Layout with Flexbox

CSS Flexbox

The CSS `display: flex` property sets an HTML element as a block level flex container which takes the full width of its parent container. Any child elements that reside within the flex container are called flex items.
Flex items change their size and location in response to the size and position of their parent container.

```css
div {
  display: flex;
}
```

justify-content Property

The CSS `justify-content` flexbox property defines how the browser distributes space between and around content items along the main-axis of their container. This is when the content items do not use all available space on the major-axis (horizontally).
`justify-content` can have the values of:

- `flex-start`
- `flex-end`
- `center`
- `space-between`
- `space-around`

```css
/* Items based at the center of the parent container: */
div {
  display: flex;
  justify-content: center;
}

/* Items based at the upper-left side of the parent
container: */
div {
  display: flex;
  justify-content: flex-start;
}
```

flex Property

The `flex` CSS property specifies how a flex item will grow or shrink so as to fit within the space available in its `flex` container. This is a shorthand property that declares the following properties in order on a single line:

- `flex-grow`
- `flex-shrink`
- `flex-basis`

```css
/* Three properties declared on three lines: */
.first-flex-item {
  flex-grow: 2;
  flex-shrink: 1;
  flex-basis: 150px;
}


/* Same three properties declared on one line: */
.first-flex-item {
  flex: 2 1 150px;
}
```

flex-direction Property

The `flex-direction` CSS property specifies how flex items are placed in the flex container - either vertically or horizontally. This property also determines whether those flex items appear in order or in reverse order.

```css
div {
  display: flex;
  flex-direction: row-reverse;
}
```

align-content Property

The `align-content` property modifies the behavior of the flex-wrap property. It determines how to space rows from top to bottom (ie. along the cross axis). Multiple rows of items are needed for this property to take effect.

`flex-grow` Property

The CSS `flex-grow` property allows flex items to grow as the parent container increases in size horizontally. This property accepts numerical values and specifies how an element should grow relative to its sibling elements based on this value. The default value for this property is `0`.

```css
.panelA {
  width: 100px;
  flex-grow: 1;
}


/* This panelB element will stretch twice wider than the
panelA element */
.panelB {
  width: 100px;
  flex-grow: 2;
}
```

`flex-shrink` Property

The CSS `flex-shrink` property determines how an element should shrink as the parent container decreases in size horizontally. This property accepts a numerical value which specifies the ratios for the shrinkage of a flex item compared to its other sibling elements within its parent container. The default value for this property is `1`.

```css
.container {
  display: flex;
}

.item-a {
  flex-shrink: 1;
  /* The value 1 indicates that the item should shrink. */
}

.item-b {
  flex-shrink: 2;
  /* The value 2 indicates that the item should shrink twice
than the element item-a. */
}
```

Css flex-basis property

The `flex-basis` CSS property sets the initial base size for a flex item before any other space is distributed according to other flex properties.

```
// Default Syntax
flex-basis: auto;
```

The CSS `flex-flow` property

The CSS property `flex-flow` provides a shorthand for the properties `flex-direction` and `flex-wrap`. The value of the `flex-direction` property specifies the direction of the flex items and the value of the `flex-wrap` property allows flex items to move to the next line instead of shrinking to fit inside the flex container. The `flex-flow` property should be declared on the flex container.

```
// In this example code block, "column" is the value of the
property "flex-direction" and "wrap" is the value of the
property "flex-wrap".

.container {
  display: flex;
  flex-flow: column wrap;
}
```

CSS `display: inline-flex` property

The CSS `display: inline-flex` property sets an HTML element as an inline flex container which takes only the required space for the content. Any child elements that reside within the flex container are called flex items. Flex items change their size and location in response to the size and position of their parent container.

```
.container{
  display: inline-flex;
}
```

Flexbox Properties align-items

When working with CSS flexbox `align-items` is used to align flex items vertically within a parent container.

Css flex-wrap property

The `flex-wrap` property specifies whether flex items should wrap or not. This applies to flex items only. Once you tell your container to `flex-wrap` , wrapping become a priority over shrinking. Flex items will only begin to wrap if their combined `flex-basis` value is `greater` than the current size of their flex container.

```
.container {
  display: flex;
  flex-wrap: wrap;
  width: 200px;
}
```

↓ Print   ⌗ Share ▼