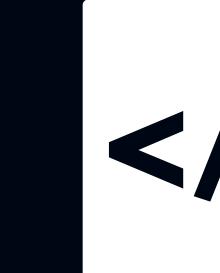




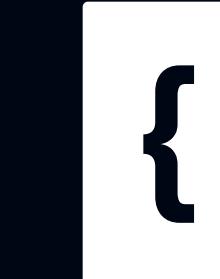
INM120

Introduction to Adaptive Web Design

WEB LANGUAGES



- **HTML:** which stands for Hypertext Markup Language, is responsible for defining the content found on a webpage.



- **Cascading Style Sheets (CSS):** are crucial in determining a webpage's visual aspects and aesthetics, including colour schemes, typography, and animations.



- **JavaScript (JS):** adds interactivity and dynamic functionality to a web page.

HISTORY OF HTML

Tim Berners-Lee, a physicist, worked at CERN, the European Organization for Nuclear Research. A renowned international research centre primarily focused on particle physics.

The need for efficient document sharing among scientists worldwide inspired Tim Berners-Lee's work at CERN. It was here that he developed the concepts that would lead to the creation of the World Wide Web, making CERN an unexpected birthplace of the modern Internet.

In the late 1980s and early 1990s, he introduced ENQUIRE, a precursor to the World Wide Web, emphasizing hypertext linking. Berners-Lee later formalized HTML, created the first web browser and server, and launched the World Wide Web, revolutionizing information sharing.

HTML, or Hypertext Markup Language, is the foundation of the World Wide Web. It was invented to structure and format text-based content on web pages, including images, links, and multimedia elements. HTML has evolved over the years, with HTML5 being a significant milestone in enhancing web capabilities.



HYPERTEXT

"Hypertext" means text that's connected or linked together. Think of it like this:

When reading a book, you can only go from page to page in a straight line. But with hypertext, when reading something on a computer or the internet, you can click on words or phrases to jump to other related information.

It's like having a web of information where everything is connected, and you can explore it in any order you want. So, hypertext allows you to quickly jump between different parts of a document or website, making it a flexible and interactive way to read and learn on the internet.

WHAT DOES HTML LOOK LIKE

HTML consists of special codes enclosed in angle brackets, known as tags. These tags structure and format content on a web page. For example, **<h1>** tags create headings, **<p>** tags define paragraphs, and **<a>** tags make links.



HTML TAGS

HTML tags come in pairs, with an opening and closing tag. The opening tag indicates the beginning of an element, and the closing tag marks the end of that element. The content you want to affect, or style, is placed between these two tags. Let's explore this further:

Opening Tag: The opening tag consists of the tag name enclosed in angle brackets `<>`. It typically includes attributes that provide additional information about the element. Attributes are included within the opening tag and are written as name-value pairs (e.g., `attribute="value"`). Here's an example:

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

In this example, `<a>` is the opening tag for an anchor (hyperlink) element, and `href` is an attribute specifying the link's destination.

HTML TAGS

Closing Tag: The closing tag looks very similar to the opening tag but includes a forward slash (`/`) before the tag name. It indicates where the element's content ends. For example:

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

In this case, `` is the closing tag for the anchor element (`<a>`). Everything between `<a>` and `` is part of the link, including the text "Visit Example.com."

Content: The content of an element goes between the opening and closing tags. It's the part of the web page that you want to structure or style. For instance:

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

In this example, the content between `<a>` and `` is the text "Visit Example.com" which will be displayed as a link on the web page.

HTML TAGS & NESTING

Using opening and closing tags correctly is crucial in HTML because they define the structure of your web page and how its content is presented.

In general, **nesting** refers to placing one HTML element inside another to create a hierarchy of content.

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

In this example, the `<a>` element **is nested inside** the `<p>` element.

Nesting is a valuable concept because it empowers you to build intricate arrangements and designs on a web page, just like how you group elements in software like Photoshop or Illustrator to organize content, apply styles, and establish connections between various components. Nesting in HTML lets you achieve similar outcomes. Ensuring you correctly nest elements is vital to guarantee that your HTML code is structured effectively and performs as expected.

EMPTY ELEMENTS

An **empty element**, also known as a **void element** or **self-closing element**, is an HTML element that doesn't have any content between opening and closing tags because it doesn't require or allow any content. Instead, it usually consists of a **single opening tag** with optional attributes.

They are written in a specific format, typically with a **trailing slash /** just before the closing angle bracket **>**, although this slash is **optional** in HTML5. Here are some common examples of empty elements:

Image Element: The **** element is used to display images on a web page. It doesn't have a closing tag because it doesn't contain any text or content. For example:

```

```

Line Break Element : The **
** element is used to insert a line break or newline within text. It's a self-closing element, and it doesn't have a closing tag. For example:

```
<p>This is some text.<br/>Here's a new line of text.</p>
```

BASIC PAGE OUTLINE

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Your Page Title</title>
  </head>
  <body>
  </body>
</html>
```

Let's break down these components:

<!DOCTYPE html>: This declaration specifies the document type and version of HTML being used. In modern web development, <!DOCTYPE html> is the standard declaration for HTML5.

<html>: The <html> element is the root of the HTML document and contains all other elements. It serves as the container for the entire web page.

<head>: The <head> element contains metadata about the document, such as character encoding and the page title. This section is not visible on the web page itself but provides important information to browsers and search engines.

HEAD VS BODY



<head>: The `<head>` element is like the hidden control centre of an HTML document, much like the human brain's control centre. While it remains unseen on the web page, it stores vital information, such as character encoding and the page title. This hidden data is crucial for browsers and search engines. It is similar to how the brain's inner workings are essential for regulating bodily functions, even though they're not visible to the outside world.

<body>: The `<body>` element contains the visible content of the web page. This is where you place text, images, links, and other elements that users will see and interact with.

Within the `<body>` element, you can structure your content using various HTML elements such as headings (`<h1>`), paragraphs (`<p>`), lists (``, ``), links (`<a>`), images (``), and more.

METADATA

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

As mentioned before the `<head>` element contains **metadata** about the document, such as character encoding and the page **title**.

<meta />: Meta tags are HTML elements used to provide metadata or information about a web page.

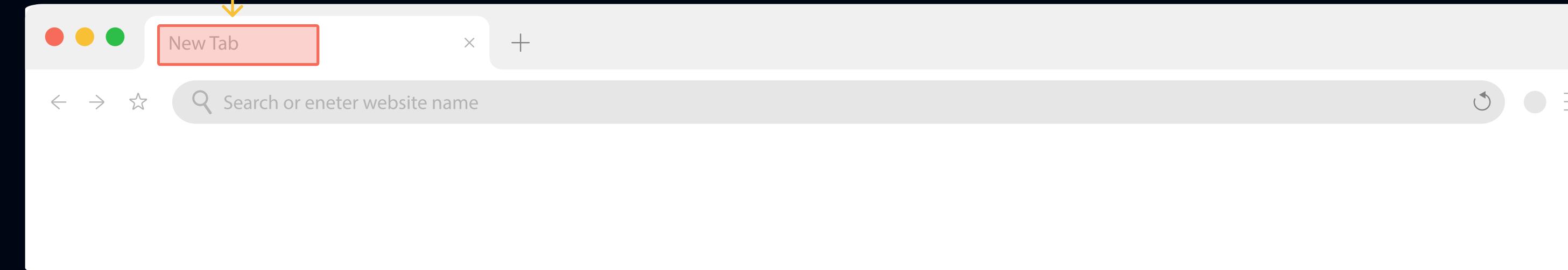
Charset Meta Tag: `<meta charset="UTF-8">` declares the character encoding used in the document. It ensures that special characters and symbols are displayed correctly.

Viewport Meta Tag: `<meta name="viewport" content="width=device-width, initial-scale=1.0">` is often used in responsive web design to control how the webpage is displayed on different devices and screen sizes.

METADATA TITLE

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

Title Meta Tag: <title>Your Page Title</title> defines the title of the web page, which appears in the browser's title bar or tab. It's also used by search engines as the page's title in search results.



OTHER HTML TAGS

HEADING

HEADING

HEADING

HEADING

HEADING

HEADING

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur non magna nec eros imperdiet efficitur. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia curae; Duis eleifend, leo dignissim bibendum porttitor, magna tortor lobortis leo, sed consequat odio velit eu est.

Headings: <h1>, <h2>, <h3>, ... <h6>
with <h1> being the highest level and
<h6> the lowest.

Paragraph: <p>: Represents a paragraph of text.

OTHER HTML TAGS

Anchor:

`<a>` Creates hyperlinks to other web pages or resources. For example:

`<p>Cats believe they're masters of disguise, hiding behind a single blade of grass and thinking, 'You can't see me!<p>`

it will appear on screen like:

Cats believe they're masters of disguise, hiding behind a single blade of grass and thinking, 'You can't see me!'

Image:

`` Embeds images on the page.

``

result will be





OTHER HTML TAGS

Unordered List:

 Defines an unordered (**bulleted**) list.

```
<ul>
  <li>Is it breakfast time yet?</li>
  <li>Why do you insist on taking my picture all the time?</li>
  <li>Why won't you let me catch that bird?</li>
</ul>
```

Ordered List:

 Defines an ordered (**numbered**) list.

```
<ol>
  <li>Is it breakfast time yet?</li>
  <li>Why do you insist on taking my picture all the time?</li>
  <li>Why won't you let me catch that bird?</li>
</ol>
```

Unordered List:

- Is it breakfast time yet?
- Why do you insist on taking my picture all the time?
- Why won't you let me catch that bird?

Ordered List:

1. Is it breakfast time yet?
2. Why do you insist on taking my picture all the time?
3. Why won't you let me catch that bird?

OTHER HTML TAGS

Span:

The tag in HTML is an inline element that is used to apply styles or scripting to a specific section of text or inline content within a larger block of text. It is a container tag that doesn't add any structural meaning to the content but is often used for styling purposes or to apply JavaScript functionality.

```
<p>This is a <span style="color: blue;">blue</span> word in a sentence.</p>
```

This is a **blue** word in a sentence.



ATTRIBUTES

HTML attributes are additional pieces of information that you can include within an HTML tag to provide extra details or characteristics about an element. These attributes help define how the element should behave or be displayed. Attributes are always **specified in the opening tag** of an HTML element and consist of **a name and a value**, separated by an **equal sign =** and enclosed in **double or single quotation marks**.

```
<tagname attribute="value">Content</tagname>
```

Here are some common HTML attributes and their purposes:



id Attribute: This attribute assigns a **unique** identifier to an element on the page. It is often used for styling or scripting purposes.

class Attribute: The class attribute assigns **one or more** class names to an element. Classes are used to group elements for styling with CSS or for scripting purposes.

WHAT IS URL?

A URL, or **Uniform Resource Locator**, is a standardized address used to identify and locate resources on the internet. It serves as a reference or pointer to a specific resource, such as a web page, image, file, or any other item accessible via the World Wide Web.

A typical URL consists of several components, which provide information on how to access the resource:



WHAT IS URL?

Scheme (Protocol): The scheme specifies the protocol or method used to access the resource. Common schemes include **http://** for regular web pages, **https://** for secure web pages, **ftp://** for file transfers, and **mailto:** for email addresses.

Domain: The domain (or hostname) identifies the web server where the resource is located. It is often a **human-readable address** like "www.apple.ca" and is used to route requests to the appropriate server.

Subdomain: A subdomain is a part of a larger domain, placed before the main domain name, separated by a dot.

Top-Level Domain (TLD) or Upper-Level Domain: The top-level domain is the highest level in the domain name hierarchy. It appears at the end of a domain name and typically represents the purpose, category, or geographic location of the website or resource.

Path: The path provides the specific location of the resource on the web server's file system. It is represented as a series of directory and file names separated by slashes ("/").

Fragment: The fragment identifier, preceded by a "#" symbol, points to a specific section within a web page. It's commonly used for deep linking within a page.

ERROR 404

Error 404, also known as '404 Not Found,' is a standard HTTP status code that signifies the web server's inability to locate the requested resource or webpage. When you come across a 404 error while surfing the internet, it signifies that the URL you've either entered or clicked on doesn't correspond to an existing page or resource on the server.

The term 'index.html' is commonly used as the default filename for a website's primary or home page. When an individual accesses a directory (a folder) on a web server without specifying a particular file (e.g., <http://www.example.com/>), the server frequently searches for an 'index.html' file within that directory and offers it as the default page. In numerous instances, 'index.html' functions as the starting point for accessing a website's content.

This underscores the reason why we encourage you to refrain from altering the default name 'index' to something different.

