

# Chapter 6 Web Technology

## 6.1 Introduction: Web Development Introduction

Web development refers to creating websites and web applications. It involves coding, designing, and making websites interactive. Web developers use different languages and tools to build a site that is user-friendly and functional.

### What is the Internet?

The **Internet** is a global network that connects millions of computers and devices, allowing them to communicate and share information. It is based on the TCP/IP protocol and enables users to access websites, send emails, stream media, and more.

---

### Popular Services of the Internet

The Internet offers a wide range of services, including:

1. **World Wide Web (WWW):** Access to websites and online content.
  2. **Email:** Sending and receiving messages via platforms like Gmail and Outlook.
  3. **Social Media:** Platforms like Facebook, Twitter, and Instagram for communication.
  4. **Cloud Storage:** Online storage services like Google Drive and Dropbox.
  5. **Streaming Services:** Watching videos (YouTube, Netflix) and listening to music (Spotify).
  6. **E-commerce:** Online shopping platforms like Amazon and eBay.
  7. **Online Banking:** Internet banking and digital transactions.
  8. **Search Engines:** Finding information through Google, Bing, and Yahoo.
  9. **Online Education:** Learning through platforms like Coursera, Udemy, and Khan Academy.
  10. **VoIP & Messaging:** Online calls and chats via WhatsApp, Zoom, and Skype.
- 

### Uses of the Internet

The Internet is used in various aspects of life, including:

#### 1. Communication

- Email, instant messaging, and social networking.
- Video conferencing (Zoom, Microsoft Teams).

#### 2. Education & Research

- E-learning platforms, digital libraries, and online courses.
- Access to research papers and academic journals.

#### 3. Business & Commerce

- Online shopping, digital marketing, and e-commerce platforms.
- Remote working and freelancing opportunities.

## 4. Entertainment

- Streaming movies, music, and online gaming.
- Watching live events and web series.

## 5. Banking & Finance

- Online banking, mobile payments (PayPal, Google Pay).
- Stock market trading and cryptocurrency transactions.

## 6. Healthcare

- Online consultations, telemedicine, and medical research.
- Booking doctor appointments and accessing health records.

## 7. Government & Public Services

- E-governance, online tax filing, and passport applications.
- Public awareness campaigns and emergency alerts.

---

## Positive & Negative Impacts of the Internet

### ✔ Positive Impacts

1. **Global Connectivity:** Easy communication across the world.
2. **Education & Knowledge Sharing:** Access to free knowledge and online learning.
3. **Economic Growth:** Online businesses, freelancing, and job opportunities.
4. **Convenience & Efficiency:** Quick access to services like shopping, banking, and healthcare.
5. **Entertainment & Socialization:** Engaging in social media, gaming, and digital content.
6. **Health & Wellness:** Online medical consultations and fitness apps.
7. **Innovation & Development:** Advancements in technology and AI.

### ✘ Negative Impacts

1. **Cybersecurity Threats:** Hacking, phishing, and identity theft.
2. **Misinformation & Fake News:** Spread of false information and rumors.
3. **Privacy Issues:** Data breaches and personal information leaks.
4. **Internet Addiction:** Excessive screen time affecting mental and physical health.
5. **Online Scams & Fraud:** Financial scams and fraudulent activities.
6. **Cyberbullying & Harassment:** Online abuse and social media negativity.
7. **Unemployment in Traditional Sectors:** Automation replacing manual jobs.

## 6.2 Web Browsers and Search Engines

A web browser is a software that allows you to access websites, such as Google Chrome, Mozilla Firefox, or Safari.

A search engine, like Google or Bing, helps you find websites by searching the internet using keywords.

## 6.3 Overview of Various Internet & Web Technologies

Web technologies refer to tools and languages used to build websites. This includes HTML, CSS, JavaScript, and more. These technologies help create, design, and enhance websites to make them work effectively.

## 6.4 Content Management System (CMS)

A Content Management System (CMS) is software that allows users to create, manage, and modify content on a website without needing special technical knowledge. Examples include WordPress and Joomla.

## 6.4 HTML: The Language of the Web

HTML (HyperText Markup Language) is the standard language used to create and design webpages. It uses tags to define elements like headings, paragraphs, links, and images on a webpage.

### 6.4.1 Objective of HTML

- Insert a graphic within a web page.
- Create a link within a web page.
- Create a table within a web page.
- Use cascading style sheets.
- Create a web page.
- Publish a web page.

The objective of HTML is to create a structured, readable format for webpages and define how they should appear in a browser.

### 6.4.2 Structure of HTML

HTML consists of elements that define the structure of a webpage. Each webpage has basic tags like `<html>`, `<head>`, and `<body>`, which organize content into a readable layout.

```
<!DOCTYPE html>
<html>
<head>
  <title>My First Webpage</title>
</head>
<body>
  <h1>Welcome to My Website</h1>
  <p>This is a paragraph of text.</p>
  <a href="https://example.com">Visit Example</a>
  
</body>
</html>
```

## Example explain:

1. **<!DOCTYPE>** : represents the document type, and helps browsers to display web pages correctly. It must only appear once, at the top of the page (before any HTML tags). The **<!DOCTYPE>** declaration is not case sensitive. The **<!DOCTYPE>**
2. **<html>**: The root element that wraps the entire HTML content.
3. **<head>**: Contains metadata about the document (e.g., title, character set, styles).
4. **<title>**: Specifies the title of the document, displayed on the browser tab.
5. **<body>**: Contains the content displayed on the webpage.
6. **<h1> to <h6>**: Heading tags, with **<h1>** being the largest and **<h6>** the smallest.
7. **<p>**: Defines a paragraph.
8. **<a href="">**: Creates a hyperlink; the href attribute specifies the URL.
9. **<img src="" alt="">**: Displays an image; src specifies the image source, and alt provides alternative text.

### 6.4.4 HTML Element, Tags vs. Attributes

An HTML element is defined by a start tag, some content, and an end tag. Tags are the basic building blocks of HTML (e.g., **<p>** for a paragraph). **Attributes provide additional information about the tag (e.g., <a href="url"> for creating links). Attributes usually come in name/value pairs like: name="value"**

### 6.4.5 Basic Tags of HTML

Basic tags are used to structure a webpage:

- **<html>**: Defines the webpage.
- **<head>**: Contains metadata like the page title.
- **<title>**: Specifies the title shown in the browser tab.
- **<body>**: Contains the content of the page.

You can set colors, background images, and even add background sounds using specific attributes.

### Html tags are of two types:

1. **Paired tags** : If it has opening and closing tags. **<b> </b>**
2. **Singular tags** : If it has only opening tag. **<br> <hr> <!-- --- >**

### HTML Block and Inline Elements

Every HTML element has a default display value, depending on what type of element it is. The two most common display values are block and inline.

**Block-level Elements:** A block-level element always starts on a new line, and the browsers automatically add some space (a margin) before and after the element. A block-level element always takes up the full width available (stretches out to the left and right as far as it can).hghghgghghghhghgh

## Here are the block-level elements in HTML:

<address> <article> <aside> <blockquote> <canvas> <dd> <div>  
<dl> <dt> <fieldset> <figcaption> <figure> <footer> <form> <h1>  
<h6> <header> <hr> <li> <main> <nav> <noscript> <ol> <p> <pre>  
<section> <table> <tfooter> <ul> <video>

## Inline Elements

An inline element does not start on a new line. An inline element only takes up as much width as necessary.

<a> <abbr> <acronym> <b> <bdo> <big> <br> <cite> <code> <dfn> <button> <em> <i>  
<img> <input> <kbd> <label> <map> <object> <output> <samp> <script> <select> <small>  
<span> <strong> <sub> <sup> <textarea> <time> <tt> <var>

**Note: An inline element cannot contain a block-level element!**

## Meta Tag

It is used to make the web site or its content searchable in WWW or internet. Its offer information about page or description about page or author of page or keyword that is relevant of that page. This information is used for search engine information. The search engine interacts with the meta tag of the HTML page at first.

### 6.4.6 Heading Tag (H1 to H6) and Attributes

Heading tags <h1> to <h6> are used to define headings on a webpage. <h1> is the most important, and <h6> is the least important. The align attribute helps position the heading (left, center, right).

### 6.4.8 Paragraph Formatting (P)

The <p> tag is used to create paragraphs. You can format paragraphs with attributes like alignment and spacing.

### 6.4.9 Break Line (BR)

The <br> tag inserts a line break, which starts text on a new line.

### 6.4.10 Comment in HTML

HTML comments are enclosed within <!-- -->. Comments help developers add notes to the code, which are not visible on the webpage.

#### 6.4.10.1 Images

The HTML `<img>` tag is used to embed an image in a web page.

```

```

### Background image in a html element

```
<p style="background-image: url('img_girl.jpg');">
```

### Background image in a page

```
<style>
```

```
body {
```

```
background-image: url('img_girl.jpg');
```

```
}
```

```
</style>
```

## 6.4.11 Formatting Text (B, I, U, Mark, Sup, Sub, EM, BLOCKQUOTE, PREFORMATTED)

These tags format text in different ways, such as bold (`<b>`), italic (`<i>`), underline (`<u>`), and others like superscript (`<sup>`) or block quotes (`<blockquote>`).

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Text Formatting Example</title>
</head>
<body>

  <p><b>This text is bold.</b></p>
  <p><i>This text is italicized.</i></p>
  <p><u>This text is underlined.</u></p>
  <p><mark>This text is highlighted.</mark></p>
  <p>Math expression: E = mc<sup>2</sup></p>
  <p>Chemical formula: H<sub>2</sub>O</p>
  <p><em>This text is emphasized.</em></p>
```

```
<blockquote>
  "This is a famous quote that is indented and stands out."
</blockquote>

<pre>
This is a block of preformatted text.
Spaces    and    newlines are
preserved exactly as you write them.
</pre>

</body>
</html>
```

### 6.4.12 Ordered List - OL

An ordered list is a numbered list created using the `<ol>` tag. Items in the list are defined by the `<li>` tag. You can control the type of numbering (1, a, i) with the `type` attribute.

```
<h2>Steps to Make a Cup of Tea</h2>

<ol type="A">
  <li>Boil water in a kettle.</li>
  <li>Add tea leaves or a tea bag to a cup.</li>
  <li>Pour the hot water into the cup over the tea.</li>
  <li>Let the tea steep for 3-5 minutes.</li>
  <li>Remove the tea bag or strain the leaves.</li>
  <li>Add milk or sugar (optional).</li>
  <li>Stir the tea and enjoy!</li>
</ol>
```

#### Explanation:

- `type="1"`: Default numeric numbering (1, 2, 3,...).
- `type="A"`: Uppercase alphabetic numbering (A, B, C,...).
- `type="a"`: Lowercase alphabetic numbering (a, b, c,...).
- `type="I"`: Uppercase Roman numerals (I, II, III,...).
- `type="i"`: Lowercase Roman numerals (i, ii, iii,...).

### 6.4.13 Unordered List - UL

An unordered list is a bulleted list, created using the `<ul>` tag. Each item is defined with the `<li>` tag. You can choose different bullet types, such as a disc, circle, or square.

```
<h2>Fruits You Can Buy</h2>

<ul>
  <li>Apple</li>
  <li>Banana</li>
  <li>Orange</li>
  <li>Grapes</li>
  <li>Cherry</li>
</ul>
```

### Explanation:

- The `<ul>` tag creates an unordered list.
- The `<li>` tags are used to define each item within the list.
- By default, each item in an unordered list will be preceded by a bullet point.
- **type="square"** – start with Solid square
- **type="circle"** – start with hollow circle
- **type="disc"** – start with solid circle

### Customizing the Bullet Style:

You can customize the bullet style of an unordered list using the `list-style-type` property in CSS. Common styles include:

- **disc**: Default filled circle bullet.
- **circle**: Hollow circle bullet.
- **square**: Square bullet.

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Unordered List with Custom Bullets</title>
  <style>
    ul {
      list-style-type: square; /* Change bullet to square */
    }
  </style>
</head>
```

## 6.4.14 Address Tag

The <address> tag is used to define contact information, usually displayed in italics.

```
<address>
  ABC Company<br>
  123 Main Street, Suite 500<br>
  Springfield, USA<br>
  Phone: (123) 456-7890<br>
  Email: <a href="mailto:contact@abccompany.com">contact@abccompany.com</a>
</address>
```

## Creating Links

Links connect different parts of a webpage or separate pages. You can link to other pages within the same site or to external sites. The <a> tag is used to create links. <a href="url">link text</a>

### 1. Jump/intra/internal Link

```
<a href="#contact">Go to Contact Section</a>
```

```
<!-- Somewhere in the same page -->
```

```
<h2 id="contact">Contact Us</h2>
```

### 2. Local/ inter link : linking different pages

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

### 3. External link : linking with different website

```
<a href="https://www.example.com" target="_self">Visit Example</a>
```

## Target attribute :

**\_self** : open in same document as it was clicked

**\_blank** : open document in new tab

**\_parent** : open document in the parent frame

**\_top** : open document in the full body of window

## Absolute URL VS Relative URL

An absolute URL includes the full web address, starting with the protocol (`http://` or `https://`) and followed by the domain name and the file path.

```
<a href="https://www.example.com/about.html">Visit About Page (Absolute URL)</a>
```

A relative URL refers to the location of a resource relative to the current page or file. It does not include the domain name.

```
<a href="about.html">Visit About Page (Relative URL)</a>
```

## Components:

1. **Protocol:** `https://` → Specifies the communication method (HyperText Transfer Protocol Secure).
2. **Subdomain:** `www.` → A subdivision of the main domain, commonly used but not always required.
3. **Domain Name:** `example.com` → The primary address of the website.
  - **Second-Level Domain (SLD):** `example` → The unique part of the domain.
  - **Top-Level Domain (TLD):** `.com` → The suffix that indicates the type of domain (`.com`, `.org`, `.net`, etc.).

## Practical :

**Use image as a link ; Link to an email address; Button as a link**

## 6.4.15 Tables: Creating Tables using TH, TR, and TD Tags

Tables in HTML are created using the following tags:

- `<table>`: This tag is used to define the table.
- `<tr>`: This tag is used to define a row in the table.
- `<th>`: This tag is used to define a header cell in the table. The text in a `<th>` element is typically bold and centered by default.
- `<td>`: This tag is used to define a data cell in the table, where the actual data is displayed.

```
<h2>Student Information</h2>
```

```
<table border="1">
```

```
<tr>
```

```
<th>Name</th>
```

```
<th>Age</th>
```

```
<th>Grade</th>
```

```
</tr>
```

```
<tr>
```

```
<td>Alice</td>
```

```
<td>16</td>
```

```
<td>A</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Bob</td>
```

```
<td>17</td>
```

```
<td>B</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Charlie</td>
```

```
<td>16</td>
```

```
<td>A</td>
```

```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

#### 6.4.16 Forms: Creating Form using Textbox, Radio, Checkbox, Text Area, Button

Forms allow users to enter data. HTML provides tags like <input>, <textarea>, <button>, and others to create different types of form elements.

```
<h2>Contact Us</h2>
```

```
<form action="/submit_form" method="post">  
  <!-- Textbox for name -->  
  <label for="name">Name:</label>  
  <input type="text" id="name" name="name" required><br><br>  
  
  <!-- Radio buttons for gender -->  
  <label>Gender:</label><br>  
  <input type="radio" id="male" name="gender" value="male">  
  <label for="male">Male</label><br>  
  <input type="radio" id="female" name="gender" value="female">  
  <label for="female">Female</label><br><br>
```

```
<!-- Checkboxes for hobbies -->  
<label>Hobbies:</label><br>  
<input type="checkbox" id="reading" name="hobbies" value="Reading">  
<label for="reading">Reading</label><br>  
<input type="checkbox" id="sports" name="hobbies" value="Sports">  
<label for="sports">Sports</label><br>  
<input type="checkbox" id="traveling" name="hobbies" value="Traveling">  
<label for="traveling">Traveling</label><br><br>  
  
<!-- Textarea for message -->  
<label for="message">Message:</label><br>  
<textarea id="message" name="message" rows="4" cols="50"></textarea><br><br>  
  
<!-- Submit Button -->  
<button type="submit">Submit</button>
```

## Explanation:

- **<form>**: This tag defines the form. The `action` attribute specifies where the form data should be sent when the form is submitted, and the `method` attribute specifies the HTTP method (GET or POST).
- **<input type="text">**: This creates a text input field, where users can type their name.
- **<input type="radio">**: These create radio buttons, allowing the user to select one option from a set. Here, users can select their gender.
- **<input type="checkbox">**: These create checkboxes, allowing the user to select multiple options. In this case, users can select their hobbies.
- **<textarea>**: This creates a multi-line text input field, allowing users to type a longer message.
- **<button type="submit">**: This creates a button that, when clicked, will submit the form.

### 6.4.17 Introduction to HTML 5 Elements including Audio, Embed, Source, Track, and Video

HTML5 introduced new tags for embedding multimedia content such as `<audio>`, `<video>`, and `<source>`, allowing users to add audio and video to webpages.

```
<h2>Background Music</h2>
<audio controls>
  <source src="background-music.mp3" type="audio/mp3">
  Your browser does not support the audio element.
</audio>
```

```
<h2>Sample Video</h2>
<video controls width="600">
  <source src="sample-video.mp4" type="video/mp4">
  <source src="sample-video.ogv" type="video/ogv">
  <p>Your browser does not support the video tag.</p>
</video>
```

```
<!-- Track Example (Subtitle for Video) -->
<h2>Track Example</h2>
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <track src="subtitles-en.vtt" kind="subtitles" srclang="en" label="English">
  Your browser does not support the video element.
</video>
```

```
<h2>Embedded Content</h2>
<embed src="https://www.example.com/somefile.swf" width="600" height="400">
```

## Summary of HTML5 Elements:

- **<audio>**: Embeds audio files with play, pause, and volume controls.
- **<video>**: Embeds video files with controls for playback.
- **<source>**: Specifies multiple sources for **<audio>** and **<video>** elements.
- **<track>**: Adds text tracks (subtitles, captions) to videos or audio.
- **<embed>**: Embeds external content like Flash, interactive elements, or media.

## HTML Iframe

**An HTML iframe is used to display a web page within a web page**

```
<h1>Watch this YouTube Video:</h1>
<iframe
  width="560"
  height="315"
  src="https://www.youtube.com/embed/dQw4w9WgXcQ"
  title="YouTube video player"
  frameborder="0"
  allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope;
  allowfullscreen>
</iframe>
```

## 6.4.18 HTML 5 Graphics using Canvas and SVG Tags

HTML5 includes the `<canvas>` tag for drawing graphics and the `<svg>` tag for creating scalable vector graphics directly in the browser.

```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
```

```
</canvas>
```

```
<svg width="100" height="100">
```

```
<circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow" />
```

```
</svg>
```

```
<h2>HTML5 Canvas</h2> Copy code
<canvas id="myCanvas" width="400" height="300" style="border:1px solid #000;"></can

<script>
  // Access the canvas element
  const canvas = document.getElementById("myCanvas");
  const ctx = canvas.getContext("2d");

  // Draw a rectangle
  ctx.fillStyle = "blue";
  ctx.fillRect(50, 50, 200, 100);

  // Draw a circle
  ctx.beginPath();
  ctx.arc(150, 200, 50, 0, Math.PI * 2);
  ctx.fillStyle = "green";
  ctx.fill();
  ctx.closePath();
```

```

    // Draw text
    ctx.fillStyle = "black";
    ctx.font = "20px Arial";
    ctx.fillText("Hello Canvas!", 100, 250);
</script>

</body>
</html>

```

```

<h2>HTML5 SVG</h2>
<svg width="400" height="300" style="border:1px solid #000;">
  <!-- Draw a rectangle -->
  <rect x="50" y="50" width="200" height="100" fill="blue" />

  <!-- Draw a circle -->
  <circle cx="150" cy="200" r="50" fill="green" />

  <!-- Draw text -->
  <text x="100" y="250" font-family="Arial" font-size="20" fill="black">Hello SVG
</svg>

```

### Explanation:

- `<svg>`: Creates a container for vector graphics.
- `<rect>`: Defines a rectangle with position (x, y) and size (width, height).
- `<circle>`: Defines a circle with center (cx, cy) and radius (r).
- `<text>`: Adds text with specified font and color.

## 6.5 Cascading Style Sheets (CSS)

CSS is used to style and layout web pages. It controls the color, fonts, and spacing of elements on a webpage.

### 6.5.1 Introduction to Cascading Style Sheets (CSS)

CSS is a language used to design the look and feel of a webpage. It allows you to apply styles like fonts, colors, and layout to the HTML elements.

## 6.5.2 Inline CSS

Inline CSS is used to apply styles directly to an HTML element using the `style` attribute. This style is applied only to that specific element.

- Example:

```
html

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

## 6.5.3 Embedded CSS / Internal CSS

Embedded CSS is written within a `<style>` tag in the head section of an HTML document. It applies to all elements in the document.

```
<style>
  h1 {
    color: green;
  }
</style>
```

## 6.5.4 External CSS

External CSS is stored in a separate file with a `.css` extension and linked to the HTML document. It is used to apply styles to multiple webpages in a website.

- Example: **HTML File:**

```
html

<link rel="stylesheet" href="styles.css">
```

### CSS File (styles.css):

```
css

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

## Key Features of CSS

1. **Selectors:**
  - Used to target specific HTML elements.
  - Examples:
    - Universal Selector (\*): Targets all elements.
    - Element Selector (p): Targets <p> elements.
    - Class Selector (.classname): Targets elements with a specific class.
    - ID Selector (#idname): Targets an element with a specific ID.
2. **Box Model:**
  - CSS treats every HTML element as a rectangular box consisting of:
    - **Content:** The actual content of the box.
    - **Padding:** Space between the content and the border.
    - **Border:** A line surrounding the padding (or content if padding is absent).
    - **Margin:** Space between the border and the next element.
3. **Positioning and Layout:**
  - CSS allows positioning elements on the webpage using properties like static, relative, absolute, and fixed.

### Element of CSS

“Background-color : blue”;

“Color: blue;” => text color

Font-size : 300%;

Text-align : center;

Border : 2px solid blue;

### 6.4.3 Published and Hosting

Publishing refers to making a website available to the public. Web hosting is a service that stores website files and makes them accessible on the internet.

#### Publishing a Website

Publishing refers to the process of making a website visible and accessible to the public on the internet. Once you’ve designed and created your website using HTML, CSS, and other technologies, you need to upload the website files (such as HTML files, images, videos, and CSS files) to a server.

To publish a website, you need to:

1. **Build the Website:** First, create the website using code (HTML, CSS, JavaScript, etc.).
2. **Choose a Domain Name:** A domain name is like the address of your website on the internet. It could be something like www.example.com.
3. **Select a Web Hosting Service:** A hosting service provides the space to store your website files and makes them accessible online.
4. **Upload the Files:** You upload the website files (like HTML, CSS, and image files) to the web host's server. This can be done using tools like File Transfer Protocol (FTP) or web hosting control panels.

Once the website files are uploaded and your domain name is pointed to the hosting server, your website is **published** and can be accessed by anyone using your domain name.

## Web Hosting

Web hosting is a service that allows individuals and organizations to store their website files on a server. A **web server** is a powerful computer that serves your website files to users when they visit your website. Hosting companies provide the physical space on their servers to store your website and make it accessible over the internet.

There are several types of web hosting options:

1. **Shared Hosting:**
  - This is the most affordable option.
  - Your website shares server space with other websites. While it is cost-effective, it can affect your website's performance if the server is overloaded with traffic.
2. **VPS (Virtual Private Server) Hosting:**
  - A VPS provides more resources and privacy than shared hosting.
  - You still share a server, but the resources are divided into virtual partitions, allowing more control and better performance.
3. **Dedicated Hosting:**
  - With dedicated hosting, your website gets its own server, which provides maximum control, performance, and security.
  - It is more expensive than shared and VPS hosting.
4. **Cloud Hosting:**
  - Cloud hosting uses multiple servers that work together to host your website. It's highly scalable and reliable.
  - This option allows websites to easily adjust resources depending on traffic needs.
5. **Managed Hosting:**
  - In managed hosting, the web host manages the technical aspects like server maintenance, security, and updates.
  - This is ideal for users who want to focus on website content instead of managing hosting settings.

## Domain Name and DNS (Domain Name System)

- **Domain Name:** A domain name is a unique address for your website. It could be something like `www.yoursite.com`. You need to purchase a domain name through a domain registrar and link it to your hosting server.
- **DNS (Domain Name System):** DNS translates your domain name (e.g., `www.yoursite.com`) into an IP address (e.g., `192.168.1.1`) that computers can understand. When someone types in your domain name, the DNS system helps direct them to the correct server where your website is stored.

## Steps in Web Hosting & Publishing

1. **Register a Domain:** You first need to choose and register a domain name through domain registrars like GoDaddy, Namecheap, or Google Domains.
2. **Choose a Web Hosting Provider:** Select a hosting provider based on your needs. Some popular hosting services are Bluehost, HostGator, SiteGround, and DigitalOcean.
3. **Upload Your Website Files:** After you have a hosting account, you can upload your website files using FTP clients like FileZilla or directly through the hosting provider's control panel.

4. **Configure Domain Name:** Set up DNS settings to link your domain name with your web hosting server. This typically involves pointing your domain's nameservers to the hosting provider's nameservers.
5. **Access the Website:** Once uploaded, your website is live and can be accessed by anyone using your domain name.

### 6.4.19 Concept of Domain Name and Web Hosting

A domain name is the address of a website on the internet, like "google.com." Web hosting is where the website's files are stored and made available on the internet.

#### What is a Domain Name?

A **domain name** is the human-readable address of a website, like `www.example.com`. It replaces the need for remembering numeric IP addresses (e.g., 192.168.1.1) used by computers to locate servers on the internet.

#### Key Components of a Domain Name:

1. **Top-Level Domain (TLD):**
  - The extension at the end of a domain name (e.g., .com, .org, .edu, .net).
  - Common TLDs: .com for commercial websites, .org for organizations, .gov for government sites, etc.
2. **Second-Level Domain (SLD):**
  - The name chosen by the website owner (e.g., example in `www.example.com`).
3. **Subdomains (Optional):**
  - Additional prefixes to organize a website, such as `blog.example.com` or `shop.example.com`.

#### How Domain Names Work:

1. A user types a domain name into a browser (e.g., `www.google.com`).
  2. The browser sends a request to a **Domain Name System (DNS)** server.
  3. The DNS server translates the domain name into its associated IP address.
  4. The browser uses the IP address to retrieve the website's content from the hosting server.
- 

#### What is Web Hosting?

**Web hosting** is a service that stores a website's files (HTML, images, videos, etc.) on a server and makes them accessible on the internet. A web hosting provider allocates space on a server and ensures the site is available 24/7.

#### Types of Web Hosting:

1. **Shared Hosting:**
  - Multiple websites share the same server resources.
  - Cost-effective but may have slower performance.
2. **Virtual Private Server (VPS) Hosting:**
  - A single server is divided into virtual sections, providing more dedicated resources.
  - Suitable for medium-sized websites.
3. **Dedicated Hosting:**

- A website has an entire server to itself.
  - Offers high performance and control but is expensive.
  - 4. **Cloud Hosting:**
    - Uses multiple servers to host a website, ensuring scalability and uptime.
    - Suitable for large websites with variable traffic.
  - 5. **Managed Hosting:**
    - The hosting provider manages the server, including updates, backups, and security.
    - Ideal for businesses without technical expertise.
- 

## How Domain Names and Web Hosting Work Together

To make a website live:

1. **Register a Domain Name:**
    - You purchase a domain name from a domain registrar (e.g., GoDaddy, Namecheap).
  2. **Choose a Hosting Provider:**
    - Select a hosting plan from a provider (e.g., Bluehost, SiteGround).
  3. **Connect the Domain to the Hosting:**
    - Update the **DNS settings** of your domain to point to the hosting provider's servers.
- 

## Example Process:

1. **Domain Name Purchase:**
    - Buy `www.mywebsite.com` from a domain registrar.
  2. **Web Hosting Setup:**
    - Choose a hosting plan with a provider (e.g., shared hosting with HostGator).
  3. **DNS Configuration:**
    - Configure the domain's **nameservers** (e.g., `ns1.hostgator.com`, `ns2.hostgator.com`) to connect it to your hosting account.
  4. **Website Upload:**
    - Upload your website's files (HTML, CSS, etc.) to the hosting server via tools like **cPanel** or **FTP**.
  5. **Go Live:**
    - The website becomes accessible on the internet using your domain name.
- 

## Why Are Domain Names and Web Hosting Important?

1. **Domain Names:**
  - Provide a unique identity to your website.
  - Make it easier for users to find and remember your site.
2. **Web Hosting:**
  - Ensures your website is available on the internet.
  - Offers necessary tools for website management.