**Display block**

Today, we'll be diving into the topic of "Display block" in CSS. It's an important concept that allows us to control the layout and behavior of elements on a web page.

**Introduction**:

The "display" property in CSS determines how an element is displayed on a web page. The "block" value of the display property is used to render elements as block-level elements. Block-level elements take up the full width of their parent container and start on a new line.

**Real-life example:**

Imagine you are building a recipe website. Each recipe on the website should be displayed as a block, taking up the full width of the container. By using the "display: block" property, you can ensure that each recipe is visually separated and starts on a new line, making it easy for users to navigate and read the recipes.

**Difference from general:**

The "display: block" property is different from the default behavior of elements, which is usually "display: inline" or "display: inline-block". Block-level elements have their own dedicated space and start on a new line, whereas inline elements flow within the text content and do not create line breaks.

**Code-based example:**

Here's an example of how to use the "display: block" property in CSS:

**HTML**:-

<div class="recipe">

<h2>Pancake Recipe</h2>

<p>Ingredients: Flour, milk, eggs, sugar</p>

<p>Instructions: Mix the ingredients, heat the pan, pour the batter, flip the pancake, and enjoy!</p>

</div>

**CSS**:-

.recipe {

display: block;

background-color: #f2f2f2;

padding: 20px;

margin-bottom: 20px;

}

In this example, we have a recipe section represented by the <div> element with the class "recipe". By setting the "display" property to "block", we ensure that each recipe is displayed as a block-level element. We can then apply additional styles such as background color, padding, and margin to create a visually appealing recipe section.

**Display inline**

we'll be discussing the "display: inline" property in CSS. It's another important aspect of controlling the layout and behavior of elements.

**Introduction:**

The "display: inline" property is used to change the default block-level behavior of elements and make them inline elements. It allows elements to flow alongside each other on the same line.

**Introduction**:

Imagine you are creating a navigation menu for a website. You want the menu items to be displayed horizontally in a row, allowing users to easily navigate through the site. By applying the "display: inline" property to the menu items, you can achieve this layout and ensure they are displayed in a single line.

**Difference from general:**

By default, HTML elements are displayed as block-level elements, which means they occupy the entire width of the parent container and start on a new line. However, with the "display: inline" property, elements are displayed inline, allowing them to flow alongside each other horizontally on the same line.

**Code-based example:**

Here's an example of how to use the "display: inline" property in:

**HTML**:-

<div class="navigation">

<a href="#">Home</a>

<a href="#">About</a>

<a href="#">Services</a>

<a href="#">Contact</a>

</div>

**CSS**:-

.navigation a {

display: inline;

margin-right: 10px;

color: #333;

text-decoration: none;

}

In this example, we have a navigation menu consisting of anchor elements. By applying the "display: inline" property to the anchor elements within the navigation div, we ensure they are displayed inline. We also add some additional styles like margin, color, and text decoration to enhance the appearance of the navigation links.

**Display table**

we'll be exploring the "display: table" property in CSS. It's a powerful property that allows us to create table-like layouts for our web pages.

**Introduction**

The "display: table" property in CSS enables us to create table structures for our HTML elements, even if we're not using actual HTML tables. It gives us the flexibility to create complex layouts and arrange elements in a tabular format.

**Real-life example:**

Imagine you're designing a pricing comparison page for a software company. You want to display the features and prices of different subscription plans in a neat and organized manner, resembling a table. Using the "display: table" property, you can achieve this by creating a table-like layout without the actual HTML table structure. This makes it easier to style and customize the layout according to your design requirements.

**Difference from general:**

The "display: table" property allows us to create table-like structures without using the traditional HTML <table> element. Unlike the "display: inline" property, which is used to display elements inline with the surrounding content, "display: table" provides a way to create grid-like layouts with rows and columns, just like a table.

**Code-based example:**

Here's an example of how to use the "display: table" property in CSS:

**HTML**:-

<div class="pricing-table">

<div class="row">

<div class="cell">Plan</div>

<div class="cell">Price</div>

<div class="cell">Features</div>

</div>

<div class="row">

<div

<div class="cell">$10</div>

<div class="cell">Feature 1, Feature 2</div>

</div>

<div class="row">

class="cell">Pro</div>

<div class="cell">$20</div>

<div class="cell">Feature 1, Feature 2, Feature 3</div>

</div>

</div>

**CSS:-**

.pricing-table {

display: table;

width: 100%;

}

.row {

display: table-row;

}

.cell {

display: table-cell;

padding: 10px;

border: 1px solid #ccc;

}

In this example, we create a pricing table using the "display: table" property. We set the outer container with the class "pricing-table" to have a table-like display. The "row" class is used for each row, and the "cell" class represents each cell within the table structure. By using the appropriate display values, we can achieve a table-like layout.