

<b>COURSE TITLE</b>	<b>WEB TECHNOLOGIES I - LAB</b>
<b>COURSE CODE</b>	<b>05MB0103</b>
<b>COURSE CREDITS</b>	<b>2</b>

**Objective:**

- 1 Aim to provide an overview of web page, websites, types of websites and structures of it.
- 2 Goal is to provide a comprehensive understanding designing of attractive webpages by adding CSS.
- 3 Aim to add on the designing skills with bootstrap and add classes to make attractive designs.
- 4 Cover CDN, local installation of bootstrap and adding classes for the effective designs.
- 5 Teach scripting language, to add logics into webpages and make it responsive.

**Course Outcomes:** After completion of this course, student will be able to:

- 1 Students will have understanding of the structure of the web, webpages, websites and types of websites.
- 2 Candidates will able to make webpage, website structure design.
- 3 Students will able to develop their designing skills with CSS.
- 4 Students will able to add Bootstrap classes for effective designs.
- 5 To learn about CDN and well-designed the webpage with logics of scripting.

**Pre-requisite of course:NA**

**Teaching and Examination Scheme**

<b>Theory Hours</b>	<b>Tutorial Hours</b>	<b>Practical Hours</b>	<b>ESE</b>	<b>IA</b>	<b>CSE</b>	<b>Viva</b>	<b>Term Work</b>
0	0	4	50	30	20	0	50
<b>Contents : Unit</b>	<b>Topics</b>						<b>Contact Hours</b>
<b>Total Hours</b>							

**Suggested List of Experiments:**

Contents : Unit	Topics	Contact Hours
1	<p><b>Unit 1</b></p> <p>Create a basic HTML document structure with the necessary tags. Create a basic HTML document and Add a title to your HTML webpage., Create a heading (h1) that introduces the topic of your HTML webpage &amp; Include a paragraph (p) that provides some information about your topic in HTML document. , Create a basic HTML document structure with the necessary tags. Add a title to your webpage, Create a heading (h1) that introduces the topic of your webpage. Include a paragraph (p) that provides some information about your topic., Add an image (img) into your HTML document related to your topic. Create an unordered list (ul) of at least three items related to your topic., Add a hyperlink (a) to an external website that provides more information about your topic. Create a table (table) to organize some data related to your topic into HTML document., In your HTML document Embed a YouTube video related to your topic using the appropriate tag. Use CSS to style your webpage, changing the font, colors, and layout., Create a form (form) with input fields (input) for collecting user information related to your topic. Implement basic accessibility features such as alt text for images and labels for form inputs., Use semantic HTML elements such as header, footer, nav, and section to structure your webpage. Experiment with different HTML elements and attributes to understand their functionality., Style a paragraph element to have a different font family, font size, and font color. Create a CSS animation to make an element move or change its properties (e.g., fade in/out, slide, rotate)., Style a form to have custom borders, padding, and background color for its input fields and submit button. Style a heading element (h1) to have a different font family, font size, and font color., In HTML document add padding and a background color to a div element, Center align text within a paragraph element (p). Change the background color and font color of a button element., Style an unordered list (ul) to have square bullet points and a different font size. Create a border around an image element (img) with a specific color and width., Apply a hover effect to a link (a) to change its color when hovered over. In HTML document, set the width and height of an input field (input) within a form, style a paragraph element to have italic text., In your HTML document, Add a margin around a block-level element (div) to create space between it and surrounding elements, and apply a background image to the entire webpage body. Use CSS to hide an element on the webpage (display: none;)., Create a navigation bar with horizontal links using an unordered list (ul) and apply styling to make it visually appealing. Apply different text alignment (left, center, right, justify) to separate paragraphs., Use CSS to create a hover effect on a button element, changing its background color when hovered over. Create a CSS animation to make an element move or change its properties (e.g., fade in/out, slide, rotate)., Style a form to have custom borders, padding, and background color for its input fields and submit button.</p>	30

**Suggested List of Experiments:**

Contents : Unit	Topics	Contact Hours
2	<b>Unit 2</b> Create a basic Bootstrap template with the necessary HTML structure and Bootstrap CDN links. Implement a Bootstrap grid system to create a responsive layout with rows and columns., Use Bootstrap classes to create a navigation bar with dropdown menus. Style buttons using Bootstrap classes for different styles (primary, secondary, success, etc.)., Apply Bootstrap utility classes to style typography (e.g., text alignment, font size, font weight). Use Bootstrap forms and form components to create a simple contact form with input fields and a submit button., Implement a Bootstrap carousel to display a slideshow of images or content. Use Bootstrap modal components to create a pop-up dialog box with custom content., Create a card layout using Bootstrap cards to display information in a visually appealing manner. Apply Bootstrap responsive classes to hide or show elements based on screen size (e.g., d-none, d-md-block)., Use Bootstrap utilities to create spacing and margin between elements (e.g., m-2, mt-4). Implement a Bootstrap navbar with a toggle button for mobile navigation., Create a responsive Bootstrap table with striped rows and hover effect. Use Bootstrap alerts to display success, warning, or error messages on the webpage., Customize Bootstrap styles by overriding default styles with custom CSS. Declare and assign values to variables using different data types: string, number, Boolean, array, object., Declare a function that takes two parameters and returns their sum. Declare a function that takes a string as a parameter and returns its length., Declare a function that takes an array of numbers as a parameter and returns the sum of all elements. Write an if statement that checks if a number is greater than 10 and prints a message accordingly., Write an if-else statement that checks if a number is even or odd and prints a message accordingly. Write a switch statement that prints different messages based on the value of a variable., Write a for loop that prints numbers from 1 to 5. Write a while loop that prints numbers from 1 to 5., Write a for loop that iterates over an array and prints each element. Write a for-in loop that iterates over an object and prints each key-value pair., Use array methods like forEach(), map(), or filter() to manipulate an array. Declare an object representing a person with properties like name, age, and gender., Access and print values of object properties using dot notation and bracket notation. Get a reference to an HTML element using document.getElementById(), Change the text content of an HTML element. Add an event listener to a button element and execute a function when it's clicked., Create a new HTML element programmatically and append it to the DOM	30
<b>Total Hours</b>		<b>60</b>

**Textbook :**

- 1 HTML and CSS: Design and Build Websites, Jon Duckett, Wiley, 2011

### References:

- 1 Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics, Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics, Jennifer Niederst Robbins, O'Reilly Media, 2018
- 2 JavaScript and JQuery: Interactive Front-End Web Development, JavaScript and JQuery: Interactive Front-End Web Development, Jon Duckett, Wiley, 2014
- 3 Bootstrap 4 Quick Start: Responsive Web Design and Development Basics for Beginners, Bootstrap 4 Quick Start: Responsive Web Design and Development Basics for Beginners, Jacob Lett, Bootstrap Creative, 2018
- 4 JavaScript: The Good Parts, JavaScript: The Good Parts, Douglas Crockford, O'Reilly Media, 2008

### Suggested Theory Distribution:

The suggested theory distribution as per Bloom's taxonomy is as follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process

Distribution of Theory for course delivery					
Remember / Knowledge	Understand	Apply	Analyze	Evaluate	Higher order Thinking / Creative
10.00	20.00	25.00	25.00	10.00	10.00

### Instructional Method:

- 1 Demo
- 2 PPT
- 3 Board work

### Supplementary Resources:

- 1 <https://developer.mozilla.org/en-US/>
- 2 <https://www.w3schools.com/html/default.asp>
- 3 <https://www.freecodecamp.org/>