## Day 4

## 1. Displays

**Display Property**: The display property in CSS determines how an element is displayed on the web page. Here are the most common values:

* **block**: The element takes up the full width available, and starts on a new line. Examples include <div>, <h1>, and <p>.

div {

display: block;

}

* **inline**: The element only takes up as much width as necessary and does not start on a new line. Examples include <span>, <a>, and <strong>.

span {

display: inline;

}

* **none**: The element is not displayed at all (it will not take up any space).

.hidden {

display: none;

}

### Example:

<div class="block">Block Element</div>

<span class="inline">Inline Element</span>

<span class="inline-block">Inline Block Element</span>

## 2. Layout and Box Model

The **box model** is fundamental in CSS, as it describes how the dimensions of elements are calculated and how they interact with one another.

### Box Model Components:

* **Content**: The actual content of the box, where text and images appear.
* **Padding**: The space between the content and the border. It is transparent and adds space inside the box.
* **Border**: A line surrounding the padding (if any) and content.
* **Margin**: The space outside the border, which creates distance between elements.

### Box Model Properties:

* **Width and Height**: By default, these properties only set the content size.
* **Padding**: Use padding-top, padding-right, padding-bottom, padding-left, or the shorthand padding.
* **Border**: Use border-width, border-style, and border-color or shorthand border.
* **Margin**: Use margin-top, margin-right, margin-bottom, margin-left, or shorthand margin.

### Example:

.box {

width: 200px;

height: 100px;

padding: 10px;

border: 2px solid black;

margin: 20px;

}

## 3. Background and Borders

### Background Properties:

* **background-color**: Sets the background color of an element.
* **background-image**: Sets an image as the background.
* **background-size**: Defines the size of the background image (e.g., cover, contain).
* **background-position**: Sets the starting position of the background image.

### Border Properties:

* **border**: A shorthand property for setting width, style, and color.
* **border-radius**: Rounds the corners of the element.

### Example:

.box {

background-color: lightblue;

background-image: url('image.jpg');

background-size: cover;

border: 5px solid darkblue;

border-radius: 10px;

}

## Flexbox

**Flexbox** (Flexible Box Layout) is a one-dimensional layout method in CSS that makes it easier to design flexible and responsive layout structures. It allows elements to align, space out, and distribute space within a container.

### Key Concepts

1. **Flex Container**: An element that has a display property set to flex or inline-flex.

.flex-container {

display: flex; /\* or inline-flex \*/

}

1. **Flex Items**: The direct children of a flex container.

### Main Properties

#### Flex Container Properties

* **flex-direction**: Defines the direction in which flex items are placed in the flex container.
  + row (default): Left to right.
  + row-reverse: Right to left.
  + column: Top to bottom.
  + column-reverse: Bottom to top.

.flex-container {

flex-direction: row; /\* row, row-reverse, column, column-reverse \*/

}

* **justify-content**: Aligns flex items along the main axis.
  + flex-start: Items are packed toward the start.
  + flex-end: Items are packed toward the end.
  + center: Items are centered.
  + space-between: Items are evenly distributed; first item at the start, last item at the end.
  + space-around: Items are evenly distributed with space around them.

.flex-container {

justify-content: center; /\* flex-start, flex-end, center, space-between, space-around \*/

}

* **align-items**: Aligns flex items along the cross axis.
  + stretch (default): Items are stretched to fill the container.
  + flex-start: Items are aligned to the start of the cross axis.
  + flex-end: Items are aligned to the end.
  + center: Items are centered along the cross axis.
  + baseline: Items are aligned to their baseline.

.flex-container {

align-items: center; /\* flex-start, flex-end, center, baseline \*/

}

**4. CSS Lists**

## Different List Item Markers

The list-style-type property specifies the type of list item marker.

The following example shows some of the available list item markers

## Remove Default Settings

The list-style-type:none property can also be used to remove the markers/bullets. Note that the list also has default margin and padding. To remove this, add margin:0 and padding:0 to <ul> or <ol>:

HOW TO EDIT PHOTOS