

# Training Day2 report

11 June 2024

## HTML TAGS:

HTML tags are like keywords which defines that how web browser will format and display the content. With the help of tags, a web browser can distinguish between an HTML content and a simple content. HTML tags contain three main parts: opening tag, content and closing tag. But some HTML tags are unclosed tags. When a web browser reads an HTML document, browser reads it from top to bottom and left to right. HTML tags are used to create HTML documents and render their properties. Each HTML tags have different properties.

An HTML file must have some essential tags so that web browser can differentiate between a simple text and HTML text. You can use as many tags you want as per your code requirement.

- All HTML tags must enclosed within < > these brackets.
- Every tag in HTML perform different tasks.
- If you have used an open tag <tag>, then you must use a close tag </tag> (except some tags)

## Syntax

<tag> content </tag>

## HTML Tag Examples

NOTE: HTML Tags are always written in lowercase letters. The basic HTML tags are given below:

<p> Paragraph Tag </p>

<h2> Heading Tag </h2>

<b> Bold Tag </b>

`<i> Italic Tag </i>`

`<u> Underline Tag </u>`

## Unclosed HTML Tags

Some HTML tags are not closed, for example br and hr.

**`<br>` Tag:** br stands for break line, it breaks the line of the code.

**`<hr>` Tag:** hr stands for Horizontal Rule. This tag is used to put a line across the webpage.

## HTML Meta Tags

DOCTYPE, title, link, meta and style

## HTML Text Tags

`<p>`, `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>`, `<strong>`, `<address>`, `<q>`, `<code>`, `<ins>`, `<del>`, `<mark>`, and `<br>`

## HTML Link Tags

`<a>`

## HTML Image and Object Tags

`<img>`, `<area>`, `<map>` and `<object>`

## HTML List Tags

`<ul>`, `<ol>`, `<li>` and `<dl>`

## HTML Table Tags

table, tr, td, th, col and caption

## HTML Form Tags

form, input, textarea, select, option, button and label

## HTML Scripting Tags

script and noscript

# All html tags:

## 1. Comments:

With the help of this tag, we can add comments in the HTML. After commenting on the elements, the browser does not render it. Also, these comments parts are not displayed on the web page. But these commented codes are available in the source code.

**Syntax:** `<strong><!-- --></strong>`

## 2. Doctype:

With the help of this tag, we can declare the version of the markup language to the web browser.

**Syntax:** `< !DOCTYPE html >`

## 3. <a>:

With the help of this tag, we can create a hyperlink to the web page. The hyperlink can be done with the help of another file in a different location.

**Syntax:** `<a href="url">Link Text</a>`

## 4. <address>:

With the help of this tag, we can find the address or the contact details of the web page's author. The `<address>` tag contains addresses, cellphone numbers, emails, website connections, etc.

**Syntax:**

1. `<address>`
2. Contact Information Goes Here
3. `</address>`
5. `<area>`:

With the help of this tag, we can identify the clickable area associated with the hyperlink.

**Syntax:**`<area shape="shape-type" coords="coordinates" href="destination-url" alt="alternative-text">`

## 6. `<b>`:

With the help of this tag, we can make the text bold without any additional significance.

**Syntax:** `<b>Text to be displayed in bold</b>`

## 7. `<base>`:

With the help of this tag, we can create the target address, which is targeted by all the URLs present in the document.

**Syntax:** `<base href="base-url" target="base-target">`

## 8. `<body>`:

With the help of this tag, we can define the web page's main content. The body tag includes text, paragraphs, headings, images, tables, links, and videos.

**Syntax:**

1. `<body>`
2. `<!-- Content goes here -->`
3. `</body>`
4. `<strong> </strong>`

## 9. `<br>`:

With the help of this tag, we can produce a line break on the web page. We can use this tag basically during the writing of poems or addresses that require line breaks.

**Syntax:**

1. `<br>`
2. `<strong> </strong>`

## 10. `<button>`:

With the help of this tag, we can create a clickable button on the web page. We can use the button tag in forms or anywhere in a document.

### **Syntax:**

1. **<button>Button Text</button>**
2. **<strong> </strong>**
  11. **<code>:**

With the help of this tag, we can insert a particular code fragment, which might be HTML or CSS code.

**Syntax: <code>Code goes here</code>**

### **12. <col>:**

With the help of this tag, we can specify all the column's properties. We can specify it inside the <colgroup> element.

**Syntax:<col>**

### **13. <colgroup>:**

With the help of this tag, we can specify one or more columns in a group. With the help of this, we can apply the style to the entire column without repeating the style for every column and every row.

### **Syntax:**

1. **<colgroup>**
2. **<!-- col elements go here -->**
3. **</colgroup>**

### **14. <data>:**

With the help of this tag, we can add machine-readable translation. The processor generates it, and the browser can generate the human-readable value.

### **Syntax:**

1. **<data value="data-value">Data Label</data>**
15. **<dd>:**

With the help of this tag, we can define the definition description in the HTML. It is also used to describe an item in the list of descriptions.

### Syntax:

1. `<dl>`
  2. `<dt>Term</dt>`
  3. `<dd>Description or definition goes here</dd>`
  4. `</dl>`
16. `<del>`:

With the help of this tag, we can declare the particular section that will be deleted from the document. After doing this, the web browser strike over that particular section in the webpage.

### Syntax:

1. `<del>Deleted Text</del>`
17. `<details>`:

With the help of this tag, we can create a toggle. This toggle contains all the information but is in a closed state. When we mouse hover over it, it comes to an open state and displays all the hidden information.

### Syntax:

1. `<details>`
  2. `<summary>Summary Text</summary>`
  3. `<!-- Content goes here -->`
  4. `</details>`
18. `<div>`:

With the help of this tag, we can create a generic container that can hold the flow of the content. It also does not have the default rendering.

### Syntax:

1. `<div>`
  2. `<!-- Content goes here -->`
  3. `</div>`
19. `<form>`:

With the help of this tag, we can receive the information from the user.

### **Syntax:**

1. **<form action="form-handler-url" method="HTTP-method">**
2. **<!-- Form controls go here -->**
3. **</form>**

### **20. <head>:**

With the help of this tag, we can specify the header part of the header section of the code. It contains all the information regarding the document.

### **Syntax:**

1. **<head>**
2. **<!-- Head content goes here -->**
3. **</head>**

### **21. <header>:**

With the help of this tag, we can define the webpage's header. It also contains information regarding the title and heading of the document and section.

### **Syntax:**

1. **<header>**
2. **<!-- Header content goes here -->**
3. **</header>**

### **22. <hgroup>:**

With the help of this tag, we can represent the heading group. It is also used for group heading elements, such as headings and subheadings, from <h1> to <h6>.

### **Syntax:**

1. **<h1>Main Heading</h1>**
2. **<h2>Subheading 1</h2>**
3. **<h3>Sub-subheading 1.1</h3>**
4. **<h3>Sub-subheading 1.2</h3>**
5. **<h2>Subheading 2</h2>**

### 23. <h1> to <h6>:

With the help of this tag, we can define the web page's heading. It also makes the heading as large and bold by default. the <h1> headings are displayed in the largest font, whereas the <h6> headings are displayed in the smallest font.

#### Syntax:

1. <h1>Heading level 1</h1>
2. <h2>Heading level 2</h2>
3. <h3>Heading level 3</h3>
4. <h4>Heading level 4</h4>
5. <h5>Heading level 5</h5>
6. <h6>Heading level 6</h6>

### 24. <hr>:

With the help of this tag, we can divide the web page. It stands for horizontal rule or theme break.

#### Syntax:

1. <hr>

### 25. <html>:

With the help of this tag, all other elements are contained by it. It is also known as the root of the HTML element.

#### Syntax:

1. <!DOCTYPE html>
2. <html>
3. <!-- HTML content goes here -->
4. </html>

### 26. <i>:

With the help of this tag, we can make any content italicized. It Highlights an important term, phrase, or technical term.

#### Syntax:

1. <i>Italicized Text</i>



## 27. <iframe>:

With the help of this tag, we can embed another HTML page into the current webpage.

**Syntax:**

1. `<iframe src="url"></iframe>`

## 28. <img>:

With the help of this tag, we can embed an image into the current web page. It links an image to the image without directly inserting it into the web page.

**Syntax:**

1. ``

## 29. <input>

With the help of this tag, we can enable interactive control in the form. It also depends on the various fields of the attribute type.

**Syntax:**

1. `<input type="input-type" name="input-name" value="default-value">`

## 30. <ins>:

With the help of this tag, we can specify the block of the inserted text. It also signifies that a particular range of text has been added to the document.

**Syntax:** `<ins>Inserted Text</ins>`

## 31. <label>:

With the help of this tag, we can improve the usability for mouse users, i.e., clicking on the label will toggle the control.

**Syntax:** `<label for="input-id">Label Text</label>`

## 32. <li>:

With the help of this tag, we can represent the list of items on the web page. It contains an ordered list (an unordered or a menu) as a parent element.

**Syntax:**

1. **<ul>**
2. **<li>List Item 1</li>**
3. **<li>List Item 2</li>**
4. **<li>List Item 3</li>**
5. **</ul>**

### **33. <link>:**

With the help of this tag, we can link external resources to the current webpage.

**Syntax:** **<link rel="stylesheet" href="stylesheet.css">**

### **34. <main>:**

With the help of this tag, we can make an ordinary element the most important element. We have to write this tag inside the **<body>** tag.

**Syntax:**

1. **<main>**
2. **<!-- Main content goes here -->**
3. **</main>**

### **35. <map>:**

With the help of this tag, we can define the map image. The image map also has a clickable area.

**Syntax:**

1. **<map name="map-name">**
2. **<!-- Area tags go here -->**
3. **</map>**

### **36. <mark>:**

With the help of this tag, we can represent the text as highlighted, which will be used for reference or notation purposes. Based on the importance of the content, it highlighted the text.

**Syntax:**

1. **<mark>Highlighted Text</mark>**

### 37. <noscript>:

With the help of this, we can provide an alternative script when the user's browser does not support the particular script. It also works whenever the user's browser does not support client-side scripting or does not support any scripting language.

**Syntax:**

1. <noscript>
2. <!-- Alternate content goes here -->
3. </noscript>

### 38. <object>:

With the help of this tag, we can display multimedia like audio, videos, images, PDFs, and Flash on web pages.

**Syntax:** <object data="resource-url"></object>

### 39. <ol>:

This tag is used for the creation of the ordered list. The ordered list can be numerical or alphabetical. Inside the <ol> tag, we can create the list using the <li> tag. It will then follow the order.

**Syntax:**

1. <ol>
2. <li>First Item</li>
3. <li>Second Item</li>
4. <li>Third Item</li>
5. </ol>
- 1.

### 40. <output>:

With the help of this tag, we can create the container for the element that injects the result of the calculation.

**Syntax:**

1. <output for="input-id"></output>

#### 41. <p> :

With the help of this tag, we can define the paragraph tag. With the help of a paragraph tag, we can represent the visual media as blocks of text separated from adjacent blocks by blank lines or first-line indentation.

**Syntax:** Paragraph text goes here.

#### 42. <q>:

With the help of this tag, we can create an inline quotation on the web page. The text is contained inside the quotation marks in most browsers today.

**Syntax:** Here is a <q>short quotation</q> within a sentence.

#### 43. <script>:

With the help of this tag, we can insert the client-side scripting. It also enables the insertion of the script in the HTML web page.

**Syntax:** <script src="script.js"></script>

#### 44. <section>:

With the help of this tag, we can create the header and footer or any other section on the web page.

**Syntax:**<section> <!-- Content goes here --> </section>

#### 45. <small>:

With the help of this tag, we can reduce the font size by one size smaller than its base size, for instance, from large to small.

**Syntax:** <small>Small text goes here</small>

#### 46. <source>

With the help of this tag, we can link multiple media resources like <audio> and <video> ,

**Syntax:** <source src="media-file.mp4" type="video/mp4">

#### 47. <strong>

With the help of this tag, we can define the content as very important. The browser will display that element in bold.

**Syntax:** <strong>Emphasized text</strong>

#### 48. <span>:

It is the most generic element in the HTML. We can apply the style by grouping the elements.

**Syntax:** <span>Text or content goes here</span>

#### 49. <style>:

We can provide the style information for the document with the help of this tag.

**Syntax:**<style> CSS rules go here </style>

#### 50. <sub>:

With the help of this tag, we can display the subscript element only for typographical reasons.

**Syntax:** <sub>Subscript text</sub>

#### 51. <sup>:

With the help of this tag, we can display any text as the superscript of the inline element.

**Syntax:** <sup>Superscript text</sup>

#### 52. <table>:

With the help of this tag, we can define the data in a tabular format in HTML. It will create a two-dimensional table that has some rows and columns.

**Syntax:**<table> <!-- Table content goes here --> </table>

#### 53. <td>:

With the help of this tag, we can define the table's cell. A table row may consist of more than one <td> element. The default alignment is left in this element.

### Syntax:

1. `<tr>`
2. `<td>Table cell content</td>`
3. `</tr>`