

WEB TECHNOLOGY-I

6.1 INTRODUCTION TO WEB DEVELOPMENT

Definition: Web development is the process of building, structuring, and maintaining websites and web applications, covering everything from what a visitor sees on screen to the logic and databases running behind it.

Web development is generally split into two sides:

- **Front-end (Client-side):** everything a visitor directly sees and interacts with in the browser, built using HTML, CSS, and JavaScript.
- **Back-end (Server-side):** the logic, database, and server that process requests behind the scenes, built using languages such as PHP, and covered in more depth in Grade 12's Web Technology II.

This unit focuses on the front-end side, specifically HTML for structuring content and CSS for styling it.

6.2 WEB BROWSERS AND SEARCH ENGINES

Web Browser: application software used to request, receive, and display web pages, by interpreting their HTML, CSS, and JavaScript and rendering the result on screen. Examples:

Google Chrome, Mozilla Firefox, Microsoft Edge, Safari.

Search Engine: a web-based tool that indexes pages across the internet and lets a user search for information by typing keywords, then returns a ranked list of matching pages. Examples: Google, Bing, DuckDuckGo.



A browser and a search engine are often confused, but they do different jobs: a search engine helps a user find a web address, while a browser is the program that actually opens and displays that address.

6.3 OVERVIEW OF INTERNET AND WEB TECHNOLOGIES

The **Internet** is a global network of interconnected computer networks that communicate using a common set of rules, called protocols (mainly TCP/IP). The **World Wide Web (WWW)** is just one service that runs on top of the internet — a system of linked documents (web pages) that can be accessed through a browser. Other services, such as email and file transfer, also run over the same internet but are not part of the web itself.

Technology	Role
HTML	Structures the content of a web page (headings, paragraphs, links, images)
CSS	Styles and lays out how that content looks (colors, fonts, spacing)
JavaScript	Adds interactivity and behaviour to a page (covered in Web Technology II)
Server-side languages (e.g. PHP)	Process requests and generate dynamic content on the server
Database (e.g. MySQL)	Stores and retrieves data used by a website

6.4 CONTENT MANAGEMENT SYSTEM (CMS)

Definition: A Content Management System (CMS) is software that allows a user to create, edit, organize, and publish content on a website without needing to write code directly.

Instead of hand-writing HTML for every page, a CMS provides a visual editor, ready-made templates, and a database to store content, which makes it possible for a non-technical person to run and update a website. Common examples include WordPress, Joomla, and Drupal.

- **Content Editor:** a visual, often drag-and-drop interface for writing and formatting posts or pages
- **Themes/Templates:** ready-made designs that control how the site looks
- **Plugins/Extensions:** add-ons that extend the site's functionality, such as contact forms or SEO tools
- **User Management:** lets multiple people manage the same site with different levels of access

6.5 HTML: THE LANGUAGE OF THE WEB

Definition: HTML (Hypertext Markup Language) is the standard markup language used to structure the content of a web page, using a system of elements called tags.

HTML is not a programming language, since it has no logic, calculations, or conditions of its own. It is a markup language: it simply describes what each piece of content is (a heading, a paragraph, a link) so the browser knows how to display it.

6.5.1 Objectives of HTML

- To structure and organize the content of a web page in a way browsers can consistently understand
- To create hyperlinks that connect one page or document to another
- To embed media such as images, audio, and video directly into a page
- To provide a foundation that CSS and JavaScript can then style and control

6.5.2 Structure of an HTML Document

Every HTML page follows the same basic skeleton:

```
<!DOCTYPE html>
<html>
  <head>
    <title>My First Web Page</title>
  </head>
  <body>
    <h1>Welcome to My Website</h1>
    <p>This is a paragraph of text.</p>
  </body>
</html>
```

- **<!DOCTYPE html>**: tells the browser this document uses HTML5, so it knows how to interpret it
- **<html>**: the root element that wraps the entire page
- **<head>**: holds information about the page that isn't displayed directly, such as the title and links to CSS files
- **<title>**: sets the text shown on the browser's tab
- **<body>**: holds everything that is actually visible on the page

6.5.3 Publishing and Hosting

Once an HTML file is written, it works fine when opened directly in a browser from a computer, but it isn't accessible to anyone else yet. To make it publicly available, the files need to be uploaded to a web server through a hosting service, and given an address (domain name) that people can type into a browser. This process is covered in more depth in [Section 6.5.20](#).

6.5.4 HTML Tags vs. Attributes

	Tag	Attribute
--	-----	-----------

What it is	A keyword enclosed in angle brackets that defines an HTML element	Extra information added inside the opening tag to modify or describe the element
Syntax	<code><p>...</p></code>	<code>name="value"</code> , written inside the tag
Example	<code></code> , <code><a></code> , <code><p></code>	<code>src="pic.jpg"</code> , <code>href="page.html"</code>

Most tags come in pairs, an opening tag and a matching closing tag (`<p>...</p>`), though a few, called empty or self-closing tags, don't wrap any content, such as `<br>` and `<img>`.

6.5.5 Basic Tags: HTML, HEAD, TITLE, BODY

The `<body>` tag can take a few older attributes to control the page's overall appearance:

```
<body bgcolor="lightyellow" text="darkblue" background="paper.jpg">
  <bgsound src="tune.mid" loop="-1">
  ...
</body>
```

- **bgcolor:** sets the background color of the page
- **text:** sets the fore color (the default color of body text)
- **background:** sets an image to tile across the page background
- **<bgsound>:** plays a background sound automatically (Internet Explorer only, and long deprecated)

Note: all four are legacy HTML4 attributes. Modern HTML5 pages set colors and backgrounds through CSS instead (`background-color`, `color`, `background-image`), and `<bgsound>` has been replaced entirely by the `<audio>` element. They're still part of the CDC syllabus and worth knowing for the exam, but they won't show up in a website built today.

6.5.6 Heading Tag (H1 to H6) and Attributes

```
<h1>This is Heading 1 (Largest)</h1>
<h2>This is Heading 2</h2>
<h3 align="center">This is Heading 3, centered</h3>
<h6>This is Heading 6 (Smallest)</h6>
```

HTML provides six levels of headings, from `<h1>` (the largest and most important) to `<h6>` (the smallest). The `align` attribute could position a heading (left, center, right), but like the body attributes above, it is deprecated in HTML5 in favour of CSS's `text-align` property.

6.5.7 FONT Tag and Attributes

```
<font size="5" color="red" face="Arial">Styled text</font>
```

```
<basefont size="4">
```

```
<small>Smaller text</small>
```

```
<big>Bigger text</big>
```

- ****: sets the size (on a scale of 1 to 7), color, and typeface of the text it wraps
- **<basefont>**: set a default font size for the rest of the document
- **<small>** / **<big>**: shrink or enlarge text by one step relative to its surrounding text

Note: `<font>` and `<basefont>` are also deprecated in HTML5. Font size, color, and family are all handled through CSS today (font-size, color, font-family), but examiners still test these tags since they were core to earlier versions of HTML.

6.5.8 Paragraph Formatting (P)

```
<p>This is the first paragraph. Every browser automatically adds space before and after a paragraph.</p>
```

```
<p align="justify">This is a second, justified paragraph.</p>
```

6.5.9 Break Line (BR)

The `<br>` tag inserts a single line break, moving the following content to a new line without starting a new paragraph. It is a self-closing (empty) tag.

First line of an address

Second line of an address

Kathmandu, Nepal

6.5.10 Comment in HTML

A comment is text that the browser ignores completely, used to leave notes for anyone reading the code.

```
<!-- This is a comment and will not be shown on the page -->
```

```
<p>This text is visible.</p>
```

6.5.11 Formatting Text

Tag	Effect
<code></code>	Bold text
<code><i></code>	Italic text

<u>	Underlined text
<mark>	Highlights text, usually with a yellow background
<sup>	Superscript text, e.g. x²
<sub>	Subscript text, e.g. H₂O
	Emphasized text, rendered in italics by default and read with stress by screen readers
<blockquote>	Indents and sets off a quoted block of text
<pre>	Preformatted text — keeps exact spacing and line breaks as typed

6.5.12 Ordered List (OL)

```
<ol type="A" start="3">
  <li>Third item, labeled C</li>
  <li>Fourth item, labeled D</li>
  <li value="10">Jumps to J</li>
</ol>
```

Each item is wrapped in . The type attribute controls the numbering style: 1 (numbers, default), I (uppercase Roman numerals), i (lowercase Roman numerals), A (uppercase letters), or a (lowercase letters). start sets which number the list begins counting from, and value on an individual jumps the count to a specific number from that point on.

6.5.13 Unordered List (UL) and Definition List

```
<ul type="square">
  <li>HTML</li>
  <li>CSS</li>
</ul>

<dl>
  <dt>HTML</dt>
  <dd>The markup language used to structure a web page</dd>
</dl>
```

An unordered list uses bullet points instead of numbers, styled with the type attribute: disc (filled circle, default), circle (hollow circle), or square. A definition list is a different structure entirely, used for terms and their meanings: <dl> wraps the whole list, <dt> holds the term, and <dd> holds its description.

6.5.14 ADDRESS Tag

The <address> tag marks up contact information for the author or owner of a page or article. Browsers usually render its content in italics by default.

```
<address>

Written by Ram Sharma<br>

Kathmandu, Nepal<br>

Email: ram@example.com

</address>
```

6.5.15 Creating Links: Anchor Tag and Hyperlink

The <a> (anchor) tag creates a hyperlink, a clickable connection from one location to another, using the href attribute to specify the destination.

```
<!-- Link to another HTML document -->

<a href="about.html">About Us</a>

<!-- Link to a place within the SAME document -->

<a href="#section2">Jump to Section 2</a>

<h2 id="section2">Section 2</h2>

<!-- Link to a place in ANOTHER document -->

<a href="page2.html#top">Go to top of Page 2</a>
```

A link to a place inside the same page works by pairing href="#idName" with an element somewhere on the page that carries a matching id="idName". The same technique, with the file name added in front of the #, links to a specific spot in a different document.

6.5.16 Tables: TH, TR and TD

```
<table border="1">
  <tr>
    <th>Name</th>
    <th>Marks</th>
  </tr>
  <tr>
    <td>Sita</td>
    <td>85</td>
  </tr>
  <tr>
    <td>Hari</td>
    <td>78</td>
  </tr>
</table>
```

```
</tr>
</table>
```

- **<table>**: wraps the entire table
- **<tr>** (table row): defines one row
- **<th>** (table header): defines a header cell, usually shown bold and centered
- **<td>** (table data): defines a normal data cell

Cells can also merge across rows or columns using `rowspan` and `colspan` attributes on a `<td>` or `<th>`.

6.5.17 Forms

```
<form action="/submit" method="post">
  Name: <input type="text" name="username"><br>
  Gender: <input type="radio" name="gender" value="M"> Male
         <input type="radio" name="gender" value="F"> Female<br>
  Interests: <input type="checkbox" name="interest" value="sports"> Sports
            <input type="checkbox" name="interest" value="music"> Music<br>
  Message: <br>
  <textarea rows="4" cols="30"></textarea><br>
  <button type="submit">Submit</button>
</form>
```

A form collects input from a user and sends it to a server for processing (a topic covered fully in Grade 12). Common input types include text (a single-line text box), radio (pick exactly one from a group sharing the same name), checkbox (pick any number of options), `<textarea>` (a multi-line text box), and `<button>` (a clickable control, commonly used to submit the form).

6.5.18 Introduction to HTML5 Elements: Audio, Embed, Source, Track, Video

```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <track src="subs.vtt" kind="subtitles" srclang="en">
  Your browser does not support the video tag.
</video>
```

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

```
<embed src="document.pdf" width="400" height="300">
```

- **<video>** / **<audio>**: embed playable media directly on a page, with the `controls` attribute adding play/pause/volume controls

- **<source>**: specifies one possible file (and format) for the browser to use inside a video or audio element
- **<track>**: adds subtitles or captions to a video
- **<embed>**: embeds external content, such as a PDF or a plugin-based file, into the page

6.5.19 HTML5 Graphics Using Canvas and SVG

```
<!-- Canvas: drawn using JavaScript -->
<canvas id="myCanvas" width="200" height="100"></canvas>

<!-- SVG: shapes described directly in the HTML markup -->
<svg width="120" height="120">
  <circle cx="60" cy="60" r="50" fill="skyblue" />
</svg>
```

<canvas> provides a blank drawing surface that graphics are drawn onto using JavaScript, pixel by pixel, which makes it well suited to animations, games, and charts. **<svg>** (Scalable Vector Graphics) describes shapes directly as markup (circles, paths, lines), which stay sharp at any zoom level since they're drawn mathematically rather than as pixels.

6.5.20 Concept of Domain Name and Web Hosting

Domain Name: a human-readable address (such as computerkite.com) that points to a website, standing in for the numeric IP address of the server that actually hosts it. The Domain Name System (DNS) is what translates a domain name into its corresponding IP address behind the scenes.

Web Hosting: a service that stores a website's files on a server and keeps that server connected to the internet, so the site can be accessed at any time from anywhere.

A domain name and a hosting plan are two separate things that work together: the domain is the address people type, and hosting is where the actual files live. Common types of hosting include shared hosting (many websites on one server, cheapest), VPS hosting (a partitioned slice of a server with more control), and dedicated hosting (an entire server for one website, most expensive and powerful).

6.6 CASCADING STYLE SHEETS (CSS)

Definition: CSS (Cascading Style Sheets) is a stylesheet language used to control the visual presentation of HTML content, such as colors, fonts, spacing, and layout, separately from its structure.

A CSS rule is made up of a selector (which element to style), and a declaration block of property-value pairs (how to style it):

```
selector {
  property: value;
```

```

}

/* Example */
h1 {
    color: navy;
    font-size: 28px;
}

```

CSS can be added to an HTML page in three different ways.

Inline CSS

Applied directly to a single element using its style attribute. It's quick but doesn't scale well, since a style has to be repeated on every element that needs it.

```
<p style="color: red; font-size: 18px;">This paragraph is styled inline.</p>
```

Embedded (Internal) CSS

Written inside a <style> tag in the document's <head>, and applies to the whole page it's written in.

```

<head>
  <style>
    p { color: green; }
    h1 { text-align: center; }
  </style>
</head>

```

External CSS

Written in a completely separate .css file and linked to the HTML document using a <link> tag. This is the most common approach for real websites, since one stylesheet file can style every page of a site at once.

```

<!-- Inside the HTML file's <head> -->
<link rel="stylesheet" href="style.css">

/* Inside style.css */
body {

```

```
font-family: Arial, sans-serif;
background-color: #f5f5f5;
}
```

Type	Where It's Written	Scope	Best For
Inline	Inside the element's own style attribute	Just that one element	A quick, one-off style tweak
Embedded	Inside a <style> tag in the head	The whole current page	Styles unique to a single page
External	A separate .css file	Every page that links to it	Consistent styling across an entire website

When styles conflict, CSS follows a cascade: inline styles generally override embedded styles, which override external styles, which is part of where the "cascading" in the name comes from.

SOLVED CODE EXAMPLES

Seeing several tags used together in one working page makes the individual definitions click faster than reading them in isolation.

1. A Simple Personal Profile Page

```
<!DOCTYPE html>
<html>
<head>
  <title>My Profile</title>
</head>
<body bgcolor="#f0f8ff">
  <h1>About Me</h1>
  <p>My name is <b>Sita Gurung</b>, and I study <i>Computer Science</i> in Grade
11.</p>
  <h2>My Hobbies</h2>
  <ul type="square">
    <li>Reading</li>
    <li>Coding</li>
    <li>Photography</li>
  </ul>
</body>
</html>
```

2. A Result Table with a Link Back Home

```
<h2 id="results">Exam Results</h2>
```

```

<table border="1">
  <tr>
    <th>Subject</th>
    <th>Marks</th>
  </tr>
  <tr>
    <td>Computer Science</td>
    <td>88</td>
  </tr>
  <tr>
    <td>Mathematics</td>
    <td>92</td>
  </tr>
</table>
<a href="#top">Back to Top</a>

```

3. A Simple Contact Form Styled with Internal CSS

```

<head>
  <style>
    form { background-color: #eef6ff; padding: 15px; }
    input, textarea { margin-bottom: 8px; }
  </style>
</head>
<body>
  <form action="/submit" method="post">
    Name: <input type="text" name="name"><br>
    Message:<br>
    <textarea rows="3" cols="25"></textarea><br>
    <button type="submit">Send</button>
  </form>
</body>

```