CSS Flexbox Properties

Before the Flexbox Layout module, there were four layout modes:

- Block, for sections in a webpage
- Inline, for text
- Table, for two-dimensional table data
- Positioned, for explicit position of an element

The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.

To start using the flexbox model, you need to first define a flex container by setting the display proper to flex:

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

Other CSS properties that make up a flexbox class are:

- flex-direction
- flex-wrap
- flex-flow
- justify-content
- align-items
- align-content

The Flex-Direction Property

The `flex-direction` property defines in which direction the container would love to stack the flex items (internal boxes). The values that this property can be set to are: column, column-reverse, row, row-reverse.

- The column value stacks the items vertically (from top to bottom)
- The column-reverse value stacks the items vertically (but from bottom to top).

- The row value stacks the items horizontally (from left to right)
- The row-reverse value stacks the items horizontally (but from right to left)

The Flex-Wrap Property

The `flex-wrap` property specifies whether the flex items should wrap or not.

The flex-wrap property takes the following values: wrap, nowrap, wrap-reverse.

- The wrap value specifies that the flex items will wrap if necessary
- The nowrap value specifies that the flex items will not wrap (this is default).
- wrap-reverse value specifies that the flexible items will wrap if necessary, in reverse order.

The Flex-Flow Property

The `flex-flow` property is a shorthand property for setting both the flex-direction and flex-wrap properties.

For example: Instead of

```
.flex-container {
  display: flex;
  flex-direction: row;
  flex-wrap: wrap;
}
```

you can use

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

The Justify-Content Property

The `justify-content` property is used to align the flex items. The values that this property can take are as follows: center, flex-start, flex-end, space-around, space-between.

- The center value aligns the flex items at the center of the container.
- The flex-start value aligns the flex items at the beginning of the container (this is default).
- The flex-end value aligns the flex items at the end of the container
- The space-around value displays the flex items with space before, between, and after the lines.
- The space-between value displays the flex items with space between the lines.

The Align-Items Property

The `align-items` property is used to align the flex items. The values that this property can be set to are: center, flex-start, flex-end, stretch, baseline.

- The center value aligns the flex items in the middle of the container
- The flex-start value aligns the flex items at the top of the container

- The flex-end value aligns the flex items at the base of the container
- The stretch value stretches the flex items to fill the container (this is default)
- The baseline value aligns the flex items based on their text baselines

The Align-Content Property

The `align-content` property is used to align the flex lines. The value that this property can be set to is: space-between, space-around, or stretch.

- The space-between value displays the flex lines with equal space between them.
- The space-around value displays the flex lines with space before, between and after them.
- The stretch value stretches the flex lines to take up the remaining space (this is default).

Perfect Centring

To perform perfect centring using the flexbox collection of properties, set both `justify-content` and `align-items` properties to centre.

For example:

```
.flex-container {
  display: flex;
  height: 300px;
  justify-content: center;
  align-items: center;
}
```

Click to Ask Question(s)



MR YOUNG

IGWURUTA | ISIOKPO | ELELE | IKIRI | OMOKU

mryoung.presmaltra.com.ng