Task 9-Alignment and Display (Gemini)

**Aligning Elements in CSS**

CSS provides several properties to control the alignment of elements on a web page. These properties can be used to align text within elements, align elements vertically or horizontally within their containers, and control the layout of elements.

**Text Alignment**

* **text-align:** Aligns the text content within an element.
  + left: Aligns text to the left.
  + right: Aligns text to the right.
  + center: Centers the text within the element.
  + justify: Justifies the text, aligning it to both the left and right margins.

**Vertical Alignment**

* **vertical-align:** Aligns the content of an inline element vertically relative to its baseline.
  + baseline: Aligns the element's baseline with the baseline of the parent element.
  + top: Aligns the top of the element with the top of the parent element.
  + bottom: Aligns the bottom of the element with the bottom of the parent element.
  + middle: Centers the element vertically within the parent element.

**Display Property**

The display property controls how an element is displayed. Different values can be used to create various layout styles:

* **block:** The element is displayed as a block-level element, taking up the full width of its container and starting on a new line.
* **inline:** The element is displayed as an inline element, sharing the same line with other inline elements.
* **inline-block:** The element is displayed as an inline element, but it can have a width and height set.
* **flex:** The element is displayed as a flex container, allowing for flexible layout and alignment of its child elements.

**Example HTML and CSS**

<!DOCTYPE html>

<html>

<head>

<title>Alignment Example</title>

<style>

.text-center {

    text-align: center;

}

.text-right {

    text-align: right;

}

.vertical-top {

    vertical-align: top;

}

.vertical-middle {

    vertical-align: middle;

}

.flex-container {

    display: flex;

    justify-content: space-between;

}

.flex-item {

    flex: 1;

    border: 1px solid black;

    padding: 10px;

}

</style>

</head>

<body>

<h2>Text Alignment</h2>

<p class="text-center">Centered text</p>

<p class="text-right">Right-aligned text</p>

<h2>Vertical Alignment</h2>

<div>

<span class="vertical-top">Top-aligned</span>

<span class="vertical-middle">Middle-aligned</span>

</div>

<h2>Flexbox Layout</h2>

<div class="flex-container">

<div class="flex-item">Item 1</div>

<div class="flex-item">Item 2</div>

<div class="flex-item">Item 3</div>

</div>

</body>

</html>

This example demonstrates different alignment techniques:

* Text alignment: The first two paragraphs use text-align to center and right-align the text.
* Vertical alignment: The vertical-align property is used to align the spans vertically within their parent div.
* Flexbox layout: The flex-container and flex-item classes use flexbox to create a flexible layout and distribute the items evenly within the container.