HTML TUTORIAL

1. HTML introduction done
2. HTML DOCUMENT STRUCTURE done
3. HTML headings done
4. HTML paragraph done
5. HTML Hyperlinks done
6. HTML File Links done
7. HTML Styles done
8. HTML span done
9. HTML div done
10. HTML section done
11. HTML button done
12. HTML class done
13. HTML id done
14. HTML images done
15. HTML icons done
16. HTML tables done
17. HTML media

HTML INTRODUCTION

HTML, also known as Hypertext Markup Language. Used to create web pages by using some elements like hyperlinks, video, images, headings, and some other elements. HTML uses a specific DOCUMENT STRUCTURE with a lot of elements and tags like <html>,<head>,<body>, and other tags and elements.

HTML DOCUMENT STRUCTURE

HTML stands for one of the most user-friendly and easy languages in web development. Important to have a basic knowledge of its structure.

Main elements of the HTML document Structure:

1. **Document Type Declaration (DOCTYPE):**<!DOCTYPE html> defines the browser about which version of the HTML document is used.  
    **DOCTYPE IN HTML 5:** MINI COMPILER  
     
    **DOCTYPE IN OTHER PREVIOUS HTML VERSIONS:** MINI COMPILER
2. **HTML Element:**<html> is the base of every HTML document. Which contains all the work of the page.
3. **Head Section:**<head> section contains <meta>, <title>, <links>, and other resource links too.
   1. **meta:**<meta> tag stands for Metadata which is used by browsers, search engines, and other web services so when users search on the browser related to the page they will reach that page.
   2. **title:**<title> defines the title of the document which is shown on the page’s tag.
   3. **links:**<link> used to link CDN or a CSS file to the HTML file.
      1. **CDN**CDN stands for Content Delivery Network. It’s a link to files from the server. CDN includes images, videos, text, and other resources.
      2. **CSS**CSS stands for Cascading Style Sheets. It’s a style sheet language used for styling the page by pointing the tags like <body>, <h1>, and other tags.
4. **Body Section:**Main information like video, text, image, and audio are included in the body section. this is the main tag that contains all the information on the page.

**Main compiler**

HTML Headings

HTML offers six types of headings <h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. Which represents different levels of headings

Definitions of the headings:

<h1> represents the highest level of heading in HTML, Used to define the main heading or title of the entire page.

<h2> represents the second level of the heading in HTML, used to define subheadings that are less important than the main heading (H1).

<h3> represents the third level of the heading in HTML, Used to define the subheading of the main heading on the page that are less important than h1 and h2.

<h4> represents the fourth level of the heading in HTML, mainly used to define the topic with the H3 heading.

<h5> and <h6> represent the lowest level of the heading in HTML, mostly used to define Bullet points after subheadings or headings.

Subheading definition

A mini-headline or text that found under the main headline to give more information about the related information.

sub

**Main compiler**

HTML PARAGRAPH

The <p> represents the paragraph in HTML. Used to add a paragraph text to the page.

Here's how it works:

Mini compiler

Key point:

Paragraph Spacing: The browser automatically adds some space (margin) before and after every <p> tag

Nested Tags: Nesting allows to apply one element inside another element Like <strong> element used for bolding the text.

Hr used to add a line or separate data on the page.

Br used to break the line and start a new paragraph without using another paragraph tag.

Pre used to display text with fixed width and font.

Strong tag used to make text bold or highlight the text.

Bold tag used to bold a specific part of the text.

Mini compiler

HTML HYPERLINKS

HTML Hyperlinks are used to link pages in HTML that can be added in text, buttons, or images.

Here's an example of how to create a basic hyperlink:

MINI COMPILER

In this example, "Click here," linked to "https://www.example.com."

Hyperlinks can also be used to link to the sections within the same page by using the id attribute to identify the target element. Here's an example:

MINI COMPILER

By clicking on "Go to Section 1" the page will scroll to the element with id="section1."

Make a link to open in a new tab or window by using the target attribute:

MINI COMPILER

Clicking on "Open in New Tab" will open the link in a new browser tab or window.

HTML FILE LINKS

To connect files like CSS and JavaScript to the HTML document by using the <link> and <script> elements, respectively. Here's how to do it:

Linking a CSS file:

Put the CSS file link in the <head> section of your HTML document.

Replace 'your\_css\_file.css' with the actual path of the CSS file:

Linking a JavaScript file:

Place the JavaScript link just before the closing </body> tag. This ensures that the JavaScript code is loaded after the HTML content is parsed.

Replace 'your\_js\_file.js' with the actual path to the JavaScript file:

Main compiler

HTML STYLES

In HTML, use CSS (Cascading Style Sheets) to style the webpage elements. Including its color, size, font, layout, and more.   
There are several methods to apply styles in HTML:

Inline styles: adding styles directly in HTML elements by using the 'style' attribute.

Main compiler with 3 slides

Internal styles: styles inside the <style> tag located in the <head> section of your HTML document.

External styles: make a separate CSS file with the ‘.css’ extension and connect it to the HTML document by using the <link> tag.

HTML SPAN

In HTML, the <span> element is an element that used to point a specific part in a document. Like in <p>, <h4>, <h5> and etc.

Here's an example of how to use the <span> element:

Mini compiler

In the previous example, the word 'blue' is enclosed within a <span> element with an inline style, changing the text color to blue. This enables to style only that particular part of text differently from the rest of the paragraph.

Add an id or class attribute to a <span> element, allowing to style it with CSS or using it in JavaScript.

HTML DIV

In HTML, the <div> element used to divide sections or to group elements

Here's how to use <div> element:

Mini compiler

In the previous example, the <div> element used to group the heading <h1> and paragraph <p> elements.

The <div> element can also add id or class attribute, just like other HTML elements, enabling you to apply particular styles or to use JavaScript.

HTML SECTION

In HTML, the <section> element is a semantic tag used to define a section of the page.

Here's an example of how to use <section> element:

Main compiler

In the example on the right side, the <section> element used as an enclosing tag. Inside it, there’s a heading <h2> element and a paragraph <p> element.

HTML BUTTON

In HTML, the <button> element used to create buttons on the web page. These buttons can perform trigger actions or a specific functions when clicked. The <button> element can be customized with text, icons, or other HTML elements.

Here's an example of how to use the <button> element:

Mini compiler

In the example on the right, a simple <button> element is created with the text "Click Me". The type="button" attribute is added to define that it's a button element.

The <button> element can be also used to wrap other HTML content, such as text, images, or icons:

In this example, an <img> element is placed inside the <button> element along with the text "Submit." This allows to show an icon or an image on the button.

Mini compiler 2

HTML CLASS

In HTML, the class attribute helps to add one or more class names to an element. This makes it easy to style it with CSS or to use it in JavaScript.

Here's an example:

Mini compiler

In the example above, there’s a <p> element with a class attribute set to "highlight." This lets to style or add functions to this paragraph using CSS or JavaScript.

For instance, the <div> element has two classes: "box" and "large." This way, styling elements by using multiple class names makes the styling clearer.

In CSS, elements can be targeted with a specific class using the dot:

In this CSS example, the .highlight used to style any element with the class "highlight" by making the text yellow.

HTML ID

In HTML, the id attribute gives a special name to an element. This id helps to identify and control the element with JavaScript or CSS.

Here's an example:

Mini compiler

In the example above, the <h1> element has an id attribute set to "main-heading." This id lets CSS target this heading for styling or other actions.

To target an element with a particular id in CSS, use the hash (#):

Mini compiler

In this CSS example, the #main-heading used to style the element with the id "main-heading" by making its text blue.

HTML image  
  
In HTML, <img> element used to put pictures on the webpage.

It's a simple tag that doesn't need a closing tag.

Here's how to use it:

Mini compiler

In this example, the src attribute tells the browser where to find the image. The alt attribute provides text to show if the image can't load.

The image's width and height can be decided with the width and height attributes:

Mini compiler

In this example, width="300" and height="200" of the image.

The <img> element can be styled by CSS classes or inline styles:

Mini compiler

In this example, the <img> has a class attribute "rounded" for styling.

Mini compiler

In this example, src uses a path from the website's root to link image.

HTML icons

In HTML, the web page can be attractive by adding icons. Here are two common ways to do it:

1. Using Icon Libraries: Icon libraries like Font Awesome offer a variety of icons to use for HTML. To include Font Awesome icons, follow these steps:
   1. Add the Font Awesome stylesheet to the HTML's <head> section with this line:

Mini compiler

* 1. Once the Font Awesome stylesheet is added, use icons in your HTML with the <i> element.   
     For example:  
      mini compiler

1. Using Custom Icon Images: If you have your own icon images, you can include them with the <img> element.

Mini compiler

HTML tables

In HTML, tables help to organize information into rows and columns. The <table> element is the main container for the table, and it's followed by <thead>, <tbody>, and <tfoot> elements to structure the table's content. Here's a basic table structure example:

Main compiler

* <thead> defines the table header with column headings in <th> elements.
* <tbody> contains the main data rows within <tr> (table row) and <td> (table cell) elements.
* <tfoot> use to add a table footer.