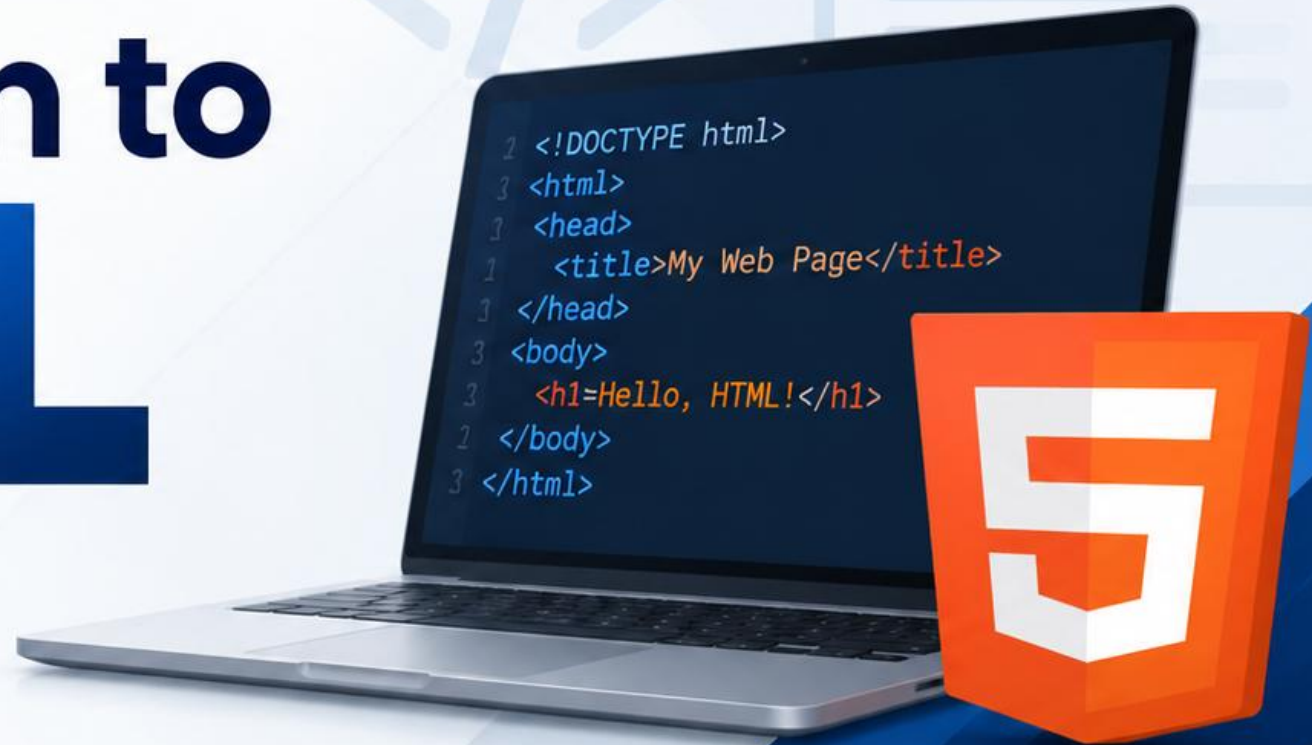


Introduction to HTML



What is HTML?

HTML (HyperText Markup Language) is the standard language for creating and structuring web pages using tags and elements. It defines how content like text, images, and links appear in a browser.



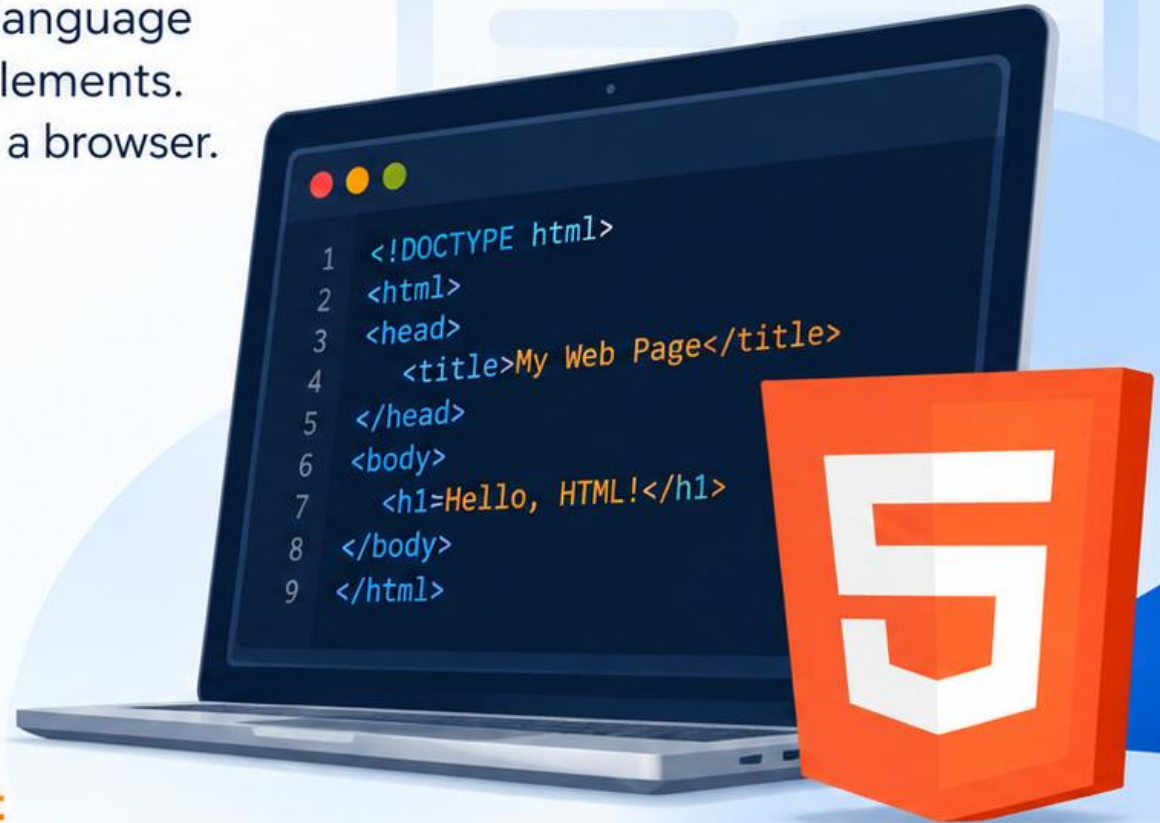
It is a **markup language**, not a programming language.



This means it annotates text to define how it is structured and displayed by **web browsers**.



It is a **static language**, meaning it does not inherently provide interactive features but can be combined with **CSS** for styling and **JavaScript** for interactivity.



Basic Structure of HTML



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>My Web Page</title>
5   </head>
6   <body>
7     <h1>Hello, HTML!</h1>
8     <p>This is a basic HTML page.</p>
9   </body>
10  </html>
```

1

<!DOCTYPE html>

Declares the **document type** and tells the browser that this is an HTML5 document.

2

<html> ... </html>

The **root element** that contains all the content of the HTML page.

3

<head> ... </head>

Contains **meta-information** about the page, such as the title, links to stylesheets, and other settings.

4

<title> ... </title>

Specifies the **title** of the web page, shown in the browser tab.

5

<body> ... </body>

Contains the **visible content** of the web page, such as text, images, links, and other elements.

There are generally two types of tags in HTML:



1 Paired Tags

These tags come in pairs, i.e. they have both opening (< >) and closing (</ >) tags.

```
1 <p>This is a paragraph.</p>
2
3 <h1>Welcome</h1>
```

2 Empty Tags

These tags are self-closing and do not require a closing tag.

```
1 <br>
2 
```

Parts of HTML Tags

Example: Paired Tag

These tags have both an opening and a closing tag.

Opening Tag

Marks the beginning of the element.

`<p`

Attribute

Provides additional information about the element.

`class="text"`

Attribute Value

The value for the attribute.

Closing Tag

Marks the end of the element.

`</p>`

Element Name

The name of the HTML element (in this example, it is a paragraph).

Content

The content of the element (e.g., text, image, or other elements).

Hello World!

Example: Empty Tag

These tags are self-closing and do not require a closing tag.

`<img`

Element Name

The name of the HTML element (in this example, it is an image).

`src="image.jpg"`

Attribute

Provides additional information about the element.

`alt="Image"`

Attribute Value

The value for the attribute.

`/>`

Self-Closing

Ends the tag. No separate closing tag is needed.

History of HTML

1991 HTML 1.0



- Created by Tim Berners-Lee at CERN.
- Introduced the basic structure of web pages.

1995 HTML 2.0



- First official standard published by the IETF.
- Introduced more elements and form capabilities.

1997 HTML 3.2



- Added tables, applets, and improved support for style and scripting.

1999 HTML 4.01



- Major improvements in structure, styling, and accessibility.
- Became a widely used standard.

2000 XHTML 1.0



- A stricter, XML-based version of HTML.
- Aimed to improve code quality and consistency.

2008 HTML5



- Introduced new semantic elements, multimedia support (audio and video), and modern web features.

2014 HTML5 Recommendation



- Officially standardized by the W3C.
- Became the current version used today.

Present Ongoing Development



- HTML continues to evolve with new features and improvements.
- Works together with CSS and JavaScript for modern web development.

HTML Text Editors

HTML can be written using any simple text editor. A text editor is a software application used to create and edit plain text files, such as HTML documents.

Common HTML Text Editors



Visual Studio Code
(VS Code)



Notepad++



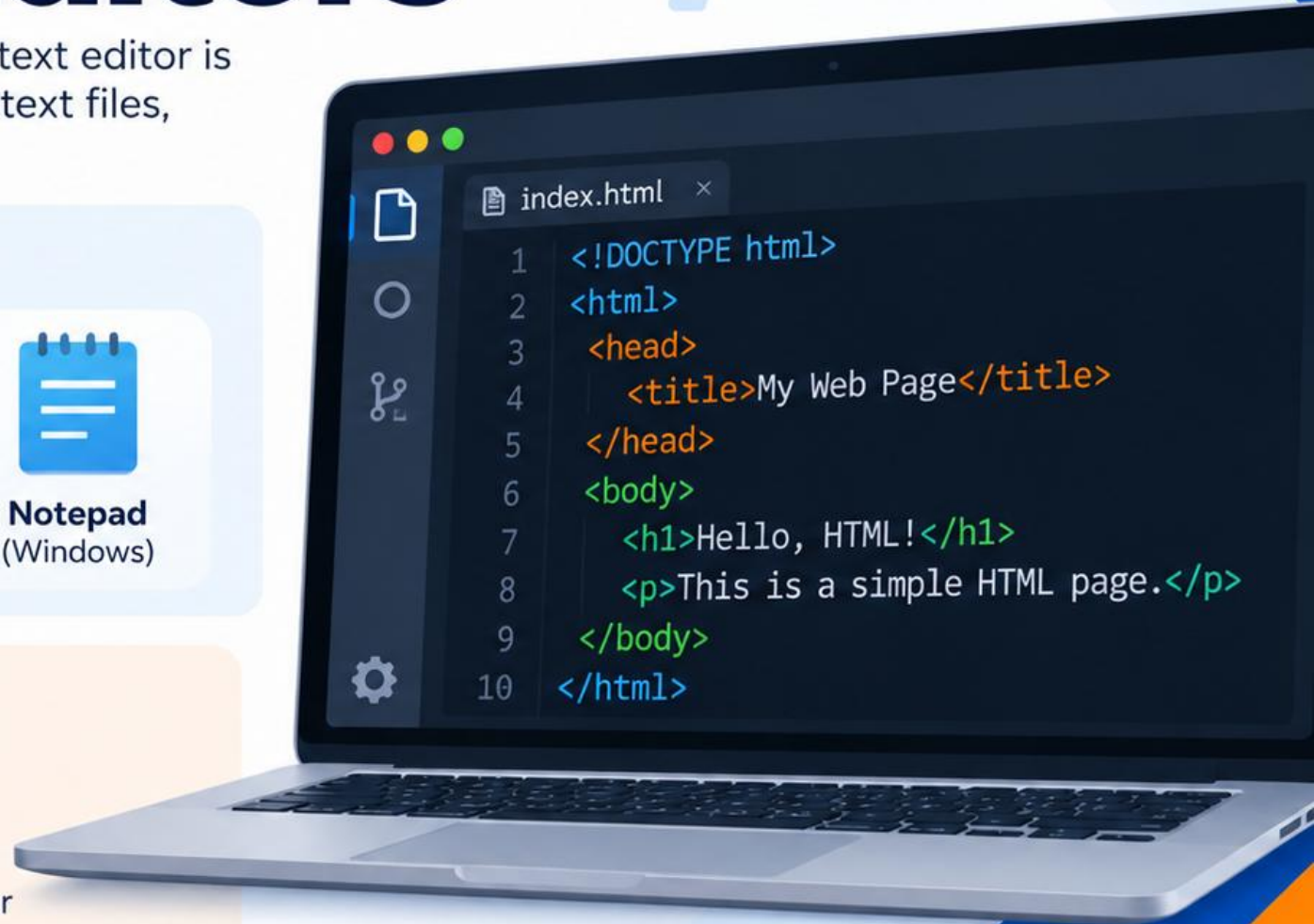
Sublime Text



Notepad
(Windows)

Key Points

- ✓ Text editors are used to write and edit HTML code.
- ✓ They save files with a **.html** or **.htm** extension.
- ✓ After saving, the HTML file can be opened in a web browser to view the result.



HTML Types of Comments

HTML comments are used to add notes or explanations in the code. They are not displayed in the web browser.

1 Single-line Comment

Used for a short comment in a single line.

```

<!-- This is a single-line comment -->
<p>This is a paragraph.</p>
```

2 Multi-line Comment

Used for a longer comment that spans multiple lines.

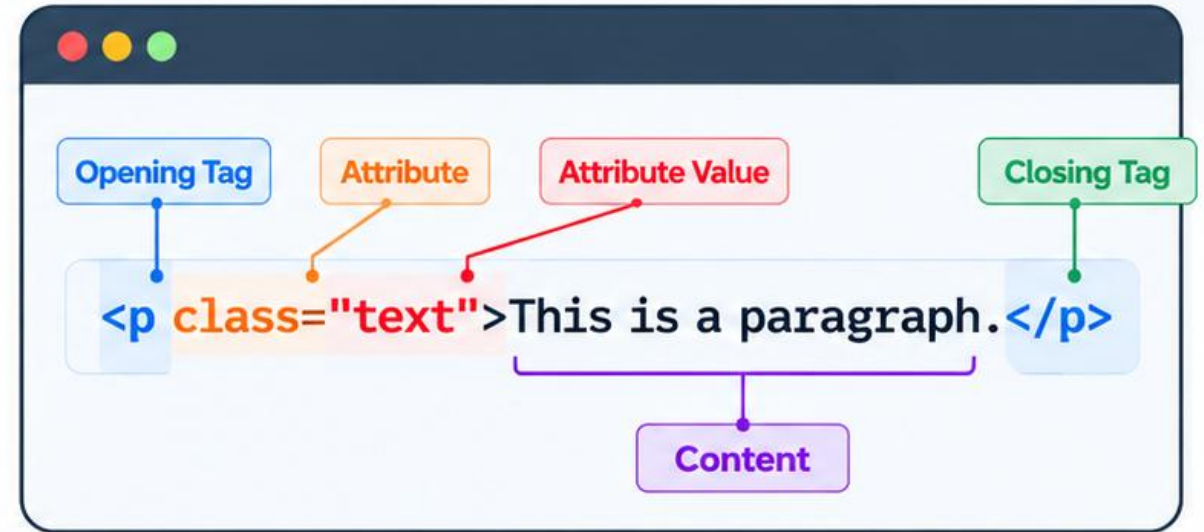
```

<!--
  This is a multi-line comment.
  It can span multiple lines.
  It will not be displayed in the browser.
-->
<h1>Welcome to My Website</h1>
```

HTML Elements

HTML elements are the basic building blocks of a webpage, defining its structure and content using start tags, content, and end tags.

- 1 HTML elements start with an opening tag `<tagname>` and end with a closing tag `</tagname>`, and can contain text, attributes, or other nested elements.
- 2 Some elements are self-closing (e.g., `<br>`, `<img>`), and browsers use elements to render the page visually.
- 3 Properly nesting elements ensures valid, accessible, and well-structured HTML.



Example: Normal Element

```
|| <h1>Welcome to My Website</h1>
```

This element has an opening tag, content, and a closing tag.

Example: Self-Closing Element

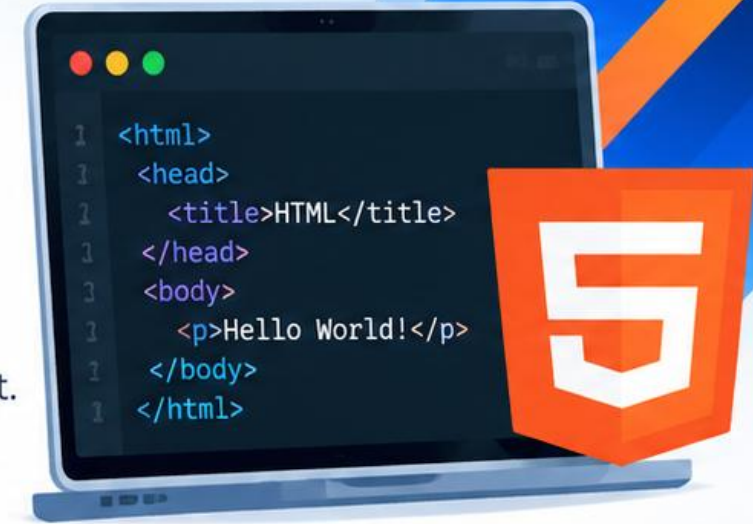
```
|| 
```

This element is self-closing and does not require a closing tag.

HTML Case Sensitivity

HTML tags are not case-sensitive, but using lowercase (e.g., `<b>`) is recommended for consistency and readability.

- HTML tags are not case-sensitive. For example, `<B>` and `<b>` both represent bold text.
- However, it's a best practice to use lowercase tags for consistency and readability.



Both work the same



```
<B>This is bold text.</B>
<b>This is bold text.</b>
```



Result in Browser

This is bold text.
This is bold text.



Recommended (Use Lowercase)



```
<b>This is bold text.</b>
```



Result in Browser

This is bold text.



Best Practice

Always use lowercase tags (e.g., `<b>`) for consistency and better readability.

HTML Nested Elements

Nested elements are elements placed inside another element. The inner element is contained within the outer element.

- 1 HTML elements can contain other elements inside them.
- 2 The inner element is called a **nested element**, and the outer element is called the **parent element**.
- 3 Properly nesting elements ensures valid, accessible, and well-structured HTML.

```
1 <div class="container">
2   <h1>Welcome to My Website</h1>
3   <p>This is a <strong>simple</strong>
4     paragraph.</p>
6 </div>
7
```

Visual Representation

`<div class="container">` (parent element)

`<h1>Welcome to My Website</h1>`
(nested element)

`<p>This is a <strong>simple</strong> paragraph.</p>`
(nested element)

Browser Output



HTML Necessary to Add an End Tag

It is important to include closing tags for non-void HTML elements, as missing tags can cause unexpected browser layout issues.

- ✓ Browsers may auto-add missing closing tags.
- ✓ This can lead to unexpected layout issues.
- ✓ Always include closing tags for non-void HTML elements.



Missing Closing Tag (Incorrect)

HTML Code

```
1 <p>This is a paragraph
2 <h1>Welcome to My Website
3
```

Browser Output

This is a paragraph

Welcome to My Website



With Closing Tag (Correct)

HTML Code

```
1 <p>This is a paragraph</p>
2 <h1>Welcome to My Website</h1>
3
4
```

Browser Output

This is a paragraph

Welcome to My Website



Key Takeaway

Always include closing tags for non-void HTML elements to ensure the browser displays the page correctly and avoid unexpected layout issues.

HTML Block-Level Elements

Block-level elements are HTML elements that always start on a new line and take up the full width of their parent container.

- 1 Block-level elements always start on a new line.
- 2 They take up the full width of their parent container (available width).
- 3 They are commonly used to structure the layout of a webpage.

Example Code

```
1 <div class="header">
2   <h1>My Website</h1>
2 </div>
3 <p>This is a paragraph.</p>
4 <div class="footer">
5   <p>Copyright 2026</p>
5 </div>
```

Browser Output (Each element takes full width)

My Website

(div - block-level element)

This is a paragraph.

(p - block-level element)

Copyright 2026

(div - block-level element)

HTML Inline Elements

Inline elements are HTML elements that do not start on a new line and only take up as much width as their content.

- 1 Inline elements do not start on a new line.
- 2 They only take up as much width as their content.
- 3 They are commonly used to format text or small parts of content within a block-level element.

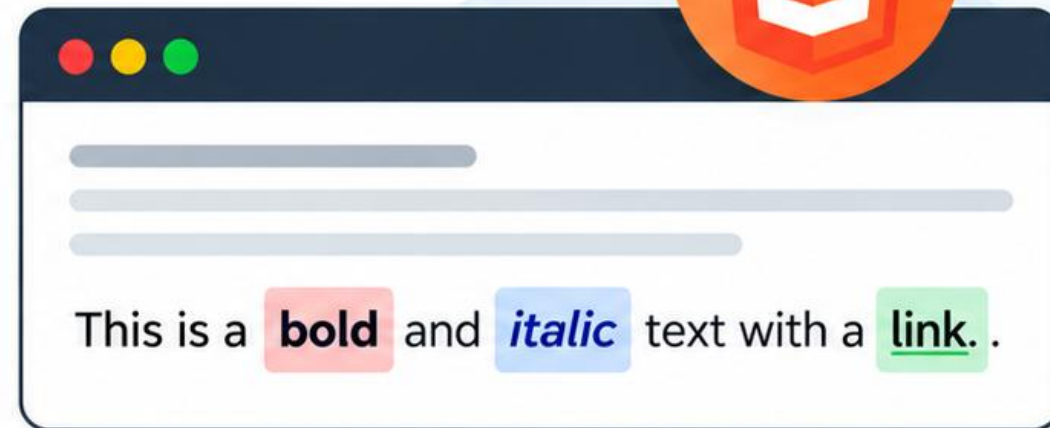
Example Code

```
1 <p>This is a <strong>bold</strong>  
2 and <em>italic</em> text with a  
3 <a href="#">link</a>.</p>  
4  
5
```

Browser Output (Same line - inline elements)



This is a **bold** and *italic* text with a [link](#).



This is a **bold** and *italic* text with a [link](#)..

HTML Attributes

HTML attributes provide additional information about an element. They are written inside the opening tag, usually in `name="value"` pairs.



1 Attributes provide extra information about an element.

2 They are written inside the opening tag.

3 Attributes are usually in **name="value"** pairs.

Example Code

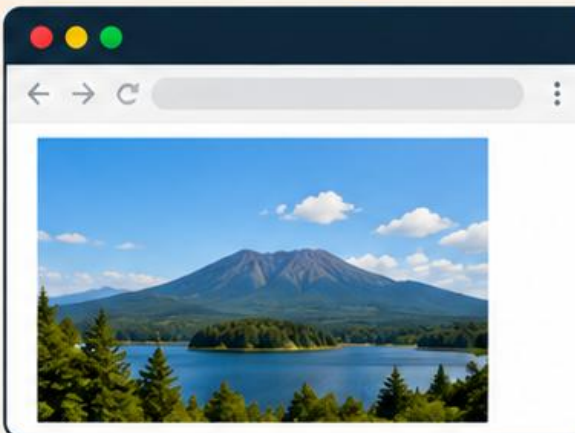
```

```

Diagram illustrating the structure of an HTML attribute pair:

- Attribute name:** `src`
- Attribute value:** `"image.jpg"`
- Another attribute:** `alt="A sample image"`
- Another attribute:** `width="300"`

Browser Output



Common HTML Attributes

id	Specifies a unique ID for an element.
class	Specifies one or more class names.
src	Specifies the source file (e.g., for images).
alt	Specifies alternative text (for images).
href	Specifies the link address (for links).
style	Specifies inline CSS styles.
title	Specifies extra information (tooltip text).

HTML Tags

HTML tags are keywords enclosed in angle brackets `< >` that define the structure and content of a webpage.

- 1 Tags tell the browser what type of content to display (e.g., headings, paragraphs, images, links).
- 2 Most tags have an opening tag `<tagname>` and a closing tag `</tagname>`.
- 3 Some tags are self-closing (void elements) and do not need a closing tag (e.g., `<br>`, `<img>`).



Example HTML Tags

```
1 <h1>Welcome to My Website</h1>
2 <p>This is a paragraph.</p>
3 <a href="https://www.example.com">Visit Link</a>
4 
5 <br>
```

Browser Output



HTML Heading Tags

Heading tags are used to define headings or titles in a webpage. They range from `<h1>` (most important) to `<h6>` (least important).



1 Heading tags are used to define headings or titles.

2 There are six levels, from `<h1>` (most important) to `<h6>` (least important).

3 They help organize content and improve readability.

Example Code

```
1 <h1>This is Heading 1</h1>
2 <h2>This is Heading 2</h2>
3 <h3>This is Heading 3</h3>
4 <h4>This is Heading 4</h4>
5 <h5>This is Heading 5</h5>
6 <h6>This is Heading 6</h6>
```

Browser Output

This is Heading 1

This is Heading 2

This is Heading 3

This is Heading 4

This is Heading 5

This is Heading 6

HTML Paragraph Tag

The `<p>` tag is used to define a paragraph of text in an HTML document.



- 1 The `<p>` tag is used to create a paragraph.
- 2 Browsers automatically add space (margin) before and after each paragraph.
- 3 It is a block-level element, which means it starts on a new line and takes the full width of its container.

Example Code

```
1 <h1>About Our University</h1>
2 <p>Our university is committed to providing
3 quality education and developing future leaders.
4 We offer various programs, state-of-the-art
5 facilities, and a supportive community.</p>
6
7 <p>Join us and be part of a brighter future.</p>
```

Browser Output



About Our University

Our university is committed to providing quality education and developing future leaders. We offer various programs, state-of-the-art facilities, and a supportive community.

Join us and be part of a brighter future.

HTML Anchor Tag

The `<a>` tag (anchor tag) is used to create hyperlinks, which link to other web pages, files, or locations within the same page.

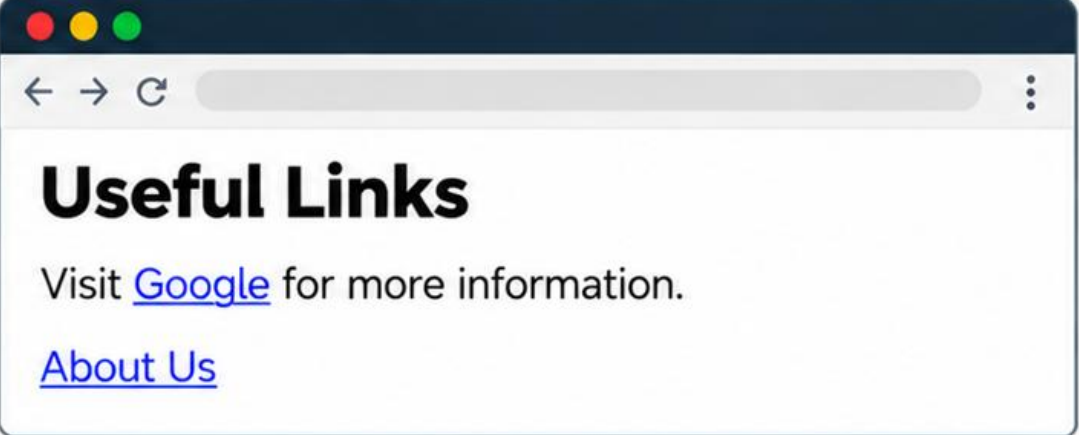


- 1 The `<a>` tag is used to create a hyperlink.
- 2 It uses the `href` attribute to specify the link destination.
- 3 It can link to external websites, internal pages, files, or sections within the same page.

Example Code

```
1 <h1>Useful Links</h1>
2
3 <p>Visit <a href="https://www.google.com">
4 Google</a> for more information.</p>
5
6 <a href="about.html">About Us</a>
```

Browser Output



HTML `<a>` target Attribute

The **target** attribute specifies where to open the linked document when a user clicks the link.



Common Values of the target Attribute

Value	Description
<code>_self</code>	Opens the linked document in the same tab (default).
<code>_blank</code>	Opens the linked document in a new tab.
<code>_parent</code>	Opens the linked document in the parent frame.
<code>_top</code>	Opens the linked document in the full body of the window.
<code>framename</code>	Opens the linked document in a specific named frame.

Example Code

```
1 <h2>Target Attribute Example</h2>
2
3 <a href="https://www.google.com"
4   target="_self">Open in Same Tab</a>
5
6 <a href="https://www.google.com"
7   target="_blank">Open in New Tab</a>
8
9 <a href="about.html"
10  target="_parent">Open in Parent</a>
11
12 <a href="about.html"
13  target="_top">Open in Full Window</a>
```

Browser Output (target="_self")



Browser Output (target="_blank")



Browser Output (target="_parent")



Browser Output (target="_top")



HTML Italic Tag

The `<i>` tag is used to define **italic text**. It makes the text slanted to the right, usually for emphasis or to indicate a different tone.



Example Code

```
1 <p>This is a normal text.</p>
2 <p>This is an <i>italic text</i>.</p>
3 <p>We can also use it in a sentence
4   like this: <i>for example</i>.</p>
```

Browser Output

This is a normal text.

This is an *italic text*.

We can also use it in a sentence
like this: *for example*.

HTML small Tag

The `<small>` tag is used to define **smaller text**. It is commonly used for side notes, copyright information, disclaimers, or any text that is less important.



Example Code

```
1 <p>This is a normal text.</p>
2 <p>This is a <small>smaller text</small>.</p>
3 <p>Copyright © 2026</p>
4 <p><small>All rights reserved.</small></p>
```

Browser Output

This is a normal text.

This is a smaller text.

Copyright © 2026

All rights reserved.

HTML ins Tag

The **<ins>** tag is used to define **inserted text**. It usually displays the text with an **underline** to indicate that it has been added to the document.



Example Code

```
1 <p>This is a normal text.</p>
2 <p>This is an <ins>inserted text</ins>.</p>
3 <p>We can also use it in a sentence
4   like this: The <ins>new version</ins>
   is now available.</p>
```

Browser Output

This is a normal text.

This is an inserted text.

We can also use it in a sentence like this:
The new version is now available.

HTML Subscript and Superscript

The `<sub>` tag is used to display **subscript text**, and the `<sup>` tag is used to display **superscript text**. They are often used in mathematical expressions, chemical formulas, and other notations.



HTML Subscript

The `<sub>` tag displays text **below** the normal line.

```
1 <p>Water chemical formula: H<sub>2</sub>O</p>
2 <p>Carbon dioxide: CO<sub>2</sub></p>
3 <p>Glucose: C<sub>6</sub>H<sub>12</sub>O<sub>6</sub></p>
```

Browser Output (Subscript)

```
Water chemical formula: H2O
Carbon dioxide: CO2
Glucose: C6H12O6
```

HTML Superscript

The `<sup>` tag displays text **above** the normal line.

```
1 <p>Square: x<sup>2</sup></p>
2 <p>Cubic: x<sup>3</sup></p>
3 <p>Area of a circle: r<sup>2</sup></p>
```

Browser Output (Superscript)

```
Square: x2
Cubic: x3
Area of a circle: r2
```

HTML Bold and Strong

The `<b>` tag is used to make text **bold** without adding extra meaning, while the `<strong>` tag is used to make text **bold** with **strong importance**.



HTML Bold Tag (`<b>`)

The `<b>` tag makes the text **bold** without giving any special meaning to the text.

```
1 <p>This is a normal text.</p>
2 <p>This is a <b>bold text</b>.</p>
2 <p>You can also use it like <b>this</b>
3   in a sentence.</p>
```

Browser Output (`<b>`)

This is a normal text.
This is a **bold text**.
You can also use it like **this** in a sentence.

HTML Strong Tag (`<strong>`)

The `<strong>` tag makes the text **bold and indicates** that the text is **important**.

```
1 <p>This is a normal text.</p>
2 <p>This is a <strong>important text</strong>.</p>
2 <p>You can also use it like <strong>this</strong>
3   in a sentence.</p>
```

Browser Output (`<strong>`)

This is a normal text.
This is a **important text**.
You can also use it like **this** in a sentence.

HTML Highlight and Emphasize Text

The `<mark>` tag is used to **highlight text**, while the `<em>` tag is used to **emphasize text**. The emphasized text is usually displayed in italic style and gives additional meaning to the content.



HTML Highlight Text (<mark>)

The `<mark>` tag is used to **highlight text**, usually to indicate important or relevant information.

```
1 <p>This is a normal text.</p>
2 <p>This is a <mark>highlighted text</mark>.</p>
2 <p>You can also use it in a sentence like this:
3   This is a <mark>key point</mark> to remember.</p>
```

Browser Output (<mark>)

This is a normal text.
This is a **highlighted text**.
You can also use it in a sentence like this:
This is a **key point** to remember.

HTML Emphasize Text ()

The `<em>` tag is used to **emphasize text**, typically displayed in italic style and gives additional meaning.

```
1 <p>This is a normal text.</p>
2 <p>This is an <em>emphasized text</em>.</p>
2 <p>You can also use it in a sentence like this:
3   This is a very <em>important</em> reminder.</p>
```

Browser Output ()

This is a normal text.
This is an *emphasized text*.
You can also use it in a sentence like this:
This is a very *important* reminder.

HTML Deleted Text

The `<del>` tag is used to define **deleted text**. It usually displays the text with a strikethrough line to indicate that it has been removed or is no longer valid.



Example Code

```
1 <p>This is a normal text.</p>
2 <p>This is a <del>deleted text</del>.</p>
3 <p>We can also use it in a sentence
4   like this: <del>old version</del> new version.</p>
```

Browser Output

This is a normal text.

This is a ~~deleted~~ text.

We can also use it in a sentence
like this: ~~old version~~ new version.