

#### 4) Flexbox items alignment properties

- The flexbox items are aligned at the start of the cross axis.
- By default, the cross axis is vertical. This means the flexbox items will be aligned vertically at the top.
- The flexbox items are aligned at the end of the cross axis.
- With these alignment properties we aligned flexbox items across main axis.

We have following properties for Flexbox items alignment :-

- a) justify-content
- b) align-items
- c) align-content
- d) align-self

##### a) justify-content

- This property aligns the flexbox items horizontally and along the main axis when flex-direction is row.
- This property aligns the flexbox items vertically and along the main axis when flex-direction is column.
- The justify-content property accepts following values : flex-start(default) / flex-end / center / space-between / space-around / space-evenly.

##### Syntax

justify-content: flex-start | flex-end | center | space-between | space-around  
| space-evenly

## Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      justify-content: center;
    }
    .items
    {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1
    {
      background-color: red;
    }
```

```
.item2    {  
    background-color: green;  
}  
.item3    {  
    background-color: yellow;  
}  
.item4    {  
    background-color: indigo;  
}  
.item5    {  
    background-color: blue;  
}  
.item6    {  
    background-color: green;  
}  
.item7    {  
    background-color: pink;  
}
```

```
</style>  
</head>  
<body>
```

```
<div class="container">  
    <div class="items item1">Item1</div>  
    <div class="items item2">Item2</div>  
    <div class="items item3">Item3</div>  
    <div class="items item4">Item4</div>  
    <div class="items item5">Item5</div>  
    <div class="items item6">Item6</div>  
    <div class="items item7">Item7</div>  
</div>
```

```
</body>  
</html>
```

### Example -- justify-content:space-evenly

With this value every flex items have equal space around it.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html,body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      justify-content: space-evenly;
    }
    .items {
      color:#fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1 {
      background-color: red;
    }
```

```
.item2    {
  background-color: green;
}
.item3    {
  background-color: yellow;
}
.item4    {
  background-color: indigo;
}
.item5    {
  background-color: blue;
}
.item6    {
  background-color: green;
}
.item7    {
  background-color: pink;
}
</style>
</head>
<body>

<div class="container">
  <div class="items item1">Item1</div>
  <div class="items item2">Item2</div>
  <div class="items item3">Item3</div>
  <div class="items item4">Item4</div>
  <div class="items item5">Item5</div>
  <div class="items item6">Item6</div>
  <div class="items item7">Item7</div>
</div>

</body>
</html>
```

## b) align-items

- This property aligns the flexbox items vertically and along the cross axis when flex-direction is row.
- This property aligns the flexbox items horizontally and along the cross axis when flex-direction is column.
- The align-items property accepts following values : flex-start / flex-end / center / baseline / stretch(default).

### Syntax

align-items: flex-start | flex-end | center | baseline | stretch



## Example -- align-items:stretch

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html,body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      align-items: stretch;
    }
    .items {
      color:#fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1 {
      background-color: red;
    }
```

```
.item2    {
    background-color: green;
}
.item3    {
    background-color: yellow;
}
.item4    {
    background-color: indigo;
}
.item5    {
    background-color: blue;
}
.item6    {
    background-color: green;
}
.item7    {
    background-color: pink;
}
</style>
</head>
<body>

<div class="container">
    <div class="items item1">Item1</div>
    <div class="items item2">Item2</div>
    <div class="items item3">Item3</div>
    <div class="items item4">Item4</div>
    <div class="items item5">Item5</div>
    <div class="items item6">Item6</div>
    <div class="items item7">Item7</div>
</div>

</body>
</html>
```



### Example -- align-items : center

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      align-items: center;
    }
    .items
    {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1
    {
      background-color: red;
    }
  </style>
</html>
```

```
.item2    {  
    background-color: green;  
}  
.item3    {  
    background-color: yellow;  
}  
.item4    {  
    background-color: indigo;  
}  
.item5    {  
    background-color: blue;  
}  
.item6    {  
    background-color: green;  
}  
.item7    {  
    background-color: pink;  
}
```

```
</style>  
</head>  
<body>
```

```
<div class="container">  
  <div class="items item1">Item1</div>  
  <div class="items item2">Item2</div>  
  <div class="items item3">Item3</div>  
  <div class="items item4">Item4</div>  
  <div class="items item5">Item5</div>  
  <div class="items item6">Item6</div>  
  <div class="items item7">Item7</div>  
</div>
```

```
</body>  
</html>
```

## align-items : baseline

- The baseline value will align flex items along their text content's baseline.
- But if content size varies among flex items, then baseline can make a noticeable difference.
- In terms of where baseline alignment occurs, that is determined by the tallest item in the row.

### Example : before baseline

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html, body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      align-items: flex-start;
    }
    .items {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
```

```
.item1    {
  background-color: red;
}
.item2    {
  background-color: green;
  font-size: 40px;
  height: 200px;
}
.item3    {
  background-color: yellow;
}
.item4    {
  background-color: indigo;
}
.item5    {
  background-color: blue;
}
.item6    {
  background-color: green;
  font-size: 10px;
}
.item7    {
  background-color: pink;
}
</style>
</head>
<body>
  <div class="container">
    <div class="items item1">Item1</div>
    <div class="items item2">Item2</div>
    <div class="items item3">Item3</div>
    <div class="items item4">Item4</div>
    <div class="items item5">Item5</div>
    <div class="items item6">Item6</div>
    <div class="items item7">Item7</div>
  </div>
</body>
</html>
```

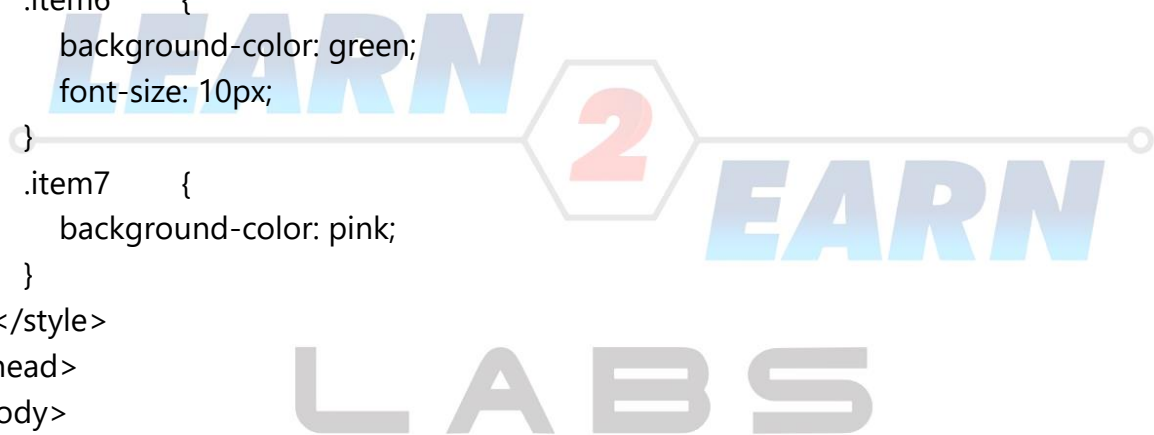


## Example -- after baseline

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html,body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      align-items: baseline;
    }
    .items {
      color:#fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1 {
      background-color: red;
    }
  </style>

```

```
.item2    {
    background-color: green;
    font-size: 40px;
    height: 200px;
}
.item3    {
    background-color: yellow;
}
.item4    {
    background-color: indigo;
}
.item5    {
    background-color: blue;
}
.item6    {
    background-color: green;
    font-size: 10px;
}
.item7    {
    background-color: pink;
}
</style>
</head>
<body>
    <div class="container">
        <div class="items item1">Item1 </div>
        <div class="items item2">Item2 </div>
        <div class="items item3">Item3 </div>
        <div class="items item4">Item4 </div>
        <div class="items item5">Item5 </div>
        <div class="items item6">Item6 </div>
        <div class="items item7">Item7 </div>
    </div>
</body>
</html>
```



### c) align-content

- This property aligns the flexbox items vertically and along the main axis when flex-direction is row.
- Align-content property always work with flex-wrap property. If we don't use flex-wrap property then align-content will not work.
- This property have following values : flex-start / flex-end / stretch(default) / space-around / space-between / space-evenly / center.
- The difference between align-items & align-content is that the align-items is use for vertical alignment of single row, while the align-content is use for multiple rows alignment.

#### Syntax

align-content: flex-start | flex-end | center | space-between | space-around  
| stretch

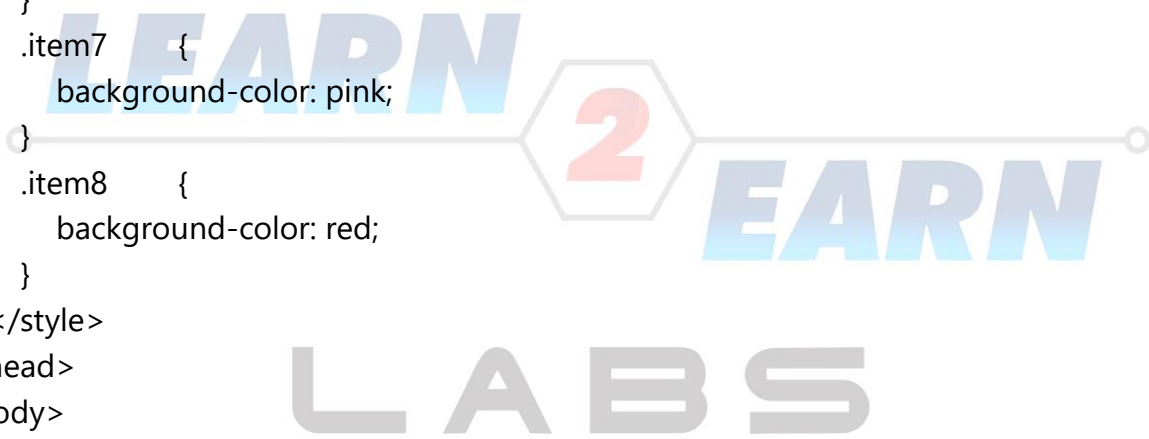


## Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      width: 300px;
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
      align-content: flex-end;
    }
    .items
    {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1
    {
      background-color: red;
    }
```



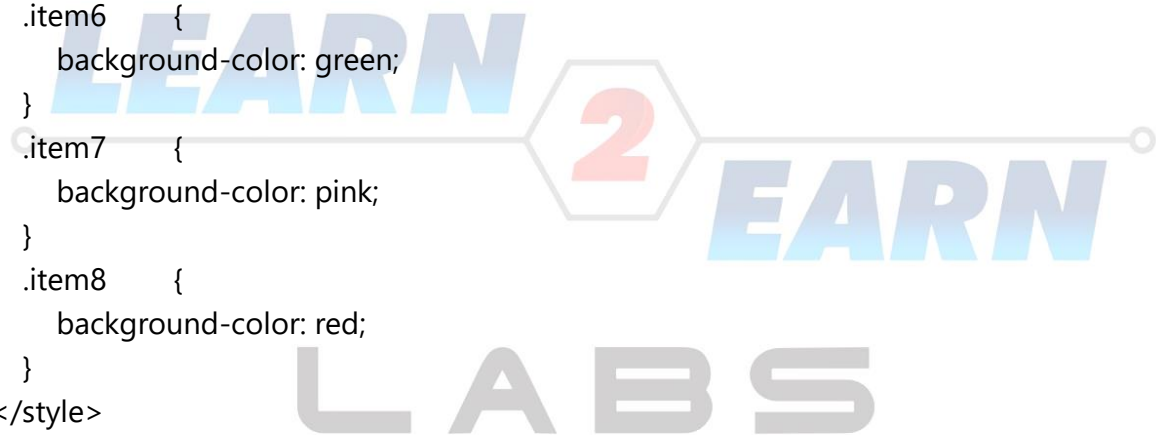
```
.item2    {
  background-color: green;
}
.item3    {
  background-color: yellow;
}
.item4    {
  background-color: indigo;
}
.item5    {
  background-color: blue;
}
.item6    {
  background-color: green;
}
.item7    {
  background-color: pink;
}
.item8    {
  background-color: red;
}
</style>
</head>
<body>
  <div class="container">
    <div class="items item1">Item1 </div>
    <div class="items item2">Item2 </div>
    <div class="items item3">Item3 </div>
    <div class="items item4">Item4 </div>
    <div class="items item5">Item5 </div>
    <div class="items item6">Item6 </div>
    <div class="items item7">Item7 </div>
    <div class="items item8">Item8 </div>
  </div>
</body>
</html>
```

The logo for 'Learn 2 Earn Labs' is centered in the background. It features the word 'LEARN' in blue, a large red number '2' inside a hexagon, the word 'EARN' in blue, and the word 'LABS' in large grey letters below. A horizontal line with circles at both ends passes behind the '2'.

### Example -- align-content : space-evenly

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      width: 300px;
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
      align-content: space-evenly;
    }
    .items
    {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="items">
      <div class="item">
        <div class="img">
          <img alt="A large, faint watermark reading 'LEARN 2 EARN LABS' is visible across the center of the page. The '2' is inside a hexagon, and 'EARN' is partially visible on the right. The text 'LABS' is at the bottom."/>
        <div class="text">
          <h1>LEARN</h1>
          <h1>EARN</h1>
          <h1>LABS</h1>
        </div>
      </div>
    </div>
  </div>
</body>
</html>
```

```
.item1    {
    background-color: red;
}
.item2    {
    background-color: green;
}
.item3    {
    background-color: yellow;
}
.item4    {
    background-color: indigo;
}
.item5    {
    background-color: blue;
}
.item6    {
    background-color: green;
}
.item7    {
    background-color: pink;
}
.item8    {
    background-color: red;
}
</style>
</head>
<body>
    <div class="container">
        <div class="items item1">Item1</div>
        <div class="items item2">Item2</div>
        <div class="items item3">Item3</div>
        <div class="items item4">Item4</div>
        <div class="items item5">Item5</div>
        <div class="items item6">Item6</div>
        <div class="items item7">Item7</div>
        <div class="items item8">Item8</div>
    </div>
</body>
</html>
```



### Example -- Important

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html,body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      width: 300px;
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
      align-content: space-between;
    }
    .items {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1 {
      background-color: red;
    }
    .item2 {
      background-color: green;
    }
  </style>
</html>
```

```
.item3    {
  background-color: yellow;
}
.item4    {
  background-color: indigo;
}
.item5    {
  background-color: blue;
}
.item6    {
  background-color: green;
}
.item7    {
  background-color: pink;
}
.item8    {
  background-color: red;
}
</style>
</head>
<body>
  <div class="container">
    <div class="items item1">Item1</div>
    <div class="items item2">Item2</div>
    <div class="items item3">Item3</div>
    <!-- <div class="items item4">Item4</div>
    <div class="items item5">Item5</div>
    <div class="items item6">Item6</div>
    <div class="items item7">Item7</div>
    <div class="items item8">Item8</div> -->
  </div>
</body>
</html>
```

Note :- Here, "align-content: space-between" not working properly cause "align-content" would not be working for single row flex-items.

#### d) align-self

- This property works for the alignment of flex items vertically when "flex-direction is row".
- If the container has align-items: center and the target has align-self: baseline , only the target will be at the baseline of the cross axis. By default, this means it will be aligned along the baseline of the text.
- This property have following values : auto / flex-start / flex-end / center / stretch / baseline

#### Syntax

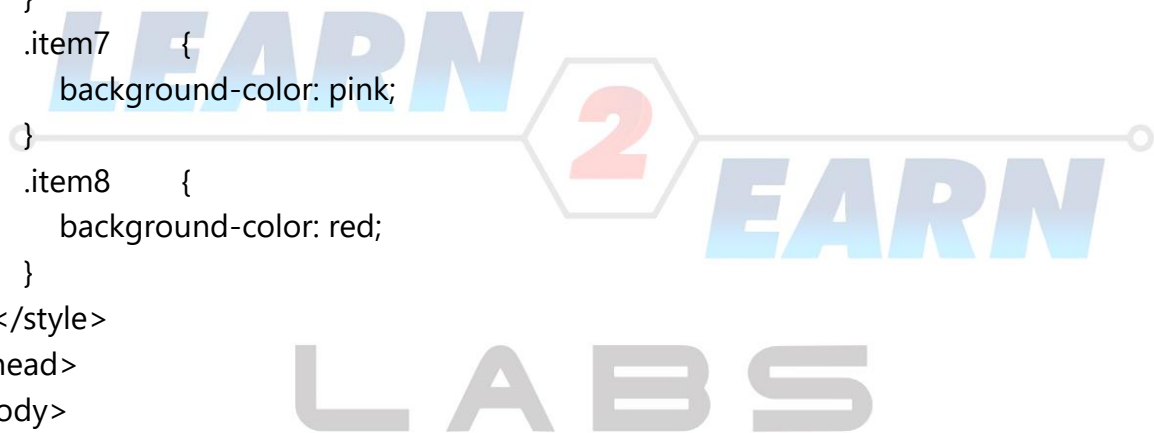
align-self : flex-start(default) / flex-end / center / stretch / baseline / auto



## Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      width: 300px;
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
      align-items: flex-start;
    }
    .items
    {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1
    {
      background-color: red;
      align-self: center;
    }
```

```
.item2    {
  background-color: green;
}
.item3    {
  background-color: yellow;
}
.item4    {
  background-color: indigo;
}
.item5    {
  background-color: blue;
}
.item6    {
  background-color: green;
}
.item7    {
  background-color: pink;
}
.item8    {
  background-color: red;
}
</style>
</head>
<body>
  <div class="container">
    <div class="items item1">Item1</div>
    <div class="items item2">Item2</div>
    <div class="items item3">Item3</div>
    <div class="items item4">Item4</div>
    <div class="items item5">Item5</div>
    <div class="items item6">Item6</div>
    <div class="items item7">Item7</div>
    <div class="items item8">Item8</div>
  </div>
</body>
</html>
```





### Example -- align-self:baseline

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      width: 300px;
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
      align-items: center;
    }
    .items
    {
      color:#fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1
    {
      background-color: red;
      font-size: 10px;
    }
```

```
.item2    {  
    background-color: green;  
}  
.item3    {  
    background-color: yellow;  
    height: 100px;  
    font-size: 20px;  
    align-self: baseline;  
}  
</style>  
</head>  
<body>  
  
<div class="container">  
    <div class="items item1">Item1 </div>  
    <div class="items item2">Item2 </div>  
    <div class="items item3">Item3 </div>  
</div>  
  
</body>  
</html>
```

LEARN  
2  
EARN  
LABS

### Example -- align-self:auto

When we apply "align-self:auto" then the flex\_items(child container) inherits the parent align-item property.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html, body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      width: 300px;
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
      align-items: center;
    }
    .items {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
```

```
.item1    {  
    background-color: red;  
}  
.item2    {  
    background-color: green;  
}  
.item3    {  
    background-color: yellow;  
    align-self: auto;  
}  
</style>  
</head>  
<body>  
  
<div class="container">  
    <div class="items item1">Item1</div>  
    <div class="items item2">Item2</div>  
    <div class="items item3">Item3</div>  
</div>  
  
</body>  
</html>
```

LABS

### Example -- align-self:stretch

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      width: 300px;
      height: 300px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
      align-items: center;
    }
    .items
    {
      color:#fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1
    {
      background-color: red;
    }
```

```
.item2    {  
    background-color: green;  
}  
.item3    {  
    background-color: yellow;  
    align-self: stretch;  
}  
</style>  
</head>  
<body>  
  
    <div class="container">  
        <div class="items item1">Item1 </div>  
        <div class="items item2">Item2 </div>  
        <div class="items item3">Item3 </div>  
    </div>  
</body>  
</html>
```



LABS

## 5) Ordering Flexbox items

- The order CSS property sets the order to layout an item in a flex container. Items in a container are sorted by ascending order value and then by their source code order.
- For Ordering flex-items we use "order" property.
- By default the order of every flex-items is 0.

### Syntax

`order : order_number`



## Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html,body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      width: 600px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
    }
    .items {
      color:#fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1 {
      background-color: red;
    }
  </style>
</head>
```



```
.item2    {
    background-color: green;
    order: 1;
}
.item3    {
    background-color: yellow;
}
.item4    {
    background-color: indigo;
    order: 2;
}
.item5    {
    background-color: blue;
}
</style>
</head>
<body>
<div class="container">
    <div class="items item1">Item1 </div>
    <div class="items item2">Item2 </div>
    <div class="items item3">Item3 </div>
    <div class="items item4">Item4 </div>
    <div class="items item5">Item5 </div>
</div>
</body>
</html>
```

### Example -- When order is -1

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      width: 600px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
    }
    .items
    {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
    }
    .item1
    {
      background-color: red;
    }
```

```
.item2    {
    background-color: green;
    order: 1;
}
.item3    {
    background-color: yellow;
}
.item4    {
    background-color: indigo;
    order: 2;
}
.item5    {
    background-color: blue;
    order: -1;
}
</style>
</head>
<body>

<div class="container">
    <div class="items item1">Item1 </div>
    <div class="items item2">Item2 </div>
    <div class="items item3">Item3 </div>
    <div class="items item4">Item4 </div>
    <div class="items item5">Item5 </div>
</div>

</body>
</html>
```

## 6) flex-grow / flex-basis / flex-shrink

### a) flex-grow

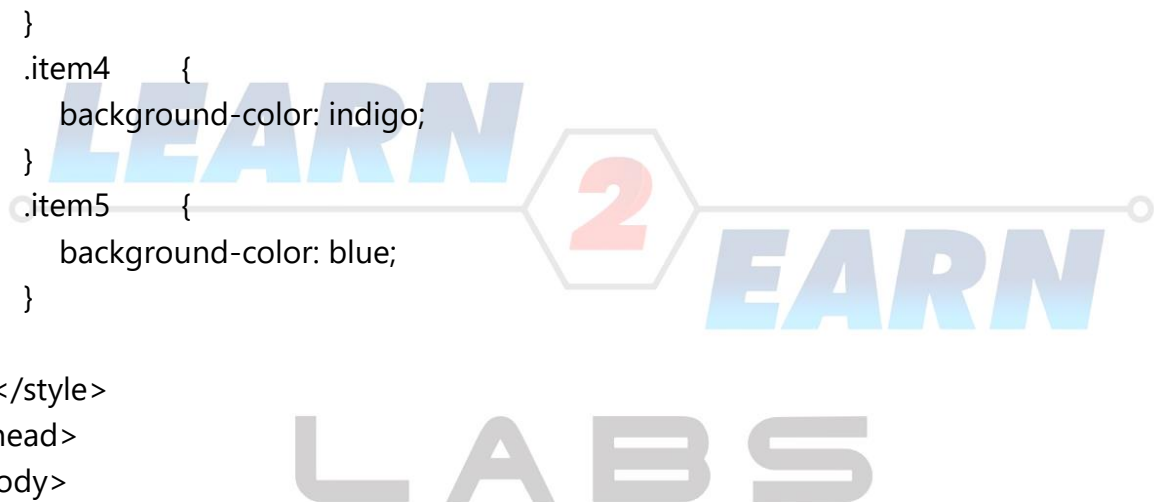
- This property applies on the flex-items.
- The default value of "flex-grow" property is 0.
- When we apply "flex-grow:1" then it gives equal width of all flex-items.

#### Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html,body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      width: 600px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
    }
    .items {
      color:#fff;
      font-size: 20px;
```

```
font-weight: bold;
padding: 10px;
margin: 10px;
flex-grow: 1;
}
.item1 {
  background-color: red;
}
.item2 {
  background-color: green;
}
.item3 {
  background-color: yellow;
}
.item4 {
  background-color: indigo;
}
.item5 {
  background-color: blue;
}

</style>
</head>
<body>
  <div class="container">
    <div class="items item1">Item1 </div>
    <div class="items item2">Item2 </div>
    <div class="items item3">Item3 </div>
    <div class="items item4">Item4 </div>
    <div class="items item5">Item5 </div>
  </div>
</body>
</html>
```

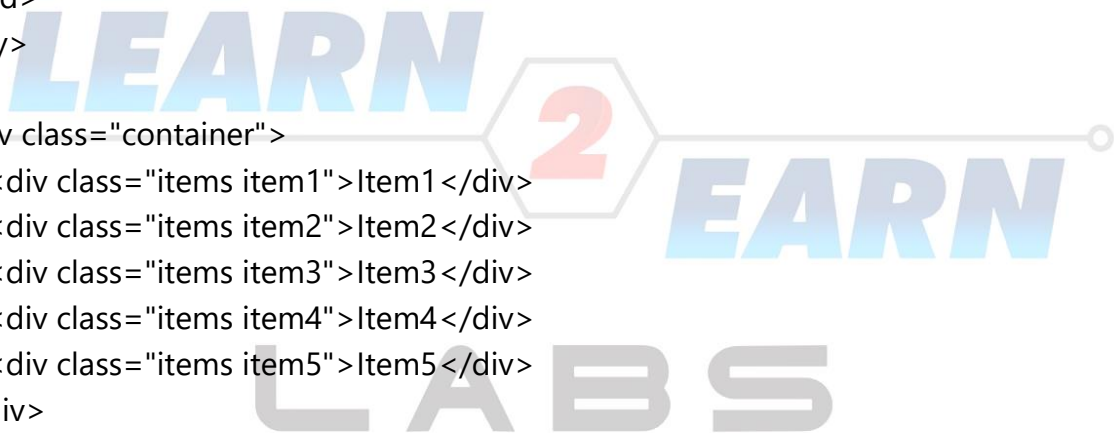


### Example -- for individual item

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      width: 600px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
    }
    .items
    {
      color:#fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
      flex-grow: 1;
    }
    .item1
    {
      background-color: red;
    }
```

```
.item2    {
  background-color: green;
  flex-grow: 4;
}
.item3    {
  background-color: yellow;
}
.item4    {
  background-color: indigo;
}
.item5    {
  background-color: blue;
}
</style>
</head>
<body>
  <div class="container">
    <div class="items item1">Item1 </div>
    <div class="items item2">Item2 </div>
    <div class="items item3">Item3 </div>
    <div class="items item4">Item4 </div>
    <div class="items item5">Item5 </div>
  </div>

</body>
</html>
```



### b) flex-basis

- This property generally deals with the width of flex-items.
- This property has no effect on resizing browser window till there is an empty space inside the flex\_container row.
- This property is best for making websites responsive.

## Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html, body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      width: 600px;
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
    }
    .items {
      height: 60px;
      color: #fff;
      font-size: 20px;
    }
  </style>
</html>
```



```
font-weight: bold;
padding: 10px;
margin: 10px;
flex-grow: 1;
}
.item1 {
  background-color: red;
}
.item2 {
  background-color: green;
}
.item3 {
  background-color: yellow;
}
.item4 {
  background-color: indigo;
  flex-basis: 200px;
}
.item5 {
  background-color: blue;
}
</style>
</head>
<body>
```

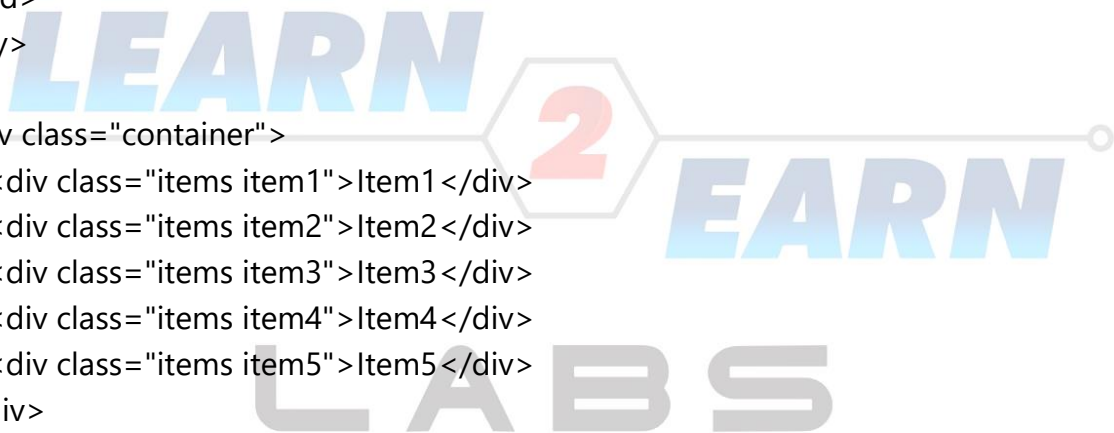
```
<div class="container">
  <div class="items item1">Item1 </div>
  <div class="items item2">Item2 </div>
  <div class="items item3">Item3 </div>
  <div class="items item4">Item4 </div>
  <div class="items item5">Item5 </div>
</div>
</body>
</html>
```

### Example -- Responsive Layout 1 / max-width case

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    *
    {
      box-sizing: border-box;
    }
    html,body
    {
      margin: 0px;
      padding: 0px;
    }
    .container
    {
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
    }
    .items
    {
      height: 60px;
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
      flex-basis: 100px;
    }
    .item1
    {
      background-color: red;
    }
```

```
.item2    {
  background-color: green;
}
.item3    {
  background-color: yellow;
}
.item4    {
  background-color: indigo;
  flex-basis: 200px;
}
.item5    {
  background-color: blue;
}
</style>
</head>
<body>
  <div class="container">
    <div class="items item1">Item1 </div>
    <div class="items item2">Item2 </div>
    <div class="items item3">Item3 </div>
    <div class="items item4">Item4 </div>
    <div class="items item5">Item5 </div>
  </div>

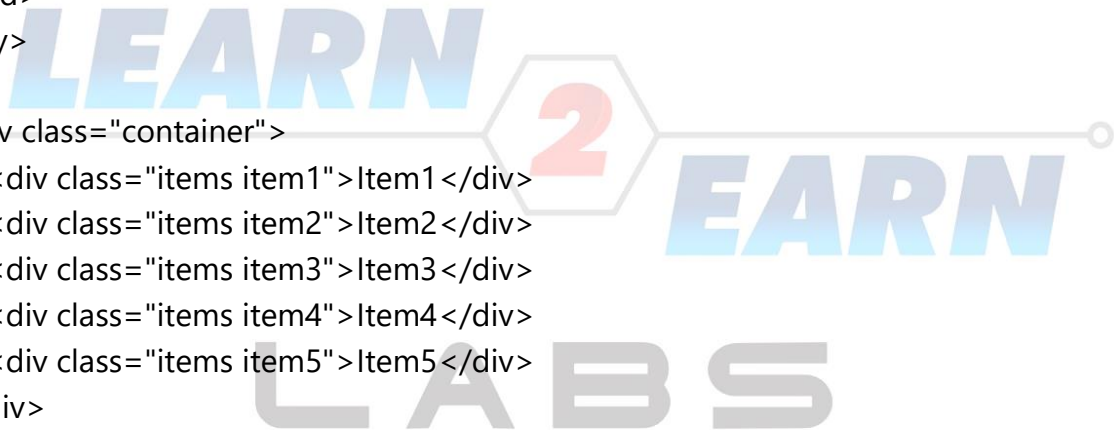
</body>
</html>
```



### Example -- Responsive Layout 2 / min-width case

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    * {
      box-sizing: border-box;
    }
    html,body {
      margin: 0px;
      padding: 0px;
    }
    .container {
      padding: 10px;
      border: 2px solid black;
      margin: 40px auto;
      background: silver;
      display: flex;
      flex-wrap: wrap;
    }
    .items {
      height: 60px;
      color:#fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: 10px;
      flex-grow: 1;
    }
    .item1 {
      background-color: red;
    }
  
```

```
.item2    {
  background-color: green;
}
.item3    {
  background-color: yellow;
}
.item4    {
  background-color: indigo;
  flex-basis: 200px;
}
.item5    {
  background-color: blue;
}
</style>
</head>
<body>
<div class="container">
  <div class="items item1">Item1 </div>
  <div class="items item2">Item2 </div>
  <div class="items item3">Item3 </div>
  <div class="items item4">Item4 </div>
  <div class="items item5">Item5 </div>
</div>
</body>
</html>
```



### c) flex-shrink

- The flex-shrink CSS property sets the flex shrink factor of a flex item. If the size of all flex items is larger than the flex container, items shrink to fit according to flex-shrink.
- In use, flex-shrink is used alongside the other flex properties flex-grow and flex-basis, and normally defined using the flex shorthand.

## Example -- No-shrinking

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container
    {
      background-color: olivedrab;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
    }
    .items    {
      color: white;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      margin: 0px;
      height: 60px;
      width: 200px;
    }
  </style>

```

```
.item1    {
  background: blue;
  flex-shrink: 0;
}
.item2    {
  background: red;
}
.item3    {
  background: green;
}
.item4    {
  background: yellow;
}
</style>
</head>
<body>
<h1>CSS Flex-Shrink Property</h1>

<div class="container">
  <div class="items item1">One</div>
  <div class="items item2">Two</div>
  <div class="items item3">Three</div>
  <div class="items item4">Four</div>
</div>

</body>
</html>
```

### Example -- Shrinking

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: orchid;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
    }
    .items
    {
      color: white;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      margin: 0px;
      height: 60px;
      width: 200px;
    }
    .item1 {
      background: blue;
      flex-shrink: 3;
    }
    .item2 {
      background: red;
    }
    .item3 {
      background: green;
    }
```



```
.item4    {  
    background: yellow;  
}  
</style>  
</head>  
<body>  
  <h1>CSS Flex-Shrink Property</h1>  
  <div class="container">  
    <div class="items item1">One</div>  
    <div class="items item2">Two</div>  
    <div class="items item3">Three</div>  
    <div class="items item4">Four</div>  
  </div>  
</body>  
</html>
```



### Example -- Shrinking

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: orchid;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
      width: 0 auto;
      max-height: 100px;
    }
    .items
    {
      color: white;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      margin: 0px;
      flex-grow: 1;
      flex-shrink: 1;
      flex-basis: 300px;
    }
    .item1 {
      background: blue;
    }
    .item2 {
      background: red;
      flex-shrink: 2;
    }
  </style>
</html>
```

```
.item3    {  
    background: green;  
}  
.item4    {  
    background: yellow;  
}  
</style>  
</head>  
<body>  
  
<h1>CSS Flex-Shrink Property</h1>  
  
<div class="container">  
    <div class="items item1">One</div>  
    <div class="items item2">Two</div>  
    <div class="items item3">Three</div>  
    <div class="items item4">Four</div>  
</div>  
  
</body>  
</html>
```

LABS

### Example -- Shrinking

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .flex-container {
      padding: 0;
      margin: 0;
      list-style: none;
      -ms-box-orient: horizontal;
      display: -webkit-box;
      display: -moz-box;
      display: -ms-flexbox;
      display: -moz-flex;
      display: -webkit-flex;
      display: flex;
    }
    .flex-item {
      background: tomato;
      padding: 10px;
      border: 5px solid red;
      color: white;
      font-weight: bold;
      font-size: 2em;
      text-align: center;
    }
    .flex1 { flex: 1 1 20em; }
    .flex2 { flex: 2 2 20em; }
  </style>
</head>
<body>
```

```
<h1>CSS Flexbox -- Resize The Browser</h1>  
<ul class="flex-container">  
  <li class="flex-item flex1">1</li>  
  <li class="flex-item flex2">2</li>  
</ul>  
</body>  
</html>
```



### Example -- Shorthand

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: orchid;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
    }
    .items {
      color: white;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      margin: 0px;
      height: 60px;
    }
    .item1 {
      background: blue;
      flex: 0 1 400px; /* flex-grow flex-shrink flex-basis */
    }
    .item2 {
      background: red;
    }
    .item3 {
      background: green;
    }
  </style>
</html>
```

```
.item4    {  
    background: yellow;  
}  
</style>  
</head>  
<body>  
  <h1>CSS Flex-Shrink Property</h1>  
  <div class="container">  
    <div class="items item1">One</div>  
    <div class="items item2">Two</div>  
    <div class="items item3">Three</div>  
    <div class="items item4">Four</div>  
  </div>  
</body>  
</html>
```



### Example -- Default Flex-Basis Value (i.e. auto)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: orchid;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
    }
    .items {
      color: white;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      margin: 0px;
      height: 60px;
    }
    .item1 {
      background: blue;
      flex: 0 1 auto; /* flex-grow flex-shrink flex-basis */
    }
    .item2 {
      background: red;
    }
    .item3 {
      background: green;
    }
  </style>
</html>
```



```
.item4
{
    background: yellow;
}
</style>
</head>
<body>

<h1>CSS Flex-Shrink Property</h1>

<div class="container">
    <div class="items item1">One</div>
    <div class="items item2">Two</div>
    <div class="items item3">Three</div>
    <div class="items item4">Four</div>
</div>
</body>
</html>
```



LABS

## Flexbox with "margin : auto"

### Example 1

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: greenyellow;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      height: 400px;
      display: flex;
    }
    .item {
      background: orangered;
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin: auto;
    }
  </style>
</head>
<body>
  <h1>CSS Flexbox</h1>
  <div class="container">
    <div class="item">One</div>
  </div>
</body>
</html>
```

## Example 2

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: greenyellow;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      height: 400px;
      display: flex;
    }
    .item {
      background: orangered;
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin-left: auto;
    }
  </style>
</head>
<body>
  <h1>CSS Flexbox</h1>
  <div class="container">
    <div class="item">One</div>
  </div>
</body>
</html>
```

### Example 3

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: greenyellow;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      height: 400px;
      display: flex;
    }
    .item {
      background: orangered;
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin-top: auto;
    }
  </style>
</head>
<body>
  <h1>CSS Flexbox</h1>
  <div class="container">
    <div class="item">One</div>
  </div>
</body>
</html>
```

#### Example 4

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: greenyellow;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      height: 400px;
      display: flex;
    }
    .item {
      background: orangered;
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      padding: 10px;
      margin-top: auto;
      margin-left: auto;
    }
  </style>
</head>
<body>
  <h1>CSS Flexbox</h1>
  <div class="container">
    <div class="item">One</div>
  </div>
</body>
</html>
```

### Example 5 -- Grab Free Space

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: darkgrey;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
    }
    .item {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      height: 100px;
      width: 100px;
    }
    .item1 {
      background-color: orangered;
    }
    .item2 {
      background-color: yellowgreen;
      margin-right: auto;
    }
    .item3
    {
      background-color: mediumpurple;
    }
  </style>
</html>
```

```
.item4    {  
    background-color: hotpink;  
}  
</style>  
</head>  
<body>  
    <h1>CSS Flexbox</h1>  
    <div class="container">  
        <div class="item item1">One</div>  
        <div class="item item2">Two</div>  
        <div class="item item3">Three</div>  
        <div class="item item4">Four</div>  
    </div>  
</body>  
</html>
```



## Example 6

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: darkgrey;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
    }
    .item {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      height: 100px;
      width: 100px;
    }
    .item1 {
      background-color: orangered;
    }
    .item2 {
      background-color: yellowgreen;
    }
    .item3 {
      background-color: mediumpurple;
    }
    .item4 {
      background-color: hotpink;
      margin-left: auto;
    }
  </style>
</head>
```



```
<body>
  <h1>CSS Flexbox</h1>
  <div class="container">
    <div class="item item1">One</div>
    <div class="item item2">Two</div>
    <div class="item item3">Three</div>
    <div class="item item4">Four</div>
  </div>

</body>
</html>
```



### Example 7

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: darkgrey;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
    }
    .item {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      height: 100px;
      width: 100px;
    }
    .item1 {
      background-color: orangered;
      margin-right: auto;
    }
    .item2 {
      background-color: yellowgreen;
    }
    .item3 {
      background-color: mediumpurple;
    }
  </style>
</html>
```

```
.item4
{
    background-color: hotpink;
    margin-left: auto;
}
</style>
</head>
<body>

<h1>CSS Flexbox</h1>

<div class="container">
    <div class="item item1">One</div>
    <div class="item item2">Two</div>
    <div class="item item3">Three</div>
    <div class="item item4">Four</div>
</div>
</body>
</html>
```

LEARN 2 EARN LABS

## Margin Vs Padding

Flexbox items and layouts would be affected by padding not by margin.

## Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: darkgrey;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
      flex-direction: column;
    }
    .item {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      height: 100px;
    }
    .item1 {
      background-color: orangered;
    }
    .item2 {
      background-color: yellowgreen;
      display: flex;
    }
  </style>
</head>
<body>
```

```
.item2 > .subitem    {
  background-color: burlywood;
  color: white;
  flex-grow: 1;
  border: 4px solid yellow;
  padding: 40%;
}
.item3    {
  background-color: mediumpurple;
}
</style>
</head>
<body>

<h1>CSS Flexbox</h1>
<div class="container">
  <div class="item item1">One</div>
  <div class="item item2">
    <div class="subitem subitem1">A</div>
    <div class="subitem subitem2">B</div>
    <div class="subitem subitem3">C</div>
  </div>
  <div class="item item3">Three</div>
</div>

</body>
</html>
```

## Nested Flex

## Example 1

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container
    {
      background-color: darkgrey;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
      flex-direction: column;
    }
    .item    {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      height: 100px;
    }
    .item1   {
      background-color: orangered;
    }
    .item2   {
      background-color: yellowgreen;
      display: flex;
    }
  </style>

```

```
.item2 > .subitem    {  
    background-color: burlywood;  
    color: white;  
    flex-grow: 1;  
    border: 4px solid yellow;  
}  
.item3  
{  
    background-color: mediumpurple;  
}
```

```
</style>  
</head>  
<body>
```

```
<h1>CSS Flexbox</h1>
```

```
<div class="container">
```

```
  <div class="item item1">One</div>
```

```
  <div class="item item2">
```

```
    <div class="subitem subitem1">A</div>
```

```
    <div class="subitem subitem2">B</div>
```

```
    <div class="subitem subitem3">C</div>
```

```
  </div>
```

```
  <div class="item item3">Three</div>
```

```
</div>
```

```
</body>
```

```
</html>
```

## Example 2 -- Nested Flexbox With "margin:auto"

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: darkgrey;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
      flex-direction: column; }
    .item {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      height: 100px; }
    .item1 {
      background-color: orangered;
    }
    .item2 {
      background-color: yellowgreen;
      display: flex; }
    .item2 > .subitem {
      background-color: burlywood;
      color: white;
      flex-grow: 1;
      border: 4px solid yellow;
      margin: auto;
    }
```



```
.item3    {  
    background-color: mediumpurple;  
}  
</style>  
</head>  
<body>  
    <h1>CSS Flexbox</h1>  
    <div class="container">  
        <div class="item item1">One</div>  
        <div class="item item2">  
            <div class="subitem subitem1">A</div>  
            <div class="subitem subitem2">B</div>  
            <div class="subitem subitem3">C</div>  
        </div>  
        <div class="item item3">Three</div>  
    </div>  
</body>  
</html>
```

LEARN 2 EARN  
LABS

### Example 3 -- Nested Flexbox With "margin:auto"

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container {
      background-color: darkgrey;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
      flex-direction: column; }
    .item {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      height: 100px; }
    .item1 {
      background-color: orangered;
    }
    .item2 {
      background-color: yellowgreen;
      display: flex; }
    .item2 > .subitem {
      background-color: burlywood;
      color: white;
      flex-grow: 1;
      border: 4px solid yellow;
      margin-left: auto;
    }
```

```
.item3    {  
    background-color: mediumpurple;  
}  
</style>  
</head>  
<body>  
    <h1>CSS Flexbox</h1>  
    <div class="container">  
        <div class="item item1">One</div>  
        <div class="item item2">  
            <div class="subitem subitem1">A</div>  
            <div class="subitem subitem2">B</div>  
            <div class="subitem subitem3">C</div>  
        </div>  
        <div class="item item3">Three</div>  
    </div>  
</body>  
</html>
```

LEARN 2 EARN  
LABS

#### Example 4

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .container
    {
      background-color: darkgrey;
      margin: 50px;
      padding: 0px;
      border: 4px solid black;
      display: flex;
      flex-direction: column;
    }
    .item
    {
      color: #fff;
      font-size: 20px;
      font-weight: bold;
      text-align: center;
      padding: 0px;
      height: 100px;
    }
    .item1
    {
      background-color: orangered;
    }
    .item2
    {
      background-color: yellowgreen;
      display: flex;
    }
  </style>
</html>
```

```
.item2 > .subitem
{
  background-color: burlywood;
  color: white;
  flex-grow: 1;
  border: 4px solid yellow;
}
.item3
{
  background-color: mediumpurple;
}
.subitem2
{
  margin: auto;
}
</style>
```

```
</head>
```

```
<body>
```

```
<h1>CSS Flexbox</h1>
```

```
<div class="container">
```

```
<div class="item item1">One</div>
```

```
<div class="item item2">
```

```
<div class="subitem subitem1">A</div>
```

```
<div class="subitem subitem2">B</div>
```

```
<div class="subitem subitem3">C</div>
```

```
</div>
```

```
<div class="item item3">Three</div>
```

```
</div>
```

```
</body>
```

```
</html>
```

## Website Layout Using Flexbox

```
<!DOCTYPE html>
<html>
<head>
  <title>Document</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <style>
    * {
      box-sizing: border-box;
    }
    .row {
      display: flex;
      flex-wrap: wrap;
    }
    /* Four equal columns */
    .column {
      flex: 25%;
      padding: 20px;
    }
    /* If a screen is 992px or less wide, keep two columns */
    @media screen and (max-width: 992px) {
      .column {
        flex: 50%;
      }
    }
    /* If a screen is 600px or less wide, stack the columns on top of each other instead
of next to each other */
    @media screen and (max-width: 600px) {
      .row {
        flex-direction: column;
      }
    }
  </style>
</head>
```

```
<body>
```

```
<div class="row">
```

```
<div class="column" style="background-color: lightgreen;">
```

```
<h2>Column 1</h2>
```

```
<p>Some text..</p>
```

```
</div>
```

```
<div class="column" style="background-color: lightblue;">
```

```
<h2>Column 2</h2>
```

```
<p>Some text..</p>
```

```
</div>
```

```
<div class="column" style="background-color: pink;">
```

```
<h2>Column 3</h2>
```

```
<p>Some text..</p>
```

```
</div>
```

```
<div class="column" style="background-color: lightgrey;">
```

```
<h2>Column 4</h2>
```

```
<p>Some text..</p>
```

```
</div>
```

```
</div>
```

```
</body>
```

```
</html>
```

