



## Flexbox Fundamentals

### Enabling Flexbox

To use flexbox, set the <code>display</code> property of a container to <code>flex</code> or <code>inline-flex</code> .
<code>.container { display: flex; /* or inline-flex */ }</code>
<code>flex</code> : Creates a block-level flex container.
<code>inline-flex</code> : Creates an inline-level flex container.

### Flex Direction

<code>flex-direction: row;</code>	Items are placed side by side (horizontally) - Default value.
<code>flex-direction: row-reverse;</code>	Items are placed side by side in reverse order (right to left).
<code>flex-direction: column;</code>	Items are placed top to bottom (vertically).
<code>flex-direction: column-reverse;</code>	Items are placed bottom to top (vertically reversed).

### Flex Wrap

<code>flex-wrap: nowrap;</code>	Items stay in a single line, even if they overflow the container - Default value.
<code>flex-wrap: wrap;</code>	Items wrap to the next line when they overflow the container.
<code>flex-wrap: wrap-reverse;</code>	Items wrap to the previous line when they overflow the container.

## Justify Content & Align Items

### Justify Content

<code>justify-content: flex-start;</code>	Items are aligned to the start of the container - Default value.
<code>justify-content: flex-end;</code>	Items are aligned to the end of the container.
<code>justify-content: center;</code>	Items are centered in the container.
<code>justify-content: space-between;</code>	Items are evenly distributed; first item is at the start, last item is at the end.
<code>justify-content: space-around;</code>	Items are evenly distributed with equal space around each item.
<code>justify-content: space-evenly;</code>	Items are distributed so that the spacing between any two items (neither the first nor the last) is the same.

### Align Items

<code>align-items: stretch;</code>	Items are stretched to fit the container - Default value.
<code>align-items: flex-start;</code>	Items are aligned to the top of the container.
<code>align-items: flex-end;</code>	Items are aligned to the bottom of the container.
<code>align-items: center;</code>	Items are vertically centered in the container.
<code>align-items: baseline;</code>	Items are aligned based on their baselines.

## Flex Item Properties

### Flex Grow

<code>flex-grow: 0;</code>	Item does not grow to fill available space - Default value.
<code>flex-grow: 1;</code>	Item grows to fill available space.
<code>flex-grow: n;</code>	Item grows to fill n times the available space compared to other items with <code>flex-grow</code> .

### Flex Shrink

<code>flex-shrink: 1;</code>	Item can shrink if necessary - Default value.
<code>flex-shrink: 0;</code>	Item does not shrink, even if it overflows.
<code>flex-shrink: n;</code>	Item shrinks n times more than other items with <code>flex-shrink</code> .

### Flex Shorthand

The <code>flex</code> property is a shorthand for <code>flex-grow</code> , <code>flex-shrink</code> , and <code>flex-basis</code> .
<code>flex: flex-grow flex-shrink flex-basis;</code>
<code>flex: 1;</code> - Equivalent to <code>flex: 1 1 0%</code> ; (item grows to fill space, can shrink, basis is 0).
<code>flex: 2 1 200px;</code> - Item grows twice as much as others, can shrink, and has a basis of 200px.
<code>flex: initial;</code> - Resets to default values: <code>0 1 auto</code>
<code>flex: auto;</code> - Sets to <code>1 1 auto</code>
<code>flex: none;</code> - Sets to <code>0 0 auto</code>

### Flex Basis

<code>flex-basis: auto;</code>	Item size is based on its content - Default value.
<code>flex-basis: 0;</code>	Item size is based on <code>flex-grow</code> and <code>flex-shrink</code> .
<code>flex-basis: length;</code>	Item has a fixed size (e.g., <code>flex-basis: 100px</code> ).
<code>flex-basis: content;</code>	Keyword specifying an automatic size based on the element's content.

## Advanced Flexbox

## Align Self

<code>align-self: auto;</code>	Item inherits the <code>align-items</code> value from the container - Default value.
<code>align-self: flex-start;</code>	Item is aligned to the top of the container, overriding the container's <code>align-items</code> value.
<code>align-self: flex-end;</code>	Item is aligned to the bottom of the container, overriding the container's <code>align-items</code> value.
<code>align-self: center;</code>	Item is vertically centered in the container, overriding the container's <code>align-items</code> value.
<code>align-self: baseline;</code>	Item is aligned based on its baseline, overriding the container's <code>align-items</code> value.
<code>align-self: stretch;</code>	Item is stretched to fit the container, overriding the container's <code>align-items</code> value.

## Order

<code>order: 0;</code>	Items are displayed in their source order - Default value.
<code>order: n;</code>	Items are reordered based on the numerical value of the <code>order</code> property. Lower values are displayed first.
Note	Items with the same order value will be displayed in their source order.

## Align Content

<code>align-content: stretch;</code>	Lines stretch to take up the remaining space - Default value.
<code>align-content: flex-start;</code>	Lines are packed at the start of the container.
<code>align-content: flex-end;</code>	Lines are packed at the end of the container.
<code>align-content: center;</code>	Lines are packed in the center of the container.
<code>align-content: space-between;</code>	Lines are evenly distributed; the first line is at the start, the last is at the end.
<code>align-content: space-around;</code>	Lines are evenly distributed with equal space around each line.