Measurements in CSS

In CSS (Cascading Style Sheets), there are several units of measurement that can be used to define the sizes of various elements. Here are some of the most commonly used units and their definitions:

**Pixels (px)** - This is a fixed unit of measurement that represents a single dot on a screen. It is commonly used to define the size of text and other elements.

**Points (pt)** - This is another fixed unit of measurement that is commonly used to define the size of text. One point is equal to 1/72 of an inch.

**Percentages (%)** - This unit of measurement is used to define sizes relative to the size of the parent element. For example, if you set the width of an element to 50%, it will be half the width of its parent element.

**Em** - This is a relative unit of measurement that is based on the font size of the parent element. For example, if the font size of the parent element is 16px and you set the font size of a child element to 2em, it will be 32px.

**Rem** - This is similar to the em unit, but it is based on the font size of the root element (usually the <html> element). This means that it is not affected by the font size of any parent elements.

**Viewport units** - These units are based on the size of the browser viewport. There are four viewport units: vw (viewport width), vh (viewport height), vmin (minimum of viewport width and height), and vmax (maximum of viewport width and height).

**Absolute length units** - These units are fixed and are based on physical measurements. Examples include inches (in), centimeters (cm), and millimeters (mm).

**Flexible length units** - These units are used to define flexible layouts. Examples include fr (fractional unit), which divides the available space into fractions, and min-content, which sizes an element based on its content.

Overall, the choice of unit depends on the context and the desired effect. Using relative units like percentages, ems, and rems is often recommended for responsiveness and scalability.

**Different kinds of selectors**

**Element Selector**

Selects an element on the page based on its tag name. It is represented by the tag name.

**ID Selector**

Selects an element on the page based on its ID attribute. It is represented by a hash (#) followed by the ID name.

**Class Selector**

Selects an element on the page based on its class attribute. It is represented by a dot (.) followed by the class name.

**Attribute Selector**

Selects an element on the page based on its attribute value. It is represented by square brackets ([]) with the attribute name and optional value.

**Universal Selector**

Selects all elements on the page. It is represented by an asterisk (\*).

**Descendant Selector**

Selects an element that is a descendant of another element. It is represented by a space between two selectors.

**Child Selector**

Selects an element that is a direct child of another element. It is represented by a greater than sign (>) between two selectors.

**Adjacent Sibling Selector**

Selects an element that is immediately after another element. It is represented by a plus sign (+) between two selectors.

**General Sibling Selector**

Selects an element that is a sibling of another element. It is represented by a tilde (~) between two selectors.

**Pseudo-classes**

Selects elements based on their state or position. They are represented by a colon (:) followed by the pseudo-class name.

**Pseudo-elements**

Selects parts of an element, such as the first letter or line. They are represented by a double colon (::) followed by the pseudo-element name.

Positioning in CSS

**Inline Elements** cannot have a width or height. You need first to set the display value to ‘inline-block.

The **justify-content: space-between;** CSS property aligns flex items along the main axis (horizontally in a row or vertically in a column) with equal space between them.

When you set the justify-content property to space-between, the flex items inside a container will be evenly spaced out along the main axis, and the first and last items will be positioned at the edges of the container.