Parent Element (Container)

Example

.flex-container {  
  display: flex;  
}

The flex container properties are:

* [flex-direction](https://www.w3schools.com/css/css3_flexbox_container.asp#flex-direction)
* [flex-wrap](https://www.w3schools.com/css/css3_flexbox_container.asp#flex-wrap)
* [flex-flow](https://www.w3schools.com/css/css3_flexbox_container.asp#flex-flow)
* [justify-content](https://www.w3schools.com/css/css3_flexbox_container.asp#justify-content)
* [align-items](https://www.w3schools.com/css/css3_flexbox_container.asp#align-items)
* [align-content](https://www.w3schools.com/css/css3_flexbox_container.asp#align-content)

The flex-direction Property

The flex-direction property defines in which direction the container wants to stack the flex items.

The column value stacks the flex items vertically (from top to bottom):

.flex-container {  
  display: flex;  
  flex-direction: column;  
}

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

.flex-container {  
  display: flex;  
  flex-direction: column-reverse;  
}

The row value stacks the flex items horizontally (from left to right):

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

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

.flex-container {  
  display: flex;  
  flex-direction: row-reverse;  
}

The flex-wrap Property

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

The examples below have 12 flex items, to better demonstrate the flex-wrap property.

Example

The wrap value specifies that the flex items will wrap if necessary:

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

The nowrap value specifies that the flex items will not wrap (this is default):

.flex-container {  
  display: flex;  
  flex-wrap: nowrap;  
}

The wrap-reverse value specifies that the flexible items will wrap if necessary, in reverse order:

.flex-container {  
  display: flex;  
  flex-wrap: wrap-reverse;  
}

The flex-flow Property

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

Example

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

The justify-content Property

The justify-content property is used to align the flex items:

Example

The center value aligns the flex items at the center of the container:

.flex-container {  
  display: flex;  
  justify-content: center;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss3_flexbox_justify-content_center)

Example

The flex-start value aligns the flex items at the beginning of the container (this is default):

.flex-container {  
  display: flex;  
  justify-content: flex-start;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss3_flexbox_justify-content_flex-start)

Example

The flex-end value aligns the flex items at the end of the container:

.flex-container {  
  display: flex;  
  justify-content: flex-end;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss3_flexbox_justify-content_flex-end)

Example

The space-around value displays the flex items with space before, between, and after the lines:

.flex-container {  
  display: flex;  
  justify-content: space-around;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss3_flexbox_justify-content_space-around)

Example

The space-between value displays the flex items with space between the lines:

.flex-container {  
  display: flex;  
  justify-content: space-between;  
}

The align-items Property

The align-items property is used to align the flex items.

In these examples we use a 200 pixels high container, to better demonstrate the align-items property.

Example

The center value aligns the flex items in the middle of the container:

.flex-container {  
  display: flex;  
  height: 200px;  
  align-items: center;  
}

Example

The flex-start value aligns the flex items at the top of the container:

.flex-container {  
  display: flex;  
  height: 200px;  
  align-items: flex-start;  
}

The flex-end value aligns the flex items at the bottom of the container:

.flex-container {  
  display: flex;  
  height: 200px;  
  align-items: flex-end;  
}

The stretch value stretches the flex items to fill the container (this is default):

.flex-container {  
  display: flex;  
  height: 200px;  
  align-items: stretch;  
}

The baseline value aligns the flex items such as their baselines aligns:

.flex-container {  
  display: flex;  
  height: 200px;  
  align-items: baseline;  
}

**Note:** the example uses different font-size to demonstrate that the items gets aligned by the text baseline:

The align-content Property

The align-content property is used to align the flex lines.

In these examples we use a 600 pixels high container, with the flex-wrap property set to wrap, to better demonstrate the align-content property.

The space-between value displays the flex lines with equal space between them:

.flex-container {  
  display: flex;  
  height: 600px;  
  flex-wrap: wrap;  
  align-content: space-between;  
}

The space-around value displays the flex lines with space before, between, and after them:

.flex-container {  
  display: flex;  
  height: 600px;  
  flex-wrap: wrap;  
  align-content: space-around;  
}

The stretch value stretches the flex lines to take up the remaining space (this is default):

.flex-container {  
  display: flex;  
  height: 600px;  
  flex-wrap: wrap;  
  align-content: stretch;  
}

The center value displays display the flex lines in the middle of the container:

.flex-container {  
  display: flex;  
  height: 600px;  
  flex-wrap: wrap;  
  align-content: center;  
}

The flex-start value displays the flex lines at the start of the container:

.flex-container {  
  display: flex;  
  height: 600px;  
  flex-wrap: wrap;  
  align-content: flex-start;  
}

The flex-end value displays the flex lines at the end of the container:

.flex-container {  
  display: flex;  
  height: 600px;  
  flex-wrap: wrap;  
  align-content: flex-end;  
}

Perfect Centering

In the following example we will solve a very common style problem: perfect centering.

**SOLUTION:** Set both the justify-content and align-items properties to center, and the flex item will be perfectly centered:

Example

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

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss3_flexbox_perfect_center)

The CSS Flexbox Container Properties

The following table lists all the CSS Flexbox Container properties:

|  |  |
| --- | --- |
| **Property** | **Description** |
| [align-content](https://www.w3schools.com/cssref/css3_pr_align-content.asp) | Modifies the behavior of the flex-wrap property. It is similar to align-items, but instead of aligning flex items, it aligns flex lines |
| [align-items](https://www.w3schools.com/cssref/css3_pr_align-items.asp) | Vertically aligns the flex items when the items do not use all available space on the cross-axis |
| [display](https://www.w3schools.com/cssref/pr_class_display.asp) | Specifies the type of box used for an HTML element |
| [flex-direction](https://www.w3schools.com/cssref/css3_pr_flex-direction.asp) | Specifies the direction of the flexible items inside a flex container |
| [flex-flow](https://www.w3schools.com/cssref/css3_pr_flex-flow.asp) | A shorthand property for flex-direction and flex-wrap |
| [flex-wrap](https://www.w3schools.com/cssref/css3_pr_flex-wrap.asp) | Specifies whether the flex items should wrap or not, if there is not enough room for them on one flex line |
| [justify-content](https://www.w3schools.com/cssref/css3_pr_justify-content.asp) | Horizontally aligns the flex items when the items do not use all available space on the main-axis |

Child Elements (Items)

The direct child elements of a flex container automatically becomes flexible (flex) items.

The element above represents four blue flex items inside a grey flex container.

<div class="flex-container">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
</div>

The flex item properties are:

* [order](https://www.w3schools.com/css/css3_flexbox_items.asp#order)
* [flex-grow](https://www.w3schools.com/css/css3_flexbox_items.asp#flex-grow)
* [flex-shrink](https://www.w3schools.com/css/css3_flexbox_items.asp#flex-shrink)
* [flex-basis](https://www.w3schools.com/css/css3_flexbox_items.asp#flex-basis)
* [flex](https://www.w3schools.com/css/css3_flexbox_items.asp#flex)
* [align-self](https://www.w3schools.com/css/css3_flexbox_items.asp#align-self)

The order Property

The first flex item in the code does not have to appear as the first item in the layout.

The order value must be a number, default value is 0.

The *order* property can change the order of the flex items:

<div class="flex-container">  
  <div style="order: 3">1</div>  
  <div style="order: 2">2</div>  
  <div style="order: 4">3</div>  
  <div style="order: 1">4</div>  
</div>

The flex-grow Property

The flex-grow property specifies how much a flex item will grow relative to the rest of the flex items.

The value must be a number, default value is 0.

Make the third flex item grow eight times faster than the other flex items:

<div class="flex-container">  
  <div style="flex-grow: 1">1</div>  
  <div style="flex-grow: 1">2</div>  
  <div style="flex-grow: 8">3</div>  
</div>

The flex-shrink Property

The flex-shrink property specifies how much a flex item will shrink relative to the rest of the flex items.

The value must be a number, default value is 1.

Do not let the third flex item shrink as much as the other flex items:

<div class="flex-container">  
  <div>1</div>  
  <div>2</div>  
  <div style="flex-shrink: 0">3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
  <div>7</div>  
  <div>8</div>  
  <div>9</div>  
  <div>10</div>  
</div>

The flex-basis Property

The flex-basis property specifies the initial length of a flex item.

Set the initial length of the third flex item to 200 pixels:

<div class="flex-container">  
  <div>1</div>  
  <div>2</div>  
  <div style="flex-basis: 200px">3</div>  
  <div>4</div>  
</div>

The flex Property

The flex property is a shorthand property for the flex-grow, flex-shrink, and flex-basis properties.

Make the third flex item not growable (0), not shrinkable (0), and with an initial length of 200 pixels:

<div class="flex-container">  
  <div>1</div>  
  <div>2</div>  
  <div style="flex: 0 0 200px">3</div>  
  <div>4</div>  
</div>

The align-self Property

The align-self property specifies the alignment for the selected item inside the flexible container.

The align-self property overrides the default alignment set by the container's align-items property.

In these examples we use a 200 pixels high container, to better demonstrate the align-self property:

Align the third flex item in the middle of the container:

<div class="flex-container">  
  <div>1</div>  
  <div>2</div>  
  <div style="align-self: center">3</div>  
  <div>4</div>  
</div>

Align the second flex item at the top of the container, and the third flex item at the bottom of the container:

<div class="flex-container">  
  <div>1</div>  
  <div style="align-self: flex-start">2</div>  
  <div style="align-self: flex-end">3</div>  
  <div>4</div>  
</div>

The CSS Flexbox Items Properties

The following table lists all the CSS Flexbox Items properties:

|  |  |
| --- | --- |
| **Property** | **Description** |
| [align-self](https://www.w3schools.com/cssref/css3_pr_align-self.asp) | Specifies the alignment for a flex item (overrides the flex container's align-items property) |
| [flex](https://www.w3schools.com/cssref/css3_pr_flex.asp) | A shorthand property for the flex-grow, flex-shrink, and the flex-basis properties |
| [flex-basis](https://www.w3schools.com/cssref/css3_pr_flex-basis.asp) | Specifies the initial length of a flex item |
| [flex-grow](https://www.w3schools.com/cssref/css3_pr_flex-grow.asp) | Specifies how much a flex item will grow relative to the rest of the flex items inside the same container |
| [flex-shrink](https://www.w3schools.com/cssref/css3_pr_flex-shrink.asp) | Specifies how much a flex item will shrink relative to the rest of the flex items inside the same container |
| [order](https://www.w3schools.com/cssref/css3_pr_order.asp) | Specifies the order of the flex items inside the same container |