



HTML Fundamentals

Basic Structure

<code><!DOCTYPE html></code> - Declares the document type as HTML5.
<code><html></code> - The root element of an HTML page.
<code><head></code> - Contains meta-information about the HTML page.
<code><title></code> - Specifies a title for the HTML page (which is shown in the browser's title bar or tab).
<code><body></code> - Defines the document's body, and is a container for all the visible content, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.
Example: <pre><!DOCTYPE html> <html> <head> <title>Page Title</title> </head> <body> <h1>This is a Heading</h1> <p>This is a paragraph.</p> </body> </html></pre>

Text Formatting

<code><h1></code> to <code><h6></code>	Defines HTML headings.
<code><p></code>	Defines a paragraph.
<code>
</code>	Inserts a single line break.
<code><hr></code>	Defines a thematic break (horizontal rule).
<code></code> / <code></code>	Defines bold text.
<code><i></code> / <code></code>	Defines italicized text.

Lists

<code></code>	Defines an unordered list.
<code></code>	Defines an ordered list.
<code></code>	Defines a list item.
Example of Unordered List:	<pre> Coffee Tea Milk </pre>
Example of Ordered List:	<pre> Coffee Tea Milk </pre>

CSS Fundamentals

Basic Syntax

CSS rulesets consist of a selector and a declaration block: <pre>selector { property: value; }</pre>
Example: <pre>h1 { color: blue; font-size: 24px; }</pre>

Selectors

<code>element</code>	Selects all HTML elements of the specified type.
<code>.class</code>	Selects all elements with the specified class.
<code>#id</code>	Selects the element with the specified id.
<code>*</code>	Selects all HTML elements on the page.
<code>element > element</code>	Selects all elements that are a direct child of a specified element.
<code>element + element</code>	Selects the first element that is immediately preceded by the specified element.

Box Model

The CSS box model is a container that contains multiple properties to format any HTML element:
<ul style="list-style-type: none"> Content - The content of the box, where text and images appear Padding - Clears an area around the content. The padding is transparent Border - A border that surrounds the padding and content Margin - Clears an area outside the border. The margin is transparent
<pre>.box { width: 300px; padding: 25px; border: 5px solid red; margin: 25px; }</pre>

HTML Forms

Form Elements

<code><form></code>	Defines an HTML form for user input.
<code><input></code>	Defines an input field where the user can enter data.
<code><textarea></code>	Defines a multiline text input control.
<code><label></code>	Defines a label for an <code><input></code> element.
<code><select></code>	Defines a drop-down list.
<code><button></code>	Defines a clickable button.

Input Types

<code><input type="text"></code>	- Defines a single-line text input field.
<code><input type="password"></code>	- Defines a password input field.
<code><input type="submit"></code>	- Defines a submit button for submitting the form.
<code><input type="radio"></code>	- Defines a radio button for selecting one of many choices.
<code><input type="checkbox"></code>	- Defines a checkbox for selecting zero or more options of a limited number of choices.
<code><input type="file"></code>	- Defines a file-select field and a "Browse" button for file uploads.

Form Attributes

<code>action</code>	Specifies where to send the form-data when a form is submitted.
<code>method</code>	Specifies the HTTP method to use when submitting the form-data (usually <code>GET</code> or <code>POST</code>).
<code>name</code>	Specifies a name for the form (to differentiate the forms).
<code>value</code>	Specifies the value of an input element.
<code>placeholder</code>	Specifies a short hint that describes the expected value of an input field.
<code>required</code>	Specifies that an input field must be filled out before submitting the form.

CSS Layout

Display Property

The <code>display</code> property specifies the display behavior (the type of rendering box) of an element.
Common values include: <code>block</code> , <code>inline</code> , <code>inline-block</code> , <code>flex</code> , <code>grid</code> , <code>none</code> .
<ul style="list-style-type: none"><code>block</code>: An element that takes up the full width available, with line breaks before and after the element.<code>inline</code>: An element that only takes up as much width as necessary, without forcing line breaks.<code>inline-block</code>: Similar to <code>inline</code>, but you can set a width and height for the element.<code>flex</code>: Defines a flexible container; see Flexbox section below.<code>grid</code>: Defines a grid container; see Grid section below.<code>none</code>: The element is completely removed.

Flexbox

<code>display: flex;</code>	Enables flexbox layout for the container.
<code>flex-direction</code>	Specifies the direction of the flex items (<code>row</code> , <code>column</code> , <code>row-reverse</code> , <code>column-reverse</code>).
<code>justify-content</code>	Aligns flex items along the main axis (<code>center</code> , <code>flex-start</code> , <code>flex-end</code> , <code>space-between</code> , <code>space-around</code>).
<code>align-items</code>	Aligns flex items along the cross axis (<code>center</code> , <code>flex-start</code> , <code>flex-end</code> , <code>stretch</code> , <code>baseline</code>).
Example:	<pre>.container { display: flex; justify-content: center; align-items: center; }</pre>

Grid

<code>display: grid;</code>	Enables grid layout for the container.
<code>grid-template-columns</code>	Defines the number and size of columns in the grid.
<code>grid-template-rows</code>	Defines the number and size of rows in the grid.
<code>grid-gap</code>	Specifies the size of the gap between rows and columns.
Example:	<pre>.container { display: grid; grid-template-columns: auto auto auto; grid-gap: 10px; }</pre>