How the Internet Works

The Internet is a global network that connects millions of devices and computers around the world, allowing them to communicate with each other. When you access a website, send an email, or stream a video, you're using the Internet. But how does this massive system work?

1. The Internet as a Network of Networks

- The Internet is not a single entity or network. It's a **network of networks**—a huge system of interconnected **local, regional, and global networks** that communicate using standardized protocols.
- These networks include **internet service providers (ISPs)**, large data centers, and individual devices (like your phone, laptop, or desktop computer).

2. IP Addresses and DNS (Domain Name System)

- Every device connected to the Internet is assigned an **IP address** (Internet Protocol address). This is like a postal address for your device, enabling data to be sent to and from the right locations.
- **IP addresses** are numeric, like `192.168.1.1` (IPv4), but newer systems use **IPv6**, which allows many more addresses due to the growing number of connected devices.
- Since humans find it easier to remember names than numbers, we use **domain names** (like `www.example.com`) instead of IP addresses.
- The **Domain Name System (DNS)** is a system that translates these human-friendly domain names into IP addresses so that computers can route data to the correct location.

3. Protocols: The Language of the Internet

- Communication on the Internet is governed by a set of rules called **protocols**. The most important ones include:
    - **TCP/IP (Transmission Control Protocol/Internet Protocol)**: These are the foundational protocols that define how data is packaged, addressed, transmitted, and received on the Internet.
    - **HTTP/HTTPS (HyperText Transfer Protocol)**: Used to access websites, where HTTPS is the secure version that encrypts data.
    - **FTP (File Transfer Protocol)**: Used for transferring files between computers.
    - **SMTP (Simple Mail Transfer Protocol)**: Used for sending emails.

4. Packets and Routing

- When data (such as a webpage, video, or email) is sent over the Internet, it is broken down into smaller pieces called **packets**. Each packet contains a portion of the data and the necessary information (such as the source and destination IP addresses).

- These packets travel across the Internet, passing through multiple devices (routers) along the way. **Routers** act as traffic directors, making sure packets take the best route to reach their destination.
- The Internet uses a **packet-switching** system, meaning that packets can take different routes to reach the destination and are reassembled once they arrive.

5. ISPs and Backbone Networks

- **Internet Service Providers (ISPs)** connect individual users and businesses to the Internet. For example, when you subscribe to an ISP (like Comcast, AT&T, or another local provider), they provide you with access to the Internet.
- ISPs are connected to larger regional networks and **backbone networks**, which are high-speed, long-distance networks that carry vast amounts of data across countries and continents. These backbone networks are run by major telecommunications companies and data carriers.
- The global Internet infrastructure consists of **undersea cables**, **fiber-optic lines**, and **satellites** that link these backbone networks and enable global communication.

6. Data Centers and Servers

- A **server** is a powerful computer that stores websites, applications, and services. When you access a website, your browser sends a request to a **web server** to retrieve the data.
- Servers are located in **data centers**, which are massive facilities with many computers storing data, websites, and services. Data centers are connected to the Internet backbone to ensure fast, reliable data access.
- Popular services like Google, Facebook, and Netflix have vast data centers worldwide to deliver content quickly to users.

7. How a Webpage Loads (Recap of Web Request)

- When you visit a website (e.g., `www.example.com`), your request is routed through the Internet to the server that hosts the website. The server then sends back the webpage's **HTML, CSS, and JavaScript** files in small packets.
- Your browser receives these packets, reassembles them, and renders the webpage for you to view.

8. Security and Encryption

- Security is a vital aspect of the Internet, and there are several methods used to protect data:
    - **HTTPS**: Ensures that the data sent between your browser and a web server is encrypted, protecting it from hackers.
    - **Firewalls**: Devices or software that monitor and control incoming and outgoing network traffic, protecting networks from unauthorized access.

- **VPN (Virtual Private Network)**: A tool that encrypts your Internet connection, making your activity more private and secure.

9. Cloud Computing

- The Internet also powers **cloud computing**, which allows users to access computing resources (like servers, databases, and storage) over the Internet. Services like Google Drive, AWS, and Microsoft Azure allow businesses and individuals to use remote servers to store and process data.

10. Content Delivery Networks (CDNs)

- To deliver content faster, especially to global users, many websites use **CDNs (Content Delivery Networks)**. A CDN stores copies of web content on servers in different locations worldwide.
- When a user accesses a website, the CDN delivers the content from the server that's geographically closest to the user, reducing load times.

Summary of Key Components:

1. **IP Addresses and DNS**: IP addresses identify devices, while DNS translates domain names into IP addresses.
2. **Protocols**: TCP/IP, HTTP, and others define how data moves across the Internet.
3. **Packets and Routers**: Data is broken into packets and routed across networks.
4. **ISPs and Backbone Networks**: ISPs connect you to larger networks, and backbone networks carry vast amounts of data globally.
5. **Data Centers and Servers**: Servers store and serve web content, while data centers house these servers.
6. **Security**: HTTPS, firewalls, and VPNs keep data safe.
7. **Cloud and CDNs**: Cloud services and CDNs improve the delivery and storage of content.

Real-World Example: Accessing a Website

1. **You enter `www.example.com` into your browser**.
2. **DNS** translates the domain into an IP address (e.g., `192.168.1.1`).
3. Your browser sends a **request** to the server at that IP address.
4. The server responds with the **HTML, CSS, and JavaScript** files.
5. Your browser processes the files and displays the webpage.

6. If the page contains images or other assets, additional requests are made to fetch them.

Basic HTML Tags and Anatomy

HTML (HyperText Markup Language) is the standard language used to create web pages. It consists of various tags that define the structure and content of a webpage.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>HTML5 Web and Cloud Technology</title>
</head>
<body>
    <h1>Hello world, Heading1!</h1>
</body>
</html>
```

1. `<!DOCTYPE html>`

- **What it does**: This is the document type declaration that tells the browser that the document is written in HTML5.
- **Why it matters**: It helps ensure that the webpage is rendered in standards-compliant mode across all browsers.

2. `<html lang="en">`

- **What it does**: This is the root element of an HTML document. The `lang="en"` attribute specifies that the language of the content is English.
- **Why it matters**: Declaring the language is important for accessibility and search engine optimization (SEO). It helps screen readers and search engines understand the content better.

3. `<head>`

- **What it does**: The `<head>` section contains metadata and other non-visible elements related to the document, such as character encoding, viewport settings, and the document's title.
- **Why it matters**: The content in the `<head>` section is crucial for configuring how the browser renders the page and how search engines index it.

4. **`<meta charset="UTF-8">`**

- **What it does**: Specifies the character encoding used by the document, in this case, UTF-8.
- **Why it matters**: UTF-8 is a widely used encoding format that supports virtually every character in the Unicode standard, ensuring that the webpage displays special characters and symbols correctly.

5. **`<meta http-equiv="X-UA-Compatible" content="IE=edge">`**

- **What it does**: This meta tag ensures that the page is rendered in the highest available mode of Internet Explorer (IE), specifically in "Edge mode."
- **Why it matters**: Although IE is an old browser, some sites may still need to ensure backward compatibility. This tag forces IE to use its most recent version, avoiding compatibility issues.

6. **`<meta name="viewport" content="width=device-width, initial-scale=1.0">`**

- **What it does**: Configures how the page should be displayed on mobile devices. `width=device-width` makes the page's width equal to the screen's width, and `initial-scale=1.0` sets the initial zoom level.
- **Why it matters**: It ensures the webpage is responsive and looks good on various devices, especially smartphones and tablets.

7. **`<title>HTML5 Web and Cloud Technology</title>`**

- **What it does**: This sets the title of the webpage that appears in the browser tab.
- **Why it matters**: It's an important factor for SEO and user experience since this is what people will see in their browser tabs and in search engine results.

8. **`<body>`**

- **What it does**: The `<body>` section contains the content that is visible to the user, including text, images, links, and other media.
- **Why it matters**: Anything inside this tag is rendered in the browser as the main content of the page.

9. **`<h1>Hello world, Heading1!</h1>`**

- **What it does**: This is a heading tag, and in this case, it's an `<h1>`, which is the largest and most important heading on the page. It contains the text "Hello world, Heading1!"
- **Why it matters**: Headings are used to define the structure of a webpage. Search engines give more weight to `<h1>` tags because they represent the main topic or heading of the content.

10. `</body>` and `</html>`

- **What they do**: These tags close the `<body>` and `<html>` elements, respectively, marking the end of the document.
- **Why it matters**: Closing these tags properly ensures that the document is well-formed, which is important for browser rendering and maintaining code quality.

HTML5 Tags

The detailed breakdown of each HTML5 tag along with definitions and examples:

1. Metadata Tags

1. **`<title>`**
   - **Definition**: Sets the title of the document, displayed in the browser's title bar or tab.
   - **Example**:

     ```
     <title>My Web Page</title>
     ```

2. **`<base>`**
   - **Definition**: Specifies a base URL for relative URLs in the document.
   - **Example**:

     ```
     <base href="https://www.example.com/">
     ```

3. **`<link>`**
   - **Definition**: Used to link external resources like stylesheets.
   - **Example**:

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

4. **`<meta>`**
   - **Definition**: Provides metadata about the HTML document (like charset, viewport settings).
   - **Example**:

     ```
     <meta charset="UTF-8">
     ```

5. **`<style>`**
   - **Definition**: Contains internal CSS styles.
   - **Example**:

     ```
     <style>
       body { background-color: lightblue; }
     ```

```
</style>
```

6. **`<script>`**
   - **Definition**: Used to include or reference JavaScript code.
   - **Example**:

```
<script src="script.js"></script>
```

7. **`<noscript>`**
   - **Definition**: Provides an alternative content for users with scripts disabled.
   - **Example**:

```
<noscript>Your browser does not support JavaScript!</noscript>
```

2. Content Sectioning Tags

1. **`<header>`**
   - **Definition**: Represents introductory content or a group of navigational links.
   - **Example**:

```
<header>
  <h1>Welcome to My Website</h1>
  <nav>...</nav>
</header>
```

2. **`<nav>`**
   - **Definition**: Contains navigation links.
   - **Example**:

```
<nav>
  <ul>
    <li><a href="#home">Home</a></li>
    <li><a href="#about">About</a></li>
  </ul>
</nav>
```

3. **`<section>`**
   - **Definition**: Represents a thematic grouping of content.
   - **Example**:

```
<section>
  <h2>About Us</h2>
  <p>...</p>
</section>
```

4. **`<article>`**
   - **Definition**: Represents a self-contained composition in a document.
   - **Example**:

```
<article>
  <h2>Article Title</h2>
  <p>...</p>
</article>
```

5. **`<aside>`**
   - **Definition**: Represents content related to the main content (like a sidebar).
   - **Example**:

```
<aside>
  <h2>Related Articles</h2>
  <p>...</p>
</aside>
```

6. **`<h1>` - `<h6>`**
   - **Definition**: Headings that indicate the hierarchy of the content.
   - **Example**:

```
<h1>Main Heading</h1>
<h2>Subheading</h2>
```

7. **`<hgroup>`**
   - **Definition**: Groups a set of `<h1>` to `<h6>` elements.
   - **Example**:

```
<hgroup>
  <h1>Title</h1>
  <h2>Subtitle</h2>
</hgroup>
```

8. **`<footer>`**
   - **Definition**: Represents the footer of a section or page, containing information like authorship.
   - **Example**:

```
<footer>
  <p>&copy; 2024 My Website</p>
</footer>
```

9. **`<address>`**
   - **Definition**: Provides contact information for the author or owner of a document.
   - **Example**:

```
<address>
  Contact us at: <a
href="mailto:info@example.com">info@example.com</a>
</address>
```

3. Text Content Tags

1. **<p>**
   - **Definition**: Represents a paragraph of text.
   - **Example**:

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

2. **<hr>**
   - **Definition**: Represents a thematic break in the content (horizontal rule).
   - **Example**:

```
<hr>
```

3. **<pre>**
   - **Definition**: Represents preformatted text.
   - **Example**:

```
<pre>
  This text is
  preformatted.
</pre>
```

4. **<blockquote>**
   - **Definition**: Represents a section that is quoted from another source.
   - **Example**:

```
<blockquote>
  <p>To be or not to be, that is the question.</p>
</blockquote>
```

5. **<ol>**
   - **Definition**: Represents an ordered list.
   - **Example**:

```
<ol>
  <li>First item</li>
  <li>Second item</li>
</ol>
```

6. **<ul>**
   - **Definition**: Represents an unordered list.
   - **Example**:

```
<ul>
  <li>Item 1</li>
  <li>Item 2</li>
</ul>
```

7. **`<li>`**
    - **Definition**: Represents a list item in an ordered or unordered list.
    - **Example**:

```
<li>List item</li>
```

8. **`<dl>`**
    - **Definition**: Represents a description list.
    - **Example**:

```
<dl>
  <dt>Term</dt>
  <dd>Description</dd>
</dl>
```

9. **`<dt>`**
    - **Definition**: Represents a term in a description list.
    - **Example**:

```
<dt>HTML</dt>
```

10. **`<dd>`**
    - **Definition**: Represents a description or definition of the term in a description list.
    - **Example**:

```
<dd>A markup language for creating web pages.</dd>
```

11. **`<figure>`**
    - **Definition**: Represents self-contained content, like illustrations or diagrams.
    - **Example**:

```
<figure>
  <img src="image.jpg" alt="Description">
  <figcaption>Caption for the image</figcaption>
</figure>
```

12. **`<figcaption>`**
    - **Definition**: Provides a caption for a `<figure>`.
    - **Example**:

```
<figcaption>This is a caption.</figcaption>
```

13. **`<div>`**
    - **Definition**: A generic container for flow content.
    - **Example**:

```
<div>
```

```
  <p>Content goes here.</p>
</div>
```

4. Inline Text Semantics

1. **<a>**
   - **Definition**: Represents a hyperlink.
   - **Example**:

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

2. ****
   - **Definition**: Represents emphasized text.
   - **Example**:

     ```
     <em>Important text</em>
     ```

3. ****
   - **Definition**: Represents strong importance or urgency.
   - **Example**:

     ```
     <strong>Strong text</strong>
     ```

4. **<small>**
   - **Definition**: Represents side comments or small print.
   - **Example**:

     ```
     <small>This is small text.</small>
     ```

5. **<s>**
   - **Definition**: Represents content that is no longer accurate or relevant (strikethrough).
   - **Example**:

     ```
     <s>Strikethrough text</s>
     ```

6. **<cite>**
   - **Definition**: Represents the title of a work (like a book or a movie).
   - **Example**:

     ```
     <cite>The Great Gatsby</cite>
     ```

7. **<q>**
   - **Definition**: Represents a short inline quotation.
   - **Example**:

     ```
     <q>To be or not to be.</q>
     ```

8. **<dfn>**
    - **Definition**: Represents the defining instance of a term.
    - **Example**:

```
<dfn>HTML</dfn> is a markup language.
```

9. **<abbr>**
    - **Definition**: Represents an abbreviation or acronym.
    - **Example**:

```
<abbr title="HyperText Markup Language">HTML</abbr>
```

10. **<data>**
    - **Definition**: Links content with a machine-readable value.
    - **Example**:

```
<data value="2024-09-27">September 27, 2024</data>
```

11. **<time>**
    - **Definition**: Represents a specific period in time.
    - **Example**:

```
<time datetime="2024-09-27">September 27, 2024</time>
```

12. **<code>**
    - **Definition**: Represents a fragment of computer code.
    - **Example**:

```
<code>console.log('Hello, World!');</code>
```

13. **<var>**
    - **Definition**: Represents a variable in a mathematical expression or programming context.
    - **Example**:

```
<var>x</var> = <var>y</var> + 1;
```

14. **<samp>**
    - **Definition**: Represents sample output from a computer program.
    - **Example**:

```
<samp>Error: File not found.</samp>
```

15. **<kbd>**
    - **Definition**: Represents user input from a keyboard.
    - **Example**:

```
<kbd>Ctrl + C</kbd>
```

16. **<sub>**
    - **Definition**: Represents subscripted text.
    - **Example**:

```
H<sub>2</sub>O
```

17. **<sup>**
    - **Definition**: Represents superscripted text.
    - **Example**:

```
x<sup>2</sup>
```

18. **<i>**
    - **Definition**: Represents text in an alternate voice or mood (italic).
    - **Example**:

```
<i>Italic text</i>
```

19. **<b>**
    - **Definition**: Represents text that is stylistically different (bold).
    - **Example**:

```
<b>Bold text</b>
```

20. **<u>**
    - **Definition**: Represents text with an unarticulated, non-textual annotation (underline).
    - **Example**:

```
<u>Underlined text</u>
```

21. **<mark>**
    - **Definition**: Represents text highlighted for reference.
    - **Example**:

```
<mark>Highlighted text</mark>
```

22. **<ruby>**
    - **Definition**: Represents annotations for East Asian languages.
    - **Example**:

```
<ruby>漢字<rt>かんじ</rt></ruby>
```

23. **<rt>**
    - **Definition**: Represents the text of the ruby annotation.
    - **Example**:

```
<ruby>漢字<rt>かんじ</rt></ruby>
```

24. **<rp>**
    - **Definition**: Represents parentheses around ruby annotations.
    - **Example**:

```
<ruby>漢字<rp>(</rp><rt>かんじ</rt><rp>)</rp></ruby>
```

25. **<bdi>**
    - **Definition**: Isolates a span of text that might be formatted in a different direction.
    - **Example**:

```
<bdi>Some text in another direction</bdi>
```

26. **<bdo>**
    - **Definition**: Overrides the current text direction.
    - **Example**:

```
<bdo dir="rtl">Right to Left text</bdo>
```

27. ****
    - **Definition**: A generic inline container for text.
    - **Example**:

```
<span style="color: red;">Red text</span>
```

28. **
**
    - **Definition**: Represents a line break.
    - **Example**:

```
Line 1<br>Line 2
```

29. **<wbr>**
    - **Definition**: Suggests an optional line break opportunity.
    - **Example**:

```
A very long word wbr>extraordinarily
```

5. Embedded Content

1. **<img>**
    - **Definition**: Embeds an image.
    - **Example**:

```
<img src="image.jpg" alt="Description">
```

2. **<iframe>**
   - **Definition**: Embeds another HTML page within the current page.
   - **Example**:

```
<iframe src="https://www.example.com" width="300"
height="200"></iframe>
```

3. **<embed>**
   - **Definition**: Embeds external content like multimedia (e.g., audio, video).
   - **Example**:

```
<embed src="movie.mp4" width="300" height="200">
```

4. **<object>**
   - **Definition**: Represents an external resource, often multimedia.
   - **Example**:

```
<object data="video.mp4" width="300" height="200"></object>
```

5. **<param>**
   - **Definition**: Specifies parameters for an `<object>`.
   - **Example**:

```
<object data="movie.swf" width="300" height="200">
  <param name="autoplay" value="true">
</object>
```

6. **<video>**
   - **Definition**: Embeds a video file.
   - **Example**:

```
<video controls>
  <source src="video.mp4" type="video/mp4">
  Your browser does not support the video tag.
</video>
```

7. **<audio>**
   - **Definition**: Embeds an audio file.
   - **Example**:

```
<audio controls>
  <source src="audio.mp3" type="audio/mpeg">
  Your browser does not support the audio tag.
</audio>
```

8. **<source>**
   - **Definition**: Specifies multiple media resources for `<video>` or `<audio>`.
   - **Example**:

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

9. **`<track>`**
   - **Definition**: Specifies text tracks for `<video>` and `<audio>`.
   - **Example**:

```
<video controls>
  <track kind="subtitles" src="subtitles_en.vtt" srclang="en"
label="English">
</video>
```

10. **`<canvas>`**
    - **Definition**: Represents a drawable region in HTML where you can use JavaScript to render graphics.
    - **Example**:

```
<canvas id="myCanvas" width="200" height="100"></canvas>
<script>
  var canvas = document.getElementById("myCanvas");
  var ctx = canvas.getContext("2d");
  ctx.fillStyle = "#FF0000";
  ctx.fillRect(0, 0, 200, 100);
</script>
```

11. **`<svg>`**
    - **Definition**: Embeds Scalable Vector Graphics.
    - **Example**:

```
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" fill="red" />
</svg>
```

12. **`<math>`**
    - **Definition**: Embeds mathematical expressions using MathML.
    - **Example**:

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
  <msup>
    <mi>x</mi>
    <mn>2</mn>
  </msup>
</math>
```

6. Interactive Elements

1. **`<details>`**

- **Definition**: Represents a disclosure widget from which the user can obtain additional information or controls.
- **Example**:

```
<details>
  <summary>More Info</summary>
  <p>Here is more information.</p>
</details>
```

2. **`<summary>`**
   - **Definition**: Represents a summary or heading for a `<details>` element.
   - **Example**:

```
<details>
  <summary>Details</summary>
  <p>Additional information goes here.</p>
</details>
```

3. **`<dialog>`**
   - **Definition**: Represents a dialog box or other interactive component.
   - **Example**:

```
<dialog open>
  <p>This is a dialog box.</p>
  <button>Close</button>
</dialog>
```

4. **`<menu>`**
   - **Definition**: Represents a list of commands.
   - **Example**:

```
<menu>
  <li><button>Copy</button></li>
  <li><button>Paste</button></li>
</menu>
```

5. **`<menuitem>`**
   - **Definition**: Represents a command that can be invoked from a menu.
   - **Example**:

```
<menu>
  <menuitem label="Copy" onclick="copy()"></menuitem>
</menu>
```

7. Form-Related Elements

1. **`<form>`**
   - **Definition**: Represents a section of a document that contains interactive controls for submitting data.

- **Example**:

```
<form action="/submit" method="post">
  <input type="text" name="name" placeholder="Enter your name">
  <input type="submit" value="Submit">
</form>
```

2. **<input>**
   - **Definition**: Represents an input control.
   - **Example**:

```
<input type="text" name="username">
```

3. **<textarea>**
   - **Definition**: Represents a multi-line plain-text editing control.
   - **Example**:

```
<textarea rows="4" cols="50"></textarea>
```

4. **<button>**
   - **Definition**: Represents a clickable button.
   - **Example**:

```
<button type="button">Click Me!</button>
```

5. **<select>**
   - **Definition**: Represents a control that allows the user to select an option from a dropdown list.
   - **Example**:

```
<select>
  <option value="1">Option 1</option>
  <option value="2">Option 2</option>
</select>
```

6. **<option>**
   - **Definition**: Represents an option in a `<select>` dropdown list.
   - **Example**:

```
<option value="1">First Option</option>
```

7. **<optgroup>**
   - **Definition**: Groups a set of `<option>` elements inside a `<select>`.
   - **Example**:

```
<select>
  <optgroup label="Group 1">
    <option value="1">Option 1</option>
  </optgroup>
```

```
</select>
```

8. **<label>**
   - **Definition**: Represents a caption for an item in a user interface.
   - **Example**:

```
<label for="username">Username:</label>
<input type="text" id="username" name="username">
```

9. **<fieldset>**
   - **Definition**: Groups related elements in a form.
   - **Example**:

```
<fieldset>
  <legend>Personal Information</legend>
  <input type="text" name="name" placeholder="Name">
</fieldset>
```

10. **<legend>**
    - **Definition**: Represents a caption for the `<fieldset>` element.
    - **Example**:

```
<legend>Account Information</legend>
```

11. **<progress>**
    - **Definition**: Represents the completion progress of a task.
    - **Example**:

```
<progress value="70" max="100">70%</progress>
```

12. **<meter>**
    - **Definition**: Represents a scalar measurement within a known range.
    - **Example**:

```
<meter value="0.7">70%</meter>
```

8. Scripting Tags

1. **<script>**
   - **Definition**: Embeds or references JavaScript code.
   - **Example**:

```
<script>
  console.log('Hello, World!');
</script>
```

2. **<noscript>**
   - **Definition**: Provides alternative content for users who have disabled scripts.

- **Example**:

```
<noscript>Your browser does not support JavaScript!</noscript>
```

9. Web Components Tags

1. **`<template>`**
   - **Definition**: Represents a template that holds markup that is not rendered when the page loads.
   - **Example**:

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

2. **`<slot>`**
   - **Definition**: Represents a placeholder inside a web component that can be filled with content from outside.
   - **Example**:

```
<slot></slot>
```

10. Miscellaneous Tags

- **`<canvas>`**
  - **Definition**: Represents a drawable region in HTML where graphics can be rendered using JavaScript.
  - **Example**:

```
<canvas id="myCanvas" width="200" height="200"></canvas>
```

This classification covers most of the standard HTML5 tags, each serving a unique purpose in web development.