

Flexbox in CSS

display: flex;

The `display: flex;` property is applied to a container to enable the Flexbox layout. It transforms an element into a flex container, allowing its direct children to become flexible items.

flex-direction:

This property determines the main axis of the flex container and the direction of the flex items. It accepts values such as `row` (default), `row-reverse`, `column`, and `column-reverse`.

- `row`: Items are placed along the horizontal axis.
- `row-reverse`: Items are placed along the horizontal axis in the opposite order.
- `column`: Items are placed along the vertical axis.
- `column-reverse`: Items are placed along the vertical axis in the opposite order.

justify-content:

This property aligns flex items along the main axis of the flex container. It controls the distribution of space between and around the items.

- `flex-start`: Items are packed towards the start of the main axis.
- `flex-end`: Items are packed towards the end of the main axis.
- `center`: Items are centered along the main axis.
- `space-between`: Items are evenly distributed along the main axis.
- `space-around`: Items are evenly distributed with equal space around them.
- `space-evenly`: Items are evenly distributed with equal space around them, including at the start and end.

flex-wrap:

This property determines whether flex items should wrap into multiple lines if they exceed the container's width.

- `nowrap`: Default value. Items will not wrap to the next line.
- `wrap`: Items will wrap to the next line if needed.
- `wrap-reverse`: Items will wrap to the next line in reverse order.

align-items:

This property aligns flex items along the cross axis of the flex container.

- `stretch`: Default value. Items are stretched to fill the container.
- `flex-start`: Items are aligned at the start of the cross axis.
- `flex-end`: Items are aligned at the end of the cross axis.
- `center`: Items are centered along the cross axis.
- `baseline`: Items are aligned based on their baselines.

align-self:

This property allows individual flex items to override the `align-items` value for their cross-axis alignment.

- `auto`: Uses the value of `align-items` of the parent.
- `flex-start`, `flex-end`, `center`, `baseline`, `stretch`: Similar to `align-items`, but applies only to the specific item.

flex-basis:

This property sets the initial size of a flex item along the main axis before free space is distributed.

flex-grow:

This property specifies how much a flex item should grow relative to the rest of the items in the flex container when there's extra space along the main axis.

flex-shrink:

This property specifies how much a flex item should shrink relative to the rest of the items in the flex container when there's not enough space along the main axis.

These Flexbox properties provide a powerful and responsive layout system in CSS, enabling developers to create dynamic and flexible designs. Understanding how these properties work together is crucial for mastering Flexbox layouts.

```
<!DOCTYPE html>  
<html lang="en">
```

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <style>
    .flex-container {
      display: flex;
      flex-direction: row;
      justify-content: space-between;
      align-items: center;
      flex-wrap: wrap;
    }

    .flex-item {
      width: 100px;
      height: 100px;
      margin: 10px;
      background-color: #3498db;
      color: #fff;
      display: flex;
      align-items: center;
      justify-content: center;
    }
  </style>
</head>
<body>

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

</body>
</html>
```