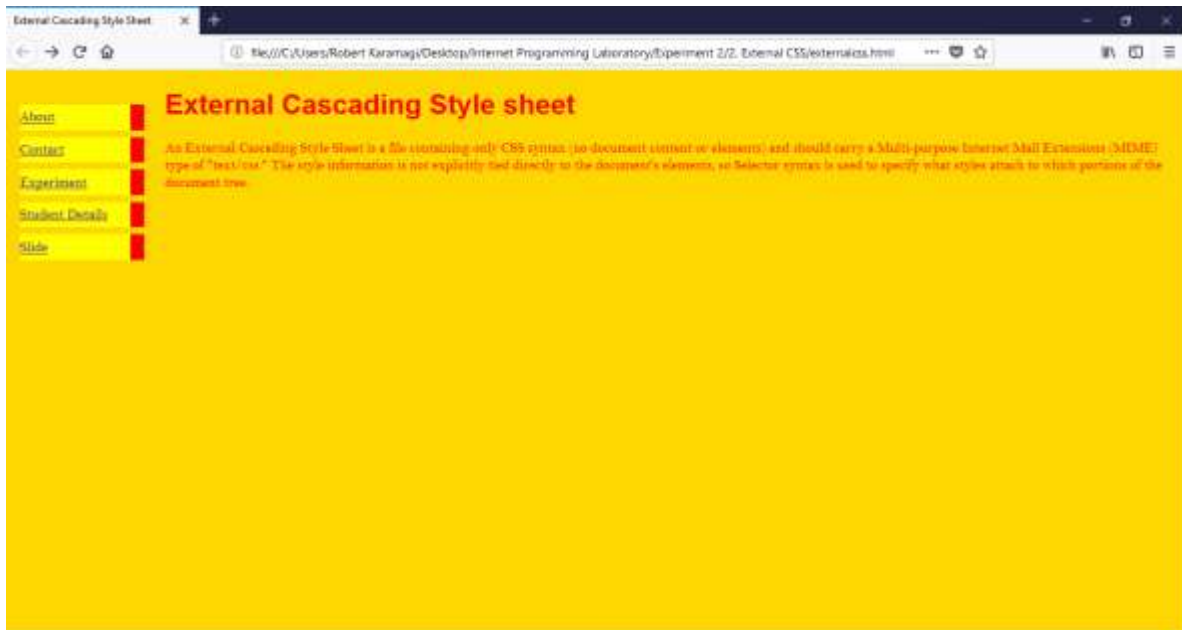
UL.navbar LI

```
{  
  BORDER-RIGHT: 1em solid;  
  PADDING-RIGHT: 0px;
```

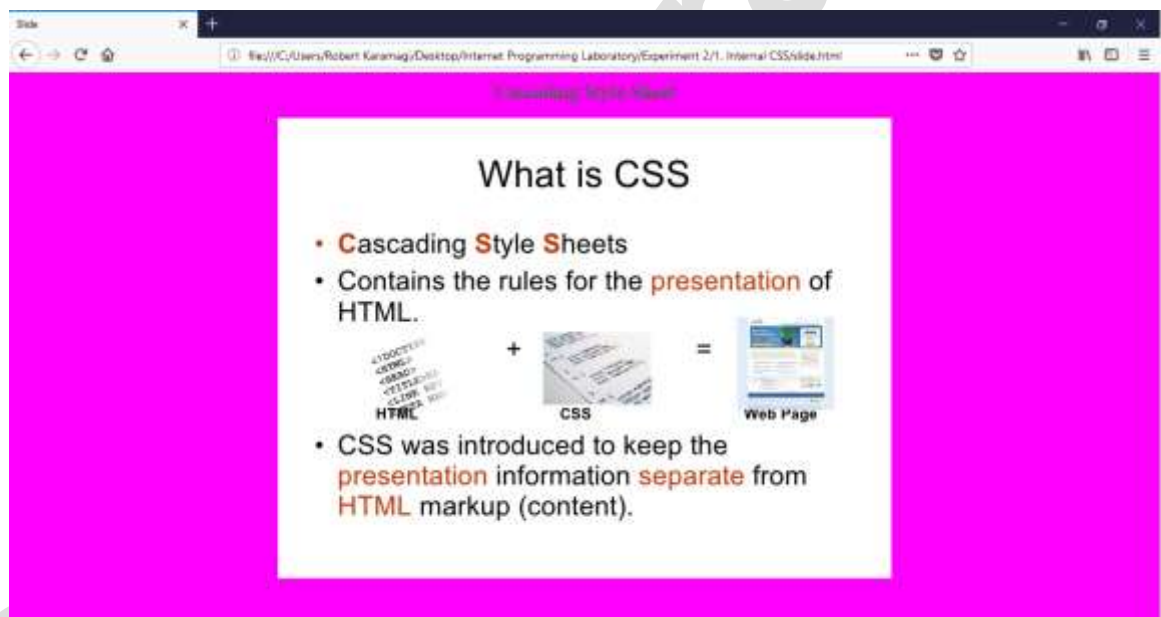
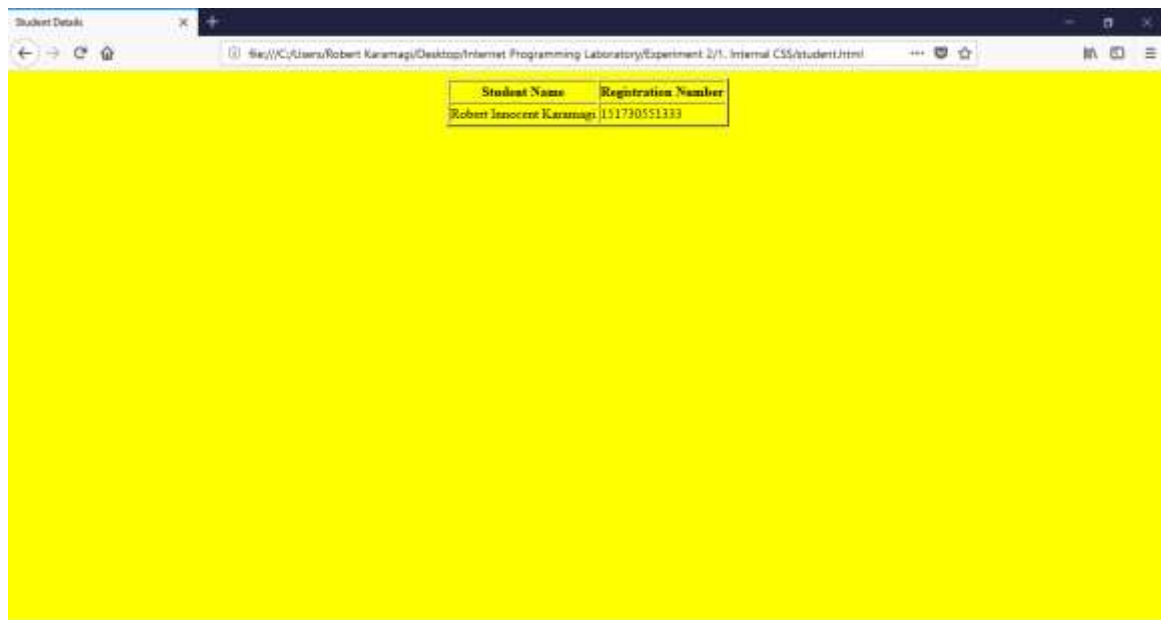
```
PADDING-LEFT: 0px;  
BACKGROUND: yellow;  
PADDING-BOTTOM: 0.3em;  
MARGIN: 0.5em 0px;  
PADDING-TOP: 0.3em  
}
```

SAMPLE OUTPUT









RESULT

Thus, the experiment to implement the usage of Internal and External Cascading Style Sheets along with HTML pages was executed successfully and the output is verified.

JAVASCRIPT FOR DISPLAYING THE DATE AND COMPARING TWO DATES

AIM

The aim of the experiment is to implement a Java Script for displaying the date and comparing two dates.

APPARATUS REQUIRED

Notepad

Internet Browser

THEORY

JavaScript, often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm. Alongside HTML and CSS, JavaScript is one of the three core technologies of World Wide Web content engineering. It is used to make dynamic webpages interactive and provide online programs, including video games. The majority of websites employ it, and all modern web browsers support it without the need for plug-ins by means of a built-in JavaScript engine.

Each of the many JavaScript engines represent a different implementation of JavaScript, all based on the ECMAScript specification, with some engines not supporting the spec fully, and with many engines supporting additional features beyond ECMA (European Computer Manufacturers Association).

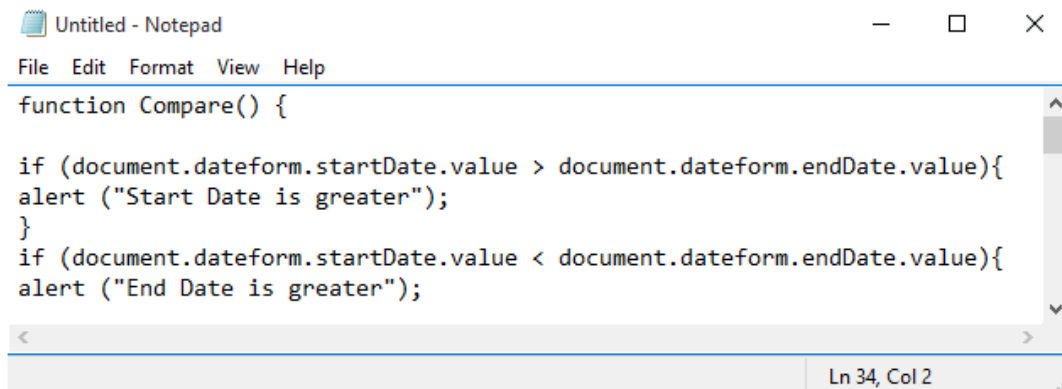
As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has an API (Application Programming Interface) for working with text, arrays, dates, regular expressions, and basic manipulation of the DOM (Document Object Model), but the language itself does not include any

I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded. Initially only implemented client-side in web browsers, JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases, and in non-web programs such as word processors and PDF (Portable Document Format) software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.

Although there are strong outward similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design; JavaScript was influenced by programming languages such as Self and Scheme.

PROCEDURE

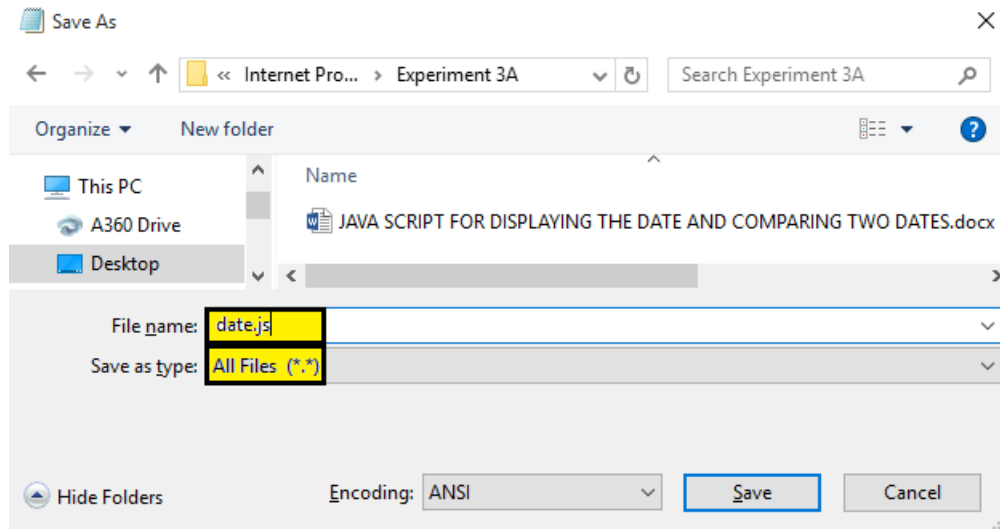
1. Open the notepad and copy the code in the “date.js” program into it.



```
function Compare() {  
  
    if (document.dateform.startDate.value > document.dateform.endDate.value){  
        alert ("Start Date is greater");  
    }  
    if (document.dateform.startDate.value < document.dateform.endDate.value){  
        alert ("End Date is greater");  
    }  
}
```

Ln 34, Col 2

2. Save the file with the .js extension as shown.



3. Copy the date.html program into the notepad and save it with the .html extension.
4. Make sure the date.html file, and date.js JavaScript file are saved into the same directory as shown below.

	date.html	06-Apr-18 9:53 AM	Firefox HTML Document	1 KB
	date.js	06-Apr-18 3:22 PM	JavaScript File	1 KB

5. Open the file named “date.html” and verify the output.

PROGRAM

date.html

```
<html>
<head>
<title>Date Comparison</title>

<script src="date.js">
```

```
</script>
```

```
</head>
```

```
<body bgcolor="lightblue">
```

```
<form name="dateform">
```

```
Start Date: <b><input type="text" name="startDate" size="14"></b><br><br>
```

```
End Date: <b><input type="text" name="endDate" size="14"></b><br/><br/>
```

```
<a href="javascript:Compare();">Compare</a><br><br>
```

```
<button onclick="DateTime()">Date and Time</button>
```

```
</form>
```

```
</body>
```

```
</html>
```

date.js

```
function Compare() {
```

```
if (document.dateform.startDate.value > document.dateform.endDate.value){
```

```
    alert ("Start Date is greater");
```

```
}
```

```
if (document.dateform.startDate.value < document.dateform.endDate.value){
```

```
    alert ("End Date is greater");
```

```
}
```

```
if (document.dateform.startDate.value == document.dateform.endDate.value){
```

```
    alert ("Dates are the same");
```

```
}
```

```
}
```

```
function DateTime(){
```

```
document.write("<html><head><title>Date and Time</title></head>")
```

```
document.write("<body bgcolor=lightgreen>")
```

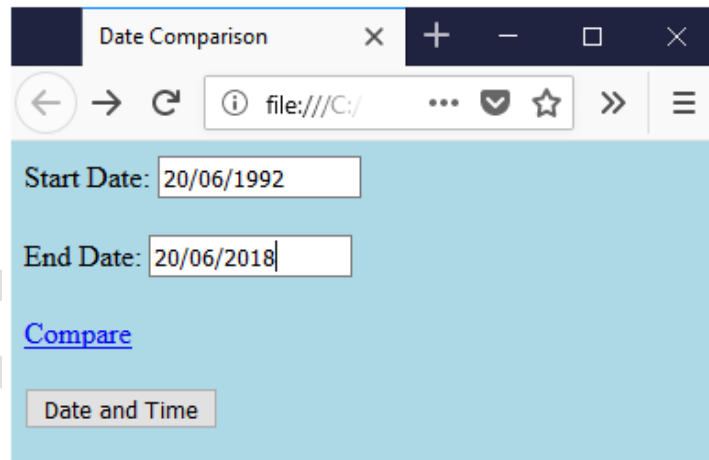
```
var currentDate = new Date(),
```

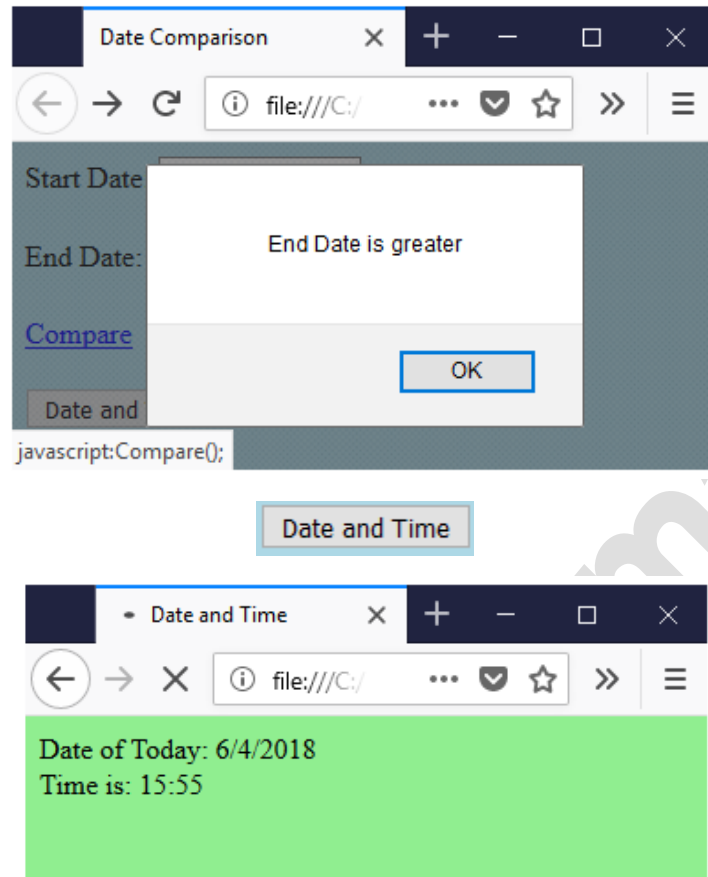
```
    day = currentDate.getDate(),
```

```
month = currentDate.getMonth()+1,
year = currentDate.getFullYear()
document.write("Date of Today: "+day + "/" + month + "/" + year)
document.write("<br>")
var currentTime = new Date(),
hours = currentTime.getHours(),
minutes = currentTime.getMinutes();

if (minutes < 10) {
    minutes = "0" + minutes;
}
document.write("Time is: "+hours + ":" + minutes)
document.write("</body></html>")
}
```

SAMPLE OUTPUT





RESULT

Thus, the experiment to implement a Java Script for displaying the date and comparing two dates was executed successfully and the output is verified.

FORM VALIDATION INCLUDING TEXT FIELDS, RADIO BUTTONS, CHECK BOXES, LIST BOXES AND OTHER CONTROLS

AIM

The aim of the experiment is to implement form validation including text fields, radio buttons, check boxes, list boxes and other controls.

APPARATUS REQUIRED

Notepad

Internet Browser

THEORY

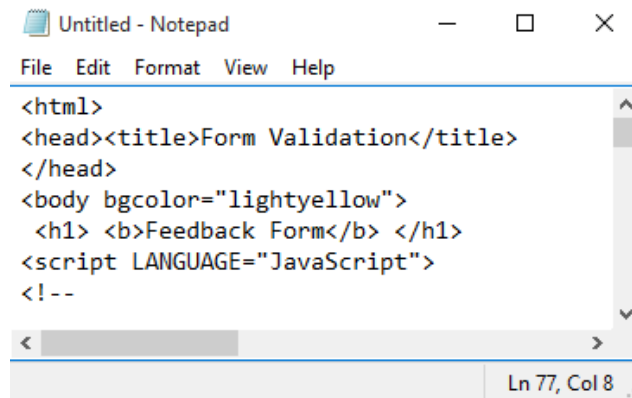
A form is simply an area that can contain form fields. When you enter data, the web application checks it to see if it is correct. If correct, the application allows the data to be submitted to the server and (usually) saved in a database; if not, it gives you error messages to explain what you've done wrong (provided you've done it right).

Form validation can be implemented in a number of different ways.

Text fields are one-line areas that allow the user to input text. Radio buttons are used when you want to let the visitor select one - and just one - option from a set of alternatives. Check boxes are used when you want to let the visitor select one or more options from a set of alternatives. List boxes are Drop-down menus and are probably the most flexible objects you can add to your forms. Depending on your settings, drop-down menus can serve the same purpose as radio buttons (one selection only) or check boxes (multiple selections allowed).

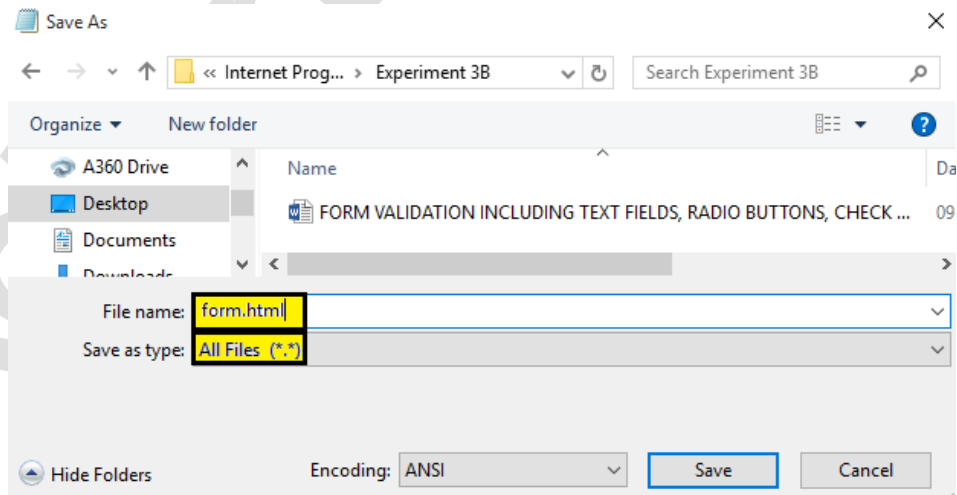
PROCEDURE

1. Open the notepad and copy the code in the “form.html” program into it.



```
<html>
<head><title>Form Validation</title>
</head>
<body bgcolor="lightyellow">
  <h1> <b>Feedback Form</b> </h1>
  <script LANGUAGE="JavaScript">
  <!--
```

2. Save the file with the .html extension as shown.



3. Open the file named “form.html” and verify the output. Make sure that you do not get any error message after clicking the “Submit” button to ensure that you have properly filled the form.

 form.html 26-Jul-17 3:28 PM Firefox HTML Document 3 KB

PROGRAM

form.html

```
<html>
<head><title>Form Validation</title>
</head>
<body bgcolor="lightyellow">
  <h1> <b>Feedback Form</b> </h1>
  <script LANGUAGE="JavaScript">
    <!--
function ValidateForm(form){
  ErrorText= "";
  if (form.username.value == "") {
    ErrorText= "\nPlease enter your name."
  }
  if ((form.email.value == "" || form.email.value.indexOf('@', 0) == -1 ||
    form.email.value.indexOf('.') == -1){
    ErrorText+="\nPlease enter a proper value for your email."
  }
  if (form.feedback.value == "") {
    ErrorText+= "\nPlease enter your feedback."
  }
  if (ErrorText!= "") {
    alert("Error :" + ErrorText);
```

```
return false;
}
if ((form.gender[0].checked == false) && (form.gender[1].checked == false)){
alert ( "Please choose your Gender: Male or Female." );
return false;
}
if (form.age.selectedIndex == 0){
alert("Please select your Age.");
return false;
}
if (ErrorText= ""){
form.submit() ;
}
if (form.terms.checked == false){
alert ( "Please check the Terms & Conditions box." );
return false;
}
}
-->
```

</script>

<form name="Feedback">

<h3>

Please fill the form by entering all the mentioned fields.

</h3>

<h4>

Name:<input type="text" value="" name="username" size=30>

E-mail: <input type="text" value="" name="email" size=30>

Gender: <input type="radio" name="gender" value="Male"> Male

<input type="radio" name="gender" value="Female"> Female

Age:

<select name="age"> <option value="">Please Select an Option:</option>

```
<option value="0-18 years">0-18 years</option>
<option value="18-30 years">18-30 years</option>
<option value="30-45 years">30-45 years</option>
<option value="45-60 years">45-60 years</option>
<option value="60+ years">60+ years</option> </select><br/>
Do you agree to the Terms and Conditions? : <input type="checkbox" name="terms"
value="Yes"> Yes<br/>
Feedback:<br/>
<textarea rows=7 cols=60 name="feedback"></textarea><br/>
<input type="button" name="SubmitButton" value="Submit"
onClick="ValidateForm(this.form)">
<input type="reset" value="Reset">
</form>
</h4>
</body>
</html>
```

SAMPLE OUTPUT

Form Validation

file:///C:/Users/Robe

Feedback Form

Please fill the form by entering all the mentioned fields.

Name:

E-mail:

Gender: ☐ Male ☐ Female

Age:

Do you agree to the Terms and Conditions? : ☐ Yes

Feedback:

Error :

Please enter your name.

Please enter a proper value for your email.

Please enter your feedback.

OK

Form Validation

file:///C:/Users/Robe

Feedback Form

Please fill the form by entering all the mentioned fields.

Name:

E-mail:

Gender: ☐ Male ☐ Female

Age:

Do you agree to the Terms and Conditions? : ☐ Yes

Feedback:

Error :
Please enter a proper value for your email.
Please enter your feedback.

OK

Form Validation

file:///C:/Users/Robe

Feedback Form

Please fill the form by entering all the mentioned fields.

Name: Robert Innocent Karamagi

E-mail: robert_karamagi@

Gender: ☐ Male ☐ Female

Age: Please Select an Option: ▾

Do you agree to the Terms and Conditions? : ☐ Yes

Feedback:

Error :

Please enter a proper value for your email.

Please enter your feedback.

OK

Form Validation

file:///C:/Users/Robe

Feedback Form

Please fill the form by entering all the mentioned fields.

Name:

E-mail:

Gender: ☐ Male ☐ Female

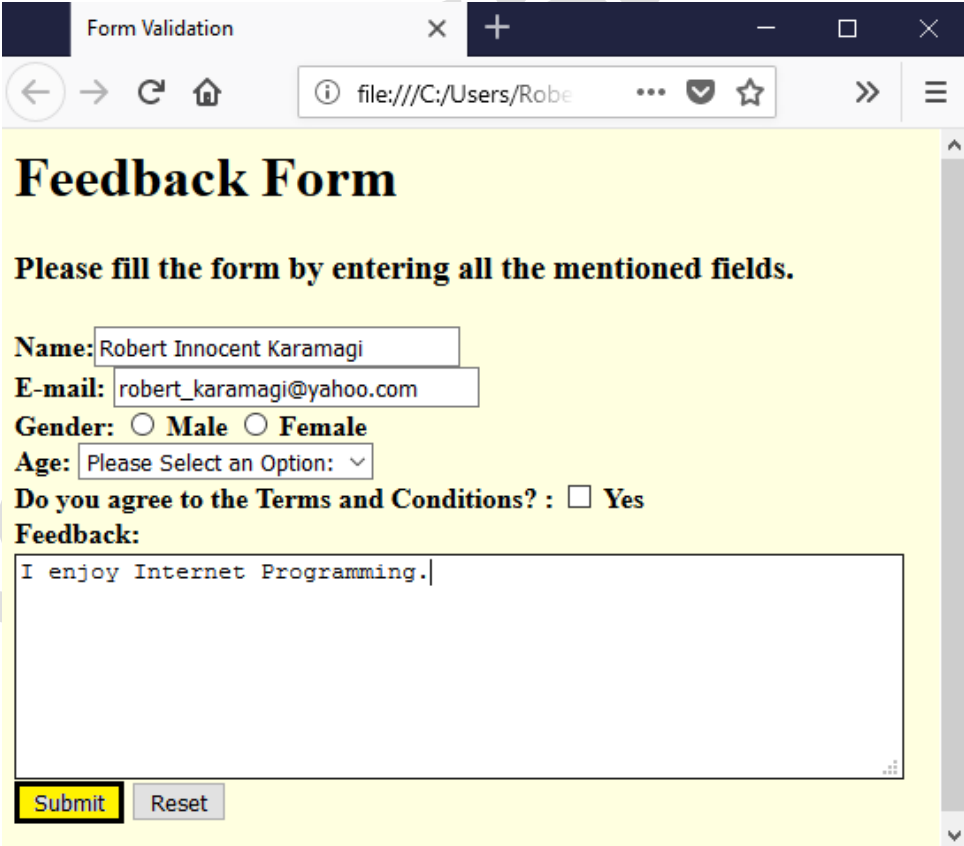
Age:

Do you agree to the Terms and Conditions? : ☐ Yes

Feedback:

Error :
Please enter your feedback.

OK



Form Validation

file:///C:/Users/Robe

Feedback Form

Please fill the form by entering all the mentioned fields.

Name: Robert Innocent Karamagi

E-mail: robert_karamagi@yahoo.com

Gender: ☐ Male ☐ Female

Age: Please Select an Option: ▾

Do you agree to the Terms and Conditions? : ☐ Yes

Feedback:

I enjoy Internet Programming.

Submit Reset

Please choose your Gender: Male or Female.

OK

Form Validation

file:///C:/Users/Robe

Feedback Form

Please fill the form by entering all the mentioned fields.

Name: Robert Innocent Karamagi

E-mail: robert_karamagi@yahoo.com

Gender: ☒ Male ☐ Female

Age: Please Select an Option: ▾

Do you agree to the Terms and Conditions? : ☐ Yes

Feedback:

I enjoy Internet Programming.

Submit Reset

Please select your Age.

OK

Form Validation

file:///C:/Users/Robe

Feedback Form

Please fill the form by entering all the mentioned fields.

Name: Robert Innocent Karamagi

E-mail: robert_karamagi@yahoo.com

Gender: ☒ Male ☐ Female

Age: 18-30 years

Do you agree to the Terms and Conditions? : ☐ Yes

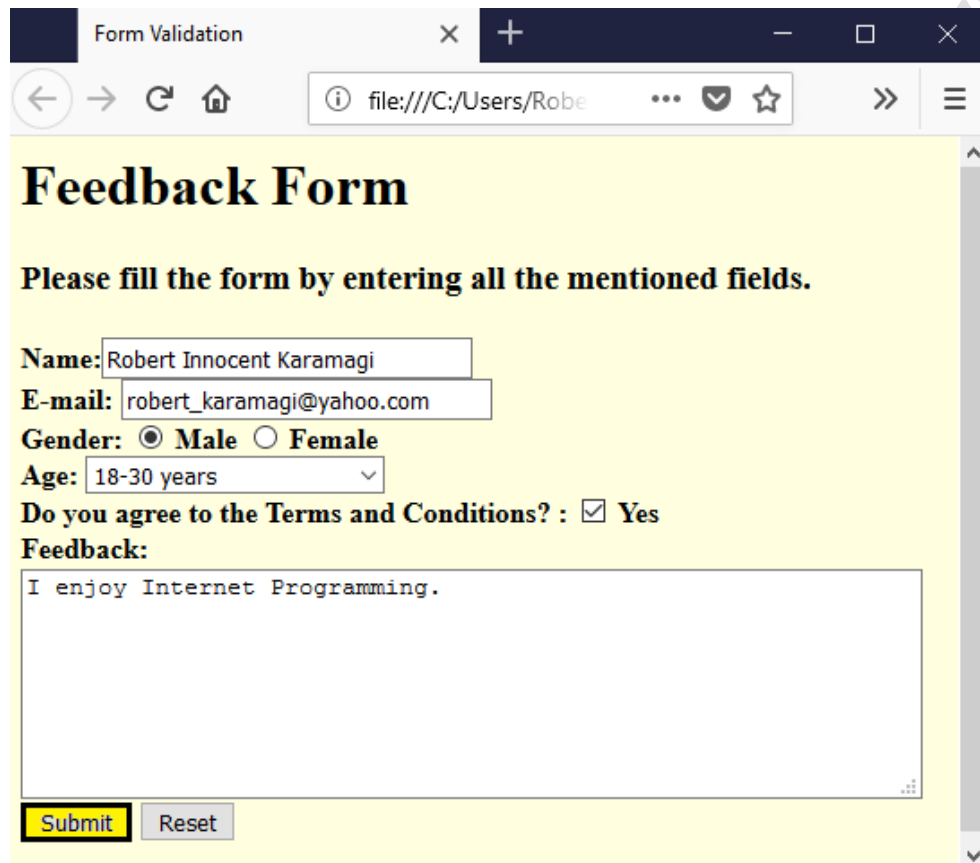
Feedback:

I enjoy Internet Programming.

Submit Reset

Please check the Terms & Conditions box.

OK



The screenshot shows a web browser window titled 'Form Validation'. The address bar displays 'file:///C:/Users/Robe'. The main content area has a yellow background and is titled 'Feedback Form'. Below the title, it says 'Please fill the form by entering all the mentioned fields.' The form contains the following fields: 'Name:' with the value 'Robert Innocent Karamagi', 'E-mail:' with the value 'robert_karamagi@yahoo.com', 'Gender:' with radio buttons for 'Male' (selected) and 'Female', 'Age:' with a dropdown menu showing '18-30 years', and 'Do you agree to the Terms and Conditions?' with a checked checkbox and the word 'Yes'. Below these is a 'Feedback:' section with a text area containing the text 'I enjoy Internet Programming.'. At the bottom of the form are two buttons: 'Submit' (highlighted in yellow) and 'Reset'.

RESULT

Thus, the experiment to implement the form validation including text fields, radio buttons, check boxes, list boxes and other controls was executed successfully and the output is verified.

Experiment 4a

Usage of Active Server Pages/Java Server Pages objects response, Request, Application, Session, Server, Active X Data Objects etc.

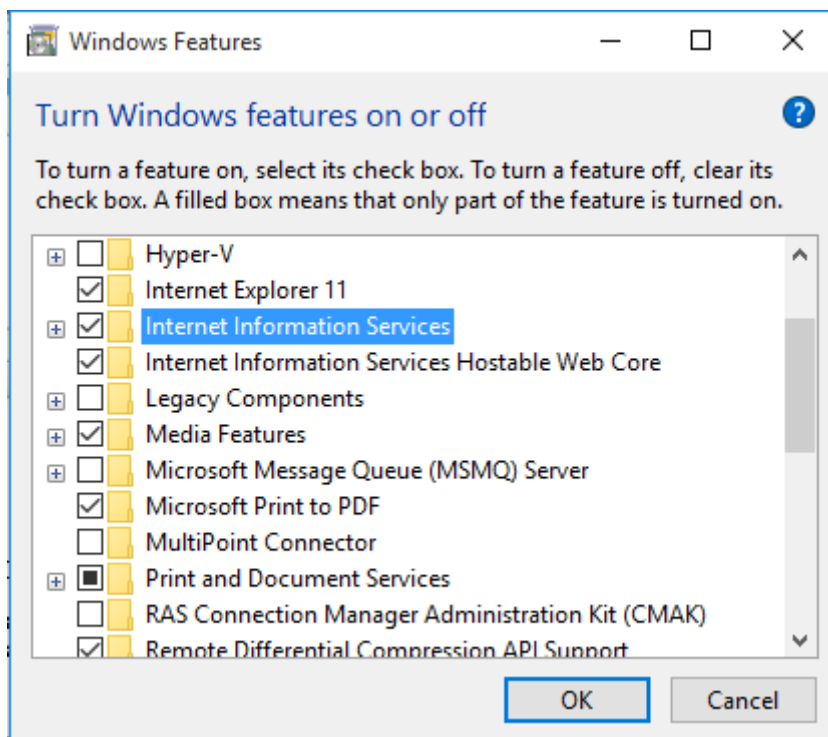
Writing online applications such as shopping, railway/air/bus ticket reservation system with the set of Active Server Pages /Java Server Pages.

Go to Windows Features in the Control Panel

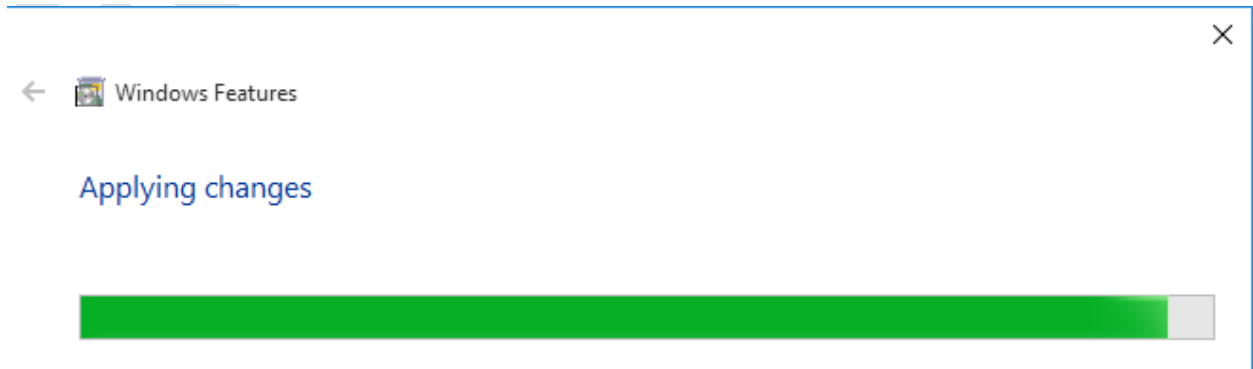
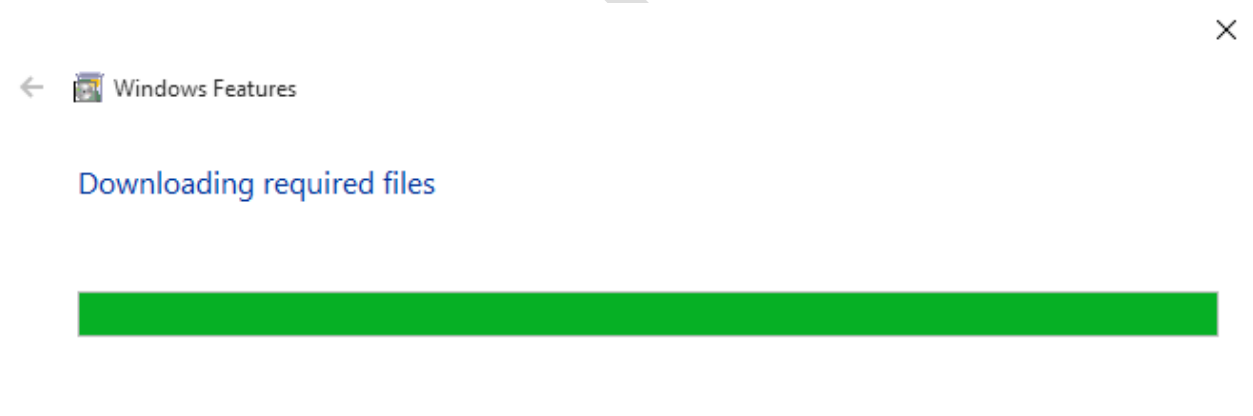
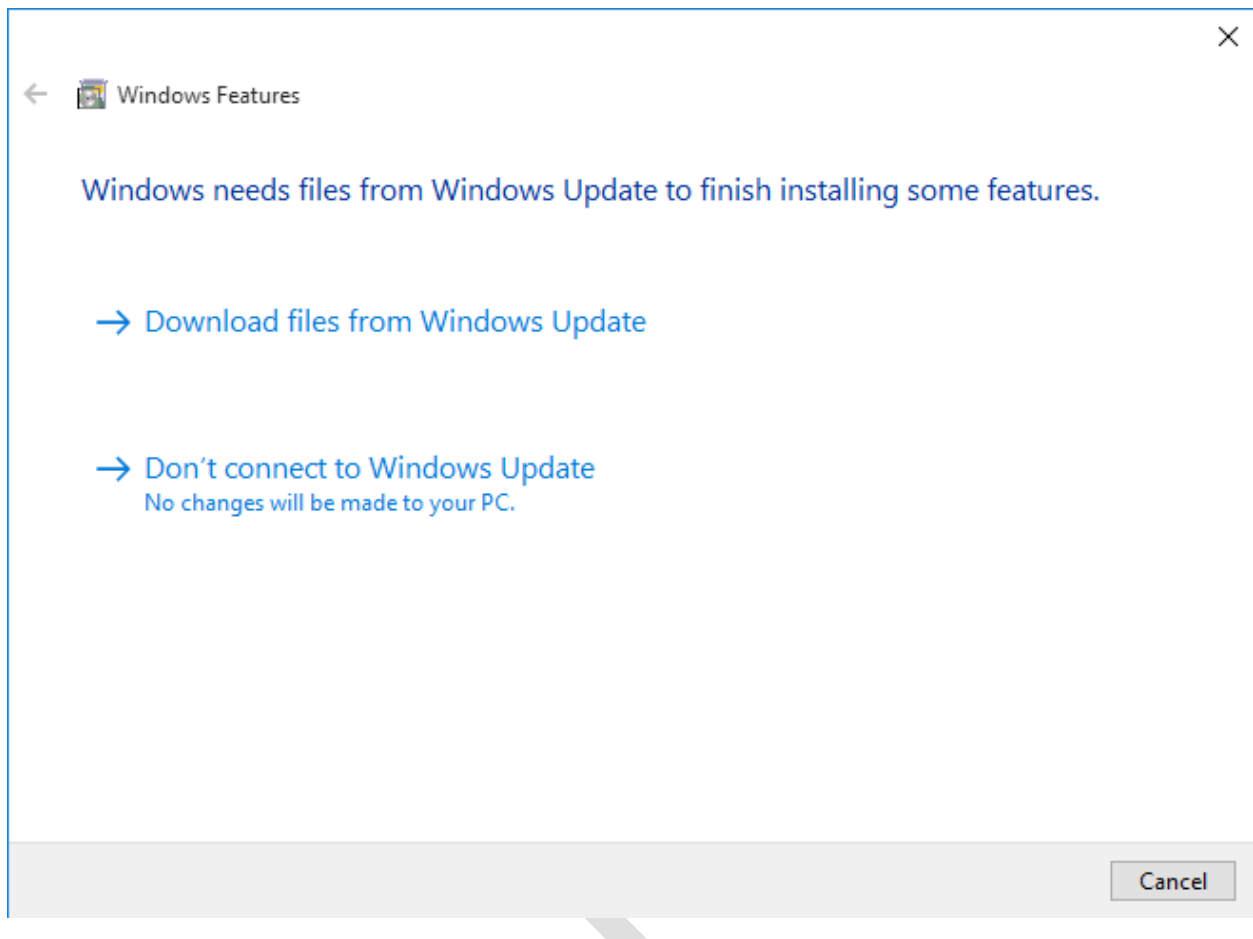
Turn on/off Windows Features

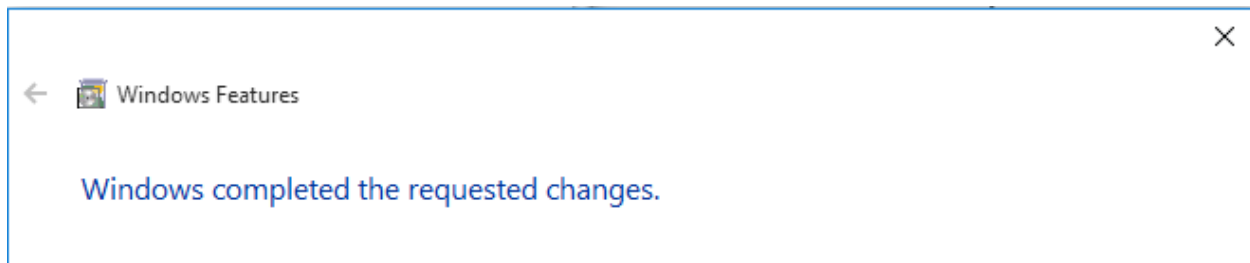
Internet Information Services

Internet Information Services Hostable Web Core

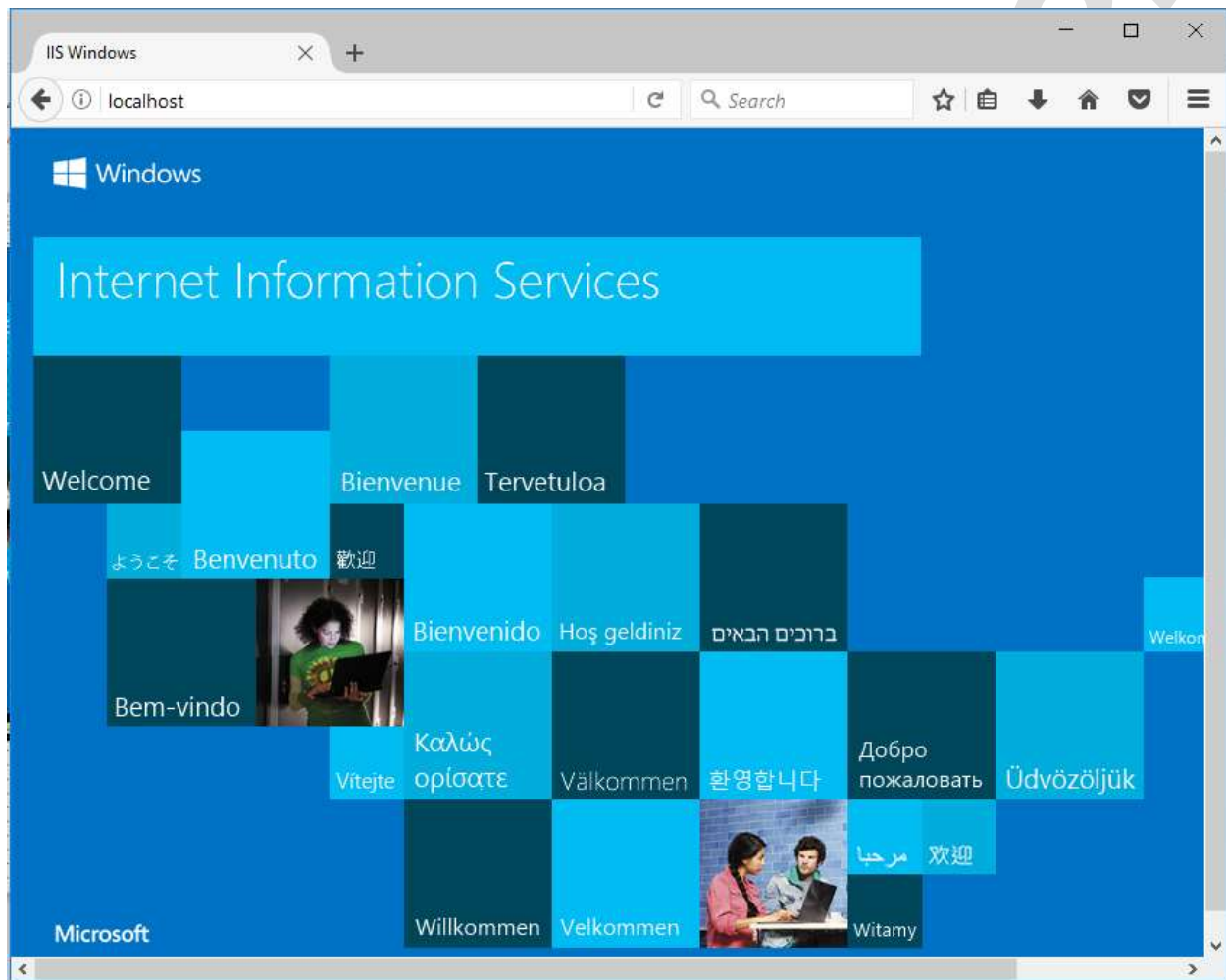


Download the files from Windows Update





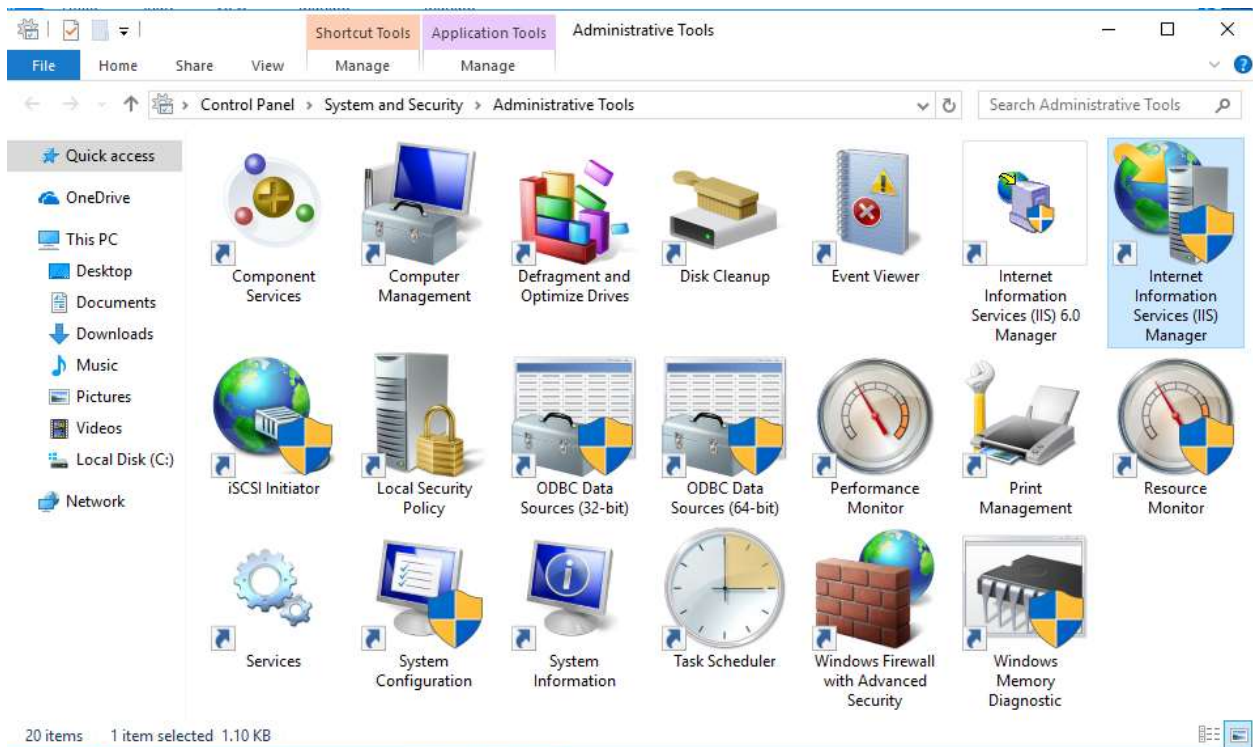
Open the Web Browser and type localhost



Go to the Control Panel

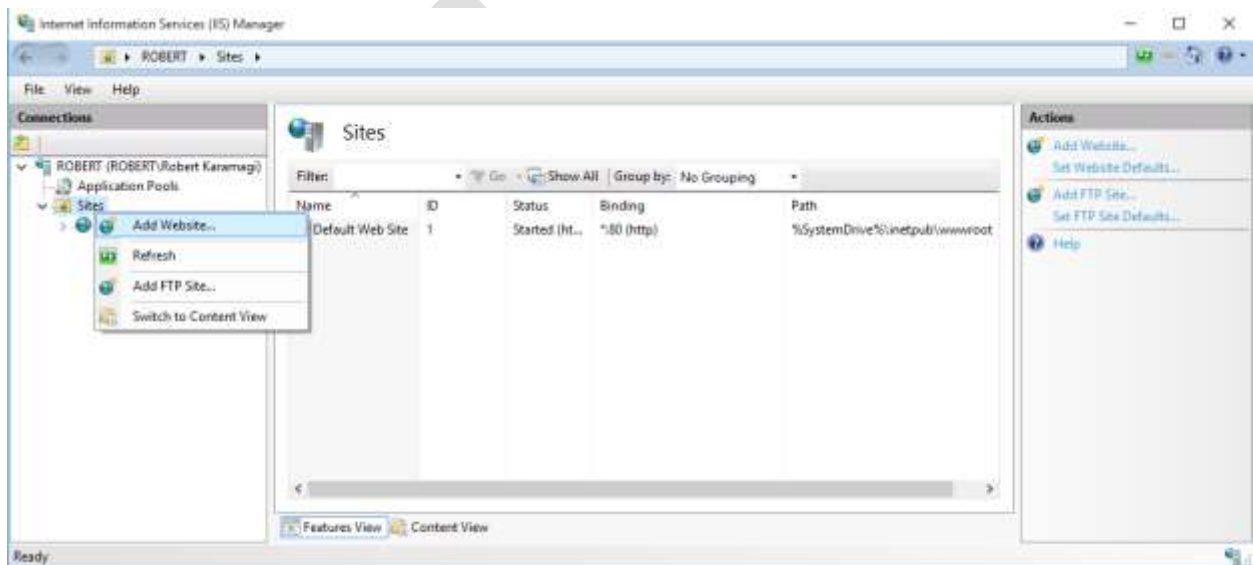
Open the System and Security and Next Open the Administrative Tools

Open the Internet Information Services (IIS) Manager.



On the Connection Tab

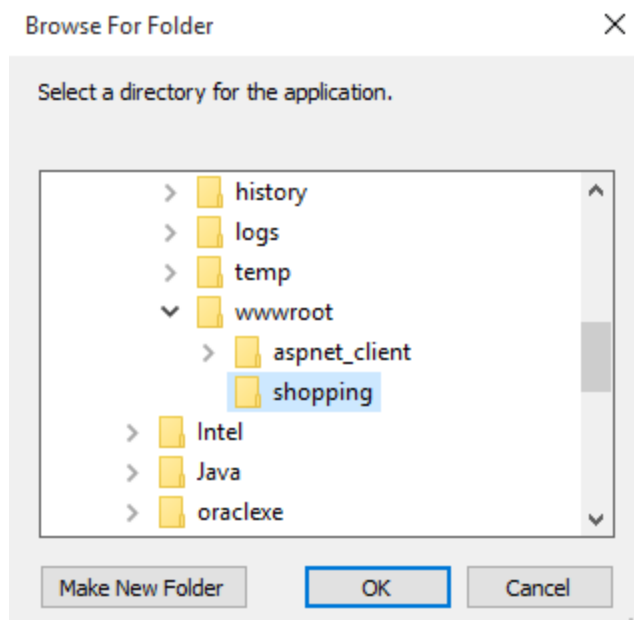
Right Click on Sites and Add Website



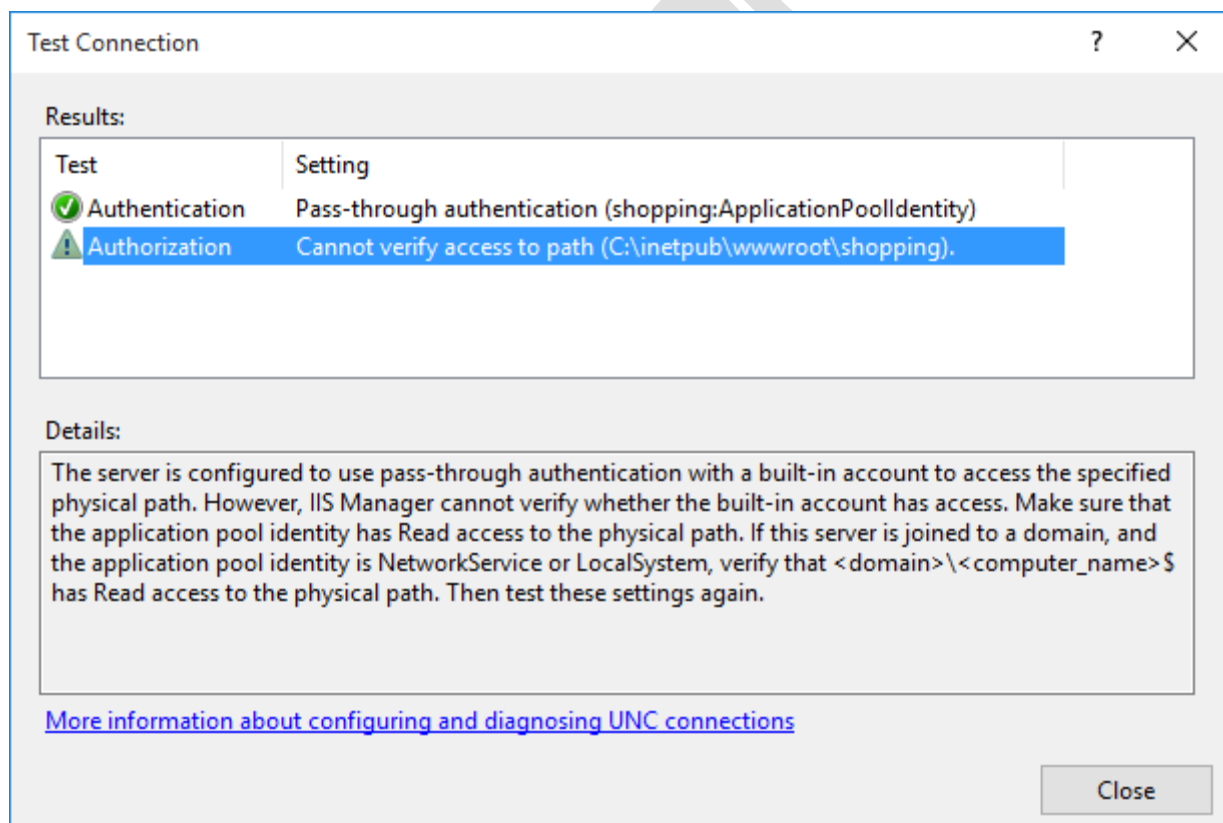
Under Site name: shopping

Content Directory

Physical Path: C:\inetpub\wwwroot\shopping



Set the Authorization



Connect as a Specific User and Set the Credentials

Connect As ? X

Path credentials:

☒ Specific user:

Set...

☐ Application user (pass-through authentication)

OK Cancel

Username:  Robert Karamagi

Set Credentials ? X

User name:

Password:

Confirm password:

OK Cancel

Under the Test Connection Tab, note that the Authorization Path is Accessible.

Test Connection



Results:

Test	Setting
✓ Authentication	User name (Robert Karamagi)
✓ Authorization	Path is accessible (C:\inetpub\wwwroot\shopping).

Details:

The specified user credentials are authorized to access the specified physical path.

[More information about configuring and diagnosing UNC connections](#)

Close

Call the hostname as www.shopping.com

Add Website



Site name: shopping Application pool: shopping Select...

Content Directory

Physical path: C:\inetpub\wwwroot\shopping ...

Connect as 'Robert Karamagi'

Connect as... Test Settings...

Binding

Type: http IP address: All Unassigned Port: 80

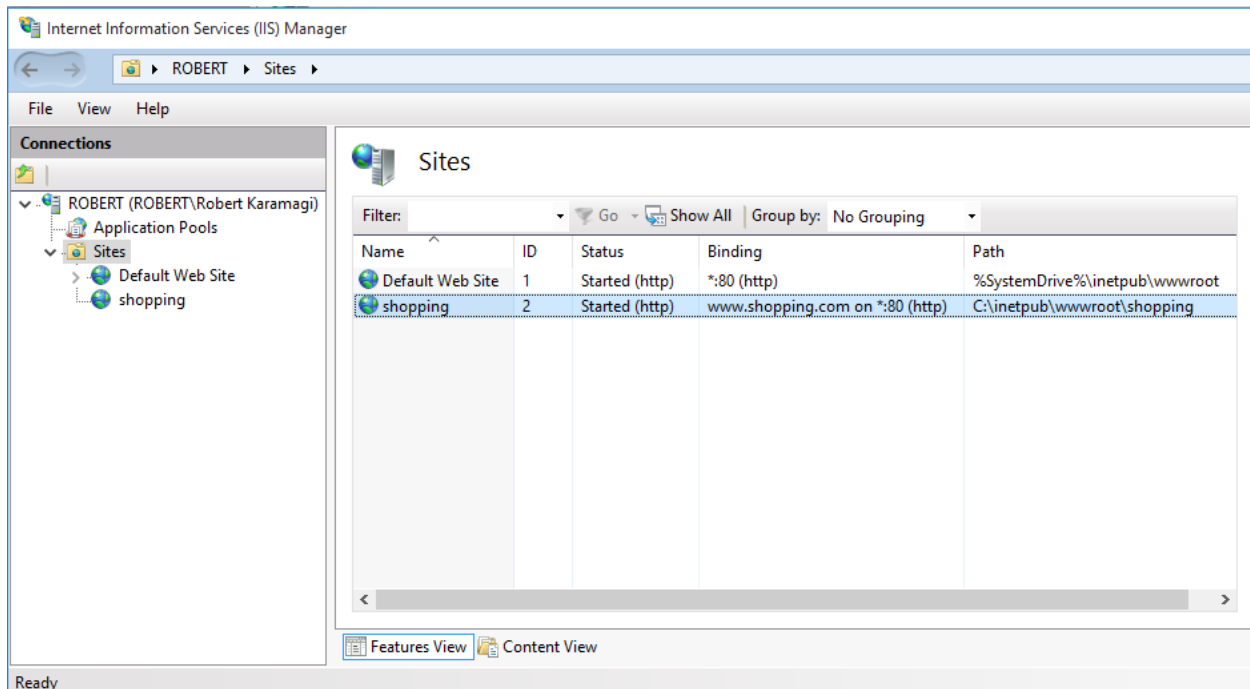
Host name: www.shopping.com

Example: www.contoso.com or marketing.contoso.com

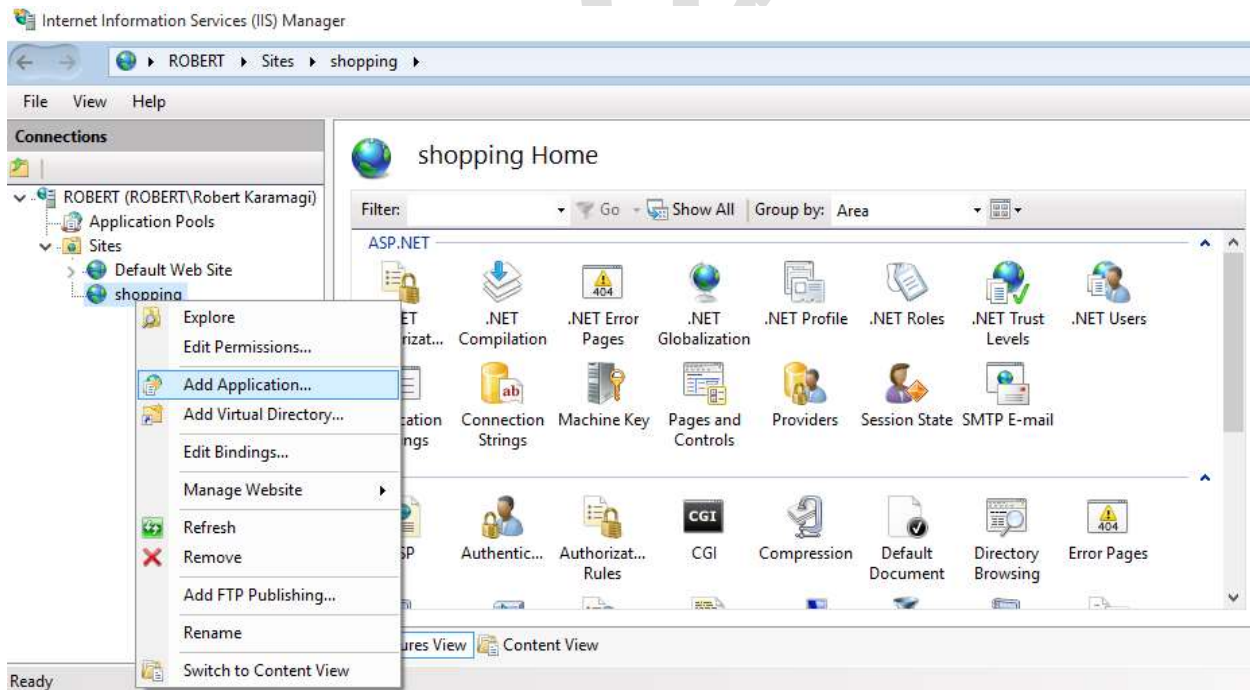
☒ Start Website immediately

OK Cancel

Note under the sites the newly created website “shopping”



Under shopping site, right click and open the Add Application option.



Set the Alias: shopping

Physical path: C:\inetpub\wwwroot\shopping

Set the Connection Settings as previously shown.

Add Application

Site name: shopping
Path: /

Alias: shopping Application pool: shopping Select...

Example: sales

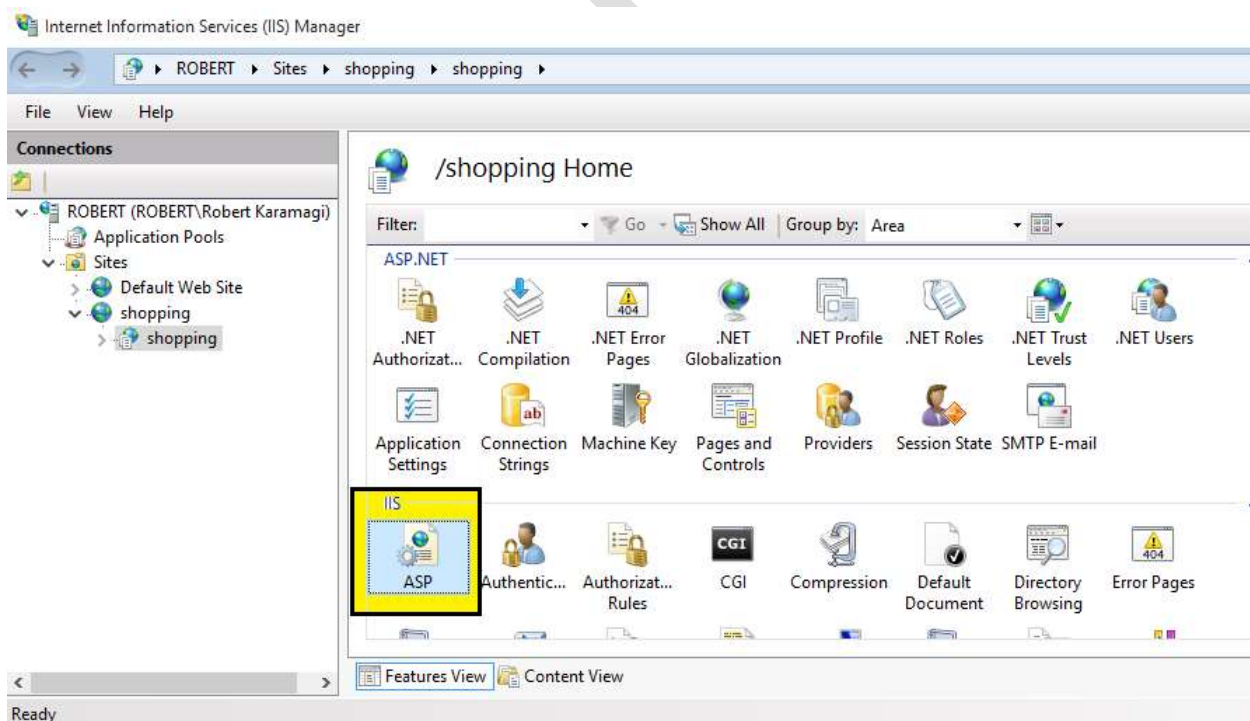
Physical path: C:\inetpub\wwwroot\shopping ...

Connect as 'Robert Karamagi'
Connect as... Test Settings...

☐ Enable Preload

OK Cancel

Under the application shopping go to the IIS section and double click on ASP icon to configure the properties for ASP applications.



Make sure under Behavior that Enable Parent Paths: False

The screenshot shows the IIS Manager console with the 'shopping' virtual directory selected. The 'ASP' feature is expanded, and the 'Behavior' tab is active. The 'Enable Parent Paths' property is highlighted in blue and set to 'False'. Below the table, a description for 'Enable Parent Paths' is provided.

Behavior	
Code Page	0
Enable Buffering	True
Enable Chunked Encoding	True
Enable HTML Fallback	True
Enable Parent Paths	False
Limits Properties	True
Locale ID	False
Restart On Config Change	True

Enable Parent Paths
Specifies whether an ASP page allows paths relative to the current directory (using the ..\ notation) or above the current directory.

Configuration: 'localhost' applicationHost.config, <location path="shopping/shopping">

Under Compilation → Debugging Properties → Send Errors To Browser: True

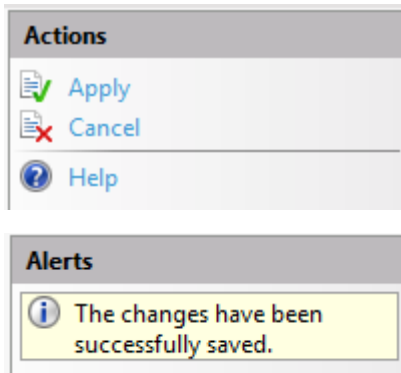
The screenshot shows the IIS Manager console with the 'shopping' virtual directory selected. The 'ASP' feature is expanded, and the 'Debugging Properties' tab under 'Compilation' is active. The 'Send Errors To Browser' property is highlighted in blue and set to 'True'. Below the table, a description for 'Send Errors To Browser' is provided.

Debugging Properties	
Calculate Line Numbers	True
Catch COM Component Exceptions	True
Enable Client-side Debugging	False
Enable Log Error Requests	True
Enable Server-side Debugging	False
Log Errors to NT Log	False
Run On End Functions Anonymously	True
Script Error Message	An error occurred on the server when processing the URL. Plea
Send Errors To Browser	True
Script Language	True

Send Errors To Browser
Specifies whether the Web server writes debugging specifications (the name, error, line number, description) to the client browser, in addition to logging them to the Windows Event Log.

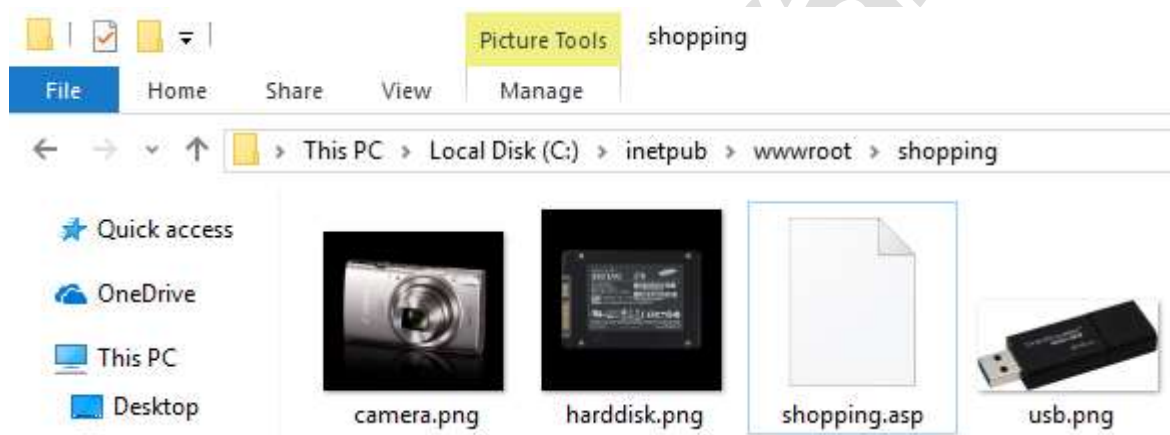
Configuration: 'localhost' applicationHost.config, <location path="shopping/shopping">

Save the changes by Applying under Actions.



Save the shopping.asp file in the path C:\inetpub\wwwroot\shopping

Add all the images usb.png, camra.png and haddisk.png to the same folder.



shopping.asp

```
<!DOCTYPE html>
```

```
<html>
```

```
<body bgcolor='cyan'>
```

```
<%response.write("<h1><font color=blue><center>Shopping Items<center></font></h1>")%>
```

```
<div align="center">
```

```
<font color=green><h2>You may purchase the following</h2></font>
```

```
<TABLE>
<TR>
<TD><IMG border="0" src="usb.png" width="270" height="200"></TD>
<TD><IMG border="0" src="camera.png" width="270" height="200"></TD>
<TD><IMG border="0" src="harddisk.png" width="270" height="200"></TD>
</TR>
</TABLE>
```

```
<TABLE border =1>
<tr>
<TH>Item</TD>
<TH>Price</TD>
</tr>

<TR>
<TD>64 GB USB Pen Drive</TD>
<TD>$20</TD></TR>
```

```
<TR>
<TD>20 MP Digital Camera</TD>
<TD>$50</TD>
</TR>
```

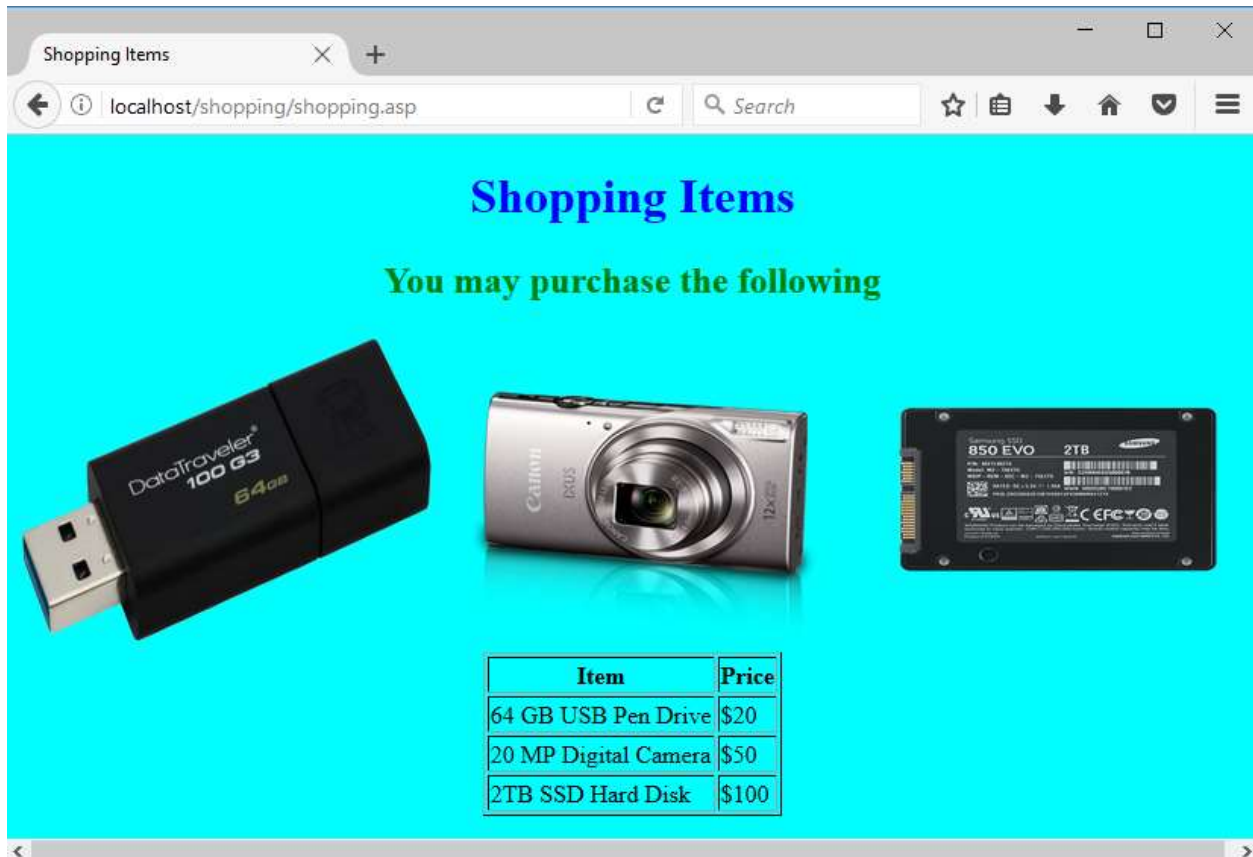
```
<TR>
<TD>2TB SSD Hard Disk</TD>
<TD>$100</TD>
</TR>
```

```
</TABLE>
```

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

</html>

Open the browser and type localhost/shopping/shopping.asp



Shopping Items

You may purchase the following



Item	Price
64 GB USB Pen Drive	\$20
20 MP Digital Camera	\$50
2TB SSD Hard Disk	\$100

ONLINE APPLICATION FOR A SHOPPING SYSTEM WITH ASP

AIM

The aim of the experiment is to write an online application for a shopping system with Active Server Pages.

APPARATUS REQUIRED

Microsoft ASP.NET

Microsoft Visual C#

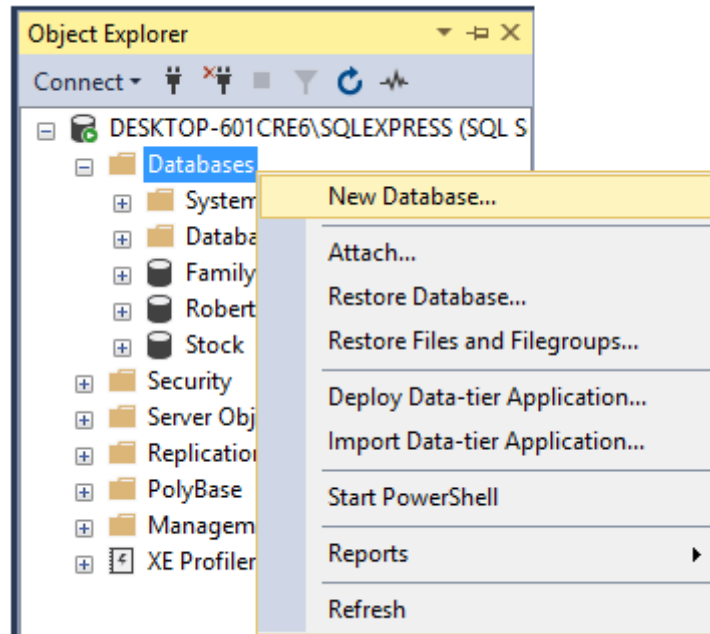
Microsoft SQL Server Management Studio

THEORY

A form is simply an area that can contain form fields. When you enter data, the web application checks it to see if it is correct. If correct, the application allows the data to be submitted to the server and (usually) saved in a database; if not, it gives you error messages to explain what you've done wrong (provided you've done it right). Form validation can be implemented in a number of different ways. Text fields are one-line areas that allow the user to input text. Radio buttons are used when you want to let the visitor select one - and just one - option from a set of alternatives. Check boxes are used when you want to let the visitor select one or more options from a set of alternatives. List boxes are Drop-down menus and are probably the most flexible objects you can add to your forms. Depending on your settings, drop-down menus can serve the same purpose as radio buttons (one selection only) or check boxes (multiple selections allowed).

PROCEDURE

1. Open the Control Panel and select the “Programs” settings.
2. Click on the “Programs and Features”.



New Database

Select a page

General

Options

Filegroups

Connection

Server:
DESKTOP-601CRE6\SQLEXPRES

Connection:
DESKTOP-601CRE6\Robert Karan

View connection properties

Progress

Ready

Script

Help

Database name:

ASPSHopping

Owner:

<default>

☒ Use full-text indexing

Database files:

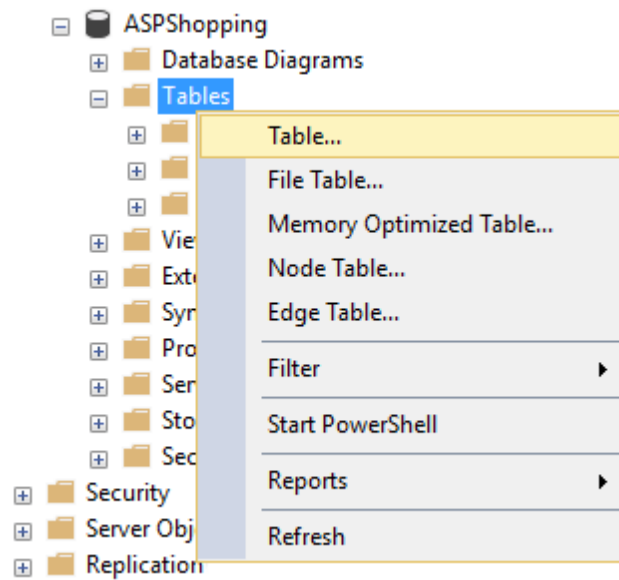
Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize
ASPSHopping	ROWS...	PRIMARY	8	By 64 MB, Unlimited
ASPSHoppin...	LOG	Not Applicable	8	By 64 MB, Unlimited

Add

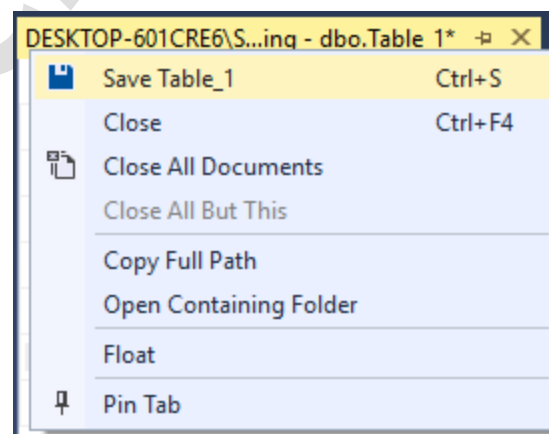
Remove

OK

Cancel



DESKTOP-601CRE6\...bo.CustomerOrder			
	Column Name	Data Type	Allow Nulls
	Item	varchar(50)	☒
	Quantity	int	☒
	AccName	varchar(50)	☒
	AccNum	varchar(50)	☒
	PIN	int	☒
	Email	varchar(50)	☐
			☐



Choose Name

Enter a name for the table:

CustomerOrder

OK Cancel

- ASPSHopping
 - Database Diagrams
 - Tables
 - System Tables
 - FileTables
 - External Tables
 - dbo.CustomerOrder
 - Columns
 - Keys

- Table...
- Design
- Select Top 1000 Rows
- Edit Top 200 Rows
- Script Table as

SQLQuery1.sql - DE...bert Karamagi (55)

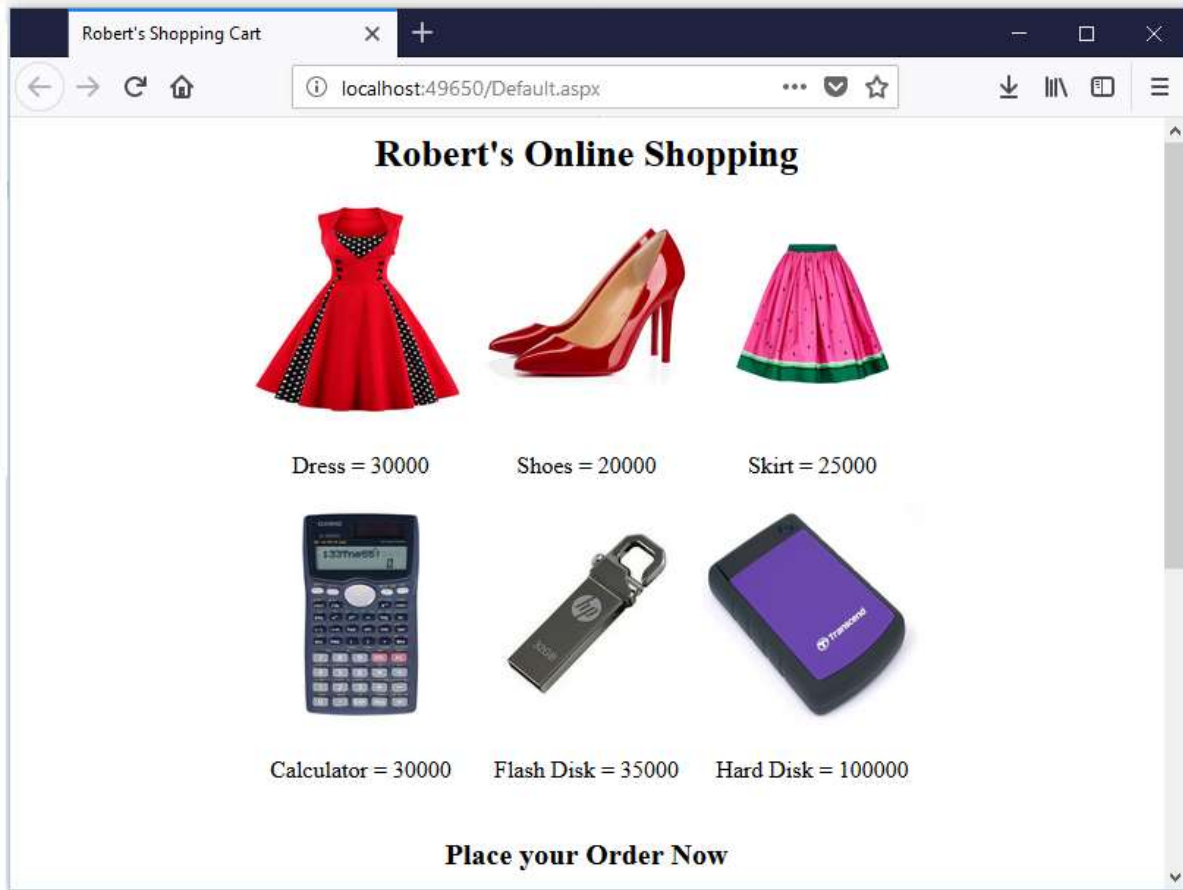
```
/****** Script for SelectTopNRows command from SSMS *****/  
SELECT TOP (1000) [Item]  
    , [Quantity]  
    , [AccName]  
    , [AccNum]  
    , [PIN]  
    , [Email]  
FROM [ASPSHopping].[dbo].[CustomerOrder]
```

100 %

Results Messages

Item	Quantity	AccName	AccNum	PIN	Email
------	----------	---------	--------	-----	-------

Query executed successfully.



Place your Order Now

Flash Disk

2

Robert Method Karamagi

9875 3539 2087 0133

•••••

robertokaramagi@gmail.com

Order Now

Enter Item

Enter Quantity

Enter Bank Account Name

Enter Bank Account Number

Enter Bank Account PIN

Enter Email Address

Order Now

Order has been placed successfully.

Results						
Messages						
	Item	Quantity	AccName	AccNum	PIN	Email
1	Flash Disk	2	Robert Method Karamagi	9875 3539 2087 0133	1234	robertokaramagi@gmail.com

Add Connection?×

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:

Microsoft SQL Server (SqlClient)Change...

Server name:

DESKTOP-601CRE6\SQLEXPRESSRefresh

Log on to the server

Authentication:

Windows Authentication

User name:

Password:

☐ Save my password

Connect to a database

☒ Select or enter a database name:

ASPSHopping

☐ Attach a database file:

Browse...

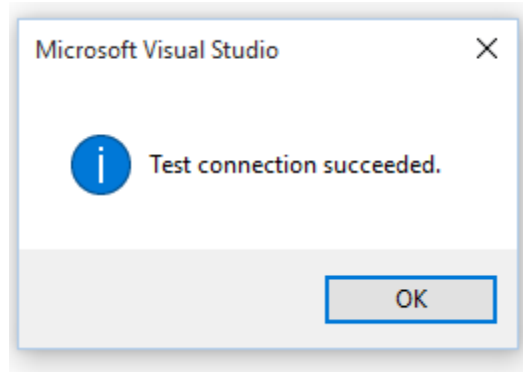
Logical name:

Advanced...

Test Connection

OK

Cancel



PROGRAM

Default.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs"
Inherits="_Default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Robert's Shopping Cart</title>
</head>
<body>
    <form id="form1" runat="server">

        <div style="margin-left: auto; margin-right: auto; text-align: center;">

            <asp:Label ID="Label1" runat="server" style="font-size:26px;font-weight:bold"
Text="Robert's Online Shopping"></asp:Label>

        </div>

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

<asp:Image ID="Image1" runat="server" Height="150px" Width="150px"
ImageUrl="~/images/dress.jpg" />

<asp:Image ID="Image2" runat="server" Height="150px" Width="150px"
ImageUrl="~/images/shoes.jpg" />

<asp:Image ID="Image3" runat="server" Height="150px" Width="150px"
ImageUrl="~/images/skirt.jpg" />

</p>

<p>

<asp:Label ID="Label2" runat="server" Text="Dress = 30000" Width="150px"
Height="18px"></asp:Label>

<asp:Label ID="Label3" runat="server" Text="Shoes = 20000"
Width="150px"></asp:Label>

<asp:Label ID="Label4" runat="server" Text="Skirt = 25000"
Width="150px"></asp:Label>

</p>

<p>

<asp:Image ID="Image4" runat="server" Height="150px" Width="150px"
ImageUrl="~/images/calculator.jpg" />

<asp:Image ID="Image5" runat="server" Height="150px" Width="150px"
ImageUrl="~/images/flashdisk.png" />

<asp:Image ID="Image6" runat="server" Height="150px" Width="150px"
ImageUrl="~/images/harddisk.jpg" />

</p>

<p>

<asp:Label ID="Label5" runat="server" Text="Calculator = 30000" Width="150px"
Height="18px"></asp:Label>

<asp:Label ID="Label6" runat="server" Text="Flash Disk = 35000" Width="150px"
Height="18px"></asp:Label>

```
<asp:Label ID="Label7" runat="server" Text="Hard Disk = 100000" Width="150px"
Height="18px"></asp:Label>
```

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

```
</p>
```

```
<asp:Label ID="Label8" runat="server" style="font-size:20px;font-weight:bold"
Text="Place your Order Now "></asp:Label>
```

```
<p>
```

```
<asp:TextBox ID="TextBoxItem" runat="server" Height="25px" placeholder="Enter
Item" style="margin-top: 0px" Width="300px"></asp:TextBox>
```

```
</p>
```

```
<p>
```

```
<asp:TextBox ID="TextBoxQuantity" runat="server" Height="25px" placeholder="Enter
Quantity" style="margin-top: 0px" Width="300px" TextMode="Number"></asp:TextBox>
```

```
</p>
```

```
<p>
```

```
<asp:TextBox ID="TextBoxAccName" runat="server" Height="25px"
placeholder="Enter Bank Account Name" style="margin-top: 0px"
Width="300px"></asp:TextBox>
```

```
</p>
```

```
<p>
```

```
<asp:TextBox ID="TextBoxAccNum" runat="server" Height="25px"
placeholder="Enter Bank Account Number" style="margin-top: 0px"
Width="300px"></asp:TextBox>
```

```
</p>
```

```
<p>
```

```
<asp:TextBox ID="TextBoxPIN" runat="server" Height="25px" placeholder="Enter
Bank Account PIN" style="margin-top: 0px" Width="300px"
TextMode="Password"></asp:TextBox>
```

```

</p>
<p>
    <asp:TextBox ID="TextBoxEmail" runat="server" Height="25px" placeholder="Enter
Email Address" style="margin-top: 0px" Width="300px" TextMode="Email"></asp:TextBox>
</p>

    <asp:Button ID="ButtonOrder" runat="server" Text="Order Now"
OnClick="ButtonOrder_Click" />

<p>
    <asp:Label ID="LabelResult" runat="server"></asp:Label>
</p>
</div>
</form>
</body>
</html>

```

Default.aspx.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.SqlClient;

public partial class _Default : System.Web.UI.Page
{

```

```
SqlConnection con = new SqlConnection("Integrated Security=SSPI;Persist Security
Info=False;Initial Catalog=ASPShopping;Data Source=DESKTOP-601CRE6\\SQLEXPRESS");
```

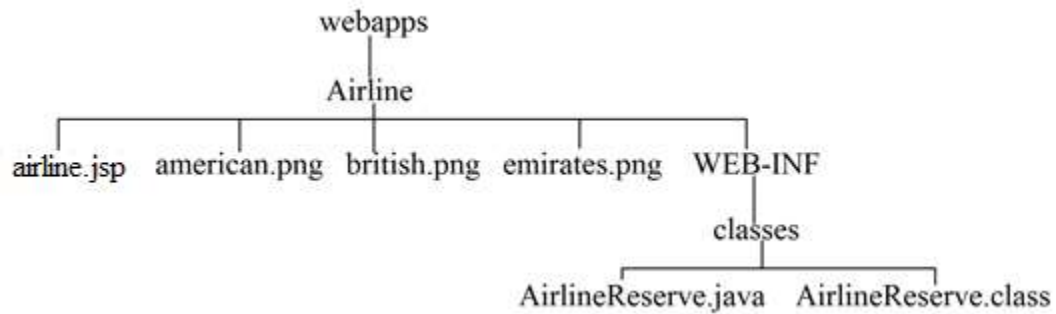
```
protected void ButtonOrder_Click(object sender, EventArgs e)
{
    string query = "insert into CustomerOrder (Item,Quantity,AccName,AccNum,PIN,Email)
values ('" + TextBoxItem.Text + "','" + TextBoxQuantity.Text + "','" + TextBoxAccName.Text +
','" + TextBoxAccNum.Text + "','" + TextBoxPIN.Text + "','" + TextBoxEmail.Text + "')";
    SqlCommand cmd = new SqlCommand(query, con);
    con.Open();
    int result = cmd.ExecuteNonQuery();
    con.Close();
    if (result > 0)
    {
        LabelResult.Text = "Order has been placed successfully.";
        TextBoxItem.Text = "";
        TextBoxQuantity.Text = "";
        TextBoxAccName.Text = "";
        TextBoxAccNum.Text = "";
        TextBoxPIN.Text = "";
        TextBoxEmail.Text = "";
    }
}
}
```

RESULT

Thus, the experiment to implement the form validation including text fields, radio buttons, check boxes, list boxes and other controls was executed successfully and the output is verified.

Airline Reservation Page

Directory Structure



Create the following Directory structure in C:\TomCat\webapps

airline.jsp

```
<head><title>Robert's Airline Resrvation</title></head>

<form name="Robert Airline" method="post" action="robertairline">

<h1 align="center">Robert's Airline Reservation</h1>

<TABLE align="center">

<TR>

<TD><IMG border="0" src="american.png" width="450" height="300"></TD>

<TD><IMG border="0" src="british.png" width="450" height="300"></TD>

<TD><IMG border="0" src="emirates.png" width="450" height="300"></TD>

</TR>

</TABLE>

<h3 align="center">Passenger Name: <input type="text" name="passenger"/> <br/>

Nationality: <input type="nationality" name="nationality"/> <br/>

    Airline:

    <select name="airline">

<option value="American Airlines">American Airlines</option>

<option value="British Airways">British Airways</option>
```

```

<option value="Fly Emirates">Fly Emirates</option>
    </select><br/>
    Destination:
    <select name="destination">
        <option value="New York">New York</option>
        <option value="London">London</option>
        <option value="Dubai">Dubai</option>
    </select><br/><br/>
Departure Times</h3>
<h4 align="center">
    <input type="radio" name="departure" value="1045" />1045 Hours<br/>
    <input type="radio" name="departure" value="1530" />1530 Hours<br/>
    <input type="radio" name="departure" value="2320" />2320 Hours<br/>
</h4>
<h3 align="center"><input type="submit" value="Book" /></h3>
</form>

```

AirlineReserve.java

```

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/robertairline")
public class AirlineReserve extends HttpServlet {

    protected void doPost(HttpServletRequest request,
        HttpServletResponse response) throws ServletException, IOException {

```

```

// read form fields

String passenger = request.getParameter("passenger");
String nationality= request.getParameter("nationality");
String airlineType = request.getParameter("airline");
    String destination = request.getParameter("destination");
    String departure = request.getParameter("departure");

System.out.println("Passenger Name: " + passenger);
System.out.println("Nationality: " + nationality);
    System.out.println("Airline: " + airlineType);
    System.out.println("Destination: " + destination);
    System.out.println("Departure Time: " + departure);
    // do some processing here...

// get response writer
PrintWriter writer = response.getWriter();

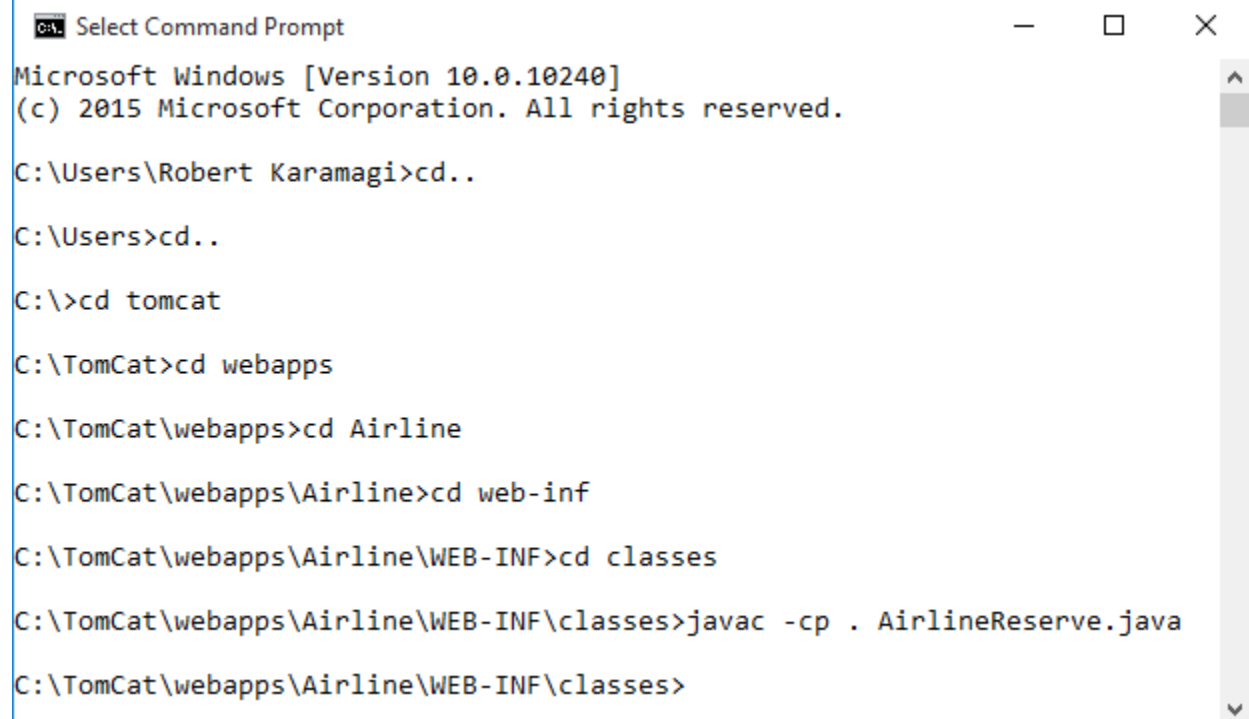
// build HTML code
String htmlResponse = "<html><head><title>Robert's Airline Reservation</title></head>";
    htmlResponse += "<h2 align= center >Ticket Information</h2>";
    htmlResponse += "<table align= center border size=1>";
    htmlResponse += "<tr><td>Passenger</td><td>" + passenger+ "</td></tr>";
    htmlResponse += "<tr><td>Nationality</td><td>" + nationality+ "</td></tr>";
    htmlResponse += "<tr><td>Airline</td><td>" + airlineType+ "</td></tr>";
    htmlResponse += "<tr><td>Destination</td><td>" + destination+ "</td></tr>";
    htmlResponse += "<tr><td>Departure Time</td><td>" + departure+ "</td></tr>";
    htmlResponse += "</table></html>";

// return response
writer.println(htmlResponse);
}

```

}

Compile the AirlineReserve.java file



```
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\Robert Karamagi>cd..

C:\Users>cd..

C:\>cd tomcat

C:\TomCat>cd webapps

C:\TomCat\webapps>cd Airline

C:\TomCat\webapps\Airline>cd web-inf

C:\TomCat\webapps\Airline\WEB-INF>cd classes

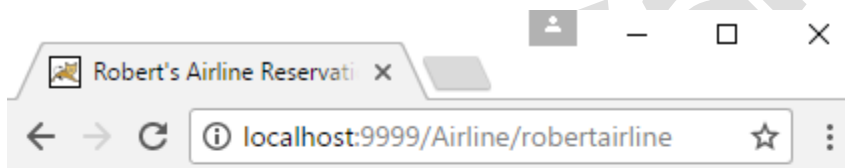
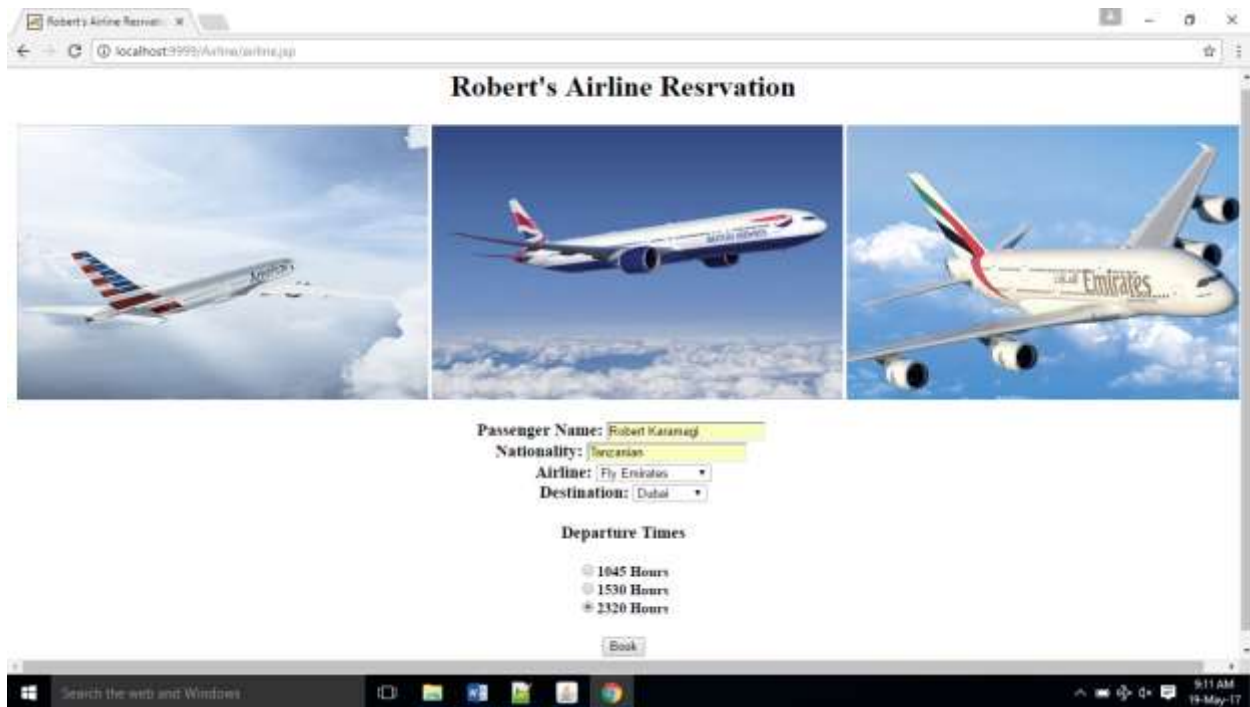
C:\TomCat\webapps\Airline\WEB-INF\classes>javac -cp . AirlineReserve.java

C:\TomCat\webapps\Airline\WEB-INF\classes>
```

Start the Apache Tomcat Server.

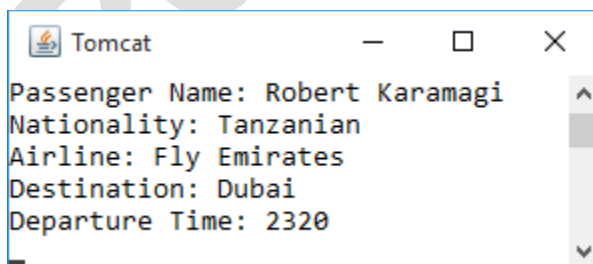
Open the Airline Reservation Form by typing in any browser

<http://localhost:9999/Airline/airline.jsp>



Ticket Information

Passenger	Robert Karamagi
Nationality	Tanzanian
Airline	Fly Emirates
Destination	Dubai
Departure Time	2320



Experiment 4b

Using sessions and cookies as part of the web application

SendCookie.jsp

```
<html>
<head>
  <title>Send Cookie</title>
</head>
<body>
  <H1>JSP is sending two cookies to the client </H1>
  <%
    Cookie userIdCookie= new Cookie("Robert", "Karamagi");
    userIdCookie.setMaxAge(24*60*60);
    Cookie createdBy= new Cookie("Innocent","Karamagi" );
    createdBy.setMaxAge(24*60*60);
    response.addCookie(userIdCookie);
    response.addCookie(createdBy);
  %>
</body>
</html>
```



General

Search

Content

Applications

Privacy

Security

Sync

Advanced

Privacy

Tracking

☒ Use Tracking Protection in Private Windows

[Learn more](#)

[Change Block List](#)

You can also [manage your Do Not Track settings](#).

History

Firefox will:

Use custom settings for history

Remember history

Never remember history

Use custom settings for history

☐ Always

☒ Remember history

☒ Remember history

☒ Accept cookies from sites

Accept third-party cookies:

Always

Keep until:

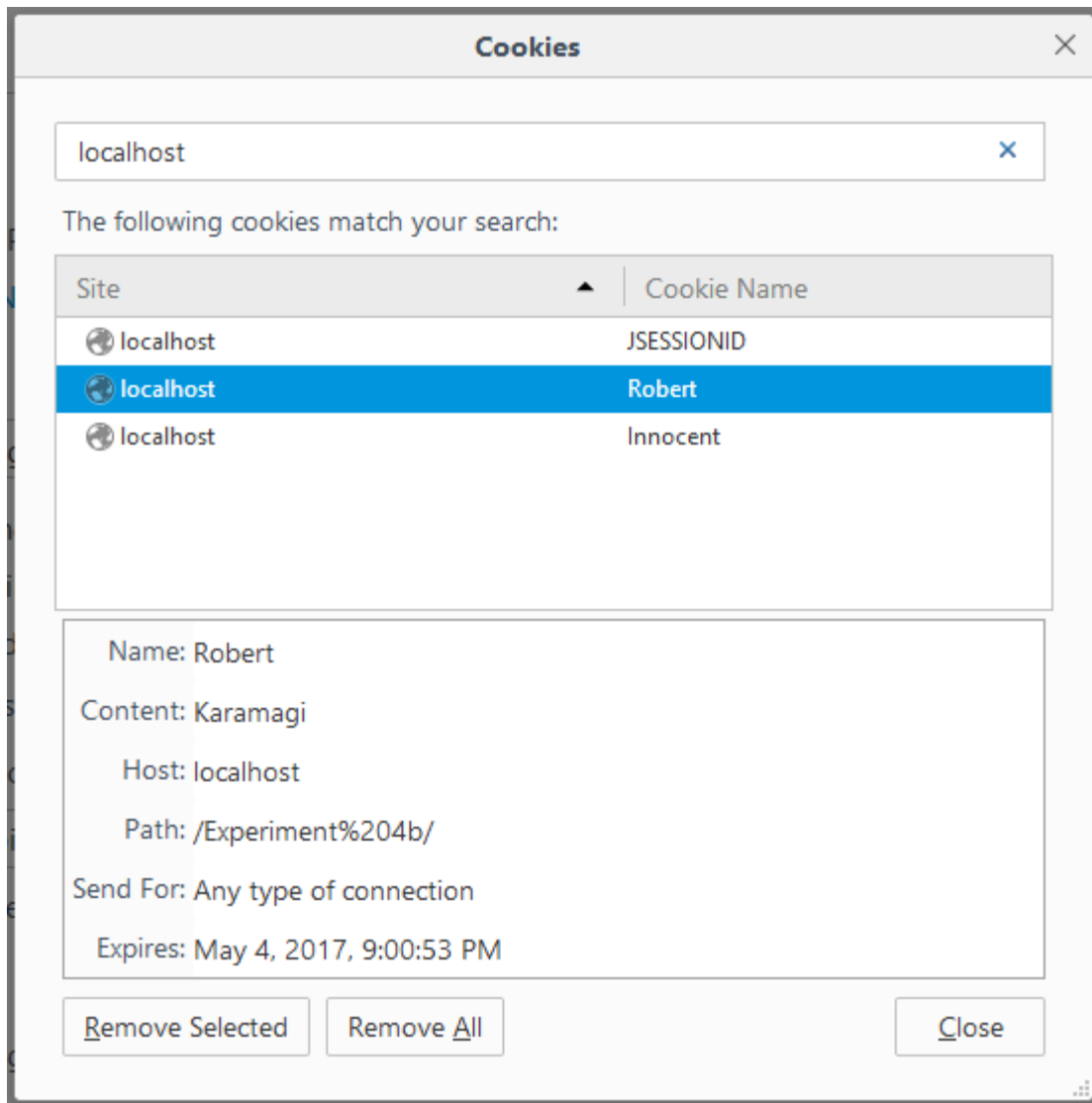
they expire

☐ Clear history when Firefox closes

[Exceptions...](#)

[Show Cookies...](#)

[Settings...](#)



ReadCookie.jsp

```
<html>
<head>
<title>Read Cookie</title>
</head>
<body>
<H1>JSP is Reading two cookies to client </H1>
<table border="1">
<tr>
<td>Cookie Name </td>
```

```

        <td>Cookie Value </td>

        <td>Cookie Max Age </td>

    </tr>

<%

    Cookie[] cookies= request.getCookies();

    // check null because there are chances that there are no cookies
    if(cookies !=null)
    {
        for(int i=0 ;i<cookies.length;i++ )
        {
            Cookie cookie = cookies[i];

            out.println(" <tr> ");

            out.println("<td>" + cookie.getName() + "</td>" );

            out.println("<td>" + cookie.getValue() + "</td>" );

            out.println("<td>" + cookie.getMaxAge() + "</td>" );

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

        }
    }

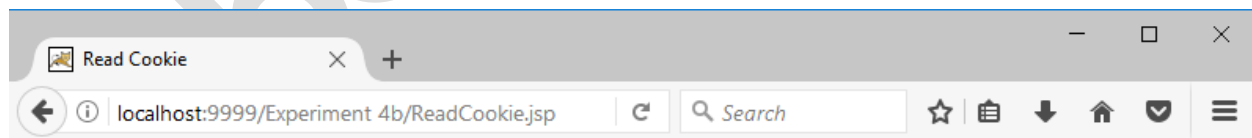
%>

</table>

</body>

</html>

```



JSP is Reading cookies

Cookie Name	Cookie Value	Cookie Max Age
Robert	Karamagi	-1
Innocent	Karamagi	-1
JSESSIONID	04D1A70CE171B408AC9DBB06FDDC9633	-1

DeleteCookie.jsp

```
<html>

<head>

  <title>Delete Cookie</title>

</head>

<body>

  <H1>JSP is deleting  userID cookie sent earlier to the client </H1>

  <%

    Cookie[] cookies= request.getCookies();

    //check null because there are chances that there are no cookies

    if(cookies !=null)

    {

      for(int i=0 ;i<cookies.length;i++ )

      {

        Cookie cookie = cookies[i];

        if(cookie.getName().equals("userID"))

        {

          cookie.setMaxAge(0);

          response.addCookie(cookie);

        }

      }

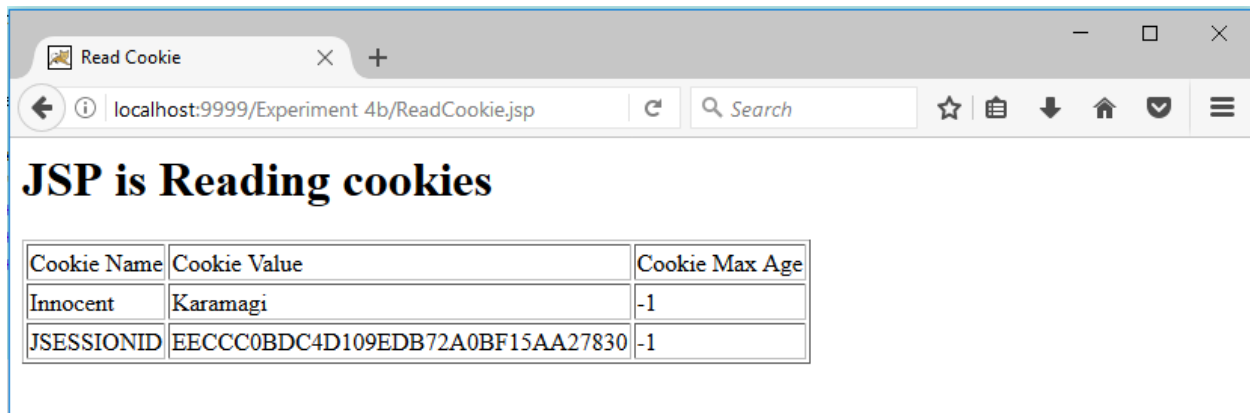
    }

  %>

</body>

</html>
```

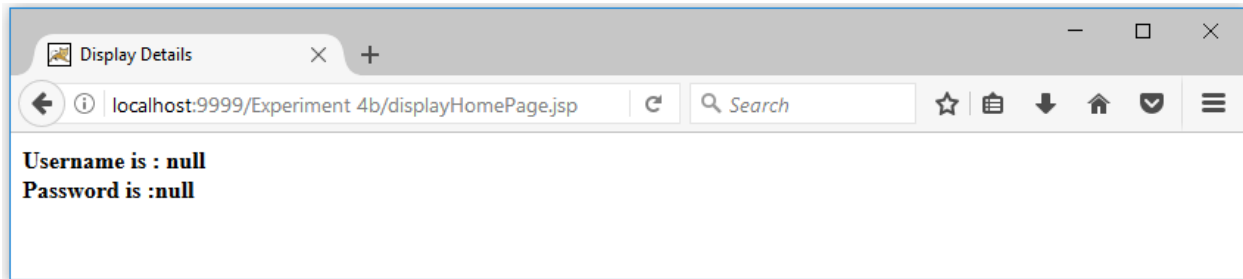




displayHomePage.jsp

```
<html>
<head>
<title>Display Details</title>
</head>
<body>
<%
String username=request.getParameter("username");
String password=request.getParameter("password");
String message="Username is : " + username + "<br/> Password is : " + password ;
String rememberMe= request.getParameter("rememberMe");
if(rememberMe!=null)
{
    Cookie usernameCookie = new Cookie("username-cookie", username);
    Cookie passwordCookie = new Cookie("password-cookie", username);
    usernameCookie.setMaxAge(24*60*60);
    passwordCookie.setMaxAge(24*60*60);
    response.addCookie(usernameCookie);
    response.addCookie(passwordCookie);
}
%>
<strong>
<%= message %>
```

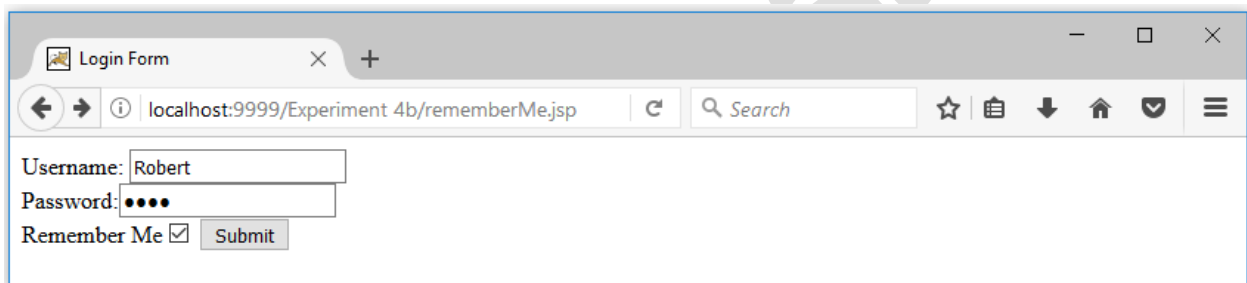
```
</strong>
</body>
</html>
```



rememberMe.jsp

```
<html>
<head>
<title>Login Form</title>
</head>
<body>
<%
    Cookie[] cookies = request.getCookies();
    String username="";
    String password = "";
    if(cookies!=null)
    {
        for(int i=0;i<cookies.length;i++){
            Cookie cookie = cookies[i];
            if(cookie.getName().equals("username-cookie"))
            {
                username= cookie.getValue();
            }
            else if(cookie.getName().equals("password-cookie"))
            {
                password= cookie.getValue();
            }
        }
    }
```

```
}  
%>  
  
<form name="loginform" action="displayHomePage.jsp" method="POST">  
    Username: <input type="text" name="username" value ="<%= username %>" />  
    <br/>  
    Password:<input type="password" name="password" value="<%= password %>" />  
    <br/>  
    Remember Me<input type="checkbox" name="rememberMe" value ="true"/>  
    <input type="submit" value="Submit"/>  
</form>  
</body>  
</html>
```



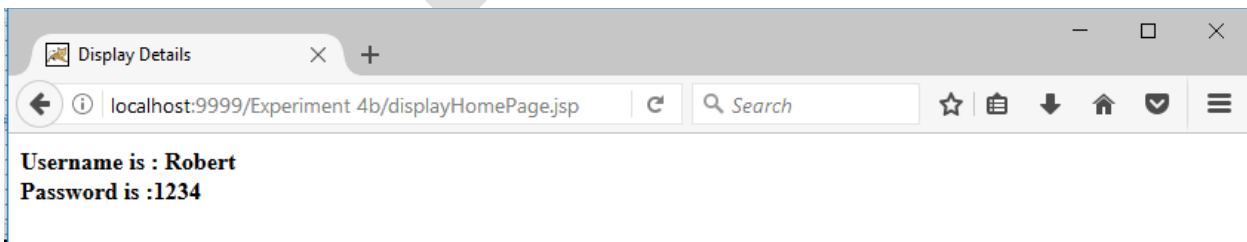
Login Form

localhost:9999/Experiment 4b/rememberMe.jsp

Username: Robert

Password: 1234

Remember Me ☒ Submit



Display Details

localhost:9999/Experiment 4b/displayHomePage.jsp

Username is : Robert

Password is :1234

General

Search

Content

Applications

Privacy

Security

Sync

Advanced

Security

General

☒ Warn you when sites try to install add-ons

Exceptions...

☒ Block dangerous and deceptive content

☒ Block dangerous downloads

☒ Warn you about unwanted and uncommon software

Logins

☒ Remember logins for sites

Exceptions...

☐ Use a master password

Change Master Password...

Saved Logins...

Saved Logins

Search

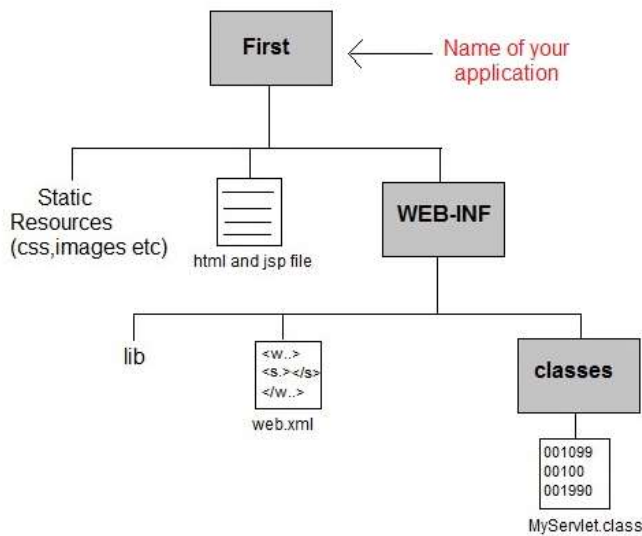
Logins for the following sites are stored on your computer:

Site	Username	Password	Last Changed	
http://localhost:9999	Robert	1234	May 3, 2017	

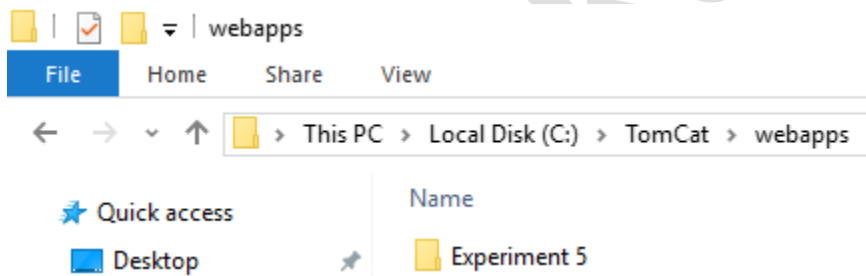
Experiment 5

Writing Servlet Program using Hypertext Transfer Protocol Servlet.

Creating the Directory Structure



Save your application in the webapps directory in the tree structure shown above.



Servlet.java

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

public class Servlet extends HttpServlet
{
    public void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException
    {
        res.setContentType("text/html");//setting the content type
```

```
PrintWriter pw=res.getWriter();//get the stream to write the data
```

```
//writing html in the stream
```

```
pw.println("<html><body>");
```

```
pw.println("Robert Innocent Karamagi");
```

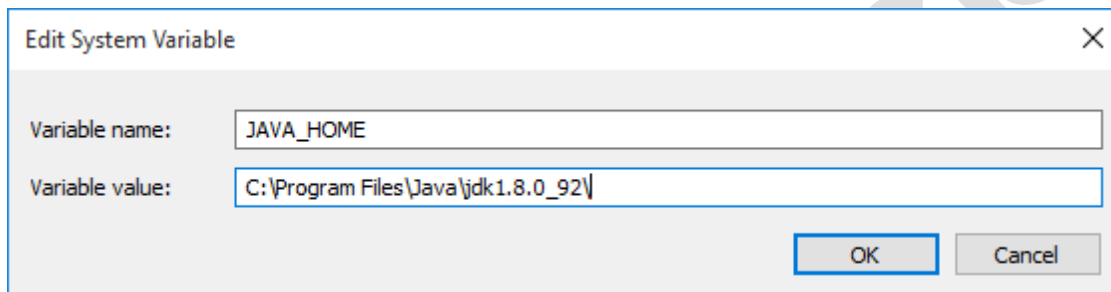
```
pw.println("</body></html>");
```

```
pw.close();//closing the stream
```

```
}
```

```
}
```

Starting up Apache Tom Cat



Configure the connector port at location C:\TomCat\conf\server.xml

Change from 8080 to 9999.

```
-->
<Connector port="9999" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" />
```

Command Prompt

Microsoft Windows [Version 10.0.10240]

(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\Robert Karamagi>cd..

C:\Users>cd..

C:\>cd TomCat

C:\TomCat>cd bin

C:\TomCat\bin>startup

Using CATALINA_BASE: "C:\TomCat"

Using CATALINA_HOME: "C:\TomCat"

Using CATALINA_TMPDIR: "C:\TomCat\temp"

Using JRE_HOME: "C:\Program Files\Java\jdk1.8.0_92\"

Using CLASSPATH: "C:\TomCat\bin\bootstrap.jar;C:\TomCat\bin\tomcat-juli.jar"

C:\TomCat\bin>

```
Tomcat
21-Mar-2017 09:03:55.503 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server version: Apache Tomcat/9.0.0.M18
21-Mar-2017 09:03:55.506 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server built: Mar 8 2017 15:28:57 UTC
21-Mar-2017 09:03:55.506 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server number: 9.0.0.0
21-Mar-2017 09:03:55.500 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log OS Name: Windows 10
21-Mar-2017 09:03:55.507 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log OS Version: 10.0
21-Mar-2017 09:03:55.507 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Architecture: amd64
21-Mar-2017 09:03:55.515 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Java Home: C:\Program Files\Java\jdk1.8.0_92\jre
21-Mar-2017 09:03:55.517 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log JVM Version: 1.8.0_92-b14
21-Mar-2017 09:03:55.518 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log JVM Vendor: Oracle Corporation
21-Mar-2017 09:03:55.518 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log CATALINA_BASE: C:\TomCat
21-Mar-2017 09:03:55.519 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log CATALINA_HOME: C:\TomCat
21-Mar-2017 09:03:55.519 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command Line arguments: -Djava.util.logging.config.file=C:\TomCat\conf\logging.properties
21-Mar-2017 09:03:55.520 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line arguments: -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager
21-Mar-2017 09:03:55.528 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line arguments: -Djdk.tls.spnegoClientAuthSize=2048
21-Mar-2017 09:03:55.531 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line arguments: -Djava.protocol.handler.pkcs=org.apache.catalina.webresources
21-Mar-2017 09:03:55.521 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line arguments: -Dcatalina.base=C:\TomCat
21-Mar-2017 09:03:55.521 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line arguments: -Dcatalina.home=C:\TomCat
21-Mar-2017 09:03:55.522 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line arguments: -Djava.io.tmpdir=C:\TomCat\temp
21-Mar-2017 09:03:55.522 INFO [main] org.apache.catalina.core.AprLifecycleListener.lifecycleEvent loaded APR based Apache Tomcat Native library 1.2.12 using APR version 1.5.2
21-Mar-2017 09:03:55.522 INFO [main] org.apache.catalina.core.AprLifecycleListener.lifecycleEvent APR capabilities: IPv6 [true], sendfile [true], accept filters [false], random [true]
21-Mar-2017 09:03:55.523 INFO [main] org.apache.catalina.core.AprLifecycleListener.lifecycleEvent APR/OpenSSL configuration: useAprConnector [false], useOpenSSL [true]
21-Mar-2017 09:03:56.154 INFO [main] org.apache.catalina.core.AprLifecycleListener.initializeSSLS OpenSSL successfully initialized (OpenSSL 1.0.2k 30 Jan 2017)
21-Mar-2017 09:03:56.294 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["http-nio-8080"]
21-Mar-2017 09:03:56.365 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["ajp-nio-8009"]
21-Mar-2017 09:03:56.365 INFO [main] org.apache.tomcat.util.net.NioSelectorPool.getSharedSelector Using a shared selector for servlet write/read
21-Mar-2017 09:03:56.365 INFO [main] org.apache.tomcat.util.net.NioSelectorPool.getSharedSelector Using a shared selector for servlet write/read
21-Mar-2017 09:03:56.365 INFO [main] org.apache.catalina.startup.Catalina.load Initialization processed in 1280 ms
21-Mar-2017 09:03:56.442 INFO [main] org.apache.catalina.core.StandardService.startInternal Starting service Catalina
21-Mar-2017 09:03:56.442 INFO [main] org.apache.catalina.core.StandardEngine.startInternal Starting Servlet Engine: Apache Tomcat/9.0.0.M18
21-Mar-2017 09:03:56.438 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory C:\TomCat\webapps\docs
21-Mar-2017 09:03:56.820 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory C:\TomCat\webapps\docs has finished in 418 ms
21-Mar-2017 09:03:56.828 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory C:\TomCat\webapps\examples
21-Mar-2017 09:03:57.175 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory C:\TomCat\webapps\examples has finished in 447 ms
21-Mar-2017 09:03:57.277 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory C:\TomCat\webapps\experiment5
21-Mar-2017 09:03:57.315 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory C:\TomCat\webapps\experiment5 has finished in 38 ms
21-Mar-2017 09:03:57.315 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory C:\TomCat\webapps\host-manager
21-Mar-2017 09:03:57.314 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory C:\TomCat\webapps\host-manager has finished in 39 ms
21-Mar-2017 09:03:57.355 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory C:\TomCat\webapps\manager
21-Mar-2017 09:03:57.398 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory C:\TomCat\webapps\manager has finished in 43 ms
21-Mar-2017 09:03:57.398 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory C:\TomCat\webapps\ROOT
21-Mar-2017 09:03:57.427 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory C:\TomCat\webapps\ROOT has finished in 29 ms
21-Mar-2017 09:03:57.438 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-nio-8080"]
21-Mar-2017 09:03:57.462 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["ajp-nio-8009"]
21-Mar-2017 09:03:57.478 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in 1891 ms
```

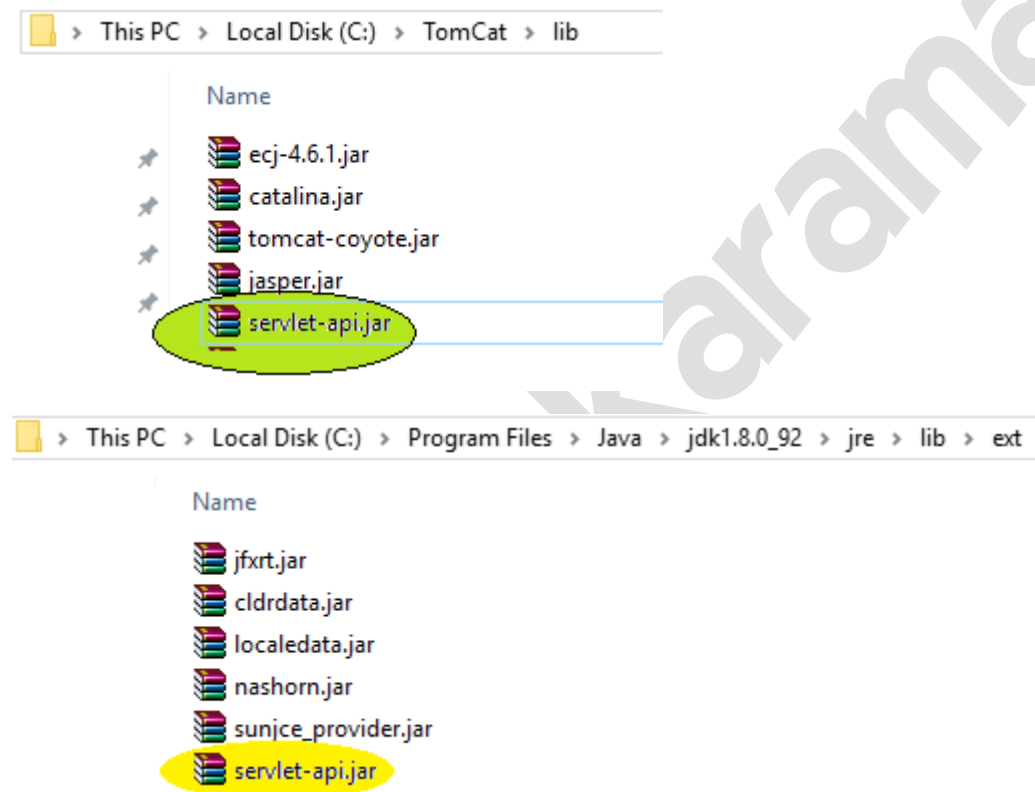
Compiling Servlet.java using Apache Tomcat: servlet-api.jar

Method 1

```
C:\TomCat\bin>cd..  
C:\TomCat>cd webapps  
C:\TomCat\webapps>cd Experiment 5  
C:\TomCat\webapps\Experiment 5>javac -cp .;C:\TomCat\lib\servlet-api.jar Servlet.java
```

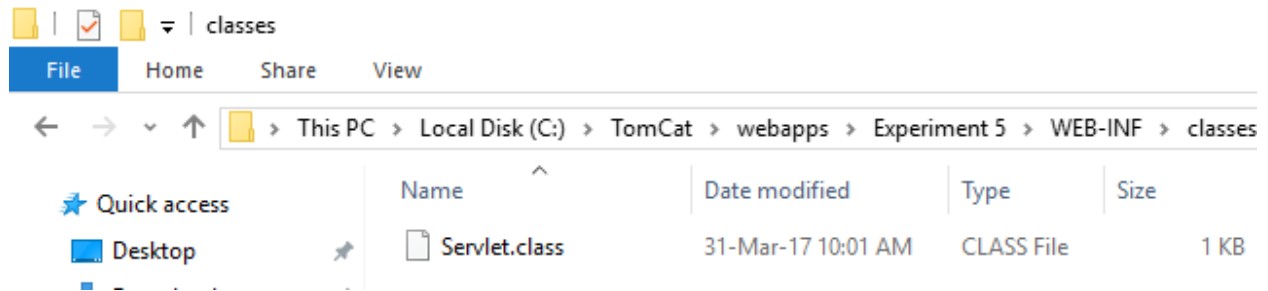
Method 2

Copy “servlet-api.jar” from “C:\TomCat\lib” to “C:\Program Files\Java\jdk1.8.0_92\jre\lib\ext”



```
C:\TomCat\webapps\Experiment 5>javac Servlet.java
```

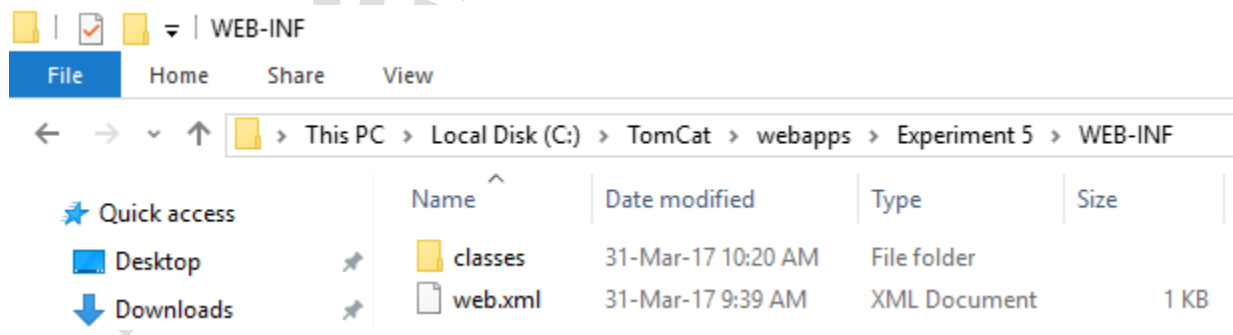
After compiling save the Servlet.class into WEB-INF/classes/ directory.



web.xml (compulsory name)

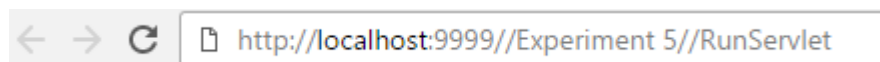
```
<web-app>
<servlet>
<servlet-name>Robert</servlet-name>
<servlet-class>Servlet</servlet-class>
</servlet>
<servlet-mapping>
fw<servlet-name>Robert</servlet-name>
<url-pattern>/RunServlet</url-pattern>
</servlet-mapping>
</web-app>
```

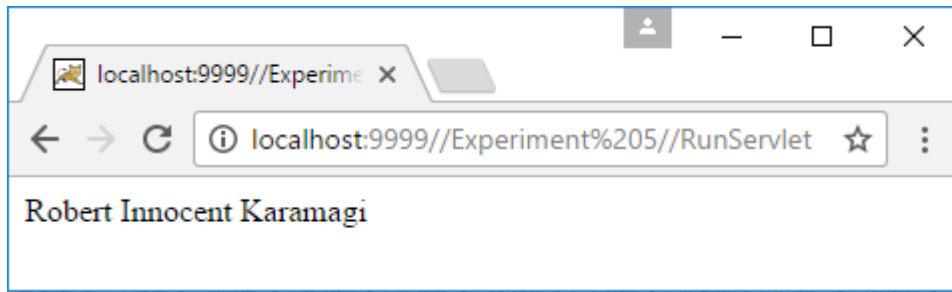
Save web.xml in the WEB-INF directory



Run the Server

<http://localhost:9999/Folder name at top of directory structure/Content at <url-pattern> in web.xml>



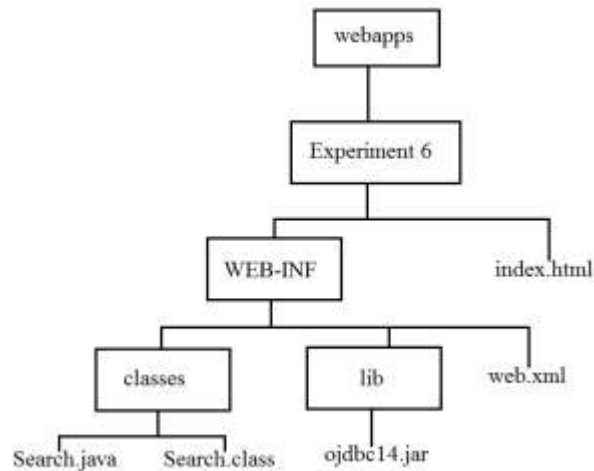


Robert Karamagi

Experiment 6

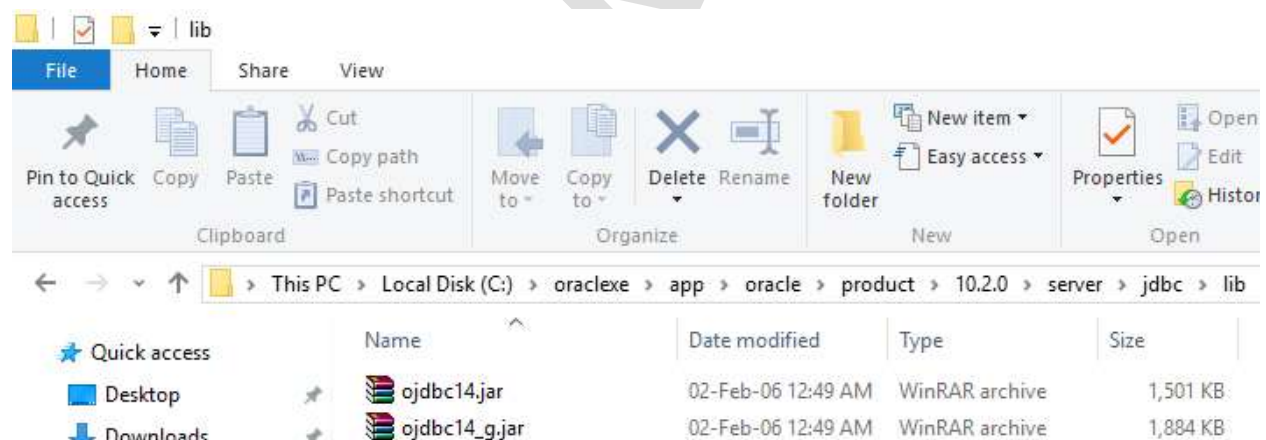
Any online application with database access.

We shall observe the following tree structure.



Copy the ojdbc.jar file i.e Oracle Java Database Connectivity Java Archive file from Oracle 10g in the location C:\oracle\app\oracle\product\10.2.0\server\jdbc\lib and paste it into your lib directory i.e.

C:\TomCat\webapps\Experiment 6\WEB-INF\lib



Search.java

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

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

public class Search extends HttpServlet
{

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

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        String rollno=request.getParameter("roll");
        int roll=Integer.valueOf(rollno);

        try
        {

            Class.forName("oracle.jdbc.driver.OracleDriver");
            Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","1234");

            PreparedStatement ps=con.prepareStatement("select * from result where rollno=?");
            ps.setInt(1,roll);

            out.print("<table width=50% border=1>");
            out.print("<caption>Result:</caption>");

            ResultSet rs=ps.executeQuery();

            /* Printing column names */

            ResultSetMetaData rsmd=rs.getMetaData();
```

```
int total=rsmd.getColumnCount();

out.print("<tr>");

for(int i=1;i<=total;i++)

{

    out.print("<th>"+rsmd.getColumnName(i)+"</th>");

}

out.print("</tr>");

/* Printing result */

while(rs.next())

{

    out.print("<tr><td>"+rs.getInt(1)+"</td><td>"+rs.getString(2)+"</td><td>"+rs.getString(3)+"</td><td>"+rs.getString(4)+"</td></tr>");

}

out.print("</table>");

}

catch (Exception e2)

{

    e2.printStackTrace();

}

finally

{

    out.close();

}

}
```

}

```
Command Prompt
C:\TomCat\webapps\Experiment 6\WEB-INF\classes>javac Search.java
C:\TomCat\webapps\Experiment 6\WEB-INF\classes>_
```

index.html

```
<html>
<body>
<form action="/Search">
Enter your Rollno:<input type="text" name="roll"/><br/>

<input type="submit" value="search"/>
</form>
</body>
</html>
```

web.xml

```
<web-app>

<servlet>
<servlet-name>Search</servlet-name>
<servlet-class>Search</servlet-class>
</servlet>

<servlet-mapping>
<servlet-name>Search</servlet-name>
<url-pattern>/Search</url-pattern>
</servlet-mapping>

</web-app>
```

Creating the Database

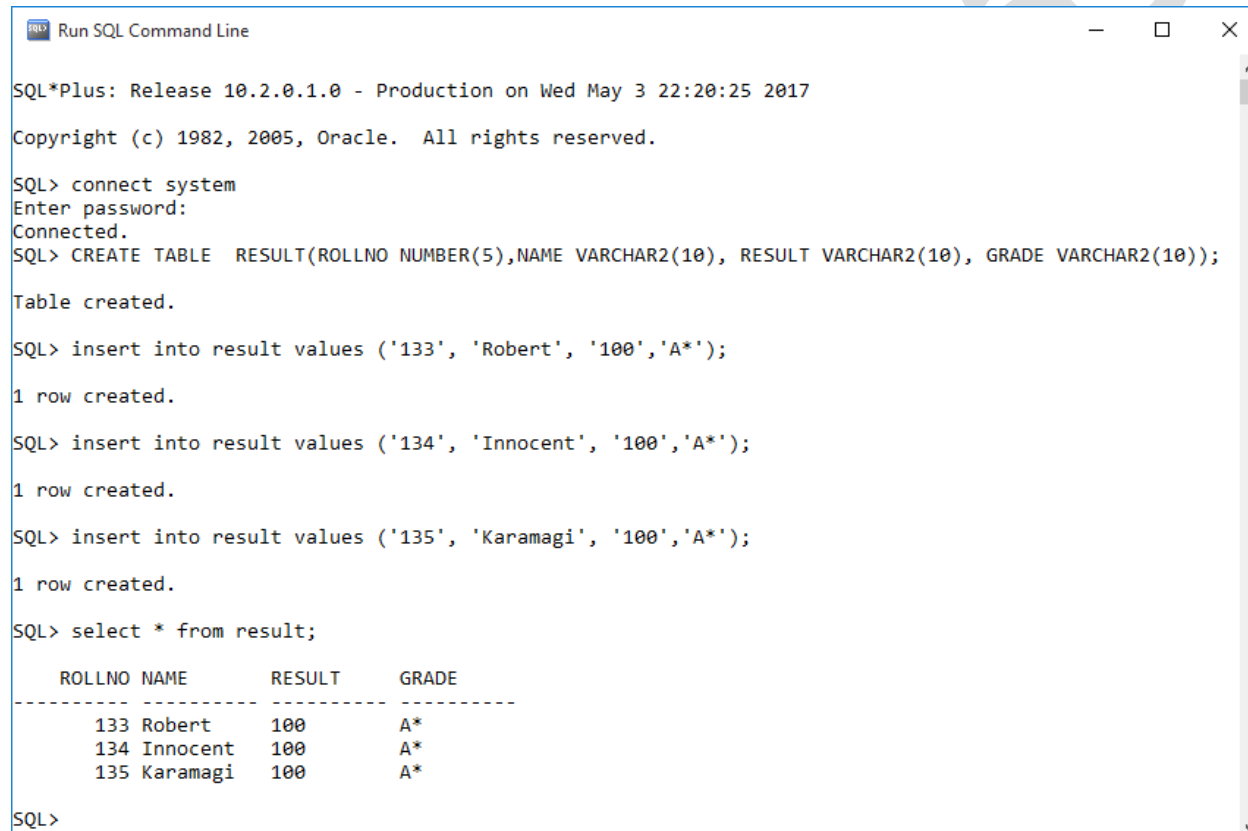
```
CREATE TABLE RESULT(ROLLNO NUMBER(5),NAME VARCHAR2(10), RESULT VARCHAR2(10),  
GRADE VARCHAR2(10));
```

```
insert into result values ('133', 'Robert', '100','A*');
```

```
insert into result values ('134', 'Innocent', '100','A*');
```

```
insert into result values ('135', 'Karamagi', '100','A*');
```

```
select * from result;
```



```
Run SQL Command Line

SQL*Plus: Release 10.2.0.1.0 - Production on Wed May 3 22:20:25 2017
Copyright (c) 1982, 2005, Oracle. All rights reserved.

SQL> connect system
Enter password:
Connected.
SQL> CREATE TABLE RESULT(ROLLNO NUMBER(5),NAME VARCHAR2(10), RESULT VARCHAR2(10), GRADE VARCHAR2(10));
Table created.

SQL> insert into result values ('133', 'Robert', '100','A*');
1 row created.

SQL> insert into result values ('134', 'Innocent', '100','A*');
1 row created.

SQL> insert into result values ('135', 'Karamagi', '100','A*');
1 row created.

SQL> select * from result;

  ROLLNO NAME      RESULT  GRADE
-----
    133 Robert      100     A*
    134 Innocent    100     A*
    135 Karamagi    100     A*

SQL>
```

Accessing the Database via the Servlet

Run the servlet by typing the command in any browser : localhost:9999/Experiment 6

http://localhost:9999//Exper X +

localhost:9999//Experiment 6/ Search

Enter your Rollno:

http://localhost:9999//Exper X +

localhost:9999//Experiment 6/Search?roll=133 Search

Result:

ROLLNO	NAME	RESULT	GRADE
133	Robert	100	A*

http://localhost:9999//Exper X +

localhost:9999//Experiment 6/ Search

Enter your Rollno:

http://localhost:9999//Exper X +

localhost:9999//Experiment 6/Search?roll=134 Search

Result:

ROLLNO	NAME	RESULT	GRADE
134	Innocent	100	A*

http://localhost:9999//Exper X +

localhost:9999//Experiment 6/ Search

Enter your Rollno:

Result:			
ROLLNO	NAME	RESULT	GRADE
135	Karamagi	100	A*

WRITING TO THE DATABASE

PROGRAM

index.html

```

<html>
<head><title>Donate to Suffering Children in Tanzania</title></head>
<body bgcolor=#D6CDB3>
<center>
<font size=10 color=grey>
<b>Poverty in Tanzania
</font>

<table align=center>
<tr>
<td><img src=tanzania1.jpg height=200 width=250></td>
<td><img src=tanzania2.jpg height=200 width=250></td>
<td><img src=tanzania3.jpg height=200 width=250></td>
</tr>
</table><br>

<h4 align=center>
Tanzania has been blood stricken by disease and famine that has left young children<br>

```

helpless and dieing of malnutrition. They have no proper shelter, food, water,

hospitals, medicine, schools and many important needs to survive.

Please donate some money to help our struggling brothers and sisters.

</h4>

```
<form action="/Donate" method="POST">  
<h3 >Donation Amount</h3>  
<input type="text" name="roll"/><br><br>  
<input type="submit" value="Donate"/>  
</form>  
</center>  
</body>  
</html>
```

Donation.java

```
import java.io.*;  
import java.sql.*;  
import javax.servlet.*;  
import javax.servlet.http.*;  
  
public class Donation extends HttpServlet{  
  
    public void doPost(HttpServletRequest req, HttpServletResponse res) throws  
        ServletException, IOException{  
        res.setContentType("text/html");  
        PrintWriter pw = res.getWriter();  
        String donate=req.getParameter("roll");  
        int roll=Integer.valueOf(donate);  
        String name=req.getParameter("name");
```

```

try{
    Class.forName("oracle.jdbc.driver.OracleDriver");
    Connection
    con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe"
    ,"system","12345");
    PreparedStatement ps=con.prepareStatement("insert into donation
    values(?)");
    ps.setInt(1,roll);
    ps.executeUpdate();
    con.commit();
    con.close();
    pw.println("<html><head><title>Thank You</title></head>");
    pw.println("<body bgcolor=#D6CDB3 align=center>");
    pw.println("<h3>Thank you for your kind donation to the suffering
    children of Tanzania.</h3>");
    pw.println("</body></html>");
}
catch (Exception e){
    e.printStackTrace();
}
finally{
    pw.close();
}
}
}

```

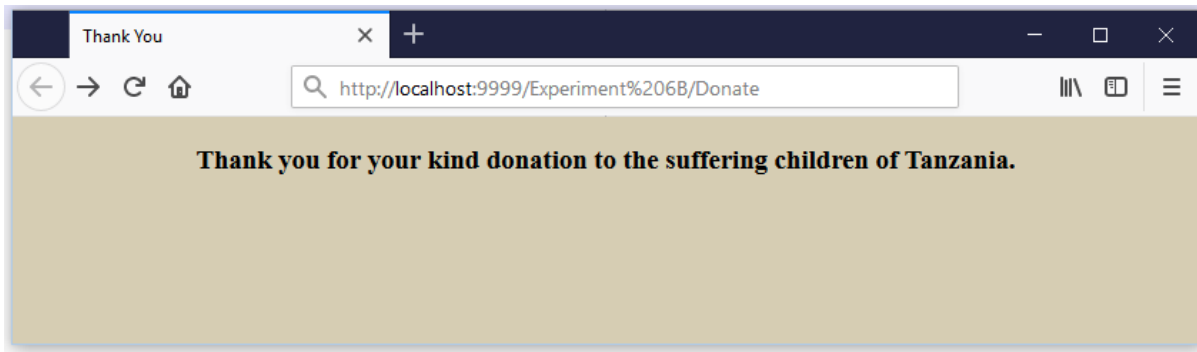
OUTPUT

Oracle Database - Table Creation

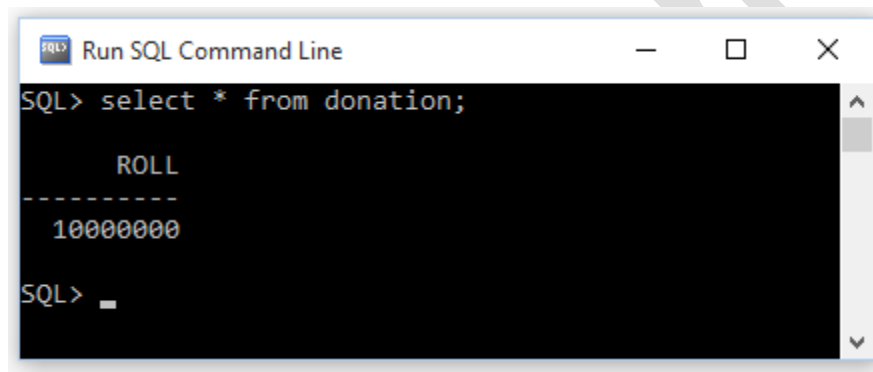
```
Run SQL Command Line
SQL> create table donation(roll number(20));
Table created.
SQL> desc donation;
Name                                         Null?    Type
-----
ROLL                                         NUMBER(20)
SQL>
```

Writing into the Database via the Servlet





Oracle Database - Table Viewing



Experiment 7

Creation of XML document for a specific domain

books.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<?xml-stylesheet type="text/css" href="books.css"?>
<books>
  <book>
    <bookname>Network Programming</bookname>
    <authorname>Robert Innocent Karamagi</authorname>
    <country>United Democratic Republic of Tanzania</country>
    <publisher>Penguin Random House</publisher>
    <price>£100</price>
    <year>2030</year>
  </book>
  <book>
    <bookname>Internet Programming</bookname>
    <authorname>Robert Innocent Karamagi</authorname>
    <country>United Democratic Republic of Tanzania</country>
    <publisher>Hachette Livre</publisher>
    <price>£100</price>
    <year>2030</year>
  </book>
</books>
```

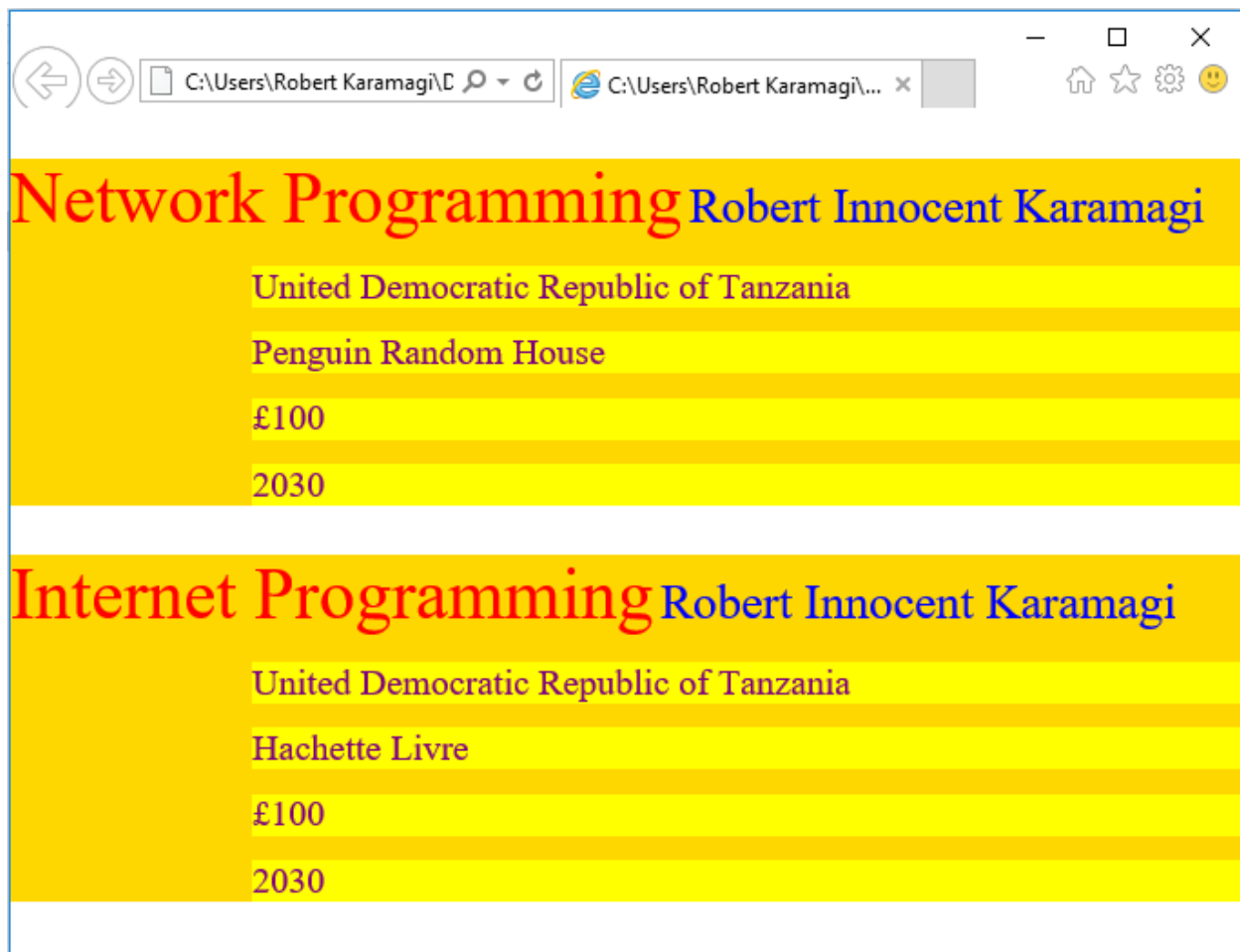
books.css

```
<style>
books
{
```

```
}
```

```
book
```

```
{
background-color:gold;
display: block;
margin-bottom: 20pt;
margin-top: 20pt;
}
bookname
{
color: red;
font-size: 30pt;
}
authorname
{
color: blue;
font-size: 20pt;
}
country,price,year,publisher
{
background-color:yellow;
display: block;
color: purple;
font-size: 15pt;
margin-left: 100pt;
margin-top: 10pt;
}
</style>
```



WRITING DTD AND XML SCHEMA FOR THE DOMAIN SPECIFIC XML DOCUMENT

DOCUMENT TYPE DEFINITION (DTD)

PROGRAM

student.xml

```
<?xml version = "1.0"?>
<?xml-stylesheet type="text/css" href= "student.css"?>
<!DOCTYPE student SYSTEM "student.dtd">
<student>
<name>
<firstname> Robert</firstname>
<lastname> Karamagi </lastname>
</name>
<address>
<street>Mbezi,Luguruni</street>
<city> Dar es Salaam </city>
<email> robertokaramagi@gmail.com </email>
<phone> 0762087332 </phone>
</address>
</student>
```

student.dtd

```
<?xml version = "1.0"?>
<!ELEMENT student (student+)>
```

```
<!ELEMENT student (name,address)>
<!ELEMENT name (firstname,lastname)>
<!ELEMENT firstname (#PCDATA)>
<!ELEMENT lastname (#PCDATA)>
<!ELEMENT address (street,city,email,phone)>
<!ELEMENT street (#PCDATA)>
<!ELEMENT city (#PCDATA)>
<!ELEMENT email (#PCDATA)>
<!ELEMENT lastname (#PCDATA)>
```

student.css

```
student{
display: block;
margin-top: 15px;
color: blue;
}
```

```
name{
display: block;
margin-left:40px;
margin-top: 30pt;
color:red;
}
```

```
firstname{
font-size: 28pt;
}
```

```
lastname{
font-size: 28pt;
```

```
}
```

```
address{  
display: block;  
margin-left:40px;  
color:green;  
}
```

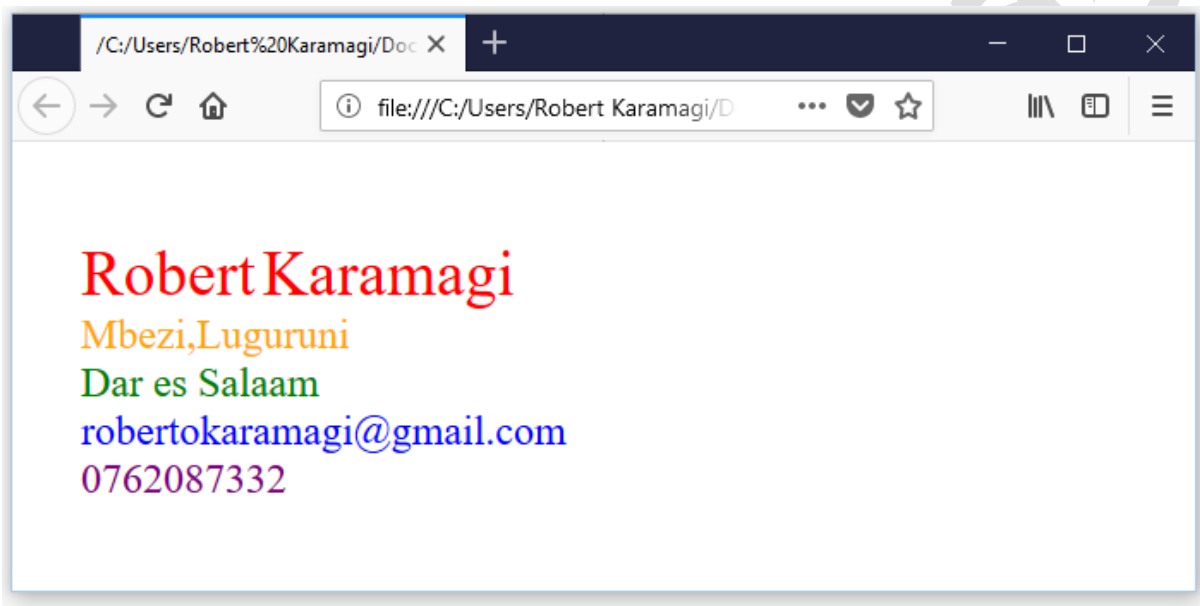
```
street{  
display: block;  
font-size: 18pt;  
color:orange;  
}
```

```
city{  
display: block;  
font-size: 18pt;  
color:green  
}
```

```
email{  
display: block;  
font-size: 18pt;  
color:blue  
}
```

```
phone{  
display: block;  
font-size: 18pt;  
color:purple  
}
```

OUTPUT



XML SCHEMA

PROGRAM

card.xml

```
<?xml-stylesheet type="text/css" href="card.css"?>
<card xmlns="https://robertkaramagi.wordpress.com/"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
    xsi:schemaLocation="test.xsd">
    <name>Robert Innocent Karamagi</name>
    <title>Student, B.E. Computer Science and Engineering</title>
    <email>robertokaramagi@gmail.com</email>
    <phone>+255762087332</phone>
  </card>
```

card.xsd

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:b="https://robertkaramagi.wordpress.com/"
  targetNamespace="https://robertkaramagi.wordpress.com/">

  <element name="card" type="b:card_type"/>
  <element name="name" type="string"/>
  <element name="title" type="string"/>
  <element name="email" type="string"/>
  <element name="phone" type="string"/>

  <complexType name="card_type">
    <sequence>
      <element ref="b:name"/>
      <element ref="b:title"/>
      <element ref="b:email"/>
      <element ref="b:phone"/>
    </sequence>
  </complexType>

</schema>
```

card.css

```
card{  
display: block;  
font-family:Lucida Console;  
font-weight: bold;  
margin-top: 50px;  
margin-left:100px;  
margin-right:620px;  
border: 5px solid yellow;  
border-radius: 20px;  
}
```

```
name{  
display: block;  
margin-left:50px;  
margin-top: 20pt;  
color:red;  
font-size: 28pt;  
}
```

```
title{  
display: block;  
margin-left:50px;  
margin-top: 20pt;  
color:green;  
font-size: 15pt;  
}
```

```
email{  
display: block;
```

```
margin-left:50px;  
margin-top: 20pt;  
color:blue;  
font-size: 12pt;  
font-style: italic;  
}
```

```
phone{  
display: block;  
margin-left:50px;  
margin-top: 20pt;  
margin-bottom:20pt;  
color:purple;  
font-size: 12pt;  
font-style: italic;  
}
```

OUTPUT

Robert Innocent Karamagi

Student, B.E. Computer Science and Engineering

robertokaramagi@gmail.com

+255762087332

Experiment 9a

Parsing an XML document using Document Object Model Parser

people.xml

```
<?xml version="1.0"?>
<people>
    <person id="1">
        <lastname>Karamagi</lastname>
        <firstname>Robert</firstname>
    </person>
    <person id="2">
        <lastname>Karamagi</lastname>
        <firstname>Innocent</firstname>
    </person>
</people>
```

MyDomParser.java

```
import java.io.IOException;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.ParserConfigurationException;
import org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.SAXException;

public class MyDomParser
{
    public static void main(String[] args)
    {
        DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
        try
        {
            DocumentBuilder builder = factory.newDocumentBuilder();
            Document doc = builder.parse("people.xml");
            NodeList personList = doc.getElementsByTagName("person");
            for(int i=0; i<personList.getLength(); i++)
            {
                Node p = personList.item(i);
                if(p.getNodeType() == Node.ELEMENT_NODE)
                {
                    Element person = (Element) p;
                    String id = person.getAttribute("id");
                    NodeList nameList = person.getChildNodes();
                    for(int j=0; j<nameList.getLength(); j++)
                    {
                        Node n = nameList.item(j);
                        if(n.getNodeType() == Node.ELEMENT_NODE)
                        {
                            Element name = (Element) n;
                            System.out.println("Person" + id + ":" + name.getTagName() + "=" + name.getTextContent());
                        }
                    }
                }
            }
        }
        catch (ParserConfigurationException | SAXException | IOException e)
        {
            e.printStackTrace();
        }
    }
}
```

```

    }
}
}
catch (ParserConfigurationException e)
{
    // TODO Auto-generated catch block
    e.printStackTrace();
}
catch (SAXException e)
{
    // TODO Auto-generated catch block
    e.printStackTrace();
}
catch (IOException e)
{
    // TODO Auto-generated catch block
    e.printStackTrace();
}
}
}
}
Linux Mint

```



The screenshot shows a terminal window titled "robert@Karamagi ~/internet/EX9A". The terminal output is as follows:

```

robert@Karamagi ~ $ cd internet
robert@Karamagi ~/internet $ cd EX9A
robert@Karamagi ~/internet/EX9A $ javac MyDomParser.java
robert@Karamagi ~/internet/EX9A $ java MyDomParser
Person1:lastname=Karamagi
Person1:firstname=Robert
Person2:lastname=Karamagi
Person2:firstname=Innocent
robert@Karamagi ~/internet/EX9A $

```

Windows 10

```
Select Command Prompt
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\Robert Karamagi>cd..

C:\Users>cd..

C:\>cd java

C:\Java>cd ex9a

C:\Java\EX9A>javac -cp . MyDomParser.java

C:\Java\EX9A>java -cp . MyDomParser
Person1:lastname=Karamagi
Person1:firstname=Robert
Person2:lastname=Karamagi
Person2:firstname=Innocent

C:\Java\EX9A>
```

input.txt

```
<?xml version="1.0"?>
<class>

  <student rollno="15173055133">
    <firstname>Robert</firstname>
    <lastname>Karamagi</lastname>
    <nickname>Rob</nickname>
    <marks>100</marks>
  </student>
</class>
```

DomParserDemo.java

```
import java.io.File;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.DocumentBuilder;
import org.w3c.dom.Document;
import org.w3c.dom.NodeList;
import org.w3c.dom.Node;
import org.w3c.dom.Element;

public class DomParserDemo {
    public static void main(String[] args){

        try {
            File inputFile = new File("input.txt");
```

```

DocumentBuilderFactory dbFactory
    = DocumentBuilderFactory.newInstance();
DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();
Document doc = dBuilder.parse(inputFile);
doc.getDocumentElement().normalize();
System.out.println("Root element : "
    + doc.getDocumentElement().getNodeName());
NodeList nList = doc.getElementsByTagName("student");
System.out.println("-----");
for (int temp = 0; temp < nList.getLength(); temp++) {
    Node nNode = nList.item(temp);
    System.out.println("\nCurrent Element : "
        + nNode.getNodeName());
    if (nNode.getNodeType() == Node.ELEMENT_NODE) {
        Element eElement = (Element) nNode;
        System.out.println("Student roll no : "
            + eElement.getAttribute("rollno"));
        System.out.println("First Name : "
            + eElement
                .getElementsByTagName("firstname")
                .item(0)
                .getTextContent());
        System.out.println("Last Name : "
            + eElement
                .getElementsByTagName("lastname")
                .item(0)
                .getTextContent());
        System.out.println("Nick Name : "
            + eElement
                .getElementsByTagName("nickname")
                .item(0)
                .getTextContent());
        System.out.println("Marks : "
            + eElement
                .getElementsByTagName("marks")
                .item(0)
                .getTextContent());
    }
}
} catch (Exception e) {
    e.printStackTrace();
}
}
}

```

```
Command Prompt

C:\Java\EX9A>javac DomParserDemo.java

C:\Java\EX9A>java -cp . DomParserDemo
Root element :class
-----
Current Element :student
Student roll no : 15173055133
First Name : Robert
Last Name : Karamagi
Nick Name : Rob
Marks : 100

C:\Java\EX9A>
```

Experiment 9b

Parsing an XML document using Simple Application Program Interface for eXtensible Markup Language Parser

sample.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<breakfast_menu>
  <food>
    <name>Robert's Eggs</name>
    <price>$5.95</price>
    <description>Our famous scrambled eggs with toast</description>
    <calories>650</calories>
  </food>

  <food>
    <name>Karamagi's Soup</name>
    <price>$6.55</price>
    <description>Chicken and Beef Soup with a Chilli Blend</description>
    <calories>1070</calories>
  </food>
</breakfast_menu>
```

ReadXMLFile.java

```
import java.io.IOException;

import javax.xml.parsers.ParserConfigurationException;
import javax.xml.parsers.SAXParser;
import javax.xml.parsers.SAXParserFactory;
import org.xml.sax.Attributes;
import org.xml.sax.SAXException;
import org.xml.sax.helpers.DefaultHandler;

public class ReadXMLFile
{
    public static void main(String[] args)
    {
        // obtain and configure a SAX based parser
        SAXParserFactory factory = SAXParserFactory.newInstance();

        //obtain object for SAX parser
        SAXParser saxParser = null;
        try {
            saxParser = factory.newSAXParser();
        } catch (ParserConfigurationException e1) {
            // TODO Auto-generated catch block
            e1.printStackTrace();
        } catch (SAXException e1) {
            // TODO Auto-generated catch block
            e1.printStackTrace();
        }

        /*
        *default handler for SAX handler class
        */
    }
}
```

*all three methods are written in handler's body
*/

```
DefaultHandler handler = new DefaultHandler()  
{
```

```
    boolean bname = false;  
    boolean bprice = false;  
    boolean bdescription = false;  
    boolean bcalories = false;
```

```
    //this method is called every time the parse gets an open tag '<'  
    //identifies which tag is being open at time by assigning an open flag  
    public void startElement(String uri, String localName, String qName,  
        Attributes attributes) throws SAXException
```

```
    {  
        if(qName.equalsIgnoreCase("name"))  
        {  
            bname=true;  
        }  
        if(qName.equalsIgnoreCase("price"))  
        {  
            bprice=true;  
        }  
        if(qName.equalsIgnoreCase("description"))  
        {  
            bdescription=true;  
        }  
        if(qName.equalsIgnoreCase("calories"))  
        {  
            bcalories=true;  
        }  
    }  
}
```

```
    // this method is called every time the parser gets a closing tag '>'  
    public void endElement(String uri, String localName, String qName) throws
```

SAXException

```
    {  
    }  
  
    //print data stored in between '<' and '>' tags  
    public void characters(char ch[], int start, int length) throws SAXException  
    {  
        if(bname)  
        {  
            System.out.println("name: "+new String(ch, start, length ));  
            bname=false;  
        }  
  
        if(bprice)  
        {  
            System.out.println("price: "+new String(ch, start, length ));  
            bprice=false;  
        }  
  
        if(bdescription)
```

```

        {
            System.out.println("description: "+new String(ch, start, length ));
            bdescription=false;
        }

        if(bcalories)
        {
            System.out.println("calories: "+new String(ch, start, length ));
            bcalories=false;
        }
    }
};

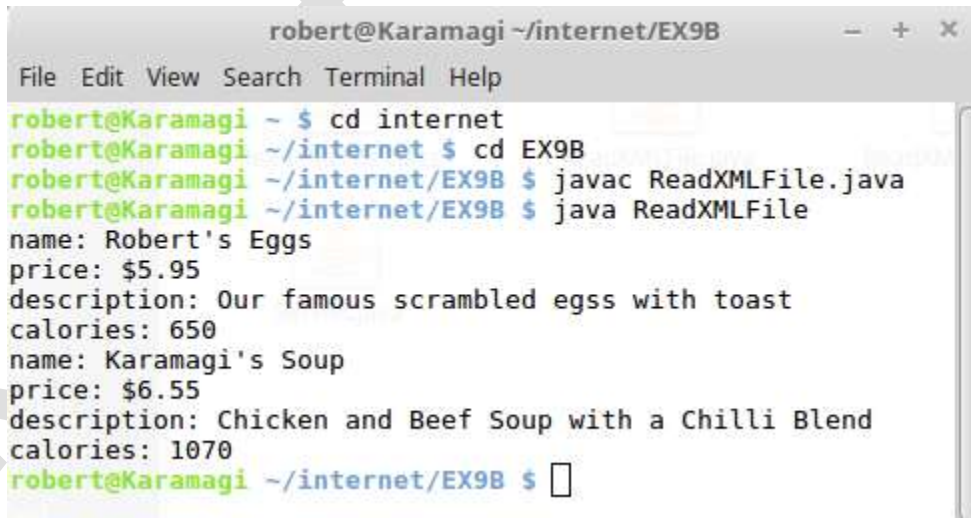
try {
    saxParser.parse("sample.xml",handler);
} catch (SAXException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}

}

}

```

Linux Mint

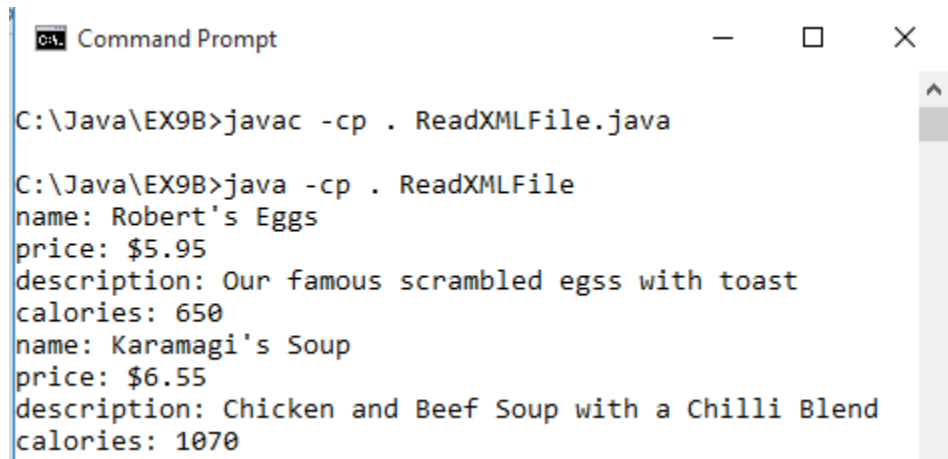


```

robert@Karamagi ~/internet/EX9B
File Edit View Search Terminal Help
robert@Karamagi ~ $ cd internet
robert@Karamagi ~/internet $ cd EX9B
robert@Karamagi ~/internet/EX9B $ javac ReadXMLFile.java
robert@Karamagi ~/internet/EX9B $ java ReadXMLFile
name: Robert's Eggs
price: $5.95
description: Our famous scrambled eggs with toast
calories: 650
name: Karamagi's Soup
price: $6.55
description: Chicken and Beef Soup with a Chilli Blend
calories: 1070
robert@Karamagi ~/internet/EX9B $

```

Windows 10



```
C:\Java\EX9B>javac -cp . ReadXMLFile.java

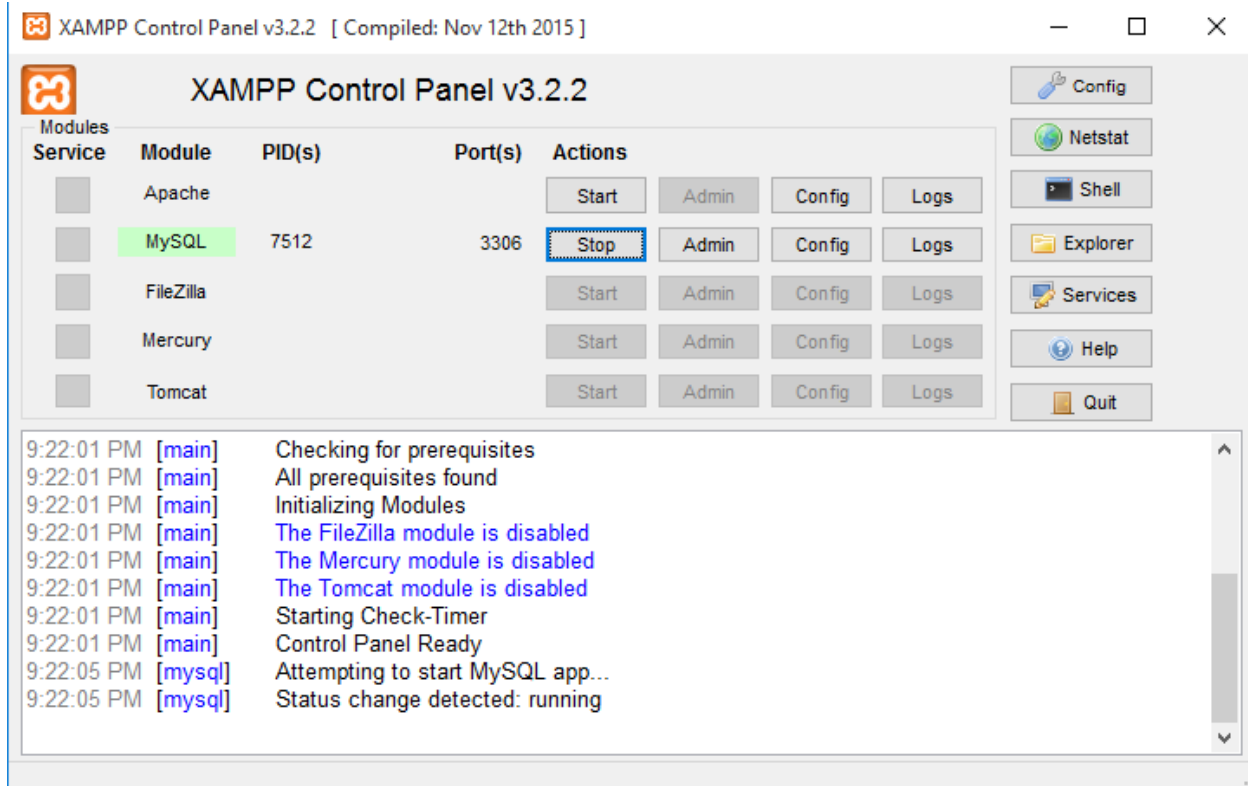
C:\Java\EX9B>java -cp . ReadXMLFile
name: Robert's Eggs
price: $5.95
description: Our famous scrambled egss with toast
calories: 650
name: Karamagi's Soup
price: $6.55
description: Chicken and Beef Soup with a Chilli Blend
calories: 1070
```

Experiment 10

Aim

Sample web application development in the open source environment

Start the mysql server using the xampp control panel



Create the database in mysql

Open the command prompt and browse using cd command to the bin directory location of mysql in xampp.

Enter the commands to create the database.

Commands

```
mysql -u root -p -h 127.0.0.1
```

```
show databases;
```

```
use mysql;
```

```
create table library (Book varchar(20) , Author varchar(20) ,ISBN varchar(20));
```

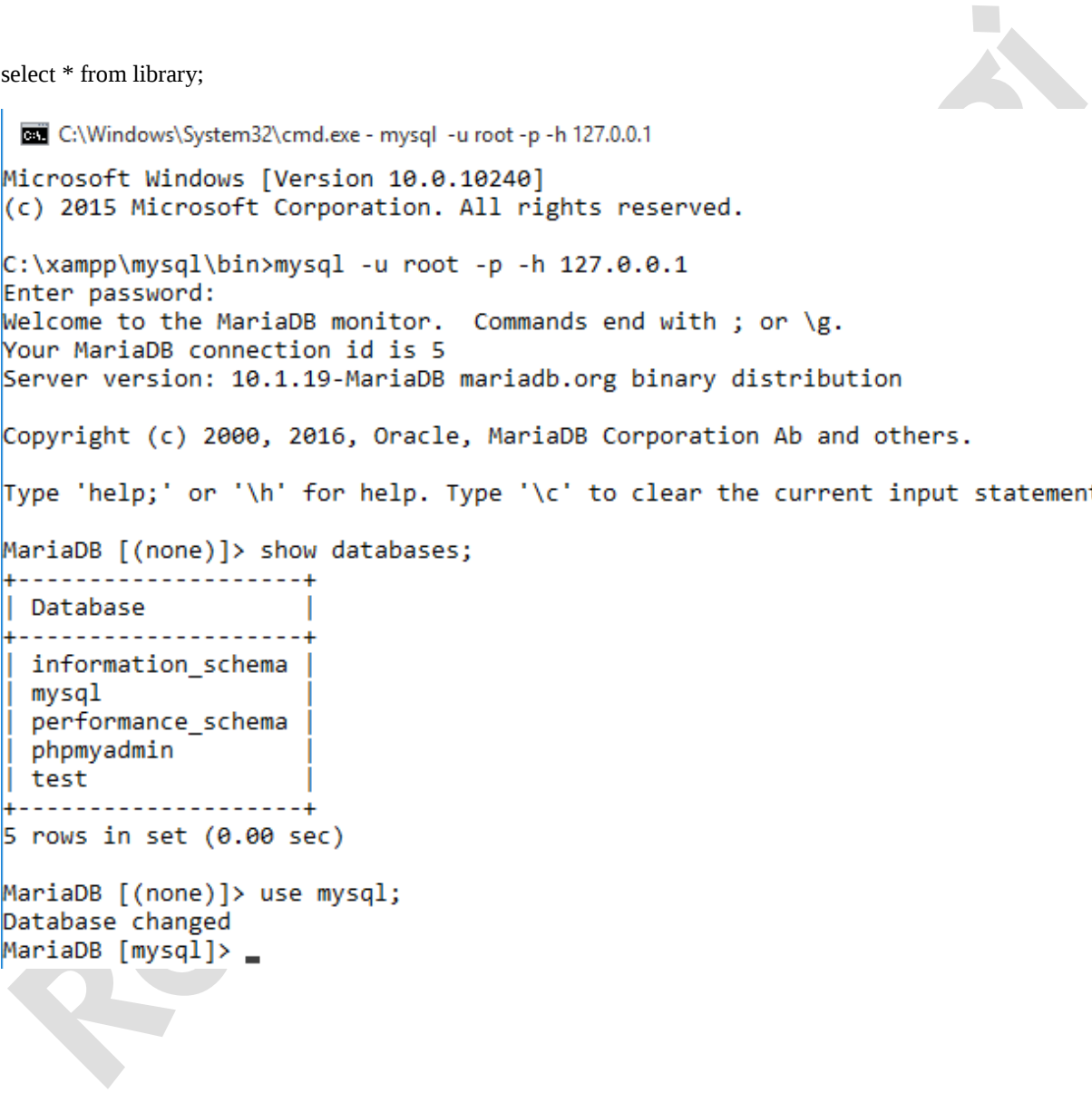
```
ALTER TABLE library ADD PRIMARY KEY (Book);
```

describe library;

insert into library values('Internet Programming','Robert','111-232-458');

insert into library values('Network Programming','Karamagi','134-869-706');

select * from library;



```
C:\Windows\System32\cmd.exe - mysql -u root -p -h 127.0.0.1
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\xampp\mysql\bin>mysql -u root -p -h 127.0.0.1
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 5
Server version: 10.1.19-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| phpmyadmin |
| test |
+-----+
5 rows in set (0.00 sec)

MariaDB [(none)]> use mysql;
Database changed
MariaDB [mysql]> _
```

```
MariaDB [mysql]> create table library (Book varchar(20) , Author varchar(20) ,ISBN varchar(20));
Query OK, 0 rows affected (0.05 sec)
```

```
MariaDB [mysql]> ALTER TABLE library ADD PRIMARY KEY (Book);
Query OK, 0 rows affected (0.19 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
MariaDB [mysql]> describe library;
```

Field	Type	Null	Key	Default	Extra
Book	varchar(20)	NO	PRI	NULL	
Author	varchar(20)	YES		NULL	
ISBN	varchar(20)	YES		NULL	

```
3 rows in set (0.01 sec)
```

```
MariaDB [mysql]> insert into library values('Internet Programming','Robert','111-232-458');
Query OK, 1 row affected (0.01 sec)
```

```
MariaDB [mysql]> insert into library values('Network Programming','Karamagi','134-869-706');
Query OK, 1 row affected (0.00 sec)
```

```
MariaDB [mysql]> select * from library;
```

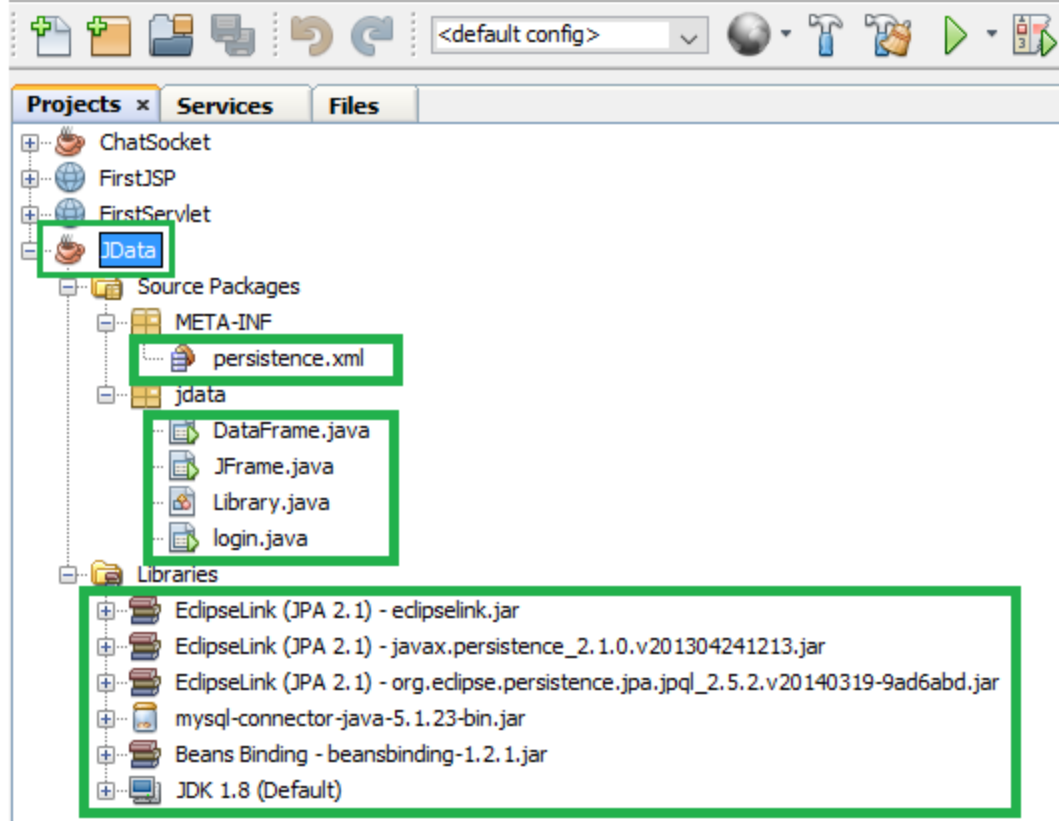
Book	Author	ISBN
Internet Programming	Robert	111-232-458
Network Programming	Karamagi	134-869-706

```
2 rows in set (0.00 sec)
```

```
MariaDB [mysql]> _
```

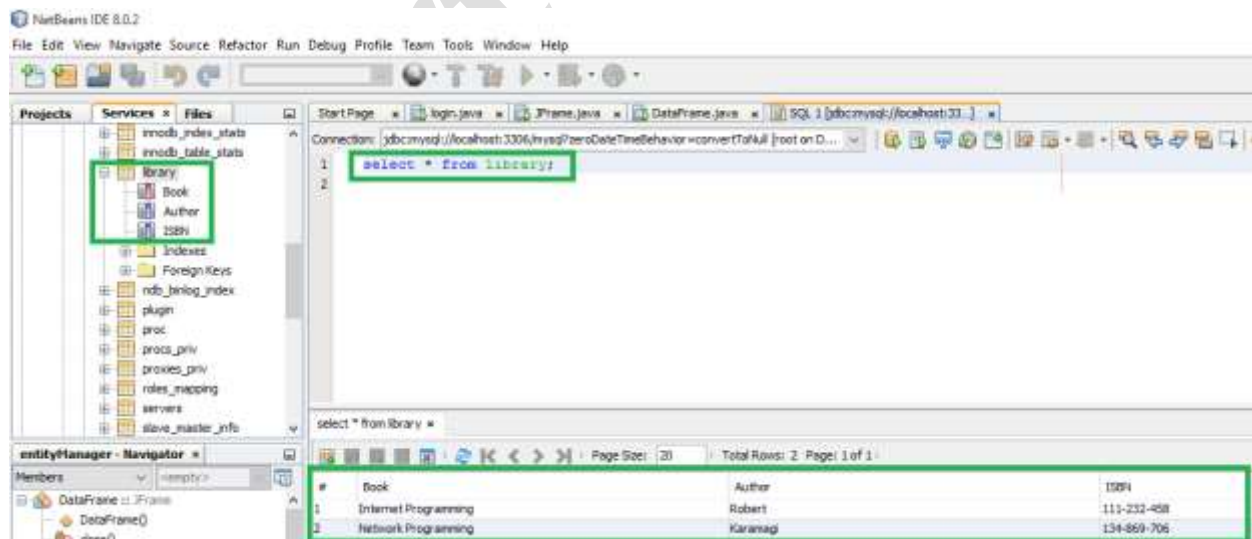
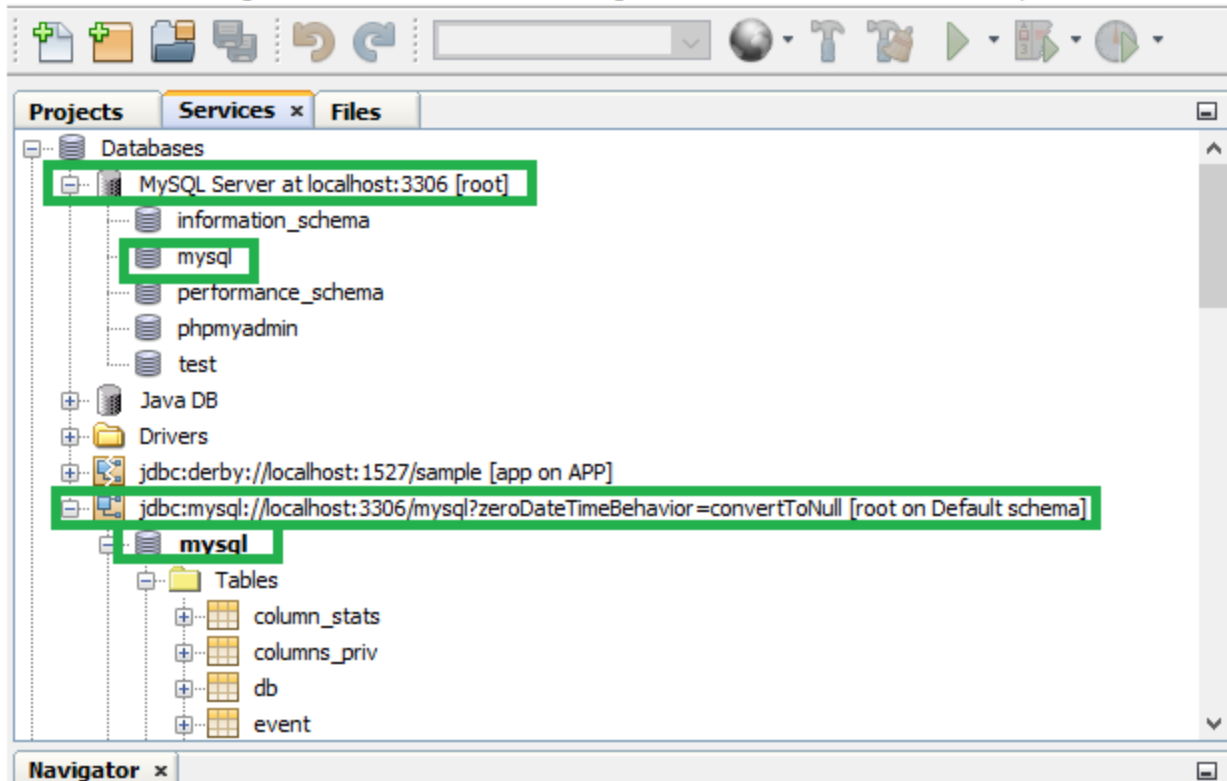
JData - NetBeans IDE 8.0.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help



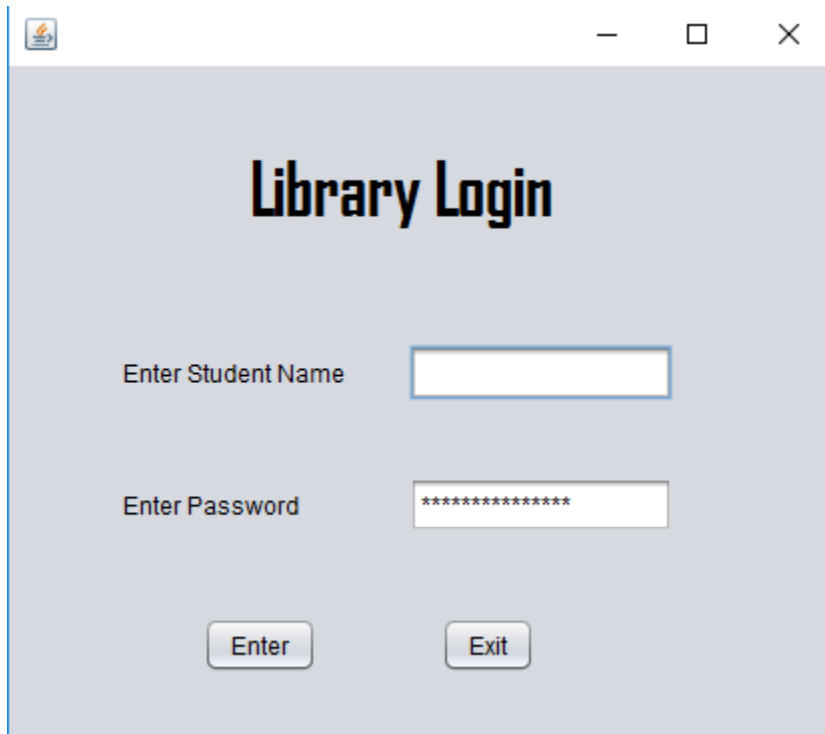
NetBeans IDE 8.0.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help



Open the NetBeans IDE.

Start the login.java file



A screenshot of a 'Library Login' window. The window has a title bar with a small icon on the left and standard minimize, maximize, and close buttons on the right. The main content area has a light gray background. At the top, the text 'Library Login' is displayed in a large, bold, black font. Below this, there are two input fields. The first is labeled 'Enter Student Name' and is an empty text box. The second is labeled 'Enter Password' and contains ten asterisks. At the bottom of the window, there are two buttons: 'Enter' and 'Exit', both with a light gray background and a thin blue border.

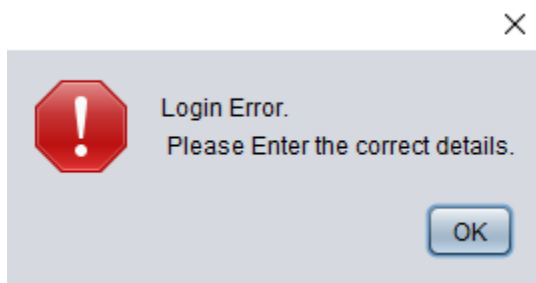
Library Login

Enter Student Name

Enter Password

Enter username as Robert and password 1234 and click Enter

If there is an error



— □ ×

Library Login

Enter Student Name

Robert

Enter Password

Enter

Exit

— □ ×

Book Details

Name

Library Books

Author

Request

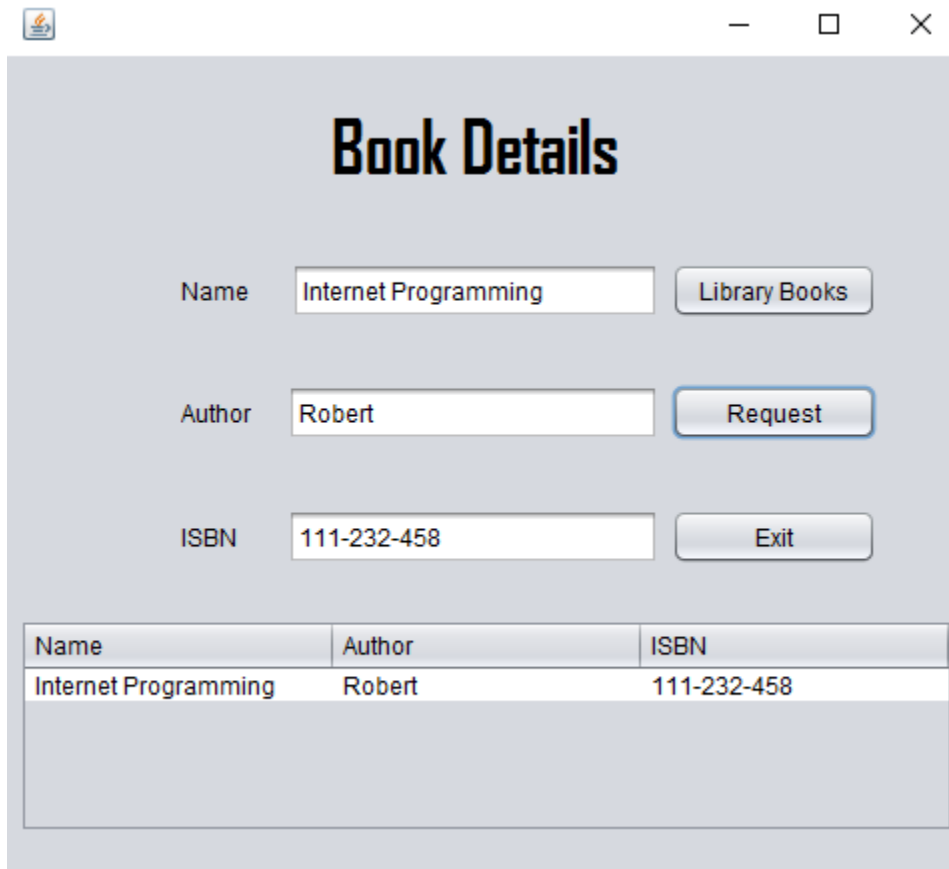
ISBN

Exit

Name	Author	ISBN

Request a book by Entering the details and clicking Request

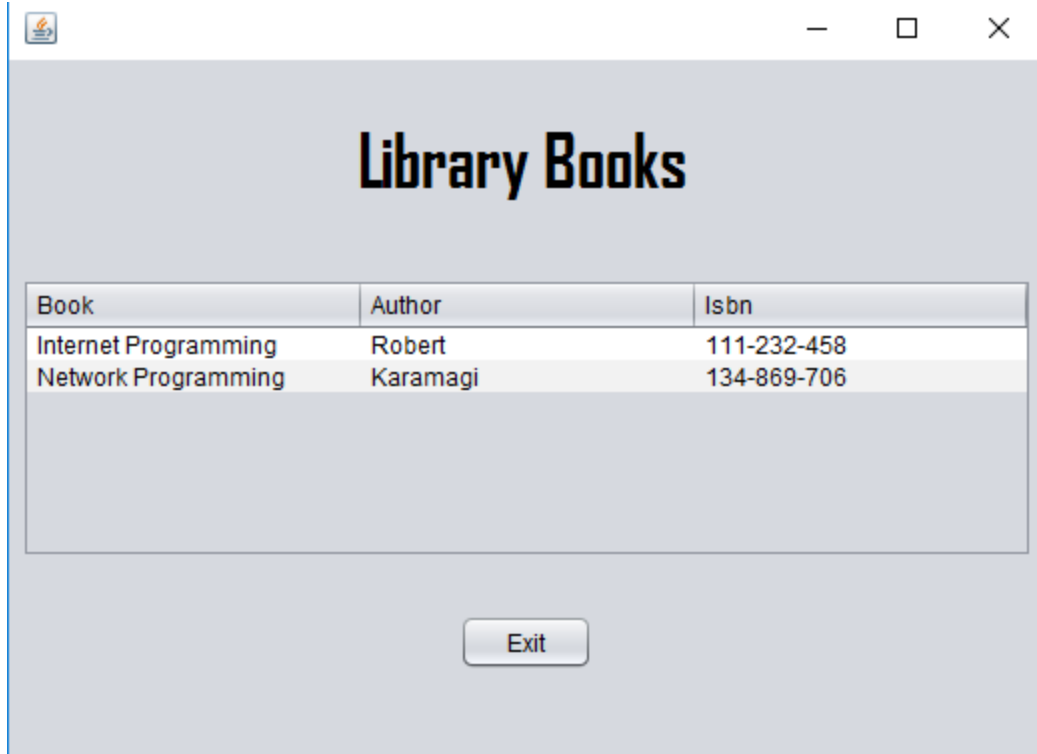
Requests shall populate in a table.



A screenshot of a Java Swing window titled "Book Details". The window has a light gray background and standard window controls (minimize, maximize, close) in the top right corner. It contains three input fields with corresponding labels: "Name" (containing "Internet Programming"), "Author" (containing "Robert"), and "ISBN" (containing "111-232-458"). To the right of the "Name" field is a button labeled "Library Books". To the right of the "Author" field is a button labeled "Request". To the right of the "ISBN" field is a button labeled "Exit". Below the input fields is a table with three columns: "Name", "Author", and "ISBN". The table contains one row of data: "Internet Programming", "Robert", and "111-232-458".

Name	Author	ISBN
Internet Programming	Robert	111-232-458

Observe the mysql database by clicking on library books.



login.java

```
/*  
 * To change this license header, choose License Headers in Project Properties.  
 * To change this template file, choose Tools | Templates  
 * and open the template in the editor.  
 */  
  
package jdata;  
  
import java.awt.Toolkit;  
import java.awt.event.WindowEvent;  
import javax.swing.JOptionPane;  
  
/**  
 *  
 * @author Robert Karamagi
```

```

*/

public class login extends javax.swing.JFrame {

    /**
     * Creates new form login
     */
    public login() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        login = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        unameTF = new javax.swing.JTextField();
        passTF = new javax.swing.JPasswordField();
        enterBtn = new javax.swing.JButton();
        exitBTn = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);

        login.setFont(new java.awt.Font("Agency FB", 1, 36)); // NOI18N
        login.setText("Library Login");

```

```
jLabel2.setText("Enter Student Name");
```

```
jLabel3.setText("Enter Password");
```

```
unameTF.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        unameTFActionPerformed(evt);  
    }  
});
```

```
passTF.setText("jPasswordField1");  
passTF.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        passTFActionPerformed(evt);  
    }  
});
```

```
enterBtn.setText("Enter");  
enterBtn.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        enterBtnActionPerformed(evt);  
    }  
});
```

```
exitBTn.setText("Exit");  
exitBTn.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        exitBTnActionPerformed(evt);  
    }  
});
```

```

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addGap(97, 97, 97)
                    .addComponent(enterBtn)
                    .addGap(62, 62, 62)
                    .addComponent(exitBTn))
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addGap(57, 57, 57)
                    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addComponent(jLabel2)
                        .addComponent(jLabel3))
                    .addGap(32, 32, 32)
                    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
                        .addComponent(unameTF, javax.swing.GroupLayout.DEFAULT_SIZE, 132,
Short.MAX_VALUE)
                        .addComponent(passTF)))
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addGap(121, 121, 121)
                    .addComponent(login)))
            .addContainerGap(81, Short.MAX_VALUE))
        );
jPanel1Layout.setVerticalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addGap(39, 39, 39)
                    .addComponent(login)

```

```

        .addGap(56, 56, 56)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

            .addComponent(jLabel2)

            .addComponent(unameTF, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(38, 38, 38)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

            .addComponent(jLabel3)

            .addComponent(passTF, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(42, 42, 42)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

            .addComponent(enterBtn)

            .addComponent(exitBTn))

        .addContainerGap(34, Short.MAX_VALUE))

    );

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );

    pack();
} // </editor-fold>

```

```
private void unameTFActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
}
```

```
private void passTFActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
}
```

```
private void exitBTnActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
  
    close();  
}
```

```
private void enterBtnActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
  
    String username = unameTF.getText();  
    String password = passTF.getText();  
  
    if (username.contains("Robert") && password.contains("1234"))  
    {  
        unameTF.setText("");  
        passTF.setText("");  
        close();  
        JFrame w = new JFrame();  
        w.setVisible(true);  
    }  
  
    else
```

```

    {
        JOptionPane.showMessageDialog(null, "Login Error.\n Please Enter the correct
details.", "", JOptionPane.ERROR_MESSAGE);

        passTF.setText("");
        unnameTF.setText("");

    }

}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
    * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    }
    catch (ClassNotFoundException ex) {
        java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
    catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
}

```

```
} catch (IllegalAccessException ex) {  
    java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
}  
} catch (javax.swing.UnsupportedLookAndFeelException ex) {  
    java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
}  
}
```

```
//</editor-fold>
```

```
//</editor-fold>
```

```
/* Create and display the form */
```

```
java.awt.EventQueue.invokeLater(new Runnable() {  
    public void run() {  
        new login().setVisible(true);  
    }  
});  
}
```

```
// Variables declaration - do not modify
```

```
private javax.swing.JButton enterBtn;  
private javax.swing.JButton exitBTn;  
private javax.swing.JLabel jLabel2;  
private javax.swing.JLabel jLabel3;  
private javax.swing.JPanel jPanel1;  
private javax.swing.JLabel login;  
private javax.swing.JPasswordField passTF;  
private javax.swing.JTextField unameTF;  
// End of variables declaration
```

```
private void close()
```

```
{
```

```
    WindowEvent winclosing = new WindowEvent(this, WindowEvent.WINDOW_CLOSING);
```

```
    Toolkit.getDefaultToolkit().getSystemEventQueue().postEvent(winclosing);
```

```
}
```

```
}
```

JFrame.java

```
package jdata;
```

```
import java.awt.Toolkit;
```

```
import java.awt.event.WindowEvent;
```

```
import javax.swing.table.DefaultTableModel;
```

```
/*
```

```
 * To change this license header, choose License Headers in Project Properties.
```

```
 * To change this template file, choose Tools | Templates
```

```
 * and open the template in the editor.
```

```
*/
```

```
/**
```

```
 *
```

```
 * @author Robert Karamagi
```

```
 */
```

```
public class JFrame extends javax.swing.JFrame {
```

```
    DefaultTableModel model;
```

```
/**
 * Creates new form JFrame
 */
public JFrame() {
    initComponents();
    model = (DefaultTableModel) table.getModel();
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    radioButton1 = new javax.swing.JRadioButton();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    nameTF = new javax.swing.JTextField();
    authorTF = new javax.swing.JTextField();
    isbnTF = new javax.swing.JTextField();
    jScrollPane2 = new javax.swing.JScrollPane();
```

```
table = new javax.swing.JTable();
borrow = new javax.swing.JButton();
exitBtn = new javax.swing.JButton();
booksBtn = new javax.swing.JButton();

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "Title 1", "Title 2", "Title 3", "Title 4"
    }
));
jScrollPane1.setViewportView(jTable1);

jRadioButton1.setText("jRadioButton1");

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jLabel1.setText("Name");
jLabel2.setText("Author");
jLabel3.setText("ISBN");

jLabel5.setFont(new java.awt.Font("Agency FB", 1, 36)); // NOI18N
jLabel5.setText("    Book Details");

table.setModel(new javax.swing.table.DefaultTableModel(
```

```

new Object [][] {

    },

    new String [] {
        "Name", "Author", "ISBN"
    }
) {
    boolean[] canEdit = new boolean [] {
        false, true, true
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});

jScrollPane2.setViewportViewView(table);

borrow.setText("Request");
borrow.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        borrowActionPerformed(evt);
    }
});

exitBtn.setText("Exit");
exitBtn.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        exitBtnActionPerformed(evt);
    }
});

booksBtn.setText("Library Books");

```

```

booksBtn.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        booksBtnActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(87, 87, 87)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel2)
                .addComponent(jLabel1)
                .addComponent(jLabel3))
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(20, 20, 20)
                    .addComponent(nameTF))
                .addGroup(layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addComponent(isbnTF)
                        .addComponent(authorTF))))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(booksBtn, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(borrow, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(exitBtn, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        )
);

```

```

        .addGap(45, 45, 45))
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup())
    .addContainerGap(127, Short.MAX_VALUE)
    .addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED_SIZE, 229,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(124, 124, 124))
    .addGroup(layout.createSequentialGroup())
    .addContainerGap()
    .addComponent(jScrollPane2)
    .addContainerGap())
);
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup())
    .addGap(23, 23, 23)
    .addComponent(jLabel5)
    .addGap(36, 36, 36)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel1)
        .addComponent(nameTF, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
    .addComponent(booksBtn))
    .addGap(33, 33, 33)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel2)
        .addComponent(authorTF, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
    .addComponent(borrow))
    .addGap(34, 34, 34)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel3)
        .addComponent(isbnTF, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

```

```

        .addComponent(exitBtn))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 27, Short.MAX_VALUE)

        .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 107,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(24, 24, 24))

    );

    pack();
} // </editor-fold>

private void borrowActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

    model.insertRow(model.getRowCount(), new Object[]
        {nameTF.getText(),authorTF.getText(),isbnTF.getText() });
}

private void exitBtnActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    close();
}

private void booksBtnActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    DataFrame df = new DataFrame();
    df.setVisible(true);

}

/**
 * @param args the command line arguments
 */

```

```

public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
    * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {
        java.util.logging.Logger.getLogger(JFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(JFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {
        java.util.logging.Logger.getLogger(JFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
        java.util.logging.Logger.getLogger(JFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
    //</editor-fold>
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new JFrame().setVisible(true);
        }
    });
}

```

```

// Variables declaration - do not modify
private javax.swing.JTextField authorTF;
private javax.swing.JButton booksBtn;
private javax.swing.JButton borrow;
private javax.swing.JButton exitBtn;
private javax.swing.JTextField isbnTF;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel5;
private javax.swing.JRadioButton jRadioButton1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JTable jTable1;
private javax.swing.JTextField nameTF;
private javax.swing.JTable table;
// End of variables declaration

private void close()
{
    WindowEvent winclosing = new WindowEvent(this, WindowEvent.WINDOW_CLOSING);
    Toolkit.getDefaultToolkit().getSystemEventQueue().postEvent(winclosing);
}
}

```

DataFrame.java

```

/*

```

* To change this license header, choose License Headers in Project Properties.

* To change this template file, choose Tools | Templates

* and open the template in the editor.

*/

```
package jdata;
```

```
import java.awt.Toolkit;
```

```
import java.awt.event.WindowEvent;
```

```
import javax.swing.JFrame;
```

```
/**
```

```
 *
```

```
 * @author Robert Karamagi
```

```
 */
```

```
public class DataFrame extends javax.swing.JFrame {
```

```
    /**
```

```
     * Creates new form DataFrame
```

```
    */
```

```
    public DataFrame() {
```

```
        initComponents();
```

```
    }
```

```
    /**
```

```
     * This method is called from within the constructor to initialize the form.
```

```
     * WARNING: Do NOT modify this code. The content of this method is always
```

```
     * regenerated by the Form Editor.
```

```
    */
```

```
    @SuppressWarnings("unchecked")
```

```
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
    private void initComponents() {
```

```
        bindingGroup = new org.jdesktop.beansbinding.BindingGroup();
```

```
entityManager = java.beans.Beans.isDesignTime() ? null :  
javax.persistence.Persistence.createEntityManagerFactory("mysql?zeroDateTimeBehavior=convertToNullPU").crea  
teEntityManager();
```

```
libraryQuery = java.beans.Beans.isDesignTime() ? null : entityManager.createQuery("SELECT l FROM  
Library l");
```

```
libraryList = java.beans.Beans.isDesignTime() ? java.util.Collections.emptyList() :  
libraryQuery.getResultList();
```

```
jScrollPane1 = new javax.swing.JScrollPane();
```

```
Table = new javax.swing.JTable();
```

```
jLabel1 = new javax.swing.JLabel();
```

```
exitBtn = new javax.swing.JButton();
```

```
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
org.jdesktop.swingbinding.JTableBinding jTableBinding =  
org.jdesktop.swingbinding.SwingBindings.createJTableBinding(org.jdesktop.beansbinding.AutoBinding.UpdateStra  
tegy.READ_WRITE, libraryList, Table);
```

```
org.jdesktop.swingbinding.JTableBinding.ColumnBinding columnBinding =  
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${book}"));
```

```
columnBinding.setColumnName("Book");
```

```
columnBinding.setColumnClass(String.class);
```

```
columnBinding =  
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${author}"));
```

```
columnBinding.setColumnName("Author");
```

```
columnBinding.setColumnClass(String.class);
```

```
columnBinding =  
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${isbn}"));
```

```
columnBinding.setColumnName("Isbn");
```

```
columnBinding.setColumnClass(String.class);
```

```
bindingGroup.addBinding(jTableBinding);
```

```
jTableBinding.bind();
```

```
jScrollPane1.setViewportViewView(Table);
```

```

jLabel1.setFont(new java.awt.Font("Agency FB", 1, 36)); // NOI18N
jLabel1.setText("Library Books");

exitBtn.setText("Exit");
exitBtn.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        exitBtnActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE, 506, Short.MAX_VALUE)
            .addContainerGap()
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel1)
                .addGap(179, 179, 179))
            .addGroup(layout.createSequentialGroup()
                .addGap(225, 225, 225)
                .addComponent(exitBtn, javax.swing.GroupLayout.PREFERRED_SIZE, 67,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(28, 28, 28)

```

```
.addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 37, Short.MAX_VALUE)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 140,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(28, 28, 28)

.addComponent(exitBtn)

.addGap(45, 45, 45))

);
```

```
bindingGroup.bind();
```

```
pack();
```

```
// </editor-fold>
```

```
private void exitBtnActionPerformed(java.awt.event.ActionEvent evt) {
```

```
    // TODO add your handling code here:
```

```
    close();
```

```
}
```

```
/**
```

```
 * @param args the command line arguments
```

```
 */
```

```
public static void main(String args[]) {
```

```
    /* Set the Nimbus look and feel */
```

```
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
```

```
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
```

```
    * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
```

```
    */
```

```
    try {
```

```
        for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
```

```
            if ("Nimbus".equals(info.getName())) {
```

```

        javax.swing.UIManager.setLookAndFeel(info.getClassName());

        break;

    }

}

} catch (ClassNotFoundException ex) {

    java.util.logging.Logger.getLogger(DataFrame.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);

} catch (InstantiationException ex) {

    java.util.logging.Logger.getLogger(DataFrame.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);

} catch (IllegalAccessException ex) {

    java.util.logging.Logger.getLogger(DataFrame.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

    java.util.logging.Logger.getLogger(DataFrame.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);

}

//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {

    public void run() {

        new DataFrame().setVisible(true);

    }

});

}

// Variables declaration - do not modify
private javax.swing.JTable Table;

private javax.persistence.EntityManager entityManager;

private javax.swing.JButton exitBtn;

private javax.swing.JLabel jLabel1;

private javax.swing.JScrollPane jScrollPane1;

```

```

private java.util.List<jdata.Library> libraryList;

private javax.persistence.Query libraryQuery;

private org.jdesktop.beansbinding.BindingGroup bindingGroup;

// End of variables declaration


private void close()
{
    WindowEvent winclosing = new WindowEvent(this, WindowEvent.WINDOW_CLOSING);
    Toolkit.getDefaultToolkit().getSystemEventQueue().postEvent(winclosing);

}
}

```

Library.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package jdata;

import java.beans.PropertyChangeListener;
import java.beans.PropertyChangeSupport;
import java.io.Serializable;
import javax.persistence.Basic;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.Table;

```

```
import javax.persistence.Transient;
```

```
/**
```

```
*
```

```
* @author Robert Karamagi
```

```
*/
```

```
@Entity
```

```
@Table(name = "library", catalog = "mysql", schema = "")
```

```
@NamedQueries({
```

```
    @NamedQuery(name = "Library.findAll", query = "SELECT l FROM Library l"),
```

```
    @NamedQuery(name = "Library.findByBook", query = "SELECT l FROM Library l WHERE l.book = :book"),
```

```
    @NamedQuery(name = "Library.findByAuthor", query = "SELECT l FROM Library l WHERE l.author = :author"),
```

```
    @NamedQuery(name = "Library.findByIsbn", query = "SELECT l FROM Library l WHERE l.isbn = :isbn"))})
```

```
public class Library implements Serializable {
```

```
    @Transient
```

```
    private PropertyChangeSupport changeSupport = new PropertyChangeSupport(this);
```

```
    private static final long serialVersionUID = 1L;
```

```
    @Id
```

```
    @Basic(optional = false)
```

```
    @Column(name = "Book")
```

```
    private String book;
```

```
    @Column(name = "Author")
```

```
    private String author;
```

```
    @Column(name = "ISBN")
```

```
    private String isbn;
```

```
    public Library() {
```

```
    }
```

```
    public Library(String book) {
```

```
        this.book = book;
```

```
}
```

```
public String getBook() {  
    return book;  
}
```

```
public void setBook(String book) {  
    String oldBook = this.book;  
    this.book = book;  
    changeSupport.firePropertyChange("book", oldBook, book);  
}
```

```
public String getAuthor() {  
    return author;  
}
```

```
public void setAuthor(String author) {  
    String oldAuthor = this.author;  
    this.author = author;  
    changeSupport.firePropertyChange("author", oldAuthor, author);  
}
```

```
public String getIsbn() {  
    return isbn;  
}
```

```
public void setIsbn(String isbn) {  
    String oldIsbn = this.isbn;  
    this.isbn = isbn;  
    changeSupport.firePropertyChange("isbn", oldIsbn, isbn);  
}
```

@Override

```
public int hashCode() {  
    int hash = 0;  
    hash += (book != null ? book.hashCode() : 0);  
    return hash;  
}
```

@Override

```
public boolean equals(Object object) {  
    // TODO: Warning - this method won't work in the case the id fields are not set  
    if (!(object instanceof Library)) {  
        return false;  
    }  
    Library other = (Library) object;  
    if ((this.book == null && other.book != null) || (this.book != null && !this.book.equals(other.book))) {  
        return false;  
    }  
    return true;  
}
```

@Override

```
public String toString() {  
    return "jdata.Library[ book=" + book + " ]";  
}  
  
public void addPropertyChangeListener(PropertyChangeListener listener) {  
    changeSupport.addPropertyChangeListener(listener);  
}  
  
public void removePropertyChangeListener(PropertyChangeListener listener) {  
    changeSupport.removePropertyChangeListener(listener);  
}
```

```
}
```

persistence.xml

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

<persistence version="1.0" xmlns="http://java.sun.com/xml/ns/persistence"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/persistence
http://java.sun.com/xml/ns/persistence/persistence_1_0.xsd">

    <persistence-unit name="mysql?zeroDateTimeBehavior=convertToNullPU" transaction-
type="RESOURCE_LOCAL">

        <provider>org.eclipse.persistence.jpa.PersistenceProvider</provider>

        <class>jdata.Library</class>

        <properties>

            <property name="javax.persistence.jdbc.url"
value="jdbc:mysql://localhost:3306/mysql?zeroDateTimeBehavior=convertToNull"/>

            <property name="javax.persistence.jdbc.user" value="root"/>

            <property name="javax.persistence.jdbc.driver" value="com.mysql.jdbc.Driver"/>

            <property name="javax.persistence.jdbc.password" value=""/>

        </properties>

    </persistence-unit>

</persistence>
```

SAMPLE WEB APPLICATION DEVELOPMENT IN THE OPEN SOURCE ENVIRONMENT

PROGRAM

index.php

```
<?php
    session_start();

    if (!isset($_SESSION['username'])) {
        $_SESSION['msg'] = "You must log in first";
        header('location: login.php');
    }
    if (isset($_GET['logout'])) {
        session_destroy();
        unset($_SESSION['username']);
        header("location: login.php");
    }
?>
<!DOCTYPE html>
<html>
<head>
    <title>Home</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>

<div class="header">
    <h2>Home Page</h2>
```

```
</div>
<div class="content">
    <!-- notification message -->
    <?php if (isset($_SESSION['success'])) : ?>
    <div class="error success" >
        <h3>
        <?php
            echo $_SESSION['success'];
            unset($_SESSION['success']);
        ?>
        </h3>
    </div>
    <?php endif ?>

    <!-- logged in user information -->
    <?php if (isset($_SESSION['username'])) : ?>
        <p>Welcome <strong><?php echo $_SESSION['username']; ?></strong></p>
        <p> <a href="index.php?logout=1" style="color: red;">logout</a> </p>
    <?php endif ?>
</div>

</body>
</html>
```

login.php

```
<?php include('server.php') ?>
<!DOCTYPE html>
<html>
<head>
    <title>Registration System - PHP and MySQL</title>
```

```
<link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
  <div class="header">
    <h2>Login</h2>
  </div>

  <form method="post" action="login.php">
    <?php include('errors.php'); ?>
    <div class="input-group">
      <label>Username</label>
      <input type="text" name="username" >
    </div>
    <div class="input-group">
      <label>Password</label>
      <input type="password" name="password">
    </div>
    <div class="input-group">
      <button type="submit" class="btn" name="login_user">Login</button>
    </div>
    <p>
      Not yet a member? <a href="register.php">Sign up</a>
    </p>
  </form>
</body>
</html>
```

register.php

```
<?php include('server.php') ?>
<!DOCTYPE html>
```

```
<html>
<head>
  <title>Registration System - PHP and MySQL</title>
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
  <div class="header">
    <h2>Register</h2>
  </div>

  <form method="post" action="register.php">
    <?php include('errors.php'); ?>
    <div class="input-group">
      <label>Username</label>
      <input type="text" name="username" value="<?php echo $username; ?>">
    </div>
    <div class="input-group">
      <label>Email</label>
      <input type="email" name="email" value="<?php echo $email; ?>">
    </div>
    <div class="input-group">
      <label>Password</label>
      <input type="password" name="password_1">
    </div>
    <div class="input-group">
      <label>Confirm password</label>
      <input type="password" name="password_2">
    </div>
    <div class="input-group">
      <button type="submit" class="btn" name="reg_user">Register</button>
    </div>
  </form>
</body>
</html>
```

```
<p>
    Already a member? <a href="login.php">Sign in</a>
</p>
</form>
</body>
</html>
```

server.php

```
<?php
session_start();

// initializing variables
$username = "";
$email    = "";
$errors = array();

// connect to the database
$db = mysqli_connect('localhost', 'root', '', 'registration');

// REGISTER USER
if (isset($_POST['reg_user'])) {
    // receive all input values from the form
    $username = mysqli_real_escape_string($db, $_POST['username']);
    $email = mysqli_real_escape_string($db, $_POST['email']);
    $password_1 = mysqli_real_escape_string($db, $_POST['password_1']);
    $password_2 = mysqli_real_escape_string($db, $_POST['password_2']);

    // form validation: ensure that the form is correctly filled ...
    // by adding (array_push()) corresponding error unto $errors array
    if (empty($username)) { array_push($errors, "Username is required"); }
```

```

if (empty($email)) { array_push($errors, "Email is required"); }
if (empty($password_1)) { array_push($errors, "Password is required"); }
if ($password_1 != $password_2) {
    array_push($errors, "The two passwords do not match");
}

// first check the database to make sure
// a user does not already exist with the same username and/or email
$user_check_query = "SELECT * FROM users WHERE username='$username' OR
email='$email' LIMIT 1";
$result = mysqli_query($db, $user_check_query);
$user = mysqli_fetch_assoc($result);

if ($user) { // if user exists
    if ($user['username'] === $username) {
        array_push($errors, "Username already exists");
    }

    if ($user['email'] === $email) {
        array_push($errors, "email already exists");
    }
}

// Finally, register user if there are no errors in the form
if (count($errors) == 0) {
    $password = md5($password_1);//encrypt the password before saving in the database

    $query = "INSERT INTO users (username, email, password)
        VALUES('$username', '$email', '$password)";
    mysqli_query($db, $query);
    $_SESSION['username'] = $username;
}

```

```
        $_SESSION['success'] = "You are now logged in";
        header('location: index.php');
    }
}

// LOGIN USER
if (isset($_POST['login_user'])) {
    $username = mysqli_real_escape_string($db, $_POST['username']);
    $password = mysqli_real_escape_string($db, $_POST['password']);

    if (empty($username)) {
        array_push($errors, "Username is required");
    }
    if (empty($password)) {
        array_push($errors, "Password is required");
    }

    if (count($errors) == 0) {
        $password = md5($password);
        $query = "SELECT * FROM users WHERE username='$username' AND
password='$password'";
        $results = mysqli_query($db, $query);
        if (mysqli_num_rows($results) == 1) {
            $_SESSION['username'] = $username;
            $_SESSION['success'] = "You are now logged in";
            header('location: index.php');
        } else {
            array_push($errors, "Wrong username/password combination");
        }
    }
}
}
```

?>

errors.php

```
<?php if (count($errors) > 0) : ?>
    <div class="error">
        <?php foreach ($errors as $error) : ?>
            <p><?php echo $error ?></p>
        <?php endforeach ?>
    </div>
<?php endif ?>
```

style.css

```
* {
    margin: 0px;
    padding: 0px;
}
body {
    font-size: 120%;
    background: #F8F8FF;
}

.header {
    width: 30%;
    margin: 50px auto 0px;
    color: white;
    background: #5F9EA0;
    text-align: center;
    border: 1px solid #B0C4DE;
    border-bottom: none;
```

```
border-radius: 10px 10px 0px 0px;
padding: 20px;
}
form, .content {
width: 30%;
margin: 0px auto;
padding: 20px;
border: 1px solid #B0C4DE;
background: white;
border-radius: 0px 0px 10px 10px;
}
.input-group {
margin: 10px 0px 10px 0px;
}
.input-group label {
display: block;
text-align: left;
margin: 3px;
}
.input-group input {
height: 30px;
width: 93%;
padding: 5px 10px;
font-size: 16px;
border-radius: 5px;
border: 1px solid gray;
}
.btn {
padding: 10px;
font-size: 15px;
color: white;
```

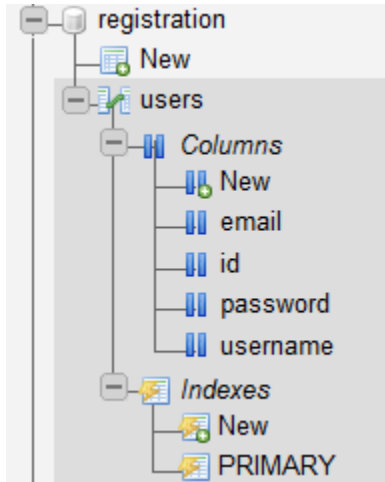
```
background: #5F9EA0;
border: none;
border-radius: 5px;
}
.error {
width: 92%;
margin: 0px auto;
padding: 10px;
border: 1px solid #a94442;
color: #a94442;
background: #f2dede;
border-radius: 5px;
text-align: left;
}
.success {
color: #3c763d;
background: #dff0d8;
border: 1px solid #3c763d;
margin-bottom: 20px;
}
```

MySQL Script

```
CREATE TABLE `users` (
  `id` int(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `username` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `password` varchar(100) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

OUTPUT

phpMyAdmin Database Structure



phpMyAdmin Table

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	username	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
☐ 3	email	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
☐ 4	password	varchar(100)	latin1_swedish_ci		No	None			Change Drop More

Register

Username

Email

Password

Confirm password

Register

Already a member? [Sign in](#)

Home Page

You are now logged in

Welcome **Robert Karamagi**

[logout](#)

phpMyAdmin Table View

+ Options

		id	username	email	password
☐	Edit Copy Delete	1	Robert Karamagi	robertokaramagi@gmail.com	827ccb0eea8a706c4c34a16891f84e7b

Robert Karamagi