



KEYSTONE
School Of Engineering



Department of Computer Engineering

Lab Manual

310257: Web Technology Laboratory

Prepared by,

**Ms. Radhika S. Malpani
Ms. Avneet Kaur
Ms. Supriya Manwar**

TE COMP (2019 Pattern)

Academic Year 2024-25 Sem II

Preface

Web technologies refer to the way computers/devices communicate with each other using markup languages. It is communication across the web, and create, deliver or manage web content using hypertext markup language (HTML). A web page is a web document which is written in HTML (hypertext markup language) It is said to have brought the world into a small village where people and devices can communicate to each other seamlessly. WWW has allowed for the access of information that would have been impossible to find or may have been difficult to find without the www.

The terms Internet and the World Wide Web are synonymous in the minds of many, but they have different meanings. The Internet is a massive network of networks that connects millions of computers worldwide. Computers connected to the Internet can communicate with one another with a number of protocols such as HTTP, SMTP (Simple Mail Transfer Protocol), FTP (File Transfer Protocol), IRC (Internet relay chat), IM (instant messaging), Telnet, and P2P (peer-to-peer). The World Wide Web is a system of interlinked hypertext documents and programs that can be accessed via the Internet primarily by using HTTP. Static pages show the same content each time they are viewed. Dynamic pages have content that can change each time they are accessed. Dynamic pages are typically written in scripting languages such as PHP, Perl, ASP, or JSP. The scripts in the pages run functions on the server that return things like the date and time, and database information. All the information is returned as HTML code, so when the page gets to your browser, all the browser has to do is translate the HTML.

Programme Outcomes: As prescribed by NBA

- 1. Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization for the solution of complex engineering problems.
- 2. Problem analysis:** Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- 3. Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety, and cultural, societal, and environmental considerations.
- 4. Conduct investigations of complex problems:** The problems that cannot be solved by straightforward application of knowledge, theories and techniques applicable to the engineering discipline.
- 5. Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling to complex engineering activities, with an understanding of the limitations.
- 6. The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- 7. Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- 8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- 9. Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- 10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with the society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- 11. Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- 12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Rubrics for Lab Assessment (CAS)

Dimension	Scale				
	1	2	3	4	5
Regularity and punctuality	Did not Perform, submitted in time	Performed and submitted later than scheduled date	Performed on schedule; submitted two weeks late	Performed on schedule; submitted one week late	Performed and submitted as per schedule
Understanding and preparation for Objective	Neither shows any understanding of the objective nor can relate it to theory.	States the objective very vaguely	Can only state the objective but shows poor understanding	Understands objective but cannot place it in context of a theory	Understands objective and can relate it to an appropriate theory topic
Participation in performance and conduction of experiment	Does not participate in experiment	Performs the experiment only with the help from supervisor/others and is confused and untidy.	Performs the experiment with some supervisory help; but forgets some crucial reading and is confused and untidy.	Performs experiment on own without supervisory help; records all readings properly but untidy.	Performs experiment on his/her own without supervisory help; records all readings properly. Keeps the setup clean and tidy.
Post experiment skills	Cannot follow the procedure and do any work	Follows procedure half-heartedly	Follows right procedure; but cannot analyze data and interpret it	Follows right procedure and can analyze data and interpret it	Follows right procedure; can analyze data and interpret it with justification

Curriculum for Third Year of Computer Engineering (2019 Course), Savitribai Phule Pune University

Suggested List of Laboratory Experiments/Assignments (All assignments are compulsory)																		
Sr. No.	Assignment Title																	
1.	<p>Case study: Before coding of the website, planning is important, students should visit different websites (Min. 5) for the different client projects and note down the evaluation results for these websites, either good website or bad website in following format:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="background-color: #90EE90;">Sr. No.</th><th style="background-color: #90EE90;">Website URL</th><th style="background-color: #90EE90;">Purpose of Website</th><th style="background-color: #90EE90;">Things liked in the website</th><th style="background-color: #90EE90;">Things disliked in the website</th><th style="background-color: #90EE90;">Overall evaluation of the website (Good/Bad)</th></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td></td><td></td><td></td></tr> </tbody> </table> <p>From the evaluation, students should learn and conclude different website design issues, which should be considered while developing a website.</p>						Sr. No.	Website URL	Purpose of Website	Things liked in the website	Things disliked in the website	Overall evaluation of the website (Good/Bad)						
Sr. No.	Website URL	Purpose of Website	Things liked in the website	Things disliked in the website	Overall evaluation of the website (Good/Bad)													
2.	<p>Implement a web page index.htm for any client website (e.g., a restaurant website project) using following:</p> <ul style="list-style-type: none"> a. HTML syntax: heading tags, basic tags and attributes, frames, tables, images, lists, links for text and images, forms etc. b. Use of Internal CSS, Inline CSS, External CSS 																	
3.	<p>Design the XML document to store the information of the employees of any business organization and demonstrate the use of:</p> <ul style="list-style-type: none"> a) DTD b) XML Schema <p>And display the content in (e.g., tabular format) by using CSS/XSL.</p>																	
4.	<p>Implement an application in Java Script using following:</p> <ul style="list-style-type: none"> a) Design UI of application using HTML, CSS etc. b) Include Java script validation c) Use of prompt and alert window using Java Script <p>e.g., Design and implement a simple calculator using Java Script for operations like addition, multiplication, subtraction, division, square of number etc.</p> <ul style="list-style-type: none"> a) Design calculator interface like text field for input and output, buttons for numbers and operators etc. b) Validate input values c) Prompt/alerts for invalid values etc. 																	
5.	<p>Implement the sample program demonstrating the use of Servlet.</p> <p>e.g., Create a database table ebookshop (book_id, book_title, book_author, book_price, quantity) using database like Oracle/MySQL etc. and display (use SQL select query) the table content using servlet.</p>																	
6.	<p>Implement the program demonstrating the use of JSP.</p> <p>e.g., Create a database table students_info (stud_id, stud_name, class, division, city) using database like Oracle/MySQL etc. and display (use SQL select query) the table content using JSP.</p>																	
7.	<p>Build a dynamic web application using PHP and MySQL.</p> <ul style="list-style-type: none"> a. Create database tables in MySQL and create connection with PHP. b. Create the add, update, delete and retrieve functions in the PHP web app interacting with MySQL database 																	

Curriculum for Third Year of Computer Engineering (2019 Course), Savitribai Phule Pune University

	8. Design a login page with entries for name, mobile number email id and login button. Use struts and perform following validations <ul style="list-style-type: none"> a. Validation for correct names b. Validation for mobile numbers c. Validation for email id d. Validation if no entered any value e. Re-display for wrongly entered values with message f. Congratulations and welcome page upon successful entries
9.	Design an application using Angular JS. e.g., Design registration (first name, last name, username, password) and login page using Angular JS.
10.	Design and implement a business interface with necessary business logic for any web application using EJB. e.g., Design and implement the web application logic for deposit and withdraw amount transactions using EJB.
11.	Mini Project: Design and implement a dynamic web application for any business functionality by using web development technologies that you have learnt in the above given assignments.

@The CO-PO Mapping Matrix

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	1	3	1	-	1	1	-	-	1	-	-
CO2	2	2	-	2	1	-	-	-	1	-	-	-
CO3	2	-	3	-	-	1	-	-	-	1	1	-
CO4	1	2	2	1	2	1	1	-	-	-	-	1

CO PO and PSO Mapping

A. Course Outcome

Course Outcome	Statement <i>At the end of the course a student will be able to (write/install/solve/apply)</i>	Bloom's Taxonomy level
310257.1	Understand the importance of website planning and website design issues.	BTL-1
310257.2	Apply the client side and server side technologies for web application development.	BTL-3
310257.3	Analyze the web technology languages, frameworks and services.	BTL-4
310257.4	Create three tier web based applications.	BTL-6

B. CO-PO mapping

Course Outcome	Program outcomes												PSO	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
310257.1	2	1	3	1	1	-	-	-	-	-	-	1	-	2
310257.2	2	2	1	2	2	-	-	-	-	-	-	2	-	3
310257.3	2	1	2	1	2	-	-	-	-	-	-	2	-	3
310257.4	1	2	2	1	2	2	3	2	1	2	1	2	-	3

INDEX

Sr. No.	Assignment Title												
1.	<p>Case study: Before coding of the website, planning is important, students should visit different websites (Min. 5) for the different client projects and note down the evaluation results for these websites, either good website or bad website in following format:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #d3d3d3;"> <th style="width: 10%;">Sr. No.</th> <th style="width: 20%;">Website URL</th> <th style="width: 20%;">Purpose of Website</th> <th style="width: 20%;">Things liked in the website</th> <th style="width: 20%;">Things disliked in the website</th> <th style="width: 20%;">Overall evaluation of the website (Good/Bad)</th> </tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td></td><td></td><td></td></tr> </tbody> </table> <p>From the evaluation, students should learn and conclude different website design issues, which should be considered while developing a website.</p>	Sr. No.	Website URL	Purpose of Website	Things liked in the website	Things disliked in the website	Overall evaluation of the website (Good/Bad)						
Sr. No.	Website URL	Purpose of Website	Things liked in the website	Things disliked in the website	Overall evaluation of the website (Good/Bad)								
2.	<p>Implement a web page index.htm for any client website (e.g., a restaurant website project) using following:</p> <ol style="list-style-type: none"> HTML syntax: heading tags, basic tags and attributes, frames, tables, images, lists, links for text and images, forms etc. Use of Internal CSS, Inline CSS, External CSS Bootstrap 												
3.	<p>Design the XML document to store the information of the employees of any business organization and demonstrate the use of:</p> <ol style="list-style-type: none"> DTD XML Schema <p>And display the content in (e.g., tabular format) by using CSS/XSL.</p>												

4.	<p>Implement an application in Java Script using following:</p> <ul style="list-style-type: none"> a) Design UI of application using HTML, CSS etc. b) Include Java script validation c) Use of prompt and alert window using Java Script <p>e.g., Design and implement a simple calculator using Java Script for operations like addition, multiplication, subtraction, division, square of number etc.</p> <ul style="list-style-type: none"> a) Design calculator interface like text field for input and output, buttons for numbers and operators etc. b) Validate input values c) Prompt/alerts for invalid values etc.
5.	<p>Implement the sample program demonstrating the use of Servlet.</p> <p>e.g., Create a database table ebookshop (book_id, book_title, book_author, book_price, quantity) using database like Oracle/MySQL etc. and display (use SQL select query) the table content using servlet.</p>
6.	<p>Implement the program demonstrating the use of JSP.</p> <p>e.g., Create a database table students_info (stud_id, stud_name, class, division, city) using database like Oracle/MySQL etc. and display (use SQL select query) the table content using JSP.</p>
7.	<p>Build a dynamic web application using PHP and MySQL.</p> <ul style="list-style-type: none"> a. Create database tables in MySQL and create connection with PHP. b. Create the add, update, delete and retrieve functions in the PHP web app interacting with MySQL database
8.	<p>Design a login page with entries for name, mobile number, email id and login button. Use struts and perform following validations</p> <ul style="list-style-type: none"> a. Validation for correct names b. Validation for mobile numbers c. Validation for email id d. Validation if no entered any value e. Re-display for wrongly entered values with message f. Congratulations and welcome page upon successful entries
9.	<p>Design an application using Angular JS.</p> <p>e.g., Design registration (first name, last name, username, password) and login page using Angular JS.</p>
10.	<p>Design and implement a business interface with necessary business logic for any web application using EJB.</p> <p>e.g., Design and implement the web application logic for deposit and withdraw amount transactions using EJB.</p>

11.	Mini Project: Design and implement a dynamic web application for any business functionality by using web development technologies that you have learnt in the above given assignments.
12	Content Beyond Syllabus : CMS – Wordpress

Assignment No: 1**Title : Study of Website****Course Objective:**

To understand the Website flow and problems.

Course Outcome: **Understand** the importance of website planning and website design issues

Problem Statement:**Case study:**

Before coding of the website, planning is important, students should visit different websites (Min. 5) for the different client projects and note down the evaluation results for these websites, either good website or bad website in following format:

Sr. No.	Website URL	Purpose of Website	Things liked in the website	Things disliked in the website	Overall evaluation of the website (Good/ Bad)

From the evaluation, students should learn and conclude different website design issues, which should be considered while developing a website.

Theory:**Website development Steps:****Step 1. Gathering Information: Purpose, Main Goals, and Target Audience**

The most important task at this point is to get a clear understanding of your future website purposes, the main goals you wish to get, and the target audience you want to attract to your site. News portal differs from the entertainment websites, and online resources for teenagers look different than sites for adults. Different types of websites provide visitors with different functionality which means that different technologies should be used according to the purposes.

Step 2. Planning: Sitemap and Wireframe Creation

At this stage of the website development cycle, the developer creates the data that can give a customer an opportunity to judge how the entire site will look like.

Step 3. Design: Page Layouts, Review, and Approval Cycle

During the design phase, your website takes shape. All the visual content, such as images, photos, and videos are created at this step. Once again, all the info that was gathered through the first phase is crucial. The customer and target audience must be kept in mind while you work on a design. Website layout is the result of designer's work. It can be a graphic sketch or an actual graphic design. The primary function of the layout is to represent the information structure, visualize the content, and demonstrate the basic functional. Layouts contain colors, logos, images and can give a general understanding of the future product.

After that, the customer can review the layout and send you his feedback. If the client is not sure about some aspects of your design, you should change the layout and send it back to him. This cycle should be repeated until the customer is completely satisfied.

Step 4. Content Writing and Assembly

Content writing and compiling usually overlaps with other stages of website creation, and its role can't be underestimated. At this step it is necessary to put in writing the very essence you'd like to communicate to the audience of your website, and add calls-to-action. Content writing involves also creation of catching headlines, text editing, writing new text, compiling the existing text, etc., which takes time and effort. As a rule, the client undertakes to provide website content ready to migrate to the site. It is better when all website content is provided before or during website coding.

Step 5. Coding

At this step, you can finally start creating the website itself. Graphic elements that have been designed during the previous stages should be used to create an actual website. Usually, the home page is created first, and then all sub-pages are added, according to the website hierarchy that was previously created in the form of a sitemap. Frameworks and CMS should be implemented to make sure that the server can handle the installation and set-up smoothly.

All static web page elements that were designed during the mock-up and layout creation should be created and tested. Then, special features and interactivity should be added. A deep understanding of every website development technology that you're going to use is crucial at this phase.

When you use CMS for site creation, you can also install CMS plugins at this step if there's a need.

The other important step is SEO (Search Engine Optimization). SEO is the optimization of website elements (e.g., title, description, keyword) that can help your site achieve higher rankings in the search engines. And, once again, valid code is pretty important for SEO.

Step 6. Testing, Review and Launch

Testing is probably the most routine part of a process. Every single link should be tested to make sure that there are no broken ones among them. You should check every form, every script, run a spell-checking software to find possible typos. Use code validators to check if your code follows the current web standards. Valid code is necessary, for example, if cross-browser compatibility is important for you.

After you check and re-check your website, it's time to upload it to a server. An FTP (File Transfer Protocol) software is used for that purpose. After you deploy the files, you should run yet another, final test to be sure that all your files have been installed correctly.

Step 7. Maintenance: Opinion Monitoring and Regular Updating

What's important to remember is that a website is more a service than a product. It's not enough to "deliver" a website to a user. You should also make sure that everything works fine, and everybody is satisfied and always be prepared to make changes in another case. Feedback system added to the site will allow you to detect possible problems the end-users face. The highest priority task in this case is to fix the problem as fast as you can. If you won't, you may find one day that your users prefer to use another website rather than put up with the inconvenience.

The other important thing is keeping your website up to date. If you use a CMS, regular updates will prevent you from bugs and decrease security risks.

Website Design Issues

While creating website some major issues of website designing must be taken into consideration, such as

Planning a website:

Planning plays an important role in the successful completion of any work. Creation of a website takes time and resources, so the plan must proceed sequentially and properly. This plan contain the following things

- Defining the purpose of website
- Knowing the audience of website
- Organizing contents of website
- Publishing of website

Defining the Purpose of Website

This step starts by defining the goals and objectives of the website. Goals are the outcomes that designers want from a website to achieve in a particular time period. Objectives are the methods that are used to achieve the goals of the website.

Knowing the Audience of Website

Audience refers to the site visitors or customers who visit the site for the purpose of viewing the products, information or buying them. Therefore assessment of the audience according to their demands, requirements and expectations is a very important thing. So the content of the website must satisfy the needs and expectations of the customer or audience.

Organizing Contents of Website

Website is made of multiple web pages that may be a combination of text, images, audio, video, animations and other multimedia elements. Generally a website has a home page and other more pages. When a visitor visits a website, the home page appears first. The contents of a website should be arranged in a logical manner from home page to end page. The information on the web page must flow from basic to more detailed content.

Publishing of Website

Publishing a website is nothing but making it available on the Internet for everyone. To do this following thing need to perform

- ***Domain registration and hosting*** – Register a domain name from the concerned authority and purchase a certain amount of hosting space having some bandwidth.
- ***Upload website on server*** – A website can be uploaded on a remote server either by using an FTP Program or using a hosting control panel.
- ***Viewing website*** – Check websites in different browsers and on different screens. If it is displayed properly on each platform, then only consider the website is working properly.

CONCLUSION/ANALYSIS

Hence, we have learned parameters which make websites good and bad.

FAQ:

Q1. List and discuss the different design issues in Web Development CO1

BTL1

Browsers.: Although Internet Explorer is still the dominant browser, Firefox and Safari are gaining in popularity. Make sure your site works in at least the 4 browsers most people use (Internet Explorer, Safari, Firefox for Windows and Firefox for the Mac).

- Operating systems. Windows XP is the most popular operating system. Make sure your site works well for Windows users.
- Connection speeds. Over 90% of home users in the US are now accessing the Internet using broadband methods (DSL/Cable modem). Although only 9.5% of active Internet users are using a

narrowband method (modems), many people use handheld devices such as the iPhone (which uses the Edge network – a narrowband connection).

- User screen sizes. 88% of users are using a display with 1024x768 pixels. Sites using a fixed-pixel length content box of 800 pixels or less risk alienating users who have larger monitors.

2) Define the terms:**BTL1****i) Website:**

A website is a collection of related web pages, including multimedia content, typically identified with a common domain name, and published on at least one web server. A website may be accessible via a public Internet Protocol (IP) network, such as the Internet, or a private local area network (LAN), by referencing a uniform resource locator (URL) that identifies the site.

ii) Web Page

A web page (also written as webpage) is a document that is suitable for the World Wide Web and web browsers. A web browser displays a web page on a monitor or mobile device.

iii) Web Server

Web server refers to server software, or hardware dedicated to running said software, that can serve contents to the World Wide Web. A web server processes incoming network requests over the HTTP protocol (and several other related protocols).

iv) URL

A URL (Uniform Resource Locator), as the name suggests, provides a way to locate a resource on the web, the hypertext system that operates over the internet.

v) Home Page

A home page is a webpage that serves as the starting point of a website. It is the default webpage that loads when you visit a web address that only contains a domain name. For example, visiting <http://techterms.com> will display the Tech Terms home page.

Assignment No. : 2

TITLE

Implement a web page index.htm for any client website (e.g., a restaurant website project) using following:

- a. **HTML** syntax: heading tags, basic tags and attributes, frames, tables, images, lists, links for text and images, forms etc.
- b. Use of Internal CSS, Inline CSS, External CSS
- c. **Bootstrap**

OBJECTIVES

1. Understand about basic concepts of html
2. Understand the basic concepts of CSS and Bootstrap

PROBLEM STATEMENT

Implement a web page index.htm for any client website (e.g., a restaurant website project) using following:

- a. HTML syntax: heading tags, basic tags and attributes, frames, tables, images, lists, links for text and images, forms etc.
- b. Use of Internal CSS, Inline CSS, External CSS.

OUTCOME

Students will be able to,

1. Design a static webpage using HTML.
2. Apply CSS to HTML pages.

SOFTWARE & HARDWARE REQUIREMENTS

Software:Notepad,Browser.

THEORY-CONCEPT

HTML: HTML is the standard markup language for creating Web pages.

- HTML stands for Hyper Text Markup Language
- HTML describes the structure of Web pages using markup
- HTML elements are the building blocks of HTML pages
- HTML elements are represented by tags
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on
- Browsers do not display the HTML tags, but use them to render the content of the page

- HTML Versions:

HTML	1991
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML 5	2014

Table.1: HTML Versions

CSS:

CSS stands for Cascading Style Sheet. It is nothing, but design language intended to simplify the process of making web pages presentable. CSS handles the feel and look part of a web page. By using CSS, one can control the color of text, style of fonts, spacing between paragraphs, layout designs.

CSS is easy to learn, easy to understand and it provides powerful control on presentation of an HTML document.

Advantages of CSS:

It saves the time, Pages load faster, Easy maintenance, Superior styles to HTML, Multiple Device Compatibility, Global web standards, Offline Browsing, Platform Independence.

CSS3 Modules:

CSS3 Modules are having old CSS specifications as well as extension features.

- Box Model
- Selectors
- Background

- Border
- Image Values and Replaced Content
- Text Effects
- 2D/3D Transformations
- Multiple Column Layout
- User Interface

THEORY

- The <!DOCTYPE html> declaration defines this document to be HTML5
- The <html> element is the root element of an HTML page
- The <head> element contains meta information about the document
- The <title> element specifies a title for the document
- The <body> element contains the visible page content
- The <h1> element defines a large heading
- The <p> element defines a paragraph
- HTML tags are element names surrounded by angle brackets: <tagname>content goes here...</tagname>

CSS can be added to HTML elements in 3 ways:

- Inline - by using the style attribute in HTML elements. An inline CSS is used to apply a unique style to a single HTML element.

Ex. <h1 style="color:blue;">This is a Blue Heading</h1>

- Internal - by using a <style> element in the <head> section. An internal CSS is used to define a style for a single HTML page. An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

Example:<style>

```
body {background-color: powderblue;}
```

```
h1 {color: blue;}  
p {color: red;}  
</style>
```

- External - by using an external CSS file. An external style sheet is used to define the style for many HTML pages. With an external style sheet, you can change the look of an entire web site, by changing one file! To use an external style sheet, add a link to it in the <head> section of the HTML page.

Example: <link rel="stylesheet" href="styles.css">

- Use the HTML <head> element to store <style> and <link> elements
- Use the CSS **color** property for text colors
- Use the CSS **font-family** property for text fonts
- Use the CSS **font-size** property for text sizes
- Use the CSS **border** property for borders
- Use the CSS **padding** property for space inside the border
- Use the CSS **margin** property for space outside the border

Bootstrap :

Bootstrap is a free front-end framework for faster and easier web development.

Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins.

Bootstrap also gives you the ability to easily create responsive designs.

What is Responsive Web Design?

Responsive web design is about creating web sites which automatically adjust themselves to look good on all devices, from small phones to large desktops.

Bootstrap 5 CDN

```
<!-- Latest compiled and minified CSS -->
```

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

<!-- Latest compiled JavaScript -->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
```

Create Your First Web Page With Bootstrap 5

1. Add the HTML5 doctype

Bootstrap 5 uses HTML elements and CSS properties that require the HTML5 doctype.

Always include the HTML5 doctype at the beginning of the page, along with the lang attribute and the correct title and character set:

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <title>Bootstrap 5 Example</title>

    <meta charset="utf-8">

  </head>

</html>
```

2. Bootstrap 5 is mobile-first

Bootstrap 5 is designed to be responsive to mobile devices. Mobile-first styles are part of the core framework.

To ensure proper rendering and touch zooming, add the following `<meta>` tag inside the `<head>` element:

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

The `width=device-width` part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).

The initial-scale=1 part sets the initial zoom level when the page is first loaded by the browser.

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>

<div class="container p-5 my-5 border">
  <h1>My First Bootstrap Page</h1>
  <p>This container has a border and some extra padding and margins.</p>
</div>

<div class="container p-5 my-5 bg-dark text-white">
  <h1>My First Bootstrap Page</h1>
  <p>This container has a dark background color and a white text, and some extra padding and margins.</p>
</div>

<div class="container p-5 my-5 bg-primary text-white">
  <h1>My First Bootstrap Page</h1>
  <p>This container has a blue background color and a white text, and some extra padding and margins.</p>
</div>

</body>
</html>
```

DESIGN / EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Write the HTML code in the notepad and save with the .html extension.
2. Write the CSS code in the notepad and save with the .css extension.
3. Import CSS file in HTML page.
4. Open HTML page in the browser.

TEST CASES

Manual testing is used to check whether CSS gets applied or not.

CONCLUSION/ANALYSIS

Hence, we have designed static web pages using HTML and CSS.

FAQS

Q1. Distinguish between HTML and HTML5

BTL2

HTML	HTML 5
Doctype declaration in Html is too longer <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">	DOCTYPE declaration in Html5 is very simple "<!DOCTYPE html>"
character encoding in Html is also longer <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">	character encoding (charset) declaration is also very simple <meta charset="UTF-8">
Audio and Video are not part of HTML4	Audio and Videos are integral part of HTML5 e.g. <audio> and <video> tags.
Vector Graphics is possible with the help of technologies such as VML, Silverlight, Flash etc	Vector graphics is integral part of HTML5 e.g. SVG and canvas
It is almost impossible to get true GeoLocation of user browsing any website especially if it comes to mobile devices.	JS GeoLocation API in HTML5 helps identify location of user browsing any website (provided user allows it)
Html5 use cookies.	It provides local storage in place of cookies.
Not possible to draw shapes like circle, rectangle, triangle.	Using Html5 you can draw shapes like circle, rectangle, triangle.
Does not allow JavaScript to run in browser. JS runs in same thread as browser interface.	Allows JavaScript to run in background. This is possible due to JS Web worker API in HTML5
Works with all old browsers	Supported by all new browser.

Q2. Explain box model of CSS.

BTL2

Ans: All HTML elements can be considered as boxes. In CSS, the term "box model" is used for design and layout of web pages.

The CSS box model is essentially a box that wraps around every HTML element.

It consists of: margins, borders, padding, and the actual content.

The image below illustrates the box model:

Content - The content of the box, where text and images appear

Padding - Clears an area around the content. The padding is transparent

Border - A border that goes around the padding and content

Margin - Clears an area outside the border. The margin is transparent

The box model allows us to add a border around elements, and to define space between elements.

Example:

```
<!DOCTYPE html>
```

```
<html>
```

```

<head>
<style>
div {
    background-color: lightgrey;
    width: 300px;
    border: 25px solid green;
    padding: 25px;
    margin: 25px;
}
</style>
</head>
<body>
<h2>Demonstrating the Box Model</h2>
<p>The CSS box model</p>
<div>This text is the actual content of the box. </div>
</body>
</html>

```

Q3. Define Image Map with example.**BTL1**

HTML `<map>` Tag:

The `<map>` tag is used to define a client-side image-map. An image-map is an image with clickable areas.

The required name attribute of the `<map>` element is associated with the ``'s `usemap` attribute and creates a relationship between the image and the map.

The `<map>` element contains a number of `<area>` elements that defines the clickable areas in the image map.

```

<!DOCTYPE html>
<html>
<body>
<p>Click on the sun or on one of the planets to watch it closer:</p>

<map name="planetmap">
    <area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm">
    <area shape="circle" coords="90,58,3" alt="Mercury" href="mercur.htm">
    <area shape="circle" coords="124,58,8" alt="Venus" href="venus.htm">
</map>
</body>
</html>

```

Q4.List the types of CSS

BTL1

1. Inline CSS
2. Internal CSS
3. External CSS

Assignment No. : 3

TITLE

Title: XML and CSS

PROBLEM STATEMENT

Design the XML document to store the information of the employees of any business organization and demonstrate the use of:

- a) DTD
- b) XML Schema

And display the content in (e.g., tabular format) by using CSS/XSL.

OUTCOMES

Students will be able to,

1. Design a static webpage using XML.
2. Apply CSS to XML pages.

SOFTWARE & HARDWARE REQUIREMENTS

Software: Notepad, Any Browser

THEORY-CONCEPT

XML stands for Extensible Markup Language. It is nothing but the text-based markup language which is derived from Standard Generalized Markup Language(SGML).

XML tags identify the data and are used to store and organize the data, rather than specifying how to display it like HTML tags, which are used to display the data. XML is not going to replace HTML in the near future, but it introduces new possibilities by adopting many successful features of HTML.

There are three important characteristics of XML that make it useful in a variety of systems and **solutions** –

- XML is extensible – **XML allows you to create your own self-descriptive tags**, or language, that suits your application.

- XML carries the data, does not present it – **XML allows you to store the data irrespective of how it will be presented.**

XML is a public standard – **XML was developed by an organization called the World Wide Web Consortium (W3C)** and is available as an open standard.

TECHNOLOGY/TOOL

The XML document have an XML declaration, but it is optional, and it is written as–

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

Where version is nothing but the version of an XML document and UTF specifies the character-encoding used in the document.

Each XML-element needs to be closed either with start or with end elements as shown below –

```
<element>.....</element>
```

An XML document can have only one root element.

```
<root>
```

```
<x>...</x>
```

```
<y>...</y>
```

```
</root>
```

XML Attributes:

Using a name/value pair, an attribute specifies a single property for an element. An XML-element can have one or more attributes. For example –

```
<a href = "http://www.google.com/">XMLTutorial</a>
```

Here href is the attribute name and http://www.google.com/ is attribute value.

DESIGN/EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Write the XML code in the notepad and save with .xml extension.
2. Write the CSS code in the notepad and save with the .css extension.
3. Import CSS file in XML page.

4. Open XML page in the browser.

TEST CASES

Manual testing is used to check whether CSS gets applied or not.

CONCLUSION/ANALYSIS

Hence, we have designed static web pages using XML and CSS

FAQ:

1. WHAT ARE DTDS? EXPLAIN HOW DO THEY WORK

BTL2

A DTD is a Document Type Definition.

A DTD defines the structure and the legal elements and attributes of an XML document.

Why Use a DTD?

With a DTD, independent groups of people can agree on a standard DTD for interchanging data.

An application can use a DTD to verify that XML data is valid.

An Internal DTD Declaration

If the DTD is declared inside the XML file, it must be wrapped inside the `<!DOCTYPE>` definition:

1. XML document with an internal DTD

```
<?xml version="1.0"?>
<!DOCTYPE note [
  <!ELEMENT note (to,from,heading,body)>
  <!ELEMENT to (#PCDATA)>
  <!ELEMENT from (#PCDATA)>
  <!ELEMENT heading (#PCDATA)>
  <!ELEMENT body (#PCDATA)>
]>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend</body>
</note>
```

The DTD above is interpreted like this:

1. **!DOCTYPE note** defines that the root element of this document is note
2. **!ELEMENT note** defines that the note element must contain four elements: "to,from,heading,body"
3. **!ELEMENT to** defines the to element to be of type "#PCDATA"

4. **!ELEMENT from** defines the from element to be of type "#PCDATA"
5. **!ELEMENT heading** defines the heading element to be of type "#PCDATA"
6. **!ELEMENT body** defines the body element to be of type "#PCDATA"

An External DTD Declaration

If the DTD is declared in an external file, the <!DOCTYPE> definition must contain a reference to the DTD file:

2. XML document with a reference to an external DTD

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

<!DOCTYPE note SYSTEM "note.dtd">

<note>

<to>Tove</to>

<from>Jani</from>

<heading>Reminder</heading>

<body>Don't forget me this weekend!</body>

</note>
```

Q2. Differentiate between HTML and XML

BTL4

HTML	XML
HTML is an abbreviation for HyperText Markup Language.	XML stands for eXtensible Markup Language.
HTML was designed to display data with focus on how data looks.	XML was designed to be a software and hardware independent tool used to transport and store data, with focus on what data is.
HTML is a markup language itself.	XML provides a framework for defining markup languages.
HTML is a presentation language.	XML is neither a programming language nor a presentation language.
HTML is case insensitive.	XML is case sensitive.
HTML is used for designing a web-page to be rendered on the client side.	XML is used basically to transport data between the application and the database.
HTML has its own predefined tags.	While what makes XML flexible is that custom tags can be defined and the tags are invented by the author of the XML document.

HTML is not strict if the user does not use the closing tags.	XML makes it mandatory for the user to close each tag that has been used.
HTML does not preserve white space.	XML preserves white space.
HTML is about displaying data,hence static.	XML is about carrying information,hence dynamic.

Q3. Why Use CSS?**BTL1**

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

HTML was NEVER intended to contain tags for formatting a web page!

HTML was created to **describe the content** of a web page, like:

```
<h1>This is a heading</h1>
```

```
<p>This is a paragraph.</p>
```

When tags like ``, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.

Assignment No. - 4

TITLE

HTML, Java Script

OBJECTIVES

1. Understand the basic concepts of JavaScript.
2. Use JavaScript for validation of data.

PROBLEM STATEMENT

Implement an application in Java Script using following:

- a) Design UI of application using HTML, CSS etc.
- b) Include Java script validation
- c) Use of prompt and alert window using Java Script

e.g., Design and implement a simple calculator using Java Script for operations like addition, multiplication, subtraction, division, square of number etc.

- d) Design calculator interface like text field for input and output, buttons for numbers and operators etc.
- e) Validate input values
- f) Prompt/alerts for invalid values etc.

OUTCOMES

Students will be able to,

1. Design a static webpage using HTML.
2. Apply JavaScript to HTML pages for validation of data.

SOFTWARE & HARDWARE REQUIREMENTS

Software's: Notepad, Any Browser

THEORY-CONCEPT

JavaScript is a programming language of HTML as well as web. It is preferred for creating network-centric applications. It is integrated and complementary with Java. As JavaScript is integrated with HTML it is very easy to implement. It is open as well as cross-platform.

Advantages of using JavaScript are –

- It requires less server interaction
- Immediate feedback to the visitors
- Increased interactivity
- Richer interfaces

Validation:

When a client enters all the necessary data and press the submit button, form validation is done at server side. If data entered by a client is incorrect or missing, the server needs to send all data back to the client and request for resubmission of the form with correct information. This is really a lengthy process which puts a lot of load(burden) on the server.

So, JavaScript provides a way to validate form's data on the client's side itself before sending it to the web server. Form validation performs two functions-

- Basic Validation – First of all the form must be checked to make sure all the mandatory fields are filled in. It would require just a loop through each field in the form and check for the data.
- Data Format Validation – **Secondly, the data that is entered must be checked for correct** format and its value. The code must include appropriate logic to test correctness of data.

TECHNOLOGY/TOOL

JavaScript can be implemented using JavaScript statements that are placed within the <script>.

You can place the <script> tags, containing your JavaScript, anywhere within your web page, but it is normally recommended that you should keep it within the <head> tags.

The script tag takes two important attributes:

- **Language** – This attribute specifies what scripting language you are using. Typically, its value will be JavaScript. Although recent versions of HTML (and XHTML, its successor) have phased out the use of this attribute.
- **Type** – This attribute is what is now recommended to indicate the scripting language in use and its value should be set to "text/javascript".

DESIGN/EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Write an HTML code in notepad and save with .html extension.
2. Write the function for validation of email id and phone no and enclosed this function in script.
3. Call this function on 'onClick' event of the submit button.
4. Open HTML page in the browser.

TEST CASES

Manual testing is used to check following validations

- All the fields like Full name, Class, Department, Address, Phone number and email id are mandatory fields.
- Phone no. should be numbers only.
- Email id should be in proper format like abc@abc.com

CONCLUSION/ANALYSIS

Hence, we validated the data using JavaScript.

FAQ**Q1. List and brief any five functions in JavaScript. BTL1**

SR. No.	Method	Description
1	charAt()	Returns the character at the specified index (position)
2	concat()	Joins two or more strings, and returns a new joined strings
3	indexOf()	Returns the position of the first found occurrence of a specified value in a string
4	lastIndexOf()	Returns the position of the last found occurrence of a specified value in a string
5	slice()	Extracts a part of a string and returns a new string

Q2. LIST AND DISCUSS ANY THREE HTTP COMMANDS BTL1

The HTTP commands are as follows:

- CONNECT Command
- DISCONNECT Command
- GET Command
- HEAD Command
- LOAD RESPONSE_INFO BODY Command
- LOAD RESPONSE_INFO HEADER Command
- POST Command
- SYNCHRONIZE REQUESTS Command

Q3. EXPLAIN DIFFERENT DATA TYPES IN JS BTL1

JS Data types

Boolean.

Null.

Undefined.

Number.

String.

Symbol.

Object.

Q4. What is the purpose of DOM Node Tree?

BTL2

Ans:

HTML DOM views the HTML document with a tree structure format and it consists of root node and child nodes.

- The node-tree is being accessed using the tree formation and the structure in which the elements get created.
- The contents that are being modified or removed using the new elements and it can be created within the limitations.
- The structure consists of a document that is the root and within it Root element <html> from where the tree starts.

Assignment No. - 05

TITLE

Servlet and MySQL(Backend)

OBJECTIVES

1. Understand about basic concepts of html, CSS
2. Understand the basic functionalities of Servlet
3. Having the knowledge of SQL query to create the database

PROBLEM STATEMENTS

Implement the sample program demonstrating the use of Servlet.

e.g., Create a database table ebookshop (book_id, book_title, book_author, book_price, quantity) using database like Oracle/MySQL etc. and display (use SQL select query) the table content using servlet.

OUTCOMES

Students will be able to,

1. Develop a dynamic webpage using JSP, HTML and Servlet.
2. Write a server side java application called Servlet to catch the data sent from the client, process it and store it on a database (MySQL).

SOFTWARE NEEDED

1. Any Operating System
2. JDK 7 or later
3. Editors; Netbeans/Eclipse
4. Web browser
5. Tomcat 7 or later

THEORY - CONCEPT

Servlets

Servlet technology is used to create web applications (resides at server side and generates dynamic web page).

Servlet technology is robust and scalable because of java language. Before Servlet, CGI (Common Gateway Interface) scripting language was popular as a server-side programming language. But there were many disadvantages of this technology.

Disadvantages

There are many interfaces and classes in the servlet API such as Servlet, GenericServlet, HttpServlet, HttpServletRequest, HttpServletResponse etc.

What is a Servlet?

Servlet can be described in many ways, depending on the context.

Servlet is a technology used to create web applications.

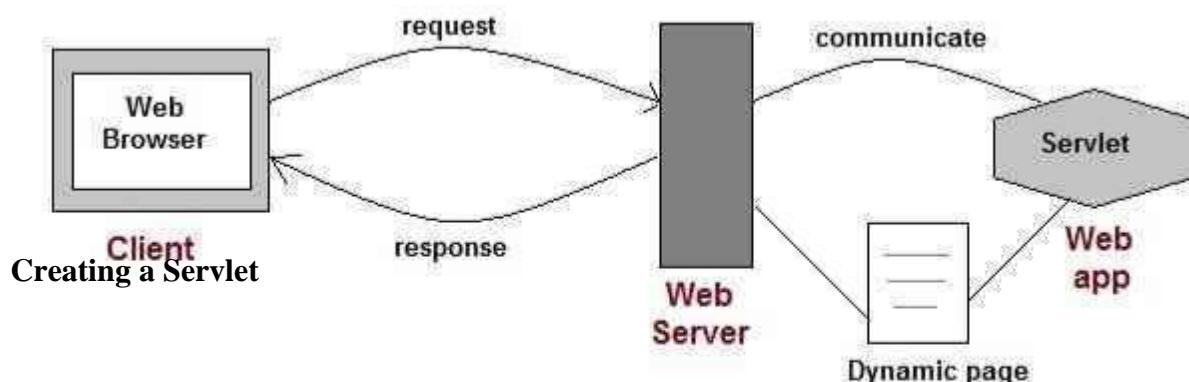
Servlet is an API that provides many interfaces and classes including documentation.

Servlet is an interface that must be implemented for creating any servlet.

Servlet is a class that extends the capabilities of the servers and responds to the incoming request.

It can respond to any type of request.

Servlet is a web component that is deployed on the server to create dynamic web page.



There are three different ways to create a servlet.

By implementing **Servlet** interface

By extending **GenericServlet** class

By extending **HttpServlet** class

But mostly a servlet is created by extending the **HttpServlet** abstract class. As discussed earlier **HttpServlet** gives the definition of service() method of the **Servlet** interface. The servlet class

that we will create should not override service() method. Our servlet class will override only the doGet() or doPost() method.

When a request comes in for the servlet, the Web Container calls the servlet's service() method and depending on the type of request the service() method calls either the doGet() or doPost() method.

NOTE: By default a request is a Get request.

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

public MyServlet extends HttpServlet

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

    {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html><body>");
        out.println("<h1>Hello Readers</h1>");
        out.println("</body></html>");
    }
}
```

Write above code in a notepad file and save it as **MyServlet.java** anywhere on your PC. Compile it(explained in next step) from there and paste the class file into WEB-INF/classes/ directory that you have to create inside **Tomcat/webapps** directory.

Compiling a Servlet

To compile a Servlet a JAR file is required. Different servers require different JAR files. In Apache Tomcat server servlet-api.jar file is required to compile a servlet class.

Create Deployment Descriptor

Deployment Descriptor(DD) is an XML document that is used by Web Container to run Servlets and JSP pages. DD is used for several important purposes such as:

Mapping URL to Servlet class.

Initializing parameters.

Defining Error page.

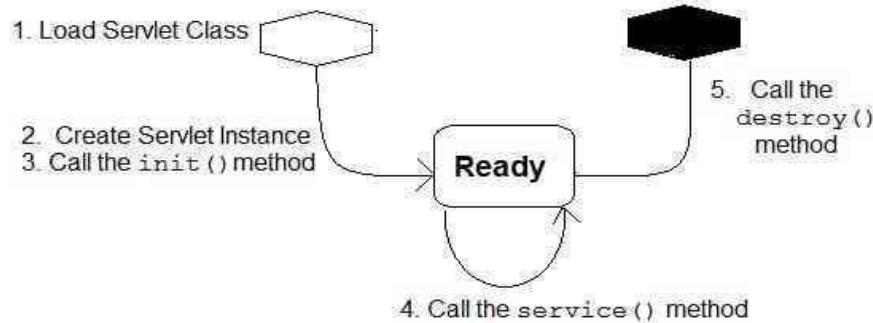
Security roles.

Declaring tag libraries.

Now we will see how to create a simple **web.xml** file for our web application.

```
First line of any xml document
<?xml version="1.0" encoding="UTF-8"?>
    root tag of web.xml file. All other tag come inside it
<web-app version="3.0"
    xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
    http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd">
    this tag maps internal name to
    fully qualified class name
    Give a internal name to your servlet
    <servlet>
        <servlet-name>hello</servlet-name>
        <servlet-class>MyServlet</servlet-class>
    </servlet>
    this tag maps internal name to
    public URL name
    servlet class that you
    have created
    URL name. This is what the user will
    see to get to the servlet.
    <servlet-mapping>
        <servlet-name>hello</servlet-name>
        <url-pattern>/hello</url-pattern>
    </servlet-mapping>
</web-app>
```

Servlet Life Cycle



Loading Servlet Class : A Servlet class is loaded when first request for the servlet is received by the Web Container.

Servlet instance creation : After the Servlet class is loaded, Web Container creates the instance of it. Servlet instance is created only once in the life cycle.

Call to the init() method : init() method is called by the Web Container on servlet instance to initialize the servlet.

Signature of init() method :

```
public void init(ServletConfig config) throws ServletException
```

Call to the service() method : The containers call the service() method each time the request for servlet is received. The service() method will then call the doGet() or doPost() methos based ont eh type of the HTTP request, as explained in previous lessons.

Signature of service() method :

```
public void service(ServletRequest request, ServletResponse response) throws
ServletException, IOException
```

Call to destroy() method: The Web Container call the destroy() method before removing servlet instance, giving it a chance for cleanup activity.

TECHNOLOGY/TOOL IN BRIEF

1. JSP and Servlets
2. IDE: NetBeans 7.0 or Later

3. Databases: MySQL

NetBeans: NetBeans is an IDE, used for quickly and easily developing java desktop, mobile, and web applications, as well as HTML5 applications with HTML, JavaScript, and CSS. Also provides a huge set of tools for PHP and C/C++ developers. It is a free and open source tool and has a great community of users and developers around the world.

MySQL: MySQL is a freely available open source Relational Database Management System (RDBMS). It uses the Structured Query Language (SQL).

SQL is the most popular language for adding, accessing and managing data in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use. MySQL is an essential part of almost every open source PHP application. Good examples for PHP & MySQL-based scripts are WordPress, Joomla, Magento and Drupal.

DESIGN / EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Design html and jsp files with an extension of.html and .jsp
2. Write database connection page using servlet
3. Set MySQL username, password and database name in database connection page
4. Start the Tomcat Server with port number
5. Open the browser and type localhost:8084

TEST CASES

Manual testing is used to validate the fields like username, password, mobile number and email id's of the users entered by user with the database.

CONCLUSION / ANALYSIS

Hence, we have performed the dynamic web application using Servlet and MySQL.

FAQs:

Q1. What is a Servlet?**BTL1**

Answer: A Servlet is a Java class that dynamically processes requests and generates responses to be sent back to the client. It is a server-side component that extends the capabilities of a server.

Q2. What is the difference between doGet() and doPost() methods in a Servlet? BTL1

Answer:

- doGet() is used to handle HTTP GET requests. It is typically used for operations that don't have side effects, such as fetching data.
- doPost() is used to handle HTTP POST requests. It is typically used for operations that have side effects, such as submitting form data or updating server-side resources.

Q3. Explain the role of web.xml in Servlet development.**BTL1**

Answer: web.xml is a deployment descriptor file for Java web applications. It is used to configure Servlets, filters, listeners, welcome pages, error pages, and other web application settings. It provides mapping between URL patterns and Servlets, among other configuration options.

Q4.What are the advantages of using Servlets?**BTL1**

Answer:

- Platform independence due to Java's write once, run anywhere nature.
- Improved performance compared to traditional CGI programs.
- Reusability and maintainability of code.
- Integration with Java EE technologies like JDBC for database access, JNDI for naming and directory services, etc.

Assignment No. - 06

TITLE

JSP, Servlet and MySQL(Backend)

OBJECTIVES

1. Understand about basic concepts of html, CSS
2. Understand the basic functionalities of JSP
3. Having the knowledge of SQL query to create the database

PROBLEM STATEMENTS

1. Implement the program demonstrating the use of JSP.
2. e.g., Create a database table students_info (stud_id, stud_name, class, division, city) using database like Oracle/MySQL etc. and display (use SQL select query) the table content using JSP.

OUTCOMES

Students will be able to,

1. Develop a dynamic webpage using JSP, HTML and Servlet.
2. Write a server side java application called JSP to catch form data sent from the client and store it on a database (MySQL).

SOFTWARE NEEDED

1. Any Operating System
2. JDK 7 or later
3. Editors; Netbeans/Eclipse
4. Web browser
5. Tomcat 7 or later

THEORY - CONCEPT

Java Server Pages (JSP): It is a server side programming technology that is used to create dynamic web-based applications. JSP have right to use the complete Java APIs, including the JDBC API to access the databases.

It is a technology that helps software developers to create dynamic web pages based on HTML, XML and other document types. It was released in 1999 by Sun Microsystems. It is just like a PHP and ASP, but it uses the Java programming language.

A JSP element is a type of java servlet that is designed to accomplish the role of a user interface for a java web application. Web developers write JSPs as text files that combine HTML or XHTML code, XML elements, and rooted JSP actions and commands.

Using JSP, you can collect input from users through web page forms, current records from a database or another source and create web pages dynamically.

JSP tags can be used for different purposes, such as retrieving information from a database or registering user preferences, accessing JavaBeans components, passing control between pages, and sharing information between requests, pages etc.

Why do we need JSP?

JSP is used for the design of dynamic web pages and servlet is used to code the logic that is present i.e. in the MVC (Model-View-Controller) architecture, the servlet is the controller and the JSP is the view.

Architecture of JSP

1. The request / response part of a JSP is defined in below architecture
2. The client initiated request for a JSP file using browser
3. Webs server (i.e, JSP Engine) invokes the JSP file and interprets the JSP file to produce a java code. The created java code will be a Servlet.
4. Once Servlet is created, JSP engine compiles the servlet. Compilation errors will be detected in this phase.
5. Now servlet class is loaded by the container and executes it.
6. Engine sends the response back to the client.

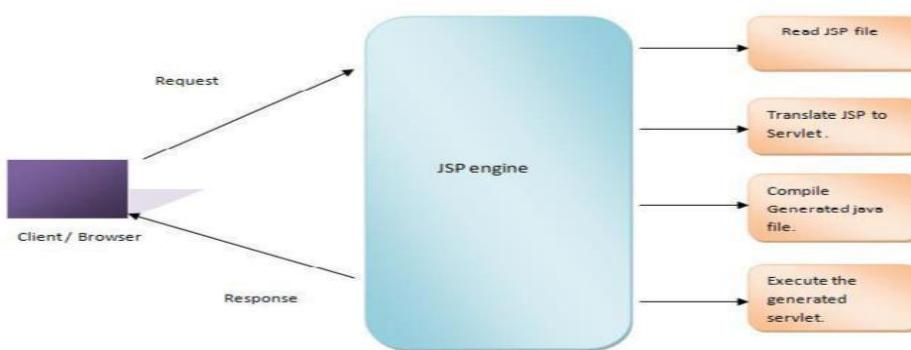


Figure.2: Architecture of JSP

Syntax of JSP:

JSP declarations is used to declare variables and methods as shown below,

```
<% text %>
```

Following is the simple and first example for JSP:

```
//Hello.jsp

<html>
  <head>
    <title> JSP File</title>
  </head>
  <body>
    <%
      out.println("Welcome to JSP Class");
    %>
  </body>
</html>
```

Output:

Welcome to JSP Class

Servlet:

A Servlet is a server side program written in Java. Servlet is a web component that is deployed on the server for creating dynamic web pages. A Java servlet is a Java program that extends the capabilities of a server. Although servlets can respond to any type of requests, they most commonly execute applications hosted on Web servers.

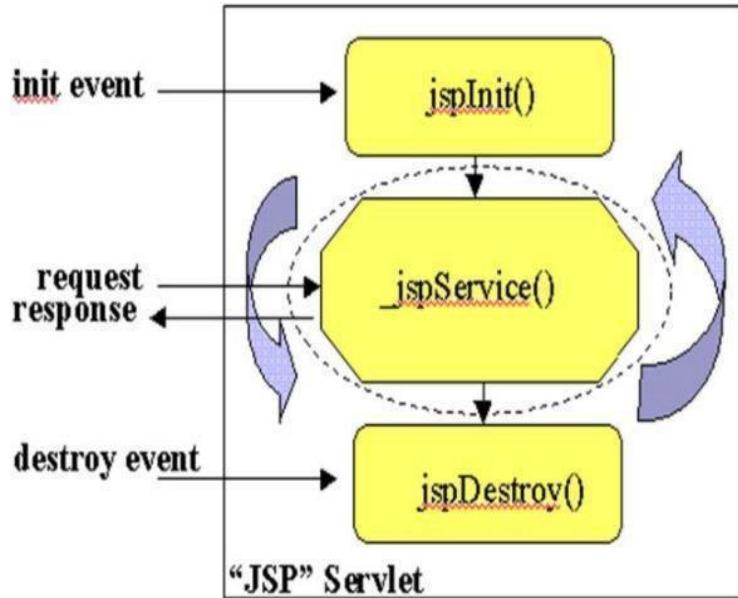


Figure.3: Functions of Servlet

TECHNOLOGY/TOOL IN BRIEF

1. JSP and Servlets
2. IDE: NetBeans 7.0 or Later
3. Databases: MySQL

NetBeans: NetBeans is an IDE, used for quickly and easily developing java desktop, mobile, and web applications, as well as HTML5 applications with HTML, JavaScript, and CSS. Also provides a huge set of tools for PHP and C/C++ developers. It is free and open source tool and has a great community of users and developers around the world.

MySQL: MySQL is a freely available open source Relational Database Management System (RDBMS). It uses the Structured Query Language (SQL).

SQL is the most popular language for adding, accessing and managing data in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use. MySQL is an essential part of almost every open source PHP application. Good examples for PHP & MySQL-based scripts are WordPress, Joomla, Magento_and Drupal.

DESIGN / EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Design html and jsp files with an extension of.html and .jsp
2. Write database connection page using servlet

3. Set MySQL username, password and database name in database connection page
4. Start the Tomcat Server with port number
5. Open the browser and type localhost:8084

TEST CASES

Manual testing is used to validate the fields like username, password, mobile number and email id's of the users entered by user with the database.

CONCLUSION / ANALYSIS

Hence, we have performed the dynamic web application using JSP and MySQL.

FAQs:

Q1. What is JSP (JavaServer Pages)?

BTL1

Answer: JSP stands for JavaServer Pages, a technology used for developing web applications that allows developers to embed Java code into HTML pages. It provides dynamic content generation capabilities similar to PHP and ASP.

Q2. How does a JSP page differ from a regular HTML page?

BTL1

Answer: A JSP page contains both HTML and Java code, allowing for dynamic content generation, whereas a regular HTML page only contains static content.

Q3. What are the main advantages of using JSP?

BTL1

Answer: Some advantages of JSP include:

- Easy integration with Java code.
- Simplified development of dynamic web pages.
- Reusability of Java components.
- Support for session management and JavaBeans.

Q4. How does a JSP page get compiled and executed?

BTL1

Answer: When a JSP page is accessed for the first time, the JSP engine compiles it into a Java servlet. This servlet is then loaded, executed, and the resulting HTML is sent back to the client. Subsequent requests use the compiled servlet directly, unless the JSP file has been modified.

Assignment No: 07

TITLE

Add dynamic web application essence using PHP, HTML and MySQL.

OBJECTIVES

To understand the principles and methodologies of PHP web based applications development process,

PROBLEM STATEMENT

Build a dynamic web application using PHP and MySQL.

- a. Create database tables in MySQL and create connection with PHP.
- b. Create the add, update, delete and retrieve functions in the PHP web app interacting with MySQL database

OUTCOMES

Students should be able to,

1. Develop web based applications using suitable client side and server side web technologies.
2. Develop solutions to complex problems using appropriate methods, technologies, frameworks, web services and content management.

SOFTWARE & HARDWARE REQUIREMENTS

Software (Minimum Requirement:):

1. Ubuntu 64 bit / Windows XP.
2. XAMPP Server

Hardware (Minimum Requirement:):

Intel p4 Machine with 1GB RAM and 32GB HDD.

THEORY-CONCEPT

1. PHP:

The PHP Hypertext Preprocessor (PHP) began as a little open source venture that advanced as an ever increasing number of individuals discovered how valuable it was. Rasmus Lerdorf released the principal form of PHP route in 1994. PHP is a recursive acronym for "PHP: Hypertext Preprocessor".

PHP is a server side scripting dialect that is installed in HTML. It is utilized to oversee dynamic substance, databases, session following, even form whole internet business locales. It is incorporated with various prevalent databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.

PHP is pleasingly zippy in its execution, particularly when gathered as an Apache module on the Unix side. The MySQL server, once began, executes even extremely complex questions with colossal outcome sets in record-setting time.

PHP bolsters a substantial number of real conventions, for example, POP3, IMAP, and LDAP. PHP4 included help for Java and conveyed question designs (COM and CORBA), making n-level improvement a plausibility out of the blue. PHP is excusing: PHP dialect tries to be as pardoning as would be prudent. PHP Syntax is C-Like.

PHP performs framework capacities, i.e. from documents on a framework it can make, open, read, compose, and close them. PHP can deal with frames, i.e. accumulate information from records, spare information to a document; through email you can send information, return information to the client.

You include, erase, adjust components inside your database through PHP. Access treats factors and set treats. Utilizing PHP, you can confine clients to get to a few pages of your site. It can encode information.

Example:

"Hello World" Script in PHP

To get a feel for PHP, first start with simple PHP scripts. Since "Hello, World!" is an essential example, first we will create a friendly little "Hello, World!" script.

As mentioned earlier, PHP is embedded in HTML. That means that in amongst your normal HTML (or XHTML if you're cutting-edge) you'll have PHP statements like this –

```
<html>
  <head>
    <title> Hello World</title>
  </head>
  <body>
    <?php echo ("Hello Php"); ?>
  </body>
</html>
```

To create and run PHP Web pages three fundamental parts should be introduced on your PC framework.

Web Server – PHP will work with for all intents and purposes all Web Server programming, including Microsoft's Internet Information Server (IIS) however then regularly utilized is unreservedly accessible Apache Server. Download Apache for nothing here – <https://httpd.apache.org/download.cgi>

Database – PHP will work with for all intents and purposes all database programming, including Oracle and Sybase yet most regularly utilized is uninhibitedly accessible MySQL database. Download MySQL for nothing here – <https://www.mysql.com/downloads/PHP Parser> – keeping in mind the end goal to process PHP content directions a parser must be introduced to create HTML yield that can be sent to the Web Browser. This instructional exercise will manage you how to introduce PHP parser on your PC.

2. MySQL:

MySQL is the most famous Open Source Relational SQL Database Management System. MySQL is outstanding amongst other RDBMS being utilized for creating different online programming applications. MySQL is created, advertised and upheld by MySQL AB, which is a Swedish organization. This instructional exercise will give you a fast begin to MySQL and make you OK with MySQL programming.

What is a Database?

A database is a different application that stores a gathering of information. Every database has at least one unmistakable APIs for making, getting to, overseeing, seeking and recreating the information it holds.

Different sorts of information stores can likewise be utilized, for example, records on the document framework or vast hash tables in memory yet information getting and composing would not be so quick and simple with those kinds of frameworks.

These days, we utilize social database administration frameworks (RDBMS) to store and oversee tremendous volumes of information. This is called a social database since every one of the information is put away into various tables and relations are set up utilizing essential keys or different keys known as Foreign Keys.

A Relational DataBase Management System (RDBMS) is a product that:

- Empowers you to execute a database with tables, segments and records.
- Ensures the Referential Integrity between columns of different tables.
- Updates the lists naturally.
- Deciphers a SQL inquiry and consolidates data from different tables.

RDBMS Terminology

Before we continue to clarify the MySQL database framework, let us modify a couple of definitions identified with the database.

- Database: A database is a gathering of tables, with related information.
- Table: A table is a grid with information. A table in a database resembles a basic spreadsheet.
- Column: One section (information component) contains information of one and a similar kind, for instance the segment postcode.
- Row: A line (= tuple, passage or record) is a gathering of related information, for instance the information of one membership.
- Redundancy: Storing information twice, needlessly to make the framework quicker.
- Essential Key: An essential key is exceptional. A key esteem can not happen twice in one table. With a key, you can just discover one column.
- Outside Key: A remote key is the connecting pin between two tables.
- Compound Key: A compound key (composite key) is a key that comprises numerous sections, since one segment isn't adequately exceptional.

- Index: A file in a database looks like a file at the back of a book.
- Referential Integrity: Referential Integrity ensures that an outside key esteem dependably indicates a current column.

MySQL Database

MySQL is a quick, simple-to-utilize RDBMS being utilized for some small and huge organizations. MySQL is produced, showcased and upheld by MySQL AB, which is a Swedish organization. MySQL is winding up so famous as a result of numerous great reasons:

- MySQL is discharged under an open-source permit. So you don't have anything to pay to utilize it.
- MySQL is a capable program in its own particular right. It handles a huge subset of the usefulness of the most costly and intense database bundles.
- MySQL utilizes a standard type of the outstanding SQL information dialect.
- MySQL takes a shot at many working frameworks and with numerous dialects including PHP, PERL, C, C++, JAVA, and so forth.
- MySQL works rapidly and functions admirably even with extensive informational indexes.
- MySQL is compatible with PHP, the most refreshing dialect for web advancement.
- MySQL underpins huge databases, up to 50 million lines or more in a table. The default document measure restrain for a table is 4GB, yet you can expand this (if your working framework can deal with it) to a hypothetical utmost of 8 million terabytes (TB).
- MySQL is adaptable. The open-source GPL permit enables developers to alter the MySQL programming to fit their own particular surroundings.

TECHNOLOGY/TOOL

1. Technology is to be used in PHP (PHP Hypertext Preprocessor) and tool XAMPP server is to be used to execute PHP web application.
2. XAMPP server embeds the PHP, MySQL and phpmyadmin, these three tools must be required to run php web application.

DESIGN/EXECUTION STEPS

For the design purpose html and CSS is to be used. For this design part contains the GUI of web applications, how its look? When users are going to use the web application.

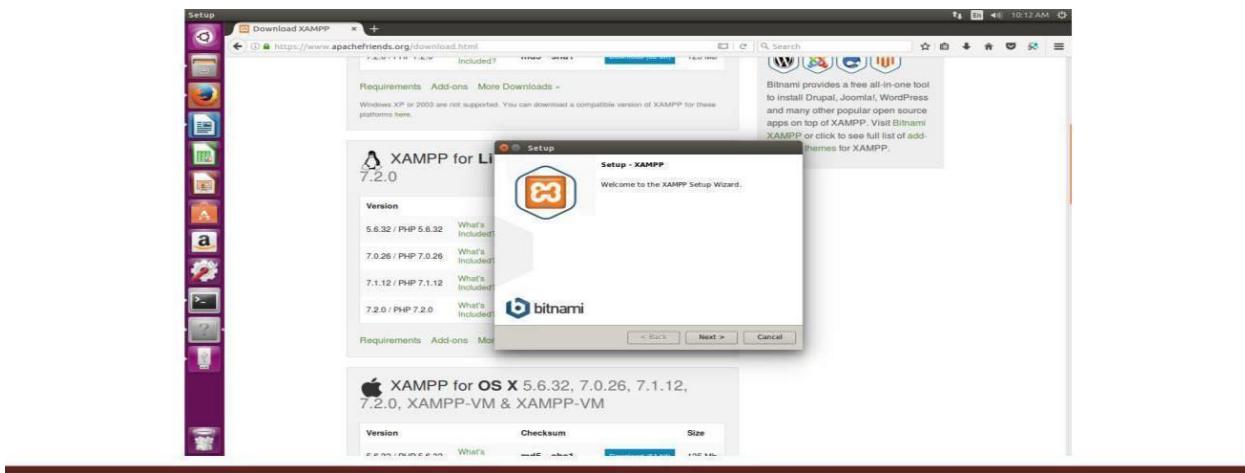
Steps to install XAMPP and configure the PHP, MYSQL server.

1. Download the XAMPP using the following link (download latest version as per your Operating system Windows/ Linux). Here we are using Linux Ubuntu Systems. Copy and paste downloaded XAMPP into home location.
<https://www.apachefriends.org/download.html>

2. Install XAMPP with the following command in the terminal(copy paste the command).

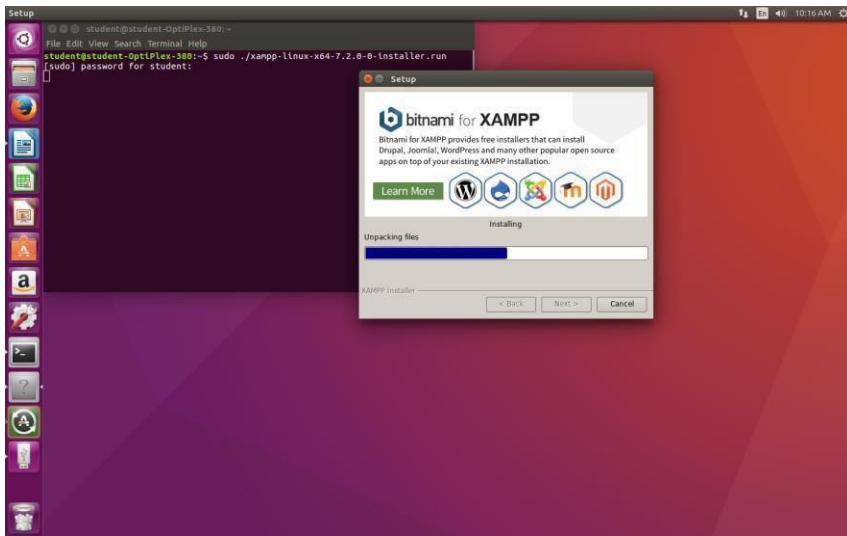
```
sudo ./xampp-linux-x64-7.2.0-0-installer.run
```

3. After the above command the following installation window will appear.

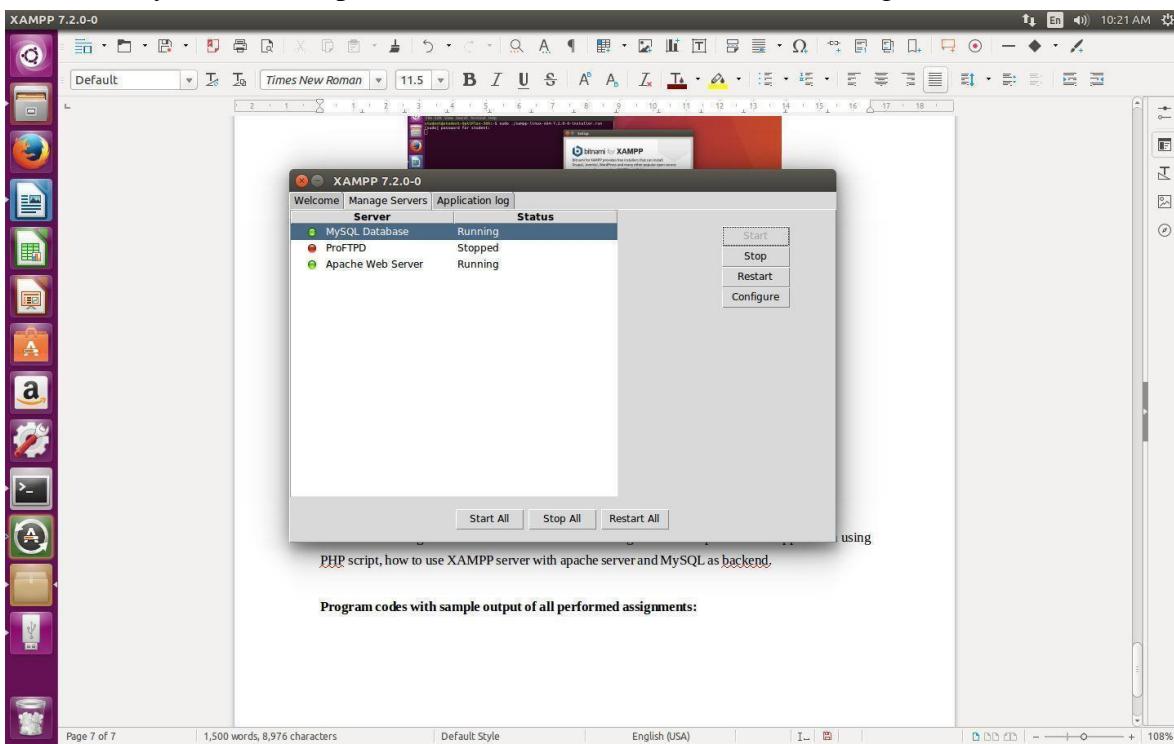


Click on next until all installation windows are completed.

During installation XAMPP look like as follows:



after successful completion of installation. It will ask to run XAMPP and here say to run. After run the MySQL and Apache web server should be in running state. See below image.



As per above image you are ready with Apache Web Server and MySQL Database.

4. Now Open browser and type “localhost” in URL with quotes. Following page should be



the Default directory is with following path

To become root just open the terminal and type the following command `sudo -i`

after you become the root just type the following command to enter the root directory

```
cd /
```

“opt” is the directory in which XAMPP is installed by default.

6. As you have successfully installed and started XAMPP now just navigate to htdocs by typing the following command into terminal:

Note: to edit or create any file in htdocs you need to be root : `sudo -i`

7. To navigate to root folder

```
cd /
```

8. To navigate to htdocs

```
cd opt/lampp/htdocs
```

9. To create a file hello.php

gedit hello.php

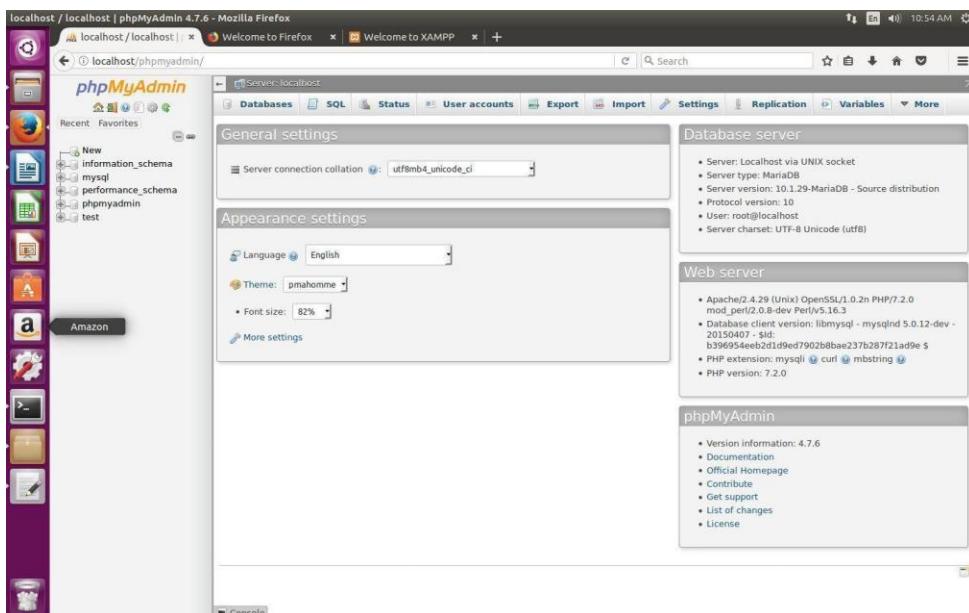
Code for Sample Hello Script

```
<?php
echo("Hello, Welcome to First PHP Application");
?>
```

10. Now go to your browser then type localhost/filename.php

11. Open phpmyadmin to create MySQL database

type in browser URL “localhost/phpmyadmin” you will see following window.



Here you can create the database and table inside the database as per your need. Here we created a “test” database with “users” table contain four columns.

To start XAMPP Manually simply call this command on root:

```
sudo /opt/lampp/lampp start
```

To stop XAMPP manually simply call this command on root:

```
sudo /opt/lampp/lampp stop
```

CONCLUSION/ANALYSIS

In this assignment, we have studied how to design and develop small web application using PHP script, XAMPP server with apache server and MySQL as backend. HTML or XHTML code, XML elements, and embedded JSP actions and commands.

FAQs:

Q1. What is PHP?

BTL1

PHP stands for Hypertext Preprocessor. It is a server-side scripting language used for web development. It is often embedded into HTML to add functionality such as dynamic content generation, form handling, and database interaction.

Q2. What is MySQL?

BTL1

MySQL is an open-source relational database management system (RDBMS) that uses Structured Query Language (SQL). It is widely used for managing databases in various applications, especially in web development alongside PHP.

Q3. How do you establish a connection between PHP and MySQL?

BTL1

To establish a connection between PHP and MySQL, you can use the mysqli **extension or the PDO (PHP Data Objects) extension**. Here's an example of using mysqli:

```
$servername = "localhost";  
  
$username = "username";  
  
$password = "password";  
  
$dbname = "database";
```

```
// Create connection  
  
$conn = new mysqli($servername, $username, $password, $dbname);  
  
// Check connection  
  
if ($conn->connect_error) {  
  
    die("Connection failed: " . $conn->connect_error);  
  
}
```

Q4. What are prepared statements in PHP MySQL?**BTL1**

Prepared statements are SQL statements that are precompiled by the database engine, allowing you to execute the same statement multiple times with different parameters. This helps prevent SQL injection attacks and improves performance. In PHP, you can use prepared statements with both mysqli **and** PDO. Here's an example using mysqli:

```
$stmt = $conn->prepare("INSERT INTO table (column1, column2) VALUES (?, ?)");  
  
$stmt->bind_param("ss", $value1, $value2);  
  
$value1 = "value1";  
  
$value2 = "value2";  
  
$stmt->execute();
```

Assignment No: 08

TITLE

Design and develop any web application using Struts framework.

OBJECTIVES

1. To impart the efficient and available client side and server side technologies.
2. To implement the communication between computing nodes using client side and server side technologies.
3. To design and implement the web services with content management.

PROBLEM STATEMENT

Design a login page with entries for name, mobile number, email id and login button. Use struts and perform following validations

- a. Validation for correct names
- b. Validation for mobile numbers
- c. Validation for email id
- d. Validation if no entered any value
- e. Re-display for wrongly entered values with message
- f. Congratulations and welcome page upon successful entries

OUTCOMES

Students should be able to,

1. Implement the effective client side and server side technologies using struts framework.
2. Solve the complex problem of development using MVC framework.

SOFTWARE & HARDWARE REQUIREMENTS

Software's: Java 1.7 or Higher, Apache Tomcat 7 or higher, Struts API's, Eclipse IDE.

THEORY

The framework plays a vital role in industries for manageable and well designed application development as well as enterprise application development. The core of the Struts framework is a flexible control layer based on standard technologies like Java Servlets, JavaBeans, Resource Bundles, and XML, as well as various Jakarta Commons packages. Struts encourages application architectures based on the

Model 2 approach, a variation of the classic Model-View-Controller(MVC)

Struts gives its own particular Controller segment and incorporates different advancements to give the Model and the View. For the Model, Struts can collaborate with standard information to advances, as JDBC and EJB, and also most any outsider bundles, as Hibernate, iBATIS, or Object Relational Bridge. For the View, Struts functions admirably with Java Server Pages, including JSTL and JSF, and in addition Velocity Templates, XSLT, and other introduction frameworks.

The Struts system gives the undetectable underpinnings each expert web application needs to survive. Struts causes you make an extensible advancement condition for your application, in view of distributed guidelines and demonstrated outline designs.

ORM remains for Object/Relational mapping. It is the customized and translucent constancy of items in a Java application in to the tables of a social database utilizing the metadata that portrays the mapping between the articles and the database. It works by changing the information starting with one portrayal then onto the next.

The Model-View-Controller Architecture

"Model-View-Controller" is a way to build applications that promotes complete separation between business logic and presentation. It is not specific to web applications, or Java, or J2EE (it predates all of these by many years), but it can be applied to building J2EE web applications.

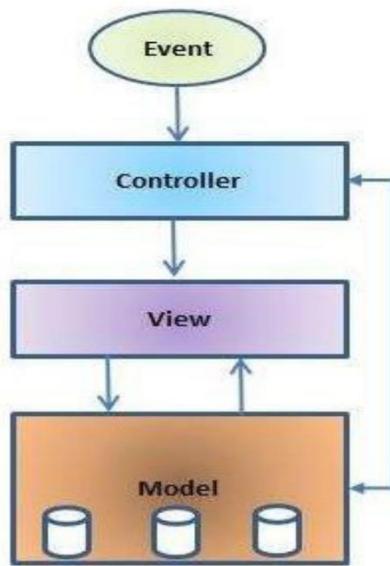


Figure.5: Basic MVC Architecture.

The "view" is the user interface, the screens that the end user of the application actually sees and interacts with. In a J2EE web application, views are JSP files. For collecting user input, you will have a JSP that generates an HTML page that contains one or more HTML forms. For displaying output (like a report), you will have a JSP generates an HTML page that probably contains one or more HTML tables. Each of these is a view: a way for the end user to interact with the system, putting data in, and getting data out.

What is Struts?

Struts is a framework that advances the utilization of the Model-View-Controller engineering for planning substantial scale applications. The structure incorporates an arrangement of custom label libraries and their related Java classes, alongside different utility classes. The most intense part of the Struts system is its help for making and preparing electronic structures. We will perceive how this functions later in this section.

Struts Tags

Common Attributes

Almost all tags provided by the Struts framework use the following attributes:

Attribute	Used for
Id	the name of a bean for temporary use by the tag
name	the name of a pre-existing bean for use with the tag
property	the property of the bean named in the name attribute for use with the tag
scope	the scope to search for the bean named in the name attribute

Table.2: Struts Framework Attributes

Creating Beans

Beans are created by Java code or tags.

Here is an example of bean creation with Java code:

- Creating a Plumber bean in the request scope Plumber aPlumber = new Plumber();
request.setAttribute("plumber", aPlumber);

Beans can be created with the <jsp:useBean></jsp:useBean> tag:

```
<!-- If we want to do <jsp:setProperty
...></jsp:setProperty> or --> <!-- <jsp:getProperty ...
--></jsp:getProperty> -->
```

```
<!-- we first need to do a <jsp:useBean ... ></jsp:useBean> -->
```

```
<jsp:useBean id="aBean" scope="session"
class="java.lang.String"> creating/using a bean in
session scope of type java.lang.String </jsp:useBean>
```

Most useful is the creation of beans with Struts tags:

```
<!-- Constant string bean -->
```

```
<bean:define id="greenBean" value="Here is a new constant string bean; pun intended."/>
```

```
<!-- Copying an already existent bean, frijole, to a new
bean, lima --> <bean:define id="lima" name="frijole"/>
```

```
<!-- Copying an already existent bean, while specifying the class -->
```

```
<bean:define id="lima" name="frijole" class="com.SomePackageName.Bbeans.LimaBean"/>
<!-- Copying a bean property to a different scope -->
<bean:define      id="goo"      name="foo"      property="geeWhiz"      scope="request"
toScope="application"/>
```

Other Bean Tags

The Struts framework provides other tags for dealing with issues concerning copying cookies, request headers, JSP implicit defined objects, request parameters, web application resources,

Struts configuration objects, and including the dynamic response data from an action. These tags are not discussed here, but it is important to be aware of their existence.

```
<bean:cookie ... >
<bean:header ... >
<bean:page ... >
<bean:parameter ... >
<bean:header ... >
<bean:resource ... >
<bean:struts ... >
```

Bean Output

The `<bean:message>` and `<bean:write>` tags from the Struts framework will write bean and application resources properties into the current `HttpServletResponse` object.

This tag allows locale specific messages to be displayed by looking up the message in the application resources .properties file.

```
<!-- looks up the error.divisionByZero resource
-->

<!-- and writes it to the HttpServletResponse >
object -->

<bean:message ... <bean:message
>           key="error.divisionByZero"/>
```

```

<!-- looks up the prompt.name resource --
- >

<!-- and writes it to the HttpServletResponse >
- object; -->

<!-- failing that, it writes the -->
- string

<!-- contained in the attribute >
arg0-->

<bean:message key="prompt.name" arg0='Enter a name:'/>

This tag writes the string equivalent of the specified bean or bean property to
the current HttpServletResponse object.

<bean:write ... >

    <!-- writes the value of
customer.getStreetAddress().toString() --> <!-- to the
HttpServletResponse object -->

    <bean:write name="customer" property="streetAddress"/>

```

Creating HTML Forms

Frequently data should be gathered from a client and handled. Without the capacity to gather client input, a web application would be futile. So as to get the clients data, a html shape is utilized. Client information can originate from a few gadgets, for example, content fields, content boxes, check takes care of, pop menus, and radio catches. The information compared to the client input is put away in an ActionForm class. A design document called struts-config.xml is utilized to characterize precisely how the client input are handled. The following diagram roughly depicts the use of Struts for using forms.

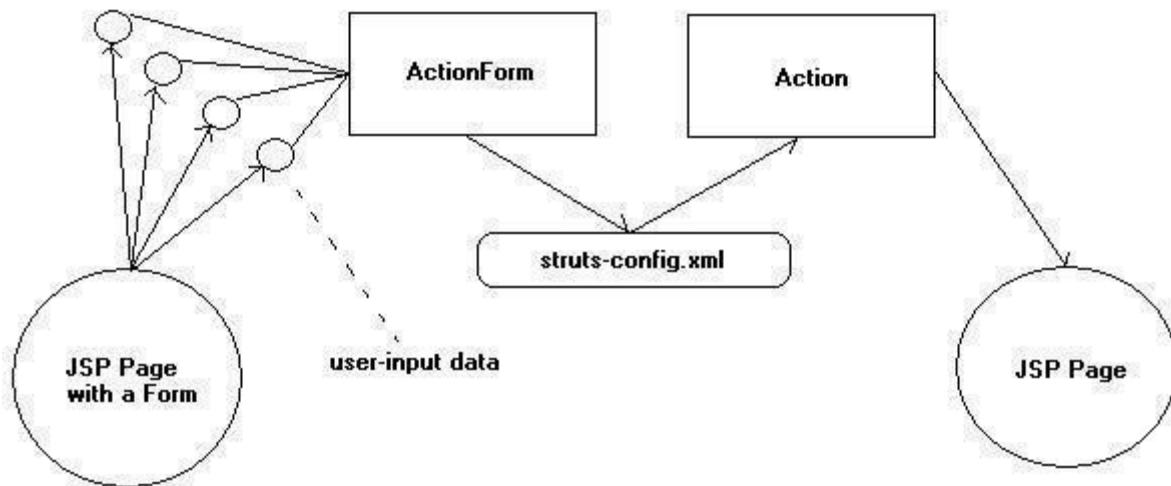


Figure.6: Struts for Using Forms

The Struts html tags are used to generate the widgets in the html that will be used in gathering the user's data. There are also tags to create a form element, html body elements, links, images, and other common html elements as well as displaying errors. Below are the tags provided by html section of the Struts framework and a short description of each.

```

<html:base>
<html:button>
<html:cancel>
<html:checkbox>
<html:multibox>
  
```

Generates a `<base>` tag. This tag should be used inside of a `<head>` tag.

Generates an `<input type="button">` tag. This tag should be used inside a `<form>` element.

Generates an `<input type="submit">` tag and causes the Action servlet not to invoke its `validate()` method. This tag should be used inside a `<form>` element.



Wheat Wood Clay



Stone Sheep

<html:checkbox> Generates an <input type="checkbox">.

<html:multibox> Generates an <input type="checkbox">. "Checkedness"

<html:errors>

<html:file>

<html:form>

<html:hidden>

The ActionForm class

depends upon whether the property array specified contains a corresponding value as the one specified for the multibox.

Generates html to display any errors that may have occurred during invocation of the validate()method.

Generates <form>.

There is a hidden element here which is invisible. :-)

Generates <input type="hidden">.

The purpose of the ActionForm class is to contain and provide validation of the user-input data.

This class is subclassed for application specific customization.

Here is a template for a customized ActionForm class with markers denoting where special items should be located in the class with \$ symbols.

```
package com.akurdi.action;

import com.opensymphony.xwork2.ActionSupport;

public class LoginAction extends ActionSupport {

    private String username;

    private String password;

    public String execute() {
        if(this.username == null || this.password == null)
```

```
{  
    return "error";  
}  
  
if (this.username.equals("admin"))  
    this.password.equals("admin123")) { return "success";  
} else {  
    addActionError(getText("error.login"));  
    return "error";  
}  
}  
  
public String getUsername() {  
    return username;  
}  
  
public void setUsername(String username) {  
    this.username = username;  
}  
  
public String getPassword()  
{  
    return password;  
}  
  
public void setPassword(String password) {  
    this.password = password;  
}  
}
```

Do this:

1. Create the directory structure. The root directory is SimpleStruts, and it has the standard WEB-INF directory with classes inside, and com.akurdi.action inside that. It also has a lib directory within WEB-INF, which is something we haven't seen before; we'll see in a minute what goes there.
2. Copy the Struts tag library descriptor files into WEB-INF. The files struts.tld, struts-bean.tld, struts-form.tld, struts-html.tld, struts-logic.tld, and struts-template.tld are available in the lib directory of your Struts installation.
3. Copy the Struts parser, struts.jar, into WEB-INF/lib/. This file is available in the lib directory of your Struts installation
4. Create the tag descriptor library file for any custom tags you may use beyond the Struts tags. In this case, the file defines no custom tags, but it's good practice to have it in place, in case you need to add your own tags later.
5. Create the struts-config.xml file.
6. There are three main sections to a struts-config.xml configuration file. They are the "Form Bean Definitions" section, the "Global Forward Definitions" section, and the "Action Mapping Definitions" section defines a forward called "success".
7. Create the web.xml file.

The web.xml web application configuration file will need to define the servlet ActionServlet, to which control will be transferred whenever an appropriate URL pattern is accessed. The servlet is defined just as any other servlet will be defined. The URL pattern is specified by a servlet mapping. For this application, the URL pattern is any requested resource that ends with a .do extension.

In order to use the Struts tags, the .tld files describing the tags will need to be included in the configuration file. The references to these tags are made just as they were for our own custom tags in the previous chapter. The Struts framework is simply a complex set of tag libraries (struts*.tld), with associated code (struts.jar).

8. The ApplicationResources.properties file provides resources that will be used by any subclassed Struts classes (for example, SetNameAction). This resources file provides a place to define prompts, labels that will display on buttons, and other information that may change. By placing this information in the ApplicationResources.properties file, recompiling any servlets used in the application can be avoided, as well as encouraging separation of logic and presentation.
9. Instances of the Name class are placed in the user sessions. Only one will exist in any particular user session. It provides methods for accessing and mutating a name.

10. The view of the application is done with the JSP index.jsp. It represents the user interface and allows the user to interact with the application.

11. Compile, create .war file and place it into the webapps folder of tomcat.

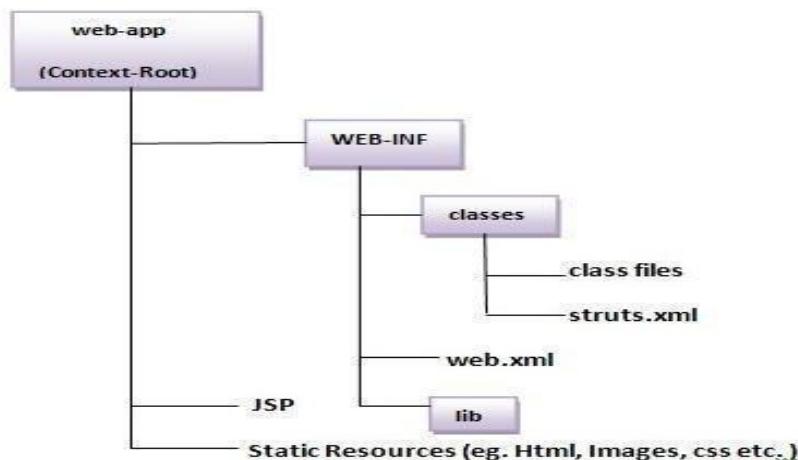
12. Go to <http://localhost:8080/Sturtsdemo/example/Login.jsp> to test your application

TECHNOLOGY/TOOL

1. Eclipse IDE
2. Apache Tomcat 7.0 or higher

DESIGN/EXECUTION STEPS

Step 1) Create the directory structure as



Step 2) Create input page as below.

Step 3) Provide the entry of Controller in (web.xml) file as given above

Step 4) Create the action class (LoginAction.java)

Step 5) Map the request in (struts-config.xml) file and define the view components.

Step 6) Load the jar files

Step 7) start server and deploy the project or create .war file paste it in webapps folder and run from manager-app.

TEST CASES

1. Manual test cases need to be performed on struts application.
2. Check whether the tomcat server is running.

3. Check whether the application running status in the manager app list is true.
4. Run the application by selecting the app in the list.
5. Check whether the application gives desired results.

CONCLUSION/ANALYSIS

Hence we have successfully tested the Struts framework and tested the results.

PROGRAM CODE: INPUT & OUTPUT

1. Deploy the application from tomcat manager.
2. Provide the login username and password for login.
3. Test the result for correct and incorrect credentials.

FAQS:

1. What is the Struts framework? BTL1
Answer: Struts is an open-source framework for developing Java web applications. It is based on the Model-View-Controller (MVC) design pattern and provides a structured approach for building web applications.

2. What are the key components of the Struts framework? BTL1

Answer: The key components of the Struts framework include:

- ActionServlet: The controller component responsible for handling incoming requests and dispatching them to appropriate Action classes.
- Action classes: These are Java classes that process requests, perform business logic, and interact with the model layer.
- ActionForm: A JavaBean class that represents the form data submitted by the user. It encapsulates the input data and provides validation.
- Struts configuration file (struts-config.xml): This XML file contains configuration settings such as mappings between URLs and Action classes, form beans, and global forwards.
- View components: JSP pages or other presentation technologies used to generate the user interface.

3. Explain the MVC architecture in the context of the Struts framework. BTL1

Answer: In the MVC architecture of Struts:

- Model: Represents the application's data and business logic. It typically consists of JavaBeans, EJBs, or other business objects.
- View: Presents the user interface to the users. In Struts, the view is usually created using JSP pages or other template technologies.
- Controller: Manages the flow of the application, handling user requests and invoking appropriate actions. In Struts, the ActionServlet serves as the controller, dispatching requests to Action classes.

4. How does Struts handle form validation?

BTL1

Answer: Struts provides built-in support for form validation using validation rules defined in the XML-based configuration file (struts-config.xml). Validation rules specify constraints on form fields such as required fields, maximum length, regular expressions, etc. When a form is submitted, Struts automatically validates the form data based on these rules. If validation fails, error messages are displayed to the user, and the input form is redisplayed with the validation errors.

Assignment No: 09

TITLE

Design and develop any web application using AngularJS.

OBJECTIVES

1. Understand the design of single-page applications and how AngularJS facilitates their development
2. Properly separate the model, view, and controller layers of your application and implement them using AngularJS
3. Master AngularJS expressions, filters, and scopes
4. Build Angular forms
5. Elegantly implement Ajax in your AngularJS applications
6. Write AngularJS directives

PROBLEM STATEMENT

Design an application using Angular JS.

e.g., Design registration (first name, last name, username, password) and login page using Angular JS.

OUTCOMES

Students can able to,

1. Implement the effective client side implementation.
Solve the complex problem of development using MVC framework.

SOFTWARE & HARDWARE REQUIREMENTS

Software's: Eclipse IDE/ Notepad/ Notepad++, Modern Web browser

THEORY-CONCEPT

AngularJS is an open-source web application framework. It was initially created in 2009 by MiskoHevery and Adam Abrons. It is presently kept up by Google. Its most recent adaptation is 1.2.21. "AngularJS is an auxiliary system for dynamic web applications. It gives you a chance to utilize HTML as your layout dialect and gives you a chance to stretch out HTML's linguistic structure to express your application parts plainly and compactly. Its information official and reliance infusion take out a significant part of the code you as of now need to compose. Also, everything occurs inside the program, making it a perfect band together with any server innovation".

General Features

- AngularJS is a productive system that can make Rich Internet Applications (RIA).
- AngularJS gives designers a choice to compose customer side applications utilizing JavaScript in a spotless Model View Controller (MVC) way.
- Applications written in AngularJS are cross-program agreeable. AngularJS consequently handles JavaScript code reasonably for every program.
- AngularJS is open source, totally free, and utilized by a great many engineers the world over. It is authorized under the Apache permit version 2.0.
- By and large, AngularJS is a system to assemble expansive scale, elite, and simple-to-keep up web applications.

Core Features:

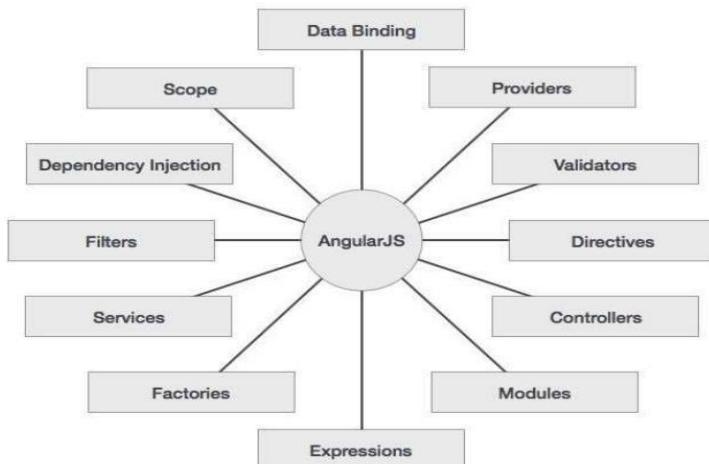


Figure.7: Architecture of AngularJS

1. **Data-authoritative:** It is the programmed synchronization of information amongst model and view parts.
2. **Scope:** These are objects that allude to the model. They go about as paste amongst controller and view.
3. **Controller:** These are JavaScript capacities bound to a specific degree.
4. **Services:** AngularJS accompanies a few implicit administrations, for example, \$http to make aXMLHttpRequests. These are singleton objects which are instantiated just once in application.
5. **Filters:** These select a subset of things from a cluster and restore another exhibit.
6. **Directives:** Directives are markers on DOM components, for example, components, characteristics, css, and that's only the tip of the iceberg. These can be utilized to make custom HTML labels that fill in as new, custom gadgets. AngularJS has worked in mandates, for example, ngBind, ngModel, and so on.
7. **Templates:** These are the rendered see with data from the controller and model. These can be a solitary record, (for example, index.html) or different perspectives in a single page utilizing partials.
8. **Routing:** It is the idea of exchanging sees.
9. **Model View Whatever:** MVW is an outline design for isolating an application into various parts called Model, View, and Controller, each with unmistakable obligations. AngularJS does not actualize MVC in the conventional sense, yet rather something nearer to MVVM (Model-View-ViewModel). The Angular JS group alludes it cleverly as Model View Whatever.
10. **Deep Linking:** Deep connecting permits to encode the condition of use in the URL with the goal that it can be bookmarked. The application would then be able to be re-established from the URL to a similar state.
11. **Dependency Injection:** AngularJS has a worked in reliance infusion subsystem that encourages the designer to make, comprehend, and test the applications effectively.

Advantages of AngularJS

- It gives the ability to make Single Page Application in a spotless and viable way.
- It gives information restricting the ability to HTML. Along these lines, it gives client a rich and responsive experience.
- AngularJS code is unit testable.

- AngularJS utilizes reliance infusion and makes use of partition of concerns.
- AngularJS gives reusable segments.
- With AngularJS, the engineers can accomplish greater usefulness with short code.
- In AngularJS, sees are unadulterated html pages, and controllers written in JavaScript do the business handling.

Model View Controller

Model View Controller or MVC as it is famously called, is a product configuration design for creating web applications. A Model View Controller design comprises the accompanying three sections.

- **Model – It is the most minimal level of the example in charge** of looking after information.
- **View – It is in charge of showing all or a part of** the information to the client.
- **Controller – It is a product Code that controls the connections between the Model and View.**

MVC is mainstream since it secludes the application rationale from the UI layer and backings detachment of concerns. The controller gets all solicitations for the application and afterward works with the model to set up any information required by the view.

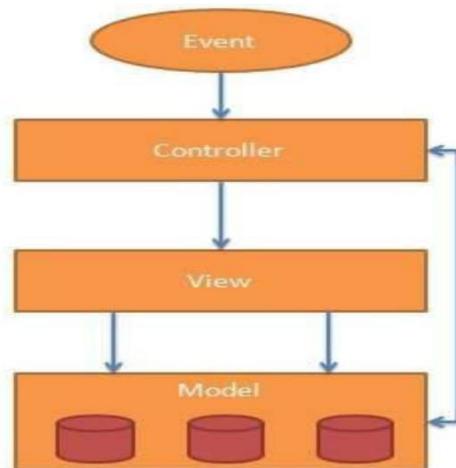


Figure. 8: Model View Controller

The Model

The model is in charge of overseeing application information. It reacts to the demand from see and to the directions from the controller to refresh itself.

The View

An introduction of information in a specific arrangement, activated by the controller's choice to exhibit the information. They are content based layout frameworks, for example, JSP, ASP, PHP and simple to incorporate with AJAX innovation.

The Controller

The controller reacts to client enter and performs communications on the information show objects. The controller gets input, approves it, and afterward performs business operations that alter the condition of the information demonstrated.

AngularJS is a MVC based structure.

- An AngularJS application comprises of following three essential parts –**ng-app** – This directive defines and links an AngularJS application to HTML.
- **ng-model** – This directive binds the values of AngularJS application data to HTML input controls.
- **ng-bind** – This directive binds the AngularJS Application data to HTML tags.

DESIGN/EXECUTION STEPS

Steps for AngularJS

1. When a link <https://angularjs.org/> is opened, there are two options to download AngularJS library –



- **View on GitHub – Click on this button to go to GitHub and get all of the latest scripts.**
- **Download AngularJS 1 – Or click on this button, a screen as below would be seen –**

Download AngularJS



- This screen gives various options of using AngularJS as follows:
- **Downloading and hosting files locally**
- 1. There are two different options: legacy and **latest**. The names itself are self-descriptive. **Legacy** has version less than 1.2.x and **latest** has 1.5.x version.
- 2. We can also go with the minified, uncompressed or zipped version.
- **CDN access – You also have access to a CDN.** The CDN will give you access around the world to regional data centers that in this case, Google host. This means using CDN moves the responsibility of hosting files from your own servers to a series of external ones. This also offers an advantage that if the visitor to your webpage has already downloaded a copy of AngularJS from the same CDN, it won't have to be re-downloaded.

Example:

Now let us write a simple example using AngularJS library. Let us create an HTML file *myfirstexample.html* as below –

```
<!doctype html>

<html>
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.2/angular.min.js"></script>
  </head>

  <body ng-app="myapp">
    <div ng-controller="HelloController">
      <h2>Welcome {{helloTo.title}} to the world of
      TutorialsPoint!</h2>
    </div>

    <script>
      angular.module("myapp",[])
        .controller("HelloController",function($scope){
          $scope.helloTo={};
          $scope.helloTo.title="AngularJS";
        });
    </script>
  </body>
</html>
```

Following sections describe the above code in detail:

1. Include AngularJS

We have included the AngularJS JavaScript file in the HTML page so we can use AngularJS –

```
<head>
  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
</head>
```

To update into the latest version of AngularJS, use the following script source.

```
<head>
```

```
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.2/angular.min.js"></script>
</head>
```

2. Point to AngularJS app

Next we tell what part of the HTML contains the AngularJS app. This done by adding the *ng-app* attribute to the root HTML element of the AngularJS app. You can either add it to *html* element or *body* element as shown **below** –

```
<body ng-app="myapp">
</body>
```

3. View

The view is this part –

```
<div ng-controller="HelloController">
<h2>Welcome {{helloTo.title}} to the world of TutorialsPoint!</h2>
</div>
```

ng-controller tells AngularJS what controller to use with this view. *helloTo.title* tells AngularJS to write the "model" value named *helloTo.title* to the HTML at this location.

4. Controller

The controller part is –

```
<script>
angular.module("myapp",[])
controller("HelloController",function($scope){
$scope.helloTo={};
$scope.helloTo.title="AngularJS";
});
</script>
```

This code registers a controller function named *HelloController* in the angular module named *myapp*. The controller function is registered in angular via the *angular.module(...).controller(...)* function call.

The `$scope` parameter passed to the controller function is the *model*. The controller function adds a `helloTo` JavaScript object, and in that object it adds a *title* field.

5. Execution

Save the above code as `myfirstexample.html` and open it in any browser.

Welcome AngularJS to the world of Tutorialspoint!

At the point when the page is stacked in the program, following things happen

HTML archive is stacked into the program, and assessed by the program. AngularJS JavaScript documents are stacked, the precise worldwide question is made. Next, JavaScript which registers controller capacities is executed.

Next AngularJS look over the HTML to search for AngularJS applications and perspectives. When `see` is found, it associates that view to the comparing controller work. Next, AngularJS executes the controller capacities. It at that point renders the perspectives with information from the model populated by the controller. The page is presently prepared.

How AngularJS	integrates with HTML
ng-app	directive indicates the start of AngularJS application.
ng-model	directive then creates a model variable named "name" which can be used
with the html	with the html page and within the div having ng-app directive.
ng-bind	then uses the name model to be displayed in the html span tag whenever
the user	input something in the text box.
Closing</div>	tag indicates the end of AngularJS application.

AngularJS directives are used to extend HTML. These are special attributes starting with `ng-` prefix. We're going to discuss following directives –

ng-app – This directive starts an **AngularJS Application**.

ng-init – This directive initializes application data.

ng-model – This directive binds the values of AngularJS application data to HTML input controls.

ng-repeat – This directive repeats html elements for each item in a collection.

TECHNOLOGY/TOOL

Any IDE or you can use web browser.

TEST CASES

Deploy the Html program run test the result for dynamic implementation of AngularJS.

CONCLUSION/ANALYSIS

With the help of this assignment it is helpful to understand features of AngularJS. MVC model structure and its use in advanced web programming is studied.

FAQs:

Q1. What is AngularJS?

BTL1

AngularJS is an open-source JavaScript framework developed and maintained by Google. It is used for building dynamic web applications, particularly single-page applications (SPAs).

Q2. What are the key features of AngularJS?

BTL1

Key features of AngularJS include two-way data binding, dependency injection, directives, MVC (Model-View-Controller) architecture, templating, and testing support.

Q3. Explain two-way data binding in AngularJS.

BTL1

Two-way data binding in AngularJS allows synchronization of data between the model and the view. Any changes made in the model are immediately reflected in the view, and vice versa, without requiring explicit DOM manipulation.

Q4. What is dependency injection in AngularJS?

BTL1

Dependency injection in AngularJS is a design pattern that helps in managing the dependencies between different components of an application. It allows objects to define their dependencies without having to create them, making the code more modular and testable.

Assignment No. - 10

TITLE

Web Application using EJB

OBJECTIVES

1. Understand about basic concepts of java beans.
2. Understand the basic functionalities of JSP, HTML.
3. Having the knowledge of JBOSS server to deploy web application.

PROBLEM STATEMENTS

Design and implement a business interface with necessary business logic for any web application using EJB.

e.g., Design and implement the web application logic for deposit and withdraw amount transactions using EJB.

OUTCOMES

Students will be able to,

1. Develop a dynamic webpage using Java Beans, HTML and JSP.
2. To understand the concepts and method of web based applications development Process using EJB.
3. Create a simple EJB 3 stateless session bean and a local Java application client which will call/invoke the bean to develop for addition of two numbers.

SOFTWARE NEEDED

1. Ubuntu 64 bit / Windows 7.
2. JDK 7 (Java SE 7)
3. EJB 3.0 (stateless session bean)
4. Eclipse luna
5. JBoss Application Server (AS) 7.1.1

THEORY – CONCEPT

Java Beans :

J2EE application container contains the components that can be used by the clients for executing the business logic .These components are known as Enterprise Java Beans (EJB) .

J2EE platform has component based architecture to provide multi-tiered, distributed and highly transactional features to enterprise level applications.

EJB mainly contains the business logic & business data. EJB component is an EJB class. It is a java class written by an EJB developer & this class implements business logic.

It is used for developing very much scalable and robust enterprise level applications to be deployed Application Server such as JBOSS, Web Logic etc.

EJB 3.0 is a large shift from EJB 2.0 and makes development of EJB based applications relatively easy.

Features of EJBs:

Some of the features of an application server include the following:

- **Client Communication:** The client, which is often a user interface, must be able to call the methods of objects on the application server via agreed-upon protocols.
- **State Management:** You'll recall our discussions on this topic in the context of JSP (JavaServer Pages) and servlet development back in Chapter 6.
- **Transaction Management:** Some operations, for example, when updating data, must occur as a unit of work. If one update fails, they all should fail.
- **Database Connection Management:** An application server must connect to a database, often using pools of database connections for optimizing resources.
- **User Authentication and Role-Based Authorization:** Users of an application must often log in for security purposes. The functionality of an application to which a user is allowed access is often based on the role associated with a user ID.
- **Asynchronous Messaging:** Applications often need to communicate with other systems in an asynchronous manner; that is, without waiting for the other system to respond. This requires an underlying messaging system that provides guaranteed delivery of these asynchronous messages.

- **Application Server Administration:** Application servers must be administered. For example, they need to be monitored and tuned.

Types of Enterprise Java Beans (EJB):

There are three types of Enterprise Java Beans namely:

1. Session Beans
2. Entity Beans
3. Message driven beans

Session Beans

- Session beans are intended to allow the application author to easily implement portions of application code in middleware and to simplify access to this code.
- Represents a single client inside the server
- The client calls the session bean to invoke methods of an application on the server
- Perform works for its client, hiding the complexity of interaction with other objects in the server
- Is not shared
- Is not persistent

When the client stops the session, the bean can be assigned to another client from the server
Session beans are divided into two types:

1. Stateless Session Bean:

Stateless Session Bean is intended to be simple and “light weight” components. The client, thereby making the server highly scalable, if required, maintains any state. Since no state is maintained in this bean type, stateless session beans are not tied to any specific client, hence any available instance of a stateless session bean can be used to service a client.

- values only for the duration of the single invocation
- Except during method invocation, all instances of stateless session bean are equivalent

Stateless Session Bean's Life Cycle:

- The client invoke the create method
- The EJB container : Instantiates the bean
- Invokes the setSessionContext Invokes ejbCreate
- The bean is ready
- While in the ready state

A client may invoke the remove method and the container calls the bean's ejbRemove method. It's never passivate .

2. Stateful Session Bean:

State ful Session Bean provides easy and transparent state management on the server side.

Because state is maintained in this bean type, the application server manages client/bean pairs.

Stateful session beans can access persistent resources on behalf of the client, but unlike entity beans, they do not actually represent the data.

Stateful Session Beans Life Cycle:

- The client invoke the create method
- The EJB container : Instantiates the bean
- The bean is read
- While in the ready state
- A client may invoke a business method

EJB container may activate a bean, moving it back to the ready stage, and then calls the bean's ejbActivate method.

A client may invoke the remove method and the container calls the bean's ejbRemove method

Difference Between Stateless and State Full EJB Are as Follows

Stateless:

1. Normally data members are not put in stateless session bean
2. Stateless beans are pooled

3. No effort for keeping client specific data
3. No Activation/Passivation in stateless session bean

Stateful:

1. Data members that represent state are present in stateful session bean
2. Stateful beans are cached
3. Setting the tag idle-timeout-seconds determines how long data is maintained in stateful session bean
4. Activation – Passivation used

An Entity Bean

- An entity bean is an object representation of persistent data maintained in a permanent data store such as a database. A primary key identifies each instance of an entity bean. Entity beans are transactional and are recoverable in the event of a system crash.
- Entity beans are representations of explicit data or collections of data, such as a row in a relational database. Entity bean methods provide procedures for acting on the data representation of the bean. An entity bean is persistent and survives if its data remains in the database.
- An entity bean can implement either bean-managed or container-managed persistence. In the case of bean-managed persistence, the implementer of an entity bean stores and retrieves the information managed by the bean through direct database calls. The bean may utilize either Java Database Connectivity (JDBC) or SQL-Java (SQLJ) for this method.
- In the case of container-managed persistence, the container provider may implement access to the database using standard APIs. The container provider can offer tools to map instance variables of an entity bean to calls to an underlying database. The container saves the data. There is no code in the bean for access the database. The container handles all database access required for the bean which create links between beans are created using a structure called abstract schema.

Enterprise Java Beans (EJB) Architecture

The EJB architecture is an extension of Web architecture. It has an additional tier. The clients of an enterprise bean can be a traditional java application like, applet, JSP or Servlet.

Like in a web application, client browser has to go all the way to web container to use a servlet or JSP, the communication between beans and clients is performed by the EJB container.

The following are the flows of the EJB architecture.

- The client is working on a web browser.
- There is a database server that hosts a database, like MySQL /Oracle.
- The J2EE server machine is running on an application server
- The client interface is provided with JSP/Servlet. The enterprise beans reside in the business tier providing to the client tier.
- The Application Server manages the relationships between the client and database machines.
- In a diagram, an enterprise bean is a non-visual component of a distributed, transaction-oriented enterprise application. Enterprise beans are typically deployed in EJB containers and run on EJB servers.

There are three types through which two or more activities may interfere:

1. Dirty read
 2. Non-Repeatable read
 3. Phantom read
- **Clustering and Load-Balancing:** Clustering is the process of combining the multiple peripherals, computers and other resources into a single unit.
 - A clustered system then works as a load balanced system. In a distributed system when a request is send to the server, an algorithm running on the server decides which server has less load and sends the request to that server. EJB container encapsulates these features to provide smooth and efficient service.
 - **Deployment Descriptor:** A deployment descriptor is an XML file packaged with the enterprise beans in an EJB JAR file or an EAR file. It contains metadata describing the contents and structure of the enterprise beans, and runtime transaction and security information for the EJB container.
 - **EJB Server:** An EJB server is a high-level process or application that provides a run-time environment to support the execution of server applications that use enterprise beans. An EJB server provides a JNDI-accessible naming service. It manages and coordinates the allocation of resources to client applications, provides access to system resources and provides a transaction service.

DESIGN / EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Design EJB project.
2. Start JBOSS & Deploy it on JBOSS server.
3. Design html and jsp files with an extension of.html and .jsp
4. Run the application in browser and get the result

TEST CASES

Manual testing is used to check the application is running properly in JBOSS server

CONCLUSION / ANALYSIS

Hence, we have created a simple EJB 3 stateless session bean and a local Java application client which will call/invoke the bean to develop for performing addition of two numbers.

FAQs:

Q1: What is EJB and how does it relate to web applications?

BTL1

Answer: Enterprise JavaBeans (EJB) is a server-side component architecture for building distributed enterprise applications in Java. EJBs are used to encapsulate business logic and provide services such as transaction management, security, and concurrency control. In the context of web applications, EJBs can be used to implement the backend business logic that interacts with databases, external services, and other components.

Q2: How do you define a session bean in EJB and what are its different types? BTL1

Answer: In EJB, a session bean is a type of enterprise bean that represents a single client session. There are three types of session beans:

Stateless Session Bean (SLSB): These beans do not maintain conversational state between client invocations. They are designed to handle individual method calls from clients efficiently.

Stateful Session Bean (SFSB): These beans maintain conversational state between multiple method invocations from the same client. They are useful for scenarios where the client needs to maintain state throughout a series of interactions with the server.

Singleton Session Bean (SSB): These beans provide a single instance shared by all clients. They are useful for implementing services that need to be accessed by multiple clients concurrently.

Q3: How does EJB facilitate transaction management in web applications? BTL1

Answer: EJB provides declarative transaction management, which allows developers to define transactional behavior using annotations or deployment descriptors. When a method of an EJB is invoked, the EJB container automatically manages the transaction lifecycle, including transaction begin, commit, and rollback. This ensures data integrity and consistency in web applications by guaranteeing that a group of operations either succeed or fail as a single unit.

Q4: Explain the role of EJB in security management within web applications. BTL1

Answer: EJB provides security features such as authentication, authorization, and secure communication. Developers can use declarative security annotations or configuration files to specify access control rules for EJB methods and resources. The EJB container enforces these rules at runtime, ensuring that only authorized clients can invoke protected methods or access sensitive data. This helps in maintaining the confidentiality, integrity, and availability of web applications.

Assignment No. : 11

TITLE

Title: MINI project

OBJECTIVES

To design and develop web applications using front end technologies and backend databases

PROBLEM STATEMENT

Mini Project: Design and implement a dynamic web application for any business functionality by using web development technologies that you have learnt in the above given assignments.

1. Define the problem you are solving. ...
2. Plan the workflow of your web application. ...
3. Wireframe/prototype your web application. ...
4. Receive Validation. ...
5. Choose your firepower. ...
6. Build your web application. ...
7. Test your web application. ...
8. Host and deploy your web application

Assignment No. : 12

TITLE

Title: Content Beyond the syllabus: Content Management System - WordPress

OBJECTIVES

To design and develop web applications using front end technologies and backend databases

PROBLEM STATEMENT

A content management system or CMS is a software that is used to build websites and create content to be published on the internet. Typically CMS allows you to create a website without writing any code.

In the early days of the web, you needed to know HTML to be able to code a website and publish your content online. That used to be hard, but CMS has made it much easier. Instead of writing your website code from scratch, the CMS software does it for you.

Think of it like driving a car. You don't need to understand the mechanics behind how it all works. Instead, you use a simplified dashboard and pedals to tell the car what to do.

With a powerful CMS platform like WordPress, you can log in to your website dashboard and use a simplified interface to create your web pages, add content, and customize the design. Then the CMS does the hard work of creating the code for you.

Website Dashboard

A good content management system allows you to use a simple user interface to create websites. As the website owner, you can log in to the website dashboard to customize your website. The dashboard can also be called the backend or admin area of a website. wordpress website dashboard

Themes to Customize the Design

CMS platforms usually let you change the look of your website by selecting a template or theme.

Themes are pre-designed templates that let you change the layout, colors, fonts, and other design aspects of a website.

WordPress has thousands of free and premium themes available. We have created our expert pick of the best WordPress themes.

Content Editor

CMS platforms include a content editor that you can use to create and publish your web pages.

Text editors are sometimes called WYSIWYG editors, which stands for What You See Is What You Get. As you edit the text, you can see exactly what it will look like on your website, instead of looking at code. Think of it like using an editor like Microsoft Word.

the WordPress content editor allows you to publish web content without coding

CMSes also allow you to upload images and other media files to your website.

Plugins and Extensions

Many CMS platforms let you add new features and modify your site in other ways by adding plugins.

Plugins on a website are like apps on a smartphone. You can easily install them to add new features and functions to your website. You can use them to add new features like a contact form or social media integration.

WordPress comes with over 57,000 free plugins. We have created a list of the must have WordPress plugins.

User Management

With a good CMS, you can easily give other people access to publish content on your website.

You can even assign them different user roles to control what they have access to.

Benefits of Using a CMS

CMS software make it easier for beginners to create websites. They open up the internet for non-techy users by allowing them to design their own websites, publish content on the internet, and build online businesses without hiring developers.

Here are just some of the benefits of using a CMS software to build your website.

No hassle publishing: A CMS allows even those without technical skills to publish content, make websites, and edit content using a simple dashboard.

Convenient content scheduling: Content management systems allow site administrators to publish content with a click of a button. You can schedule posts to be published to meet calendar deadlines, business events, or product launches.

Works with any size business: Whether you're making a website for a big name brand, or just starting a business, you can use a CMS. With the right CMS and web hosting package, you can create any type or size of website.

Affordable and easy to control: A CMS allows beginners to manage sites of all sizes without relying on high-priced web developers to perform site maintenance or make routine changes. A content management system allows you to choose who has access to your site.

Ability to Customize: It's easy to change your website design or customize it with your own logo, colors, and styles. You can also use plugins to add more features.

Conclusion : We learnt Content management system with Word press and able to design websites.