

Module 1: Web Design Fundamentals (Classes 1–3)

Module Duration: 6 Hours (3 Classes × 2 Hours)

Module Learning Objectives

By the end of this module, students will be able to:

- Understand the fundamentals of web design and web development.
 - Explain how websites work on the Internet.
 - Differentiate between frontend and backend development.
 - Create basic web pages using HTML5.
 - Build forms and tables.
 - Use semantic HTML elements.
 - Develop simple, well-structured web pages following best practices.
-

Class 1 (2 Hours)

Introduction to Web Design & Development

Learning Outcomes

After completing this class, students will be able to:

- Understand the basics of web design and development.
 - Explain how websites function.
 - Differentiate between frontend and backend technologies.
 - Install and configure Visual Studio Code.
 - Create their first HTML web page.
-

1.1 What is Web Design?

Web Design is the process of planning and creating the visual appearance, layout, and user experience of a website.

A web designer focuses on:

- Website Layout
- Colors
- Typography
- Images
- Navigation
- User Experience (UX)
- Responsive Design

Examples

- Business Website
 - Personal Portfolio
 - School Website
 - E-commerce Website
 - News Portal
 - Blog
-

1.2 What is Web Development?

Web Development is the process of building websites and web applications using programming languages.

It includes:

- Website Structure
 - Database
 - User Authentication
 - Dynamic Content
 - Server-side Programming
-

1.3 How Websites Work

A website works through communication between the browser and a web server.

Basic Process

1. User enters a website address (URL).
2. The browser sends a request to the web server.
3. The server processes the request.
4. The server returns HTML, CSS, JavaScript, images, and other files.

5. The browser displays the webpage.
-

1.4 Website Components

Domain Name

A unique website address.

Examples:

- www.google.com
 - www.microsoft.com
-

Web Hosting

A service that stores website files and makes them accessible online.

Web Server

A computer that delivers website files to users.

Examples:

- Apache
 - Nginx
 - IIS
-

Browser

Software used to open websites.

Examples:

- Google Chrome
- Microsoft Edge
- Mozilla Firefox
- Safari

1.5 Frontend vs Backend

Frontend	Backend
Visible to users	Runs on the server
HTML	PHP
CSS	Python
JavaScript	Node.js
Bootstrap	Java
React	MySQL Database

1.6 Static vs Dynamic Websites

Static Website

- Fixed content
- Fast loading
- No database
- Easy to build

Examples:

- Portfolio
 - Company Profile
-

Dynamic Website

- Content changes automatically
- Uses a database
- User login
- Admin panel

Examples:

- Facebook
- Amazon
- Online Banking

1.7 Web Browsers

Popular browsers:

- Google Chrome
- Microsoft Edge
- Mozilla Firefox
- Safari
- Opera

1.8 Developer Tools

Browser Developer Tools help developers inspect and debug web pages.

Common features:

- Inspect Element
- Console
- Network
- Sources
- Responsive Device Mode

Shortcut:

F12 or **Ctrl + Shift + I**

1.9 Installing Visual Studio Code

Steps

1. Download Visual Studio Code.
2. Install the software.
3. Install useful extensions:
 - Live Server
 - Prettier
 - HTML CSS Support

- Auto Rename Tag
 - Auto Close Tag
-

1.10 Course Structure

Modules:

- HTML5
 - CSS3
 - Bootstrap
 - JavaScript
 - PHP
 - MySQL
 - Git & GitHub
 - Final Project
-

Practical Exercise

Activity 1

Install:

- Visual Studio Code
 - Google Chrome
-

Activity 2

Create your first webpage.

```
<!DOCTYPE html>
<html>
<head>
  <title>My First Web Page</title>
</head>
<body>

<h1>Welcome to Web Design & Development</h1>

<p>This is my first HTML page.</p>
```

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

Save as:

index.html

Open it in the browser.

Homework

Create a webpage displaying:

- Your Name
 - Photo
 - Department
 - Hobbies
-

Class 2 (2 Hours)

HTML5 Basics

Learning Outcomes

Students will be able to:

- Understand HTML document structure.
 - Create text content.
 - Add images and hyperlinks.
 - Organize information using lists.
 - Write clean HTML code.
-

2.1 What is HTML?

HTML stands for **HyperText Markup Language**.

It is the standard language used to create web pages.

HTML defines:

- Structure
- Content
- Layout (with CSS support)

2.2 HTML Document Structure

```
<!DOCTYPE html>
<html>
<head>
  <title>My Website</title>
</head>

<body>

</body>

</html>
```

Explanation

- `<!DOCTYPE html>` – Declares HTML5.
- `<html>` – Root element.
- `<head>` – Contains metadata.
- `<title>` – Browser tab title.
- `<body>` – Visible webpage content.

2.3 Headings

HTML provides six heading levels.

```
<h1>Main Heading</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```

2.4 Paragraphs

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

2.5 Lists

Unordered List

```
<ul>
<li>HTML</li>
<li>CSS</li>
<li>JavaScript</li>
</ul>
```

Ordered List

```
<ol>
<li>Install VS Code</li>
<li>Create HTML File</li>
<li>Run Website</li>
</ol>
```

2.6 Links

```
<a href="https://www.google.com">
Visit Google
</a>
```

2.7 Images

```

```

2.8 Comments

```
<!-- This is a comment -->
```

Comments are ignored by the browser and are useful for documentation.

Best Practices

- Use lowercase tags.
- Close all tags properly.
- Indent code for readability.
- Use meaningful filenames.
- Include `alt` text for images.

Practical Exercise

Build a personal profile webpage containing:

- Name
- Photo
- About Me
- Education
- Skills
- Contact Information
- Social Media Links

Homework

Create a webpage about your favorite tourist destination.

Class 3 (2 Hours)

HTML Forms, Tables & Semantic HTML5

Learning Outcomes

Students will be able to:

- Create forms and collect user input.

- Display tabular data.
 - Use semantic HTML elements.
 - Embed multimedia content.
 - Build accessible and organized web pages.
-

3.1 HTML Forms

Forms collect user information.

Common Form Elements

- Text Field
- Password Field
- Email
- Phone Number
- Radio Button
- Checkbox
- Select Menu
- Textarea
- Submit Button
- Reset Button

Example

```
<form>

<label>Full Name</label>
<input type="text">

<label>Email</label>
<input type="email">

<label>Password</label>
<input type="password">

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

</form>
```

3.2 HTML Tables

Tables organize data into rows and columns.

Example

```
<table border="1">

<tr>
<th>Name</th>
<th>Department</th>
</tr>

<tr>
<td>John</td>
<td>CSE</td>
</tr>

</table>
```

3.3 Semantic HTML5

Semantic elements improve accessibility and SEO by clearly describing the purpose of content.

Common semantic tags:

- `<header>`
- `<nav>`
- `<main>`
- `<section>`
- `<article>`
- `<aside>`
- `<footer>`

Example

```
<header>
<h1>My Website</h1>
</header>

<nav>
<a href="#">Home</a>
<a href="#">About</a>
</nav>

<main>
<section>
<h2>Welcome</h2>
<p>This is the main content.</p>
</section>
</main>

<footer>
<p>&copy; 2026 My Website</p>
```

</footer>

3.4 Audio & Video

Audio

```
<audio controls>
  <source src="audio.mp3" type="audio/mpeg">
</audio>
```

Video

```
<video controls width="400">
  <source src="video.mp4" type="video/mp4">
</video>
```

3.5 HTML Best Practices

- Use semantic HTML elements.
 - Keep code properly indented.
 - Use descriptive file and folder names.
 - Add `alt` attributes to images.
 - Validate forms where appropriate.
 - Test pages in multiple browsers.
 - Organize assets (images, CSS, JavaScript) into separate folders.
-

Practical Exercise

Create a **Student Registration Form** with the following fields:

- Full Name
- Father's Name
- Mother's Name
- Date of Birth
- Gender
- Mobile Number
- Email Address
- Address
- Department (Dropdown)
- Skills (Checkboxes)

- Profile Photo Upload
 - Submit and Reset Buttons
-

Module 1 Assessment

Assessment	Marks
HTML Fundamentals Quiz	10
Practical Exercises	20
Assignments	20
Registration Form Project	40
Class Participation & Viva	10
Total	100

Module 1 Learning Outcomes

After completing **Module 1: Web Design Fundamentals**, students will be able to:

- Explain how websites are built and delivered over the Internet.
- Create well-structured HTML5 documents.
- Use headings, paragraphs, lists, links, and images effectively.
- Build tables and user-friendly HTML forms.
- Apply semantic HTML5 elements to improve accessibility and SEO.
- Develop simple, standards-compliant web pages ready for styling with CSS in the next module.