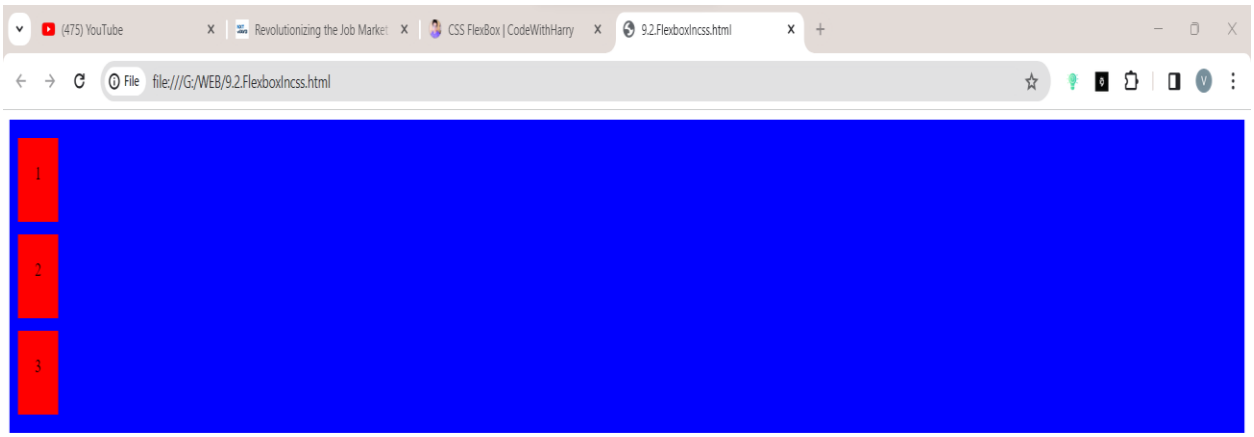
# CSS FlexBox

FlexBox aka Flexible Box Layout makes it easier to layout, align and style items in the container while maintaining the responsiveness of the website.

To create a flexbox you need to set the display of the container as flex
Eg: {display: flex;}

This element is called the flex container, and stores the sub-elements which are known as flex items

```
<head>
  <style>
    .background {
      background-color: blue;
      padding-top: 5px;
      padding-bottom: 5px;
    }
    .cubes {
      text-align: center;
      padding-top: 20px;
      height: 50px;
      width: 50px;
      background-color: red;
      margin: 10px;
    }
  </style>
</head>
<body>
  <div class="background">
    <div class="cubes">1</div>
    <div class="cubes">2</div>
    <div class="cubes">3</div>
  </div>
</body>
html>
```



## Flex Container Properties

The flex container properties are:

### 1. Flex Direction

It defines in which direction the flex elements would be displayed. It takes values like row, column or "reverse" too.

1.COLUMN:

```
    <style>
      .background {
        background-color: blue;
        padding-top: 5px;
        padding-bottom: 5px;
        display: flex;
        flex-direction: column;
      }
      .cubes {
        text-align: center;
        padding-top: 20px;
        height: 50px;
        width: 50px;
        background-color: red;
        margin: 10px;
      }
    </style>
  </head>
  <body>
    <div class="background">
      <div class="cubes">1</div>
      <div class="cubes">2</div>
      <div class="cubes">3</div>
    </div>
  </body>
</html>
```

2.ROW

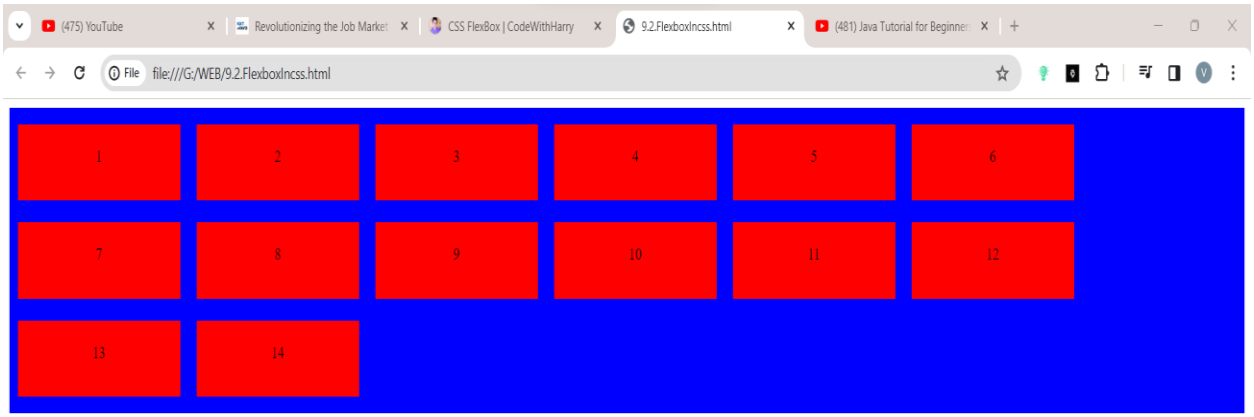
```
<head>
  <style>
    .background {
      background-color: blue;
      padding-top: 5px;
      padding-bottom: 5px;
      display: flex;
      flex-direction: row;
    }
    .cubes {
      text-align: center;
      padding-top: 20px;
      height: 50px;
      width: 50px;
      background-color: red;
      margin: 10px;
    }
  </style>
</head>
<body>
  <div class="background">
    <div class="cubes">1</div>
    <div class="cubes">2</div>
    <div class="cubes">3</div>
  </div>
</body>
</html>
```

## 2. Flex Wrap

By using this property we can make our elements responsive for different screen sizes.

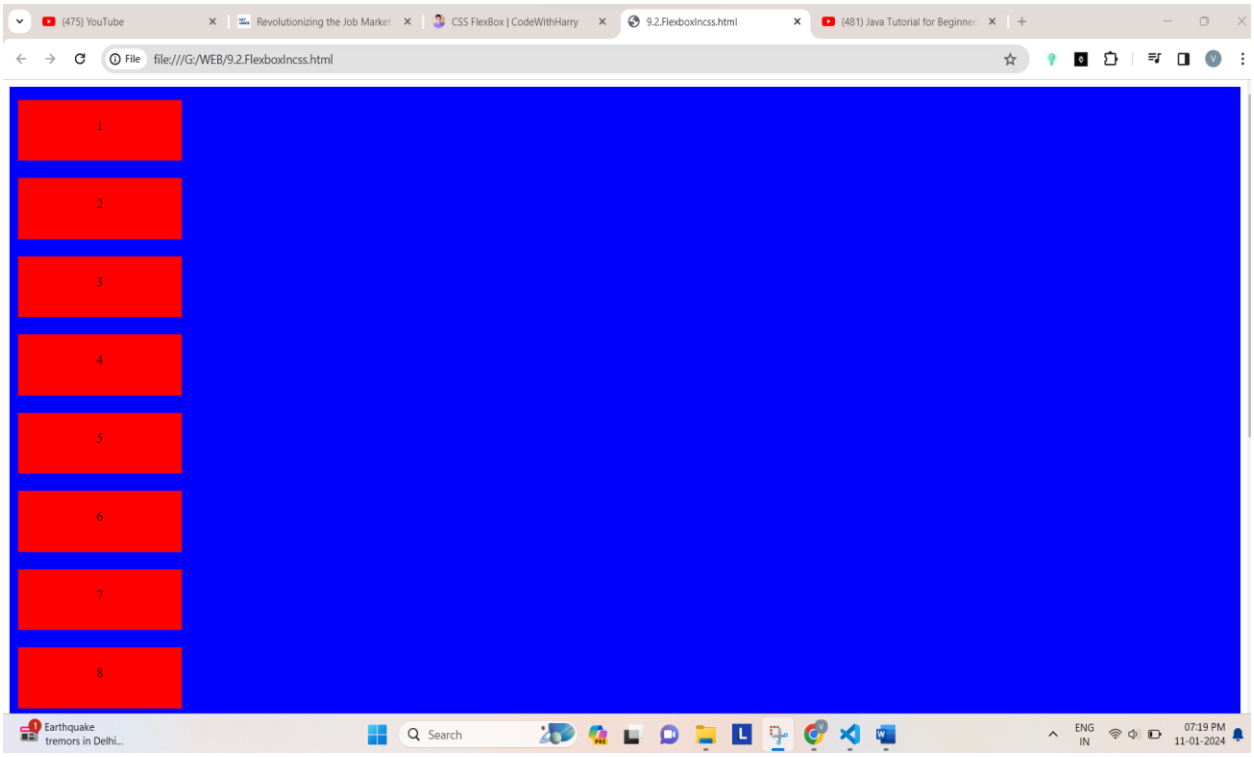
1.ROW

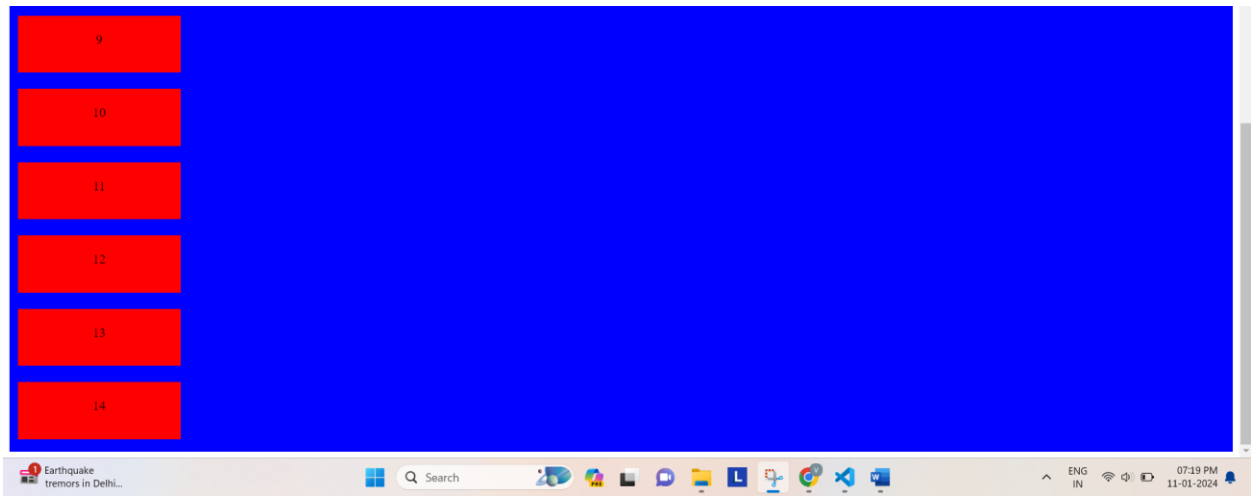
```
    <style>
      .background {
        background-color: blue;
        padding-top: 5px;
        padding-bottom: 5px;
        display: flex;
        flex-direction: row;
        flex-wrap: wrap;
      }
      .cubes {
        text-align: center;
        padding-top: 20px;
        height: 50px;
        width: 200px;
        background-color: red;
        margin: 10px;
      }
    </style>
  </head>
  <body>
    <div class="background">
      <div class="cubes">1</div>
      <div class="cubes">2</div>
      <div class="cubes">3</div>
      <div class="cubes">4</div>
      <div class="cubes">5</div>
      <div class="cubes">6</div>
      <div class="cubes">7</div>
      <div class="cubes">8</div>
      <div class="cubes">9</div>
      <div class="cubes">10</div>
      <div class="cubes">11</div>
      <div class="cubes">12</div>
      <div class="cubes">13</div>
      <div class="cubes">14</div>
    </div>
```


(475) YouTube
Revolutionizing the Job Market
CSS FlexBox | CodeWithHarry
9.2.FlexboxIncss.html
(481) Java Tutorial for Beginner
File
file:///G:/WEB/9.2.FlexboxIncss.html
1
2
3
4
5
6
7
8
9
10
11
12
13
14

2.COLUMN

```
    <style>
      .background {
        background-color: blue;
        padding-top: 5px;
        padding-bottom: 5px;
        display: flex;
        flex-direction: column;
        flex-wrap: wrap;
      }
      .cubes {
        text-align: center;
        padding-top: 20px;
        height: 50px;
        width: 200px;
        background-color: red;
        margin: 10px;
      }
    </style>
  </head>
  <body>
    <div class="background">
      <div class="cubes">1</div>
      <div class="cubes">2</div>
      <div class="cubes">3</div>
      <div class="cubes">4</div>
      <div class="cubes">5</div>
      <div class="cubes">6</div>
      <div class="cubes">7</div>
      <div class="cubes">8</div>
      <div class="cubes">9</div>
      <div class="cubes">10</div>
      <div class="cubes">11</div>
      <div class="cubes">12</div>
      <div class="cubes">13</div>
      <div class="cubes">14</div>
    </div>
```


(475) YouTube
Revolutionizing the Job Market
CSS FlexBox | CodeWithHarry
9.2.FlexboxIncss.html
(481) Java Tutorial for Beginner
File
file:///G:/WEB/9.2.FlexboxIncss.html
1
2
3
4
5
6
7
8
Earthquake
tremors in Delhi...
Search
ENG
IN
07:19 PM
11-01-2024

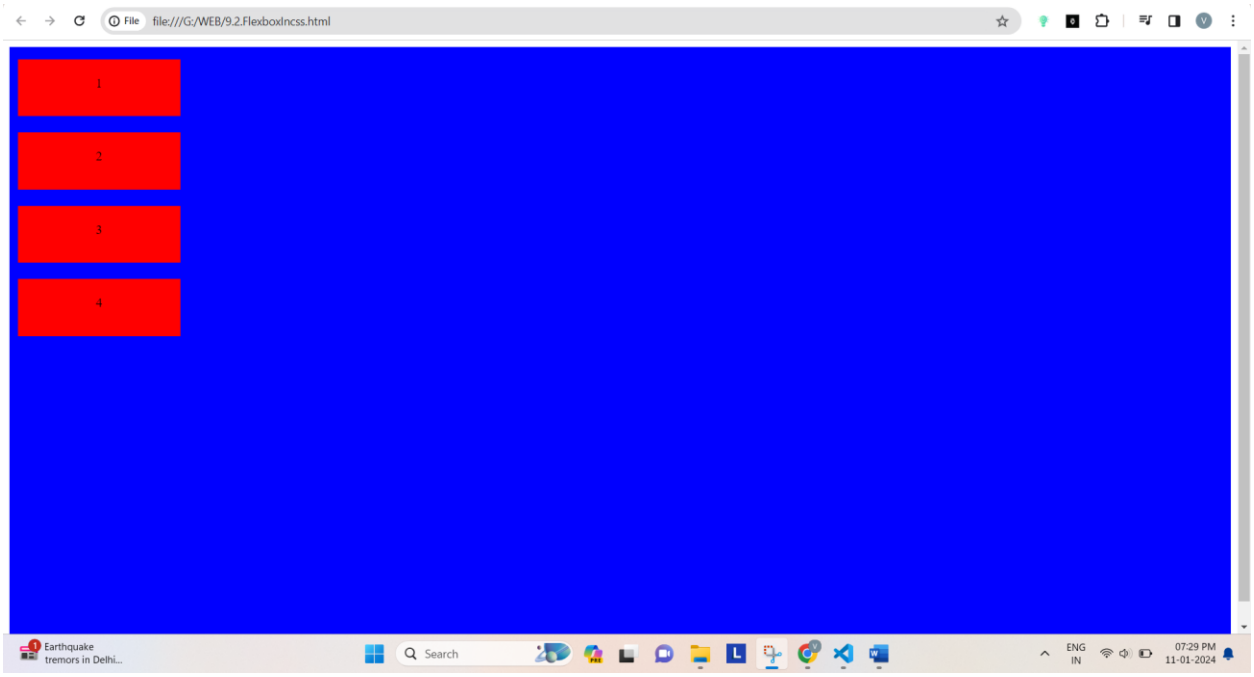

9
10
11
12
13
14
Earthquake
tremors in Delhi...
Search
ENG
IN
07:19 PM
11-01-2024

## 3. Justify Content

This property is used to set the position of content or rather align content along the main axis.

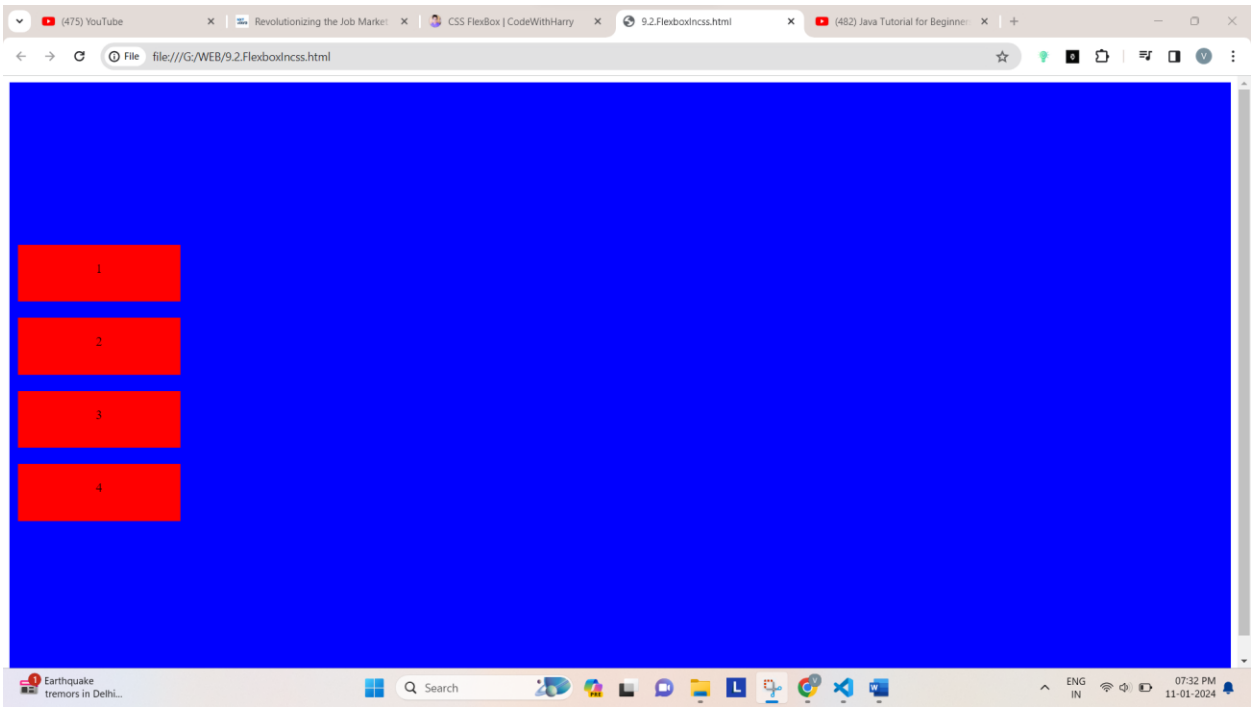
1.COLUMN START

```
<head>
  <style>
    .background {
      background-color: blue;
      height: 100vh;
      padding-top: 5px;
      padding-bottom: 5px;
      display: flex;
      flex-direction: column;
      justify-content: start;
    }
    .cubes {
      text-align: center;
      padding-top: 20px;
      height: 50px;
      width: 200px;
      background-color: red;
      margin: 10px;
    }
  </style>
</head>
<body>
  <div class="background">
    <div class="cubes">1</div>
    <div class="cubes">2</div>
    <div class="cubes">3</div>
    <div class="cubes">4</div>
  </div>
</body>
</html>
```


File
file:///G:/WEB/9.2.FlexboxIncss.html
1
2
3
4
Earthquake
tremors in Delhi...
Search
ENG
IN
07:29 PM
11-01-2024

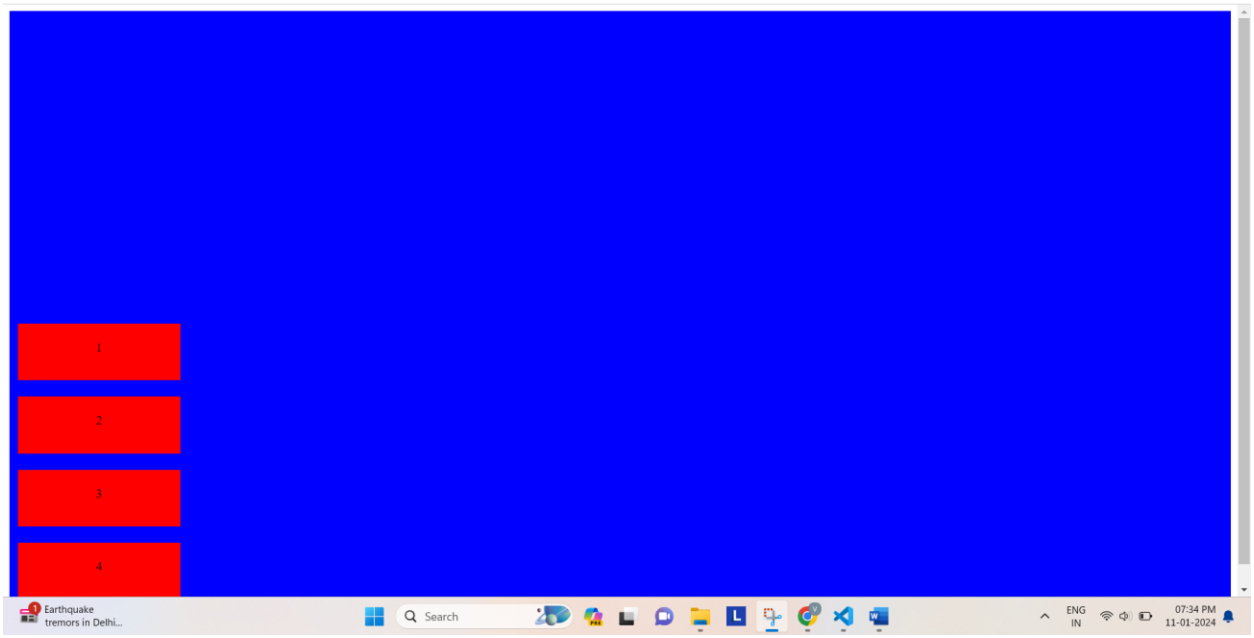
2.COLUMN-CENTER

```
<head>
  <style>
    .background {
      background-color: blue;
      height: 100vh;
      padding-top: 5px;
      padding-bottom: 5px;
      display: flex;
      flex-direction: column;
      justify-content: center;
    }
    .cubes {
      text-align: center;
      padding-top: 20px;
      height: 50px;
      width: 200px;
      background-color: red;
      margin: 10px;
    }
  </style>
</head>
<body>
  <div class="background">
    <div class="cubes">1</div>
    <div class="cubes">2</div>
    <div class="cubes">3</div>
    <div class="cubes">4</div>
  </div>
</body>
</html>
```


(475) YouTube
Revolutionizing the Job Market
CSS FlexBox | CodeWithHarry
9.2.FlexboxIncss.html
(482) Java Tutorial for Beginner
File
file:///G:/WEB/9.2.FlexboxIncss.html
1
2
3
4
Earthquake
tremors in Delhi...
Search
ENG
IN
07:32 PM
11-01-2024

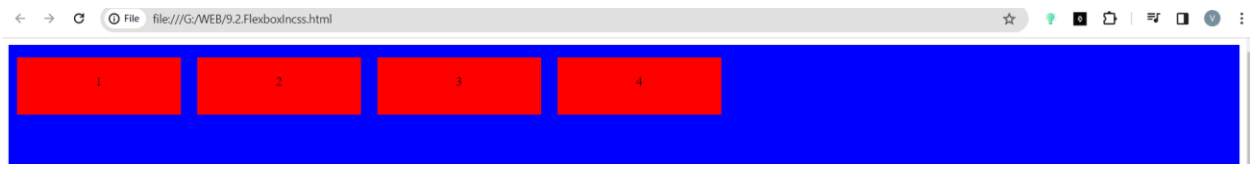
3.COLUMN-END

```
<head>
  <style>
    .background {
      background-color: blue;
      height: 100vh;
      padding-top: 5px;
      padding-bottom: 5px;
      display: flex;
      flex-direction: column;
      justify-content: end;
    }
    .cubes {
      text-align: center;
      padding-top: 20px;
      height: 50px;
      width: 200px;
      background-color: red;
      margin: 10px;
    }
  </style>
</head>
<body>
  <div class="background">
    <div class="cubes">1</div>
    <div class="cubes">2</div>
    <div class="cubes">3</div>
    <div class="cubes">4</div>
  </div>
</body>
</html>
```


1
2
3
4
Earthquake
tremors in Delhi...
Search
ENG
IN
07:34 PM
11-01-2024

1.ROW-START

```
<head>
  <style>
    .background {
      background-color: blue;
      height: 100vh;
      padding-top: 5px;
      padding-bottom: 5px;
      display: flex;
      flex-direction: row;
      justify-content: start;
    }
    .cubes {
      text-align: center;
      padding-top: 20px;
      height: 50px;
      width: 200px;
      background-color: red;
      margin: 10px;
    }
  </style>
</head>
<body>
  <div class="background">
    <div class="cubes">1</div>
    <div class="cubes">2</div>
    <div class="cubes">3</div>
    <div class="cubes">4</div>
  </div>
</body>
</html>
```

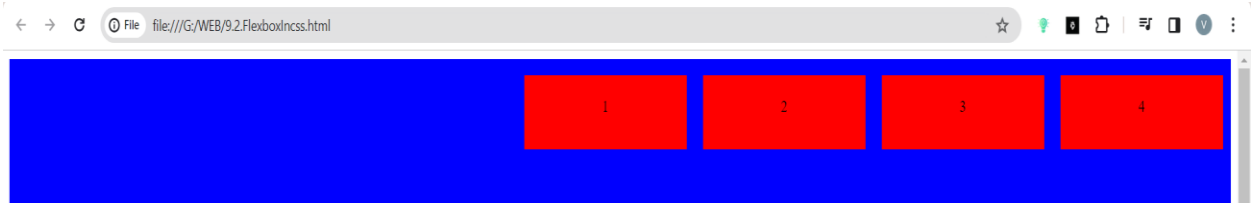


2.ROW-CENTER

```
    <style>
      .background {
        background-color: blue;
        height: 100vh;
        padding-top: 5px;
        padding-bottom: 5px;
        display: flex;
        flex-direction: row;
        justify-content: center;
      }
      .cubes {
        text-align: center;
        padding-top: 20px;
        height: 50px;
        width: 200px;
        background-color: red;
        margin: 10px;
      }
    </style>
  </head>
  <body>
    <div class="background">
      <div class="cubes">1</div>
      <div class="cubes">2</div>
      <div class="cubes">3</div>
      <div class="cubes">4</div>
    </div>
  </body>
</html>
```

3.ROW-END

```
    <style>
      .background {
        background-color: blue;
        height: 100vh;
        padding-top: 5px;
        padding-bottom: 5px;
        display: flex;
        flex-direction: row;
        justify-content: end;
      }
      .cubes {
        text-align: center;
        padding-top: 20px;
        height: 50px;
        width: 200px;
        background-color: red;
        margin: 10px;
      }
    </style>
  </head>
  <body>
    <div class="background">
      <div class="cubes">1</div>
      <div class="cubes">2</div>
      <div class="cubes">3</div>
      <div class="cubes">4</div>
    </div>
  </body>
</html>
```



## 4. Align Items

Just like the justify-content property, align-items define the alignment of the flex container but along the cross-axis.

### Difference Between Main Axis and Cross Axis:

In **flexbox**, the **main axis** and **cross axis** depend on the `flex-direction` property:

1. **Main Axis**:
   - It's the **direction in which the flex items are arranged**.
   - By default, it's **horizontal (left to right)** because `flex-direction` is set to `row` by default.
2. **Cross Axis**:
   - It's **perpendicular to the main axis**.
   - For example, if the main axis is horizontal (`row`), the cross axis will be **vertical (top to bottom)**.

### Flex Directions:

`flex-direction: row` **(default):**

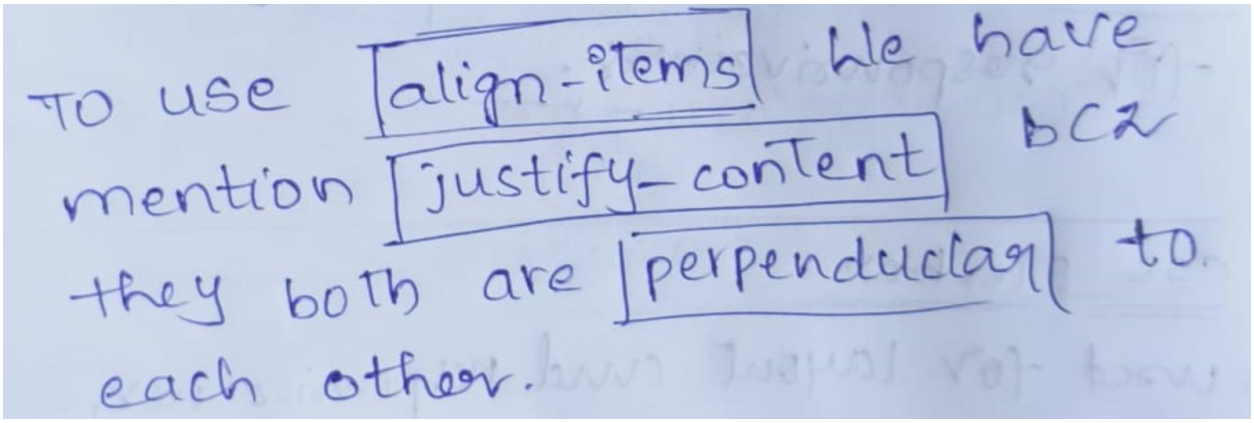
- Main Axis: **Horizontal** (left to right).
- Cross Axis: **Vertical** (top to bottom).

`flex-direction: column` **:**

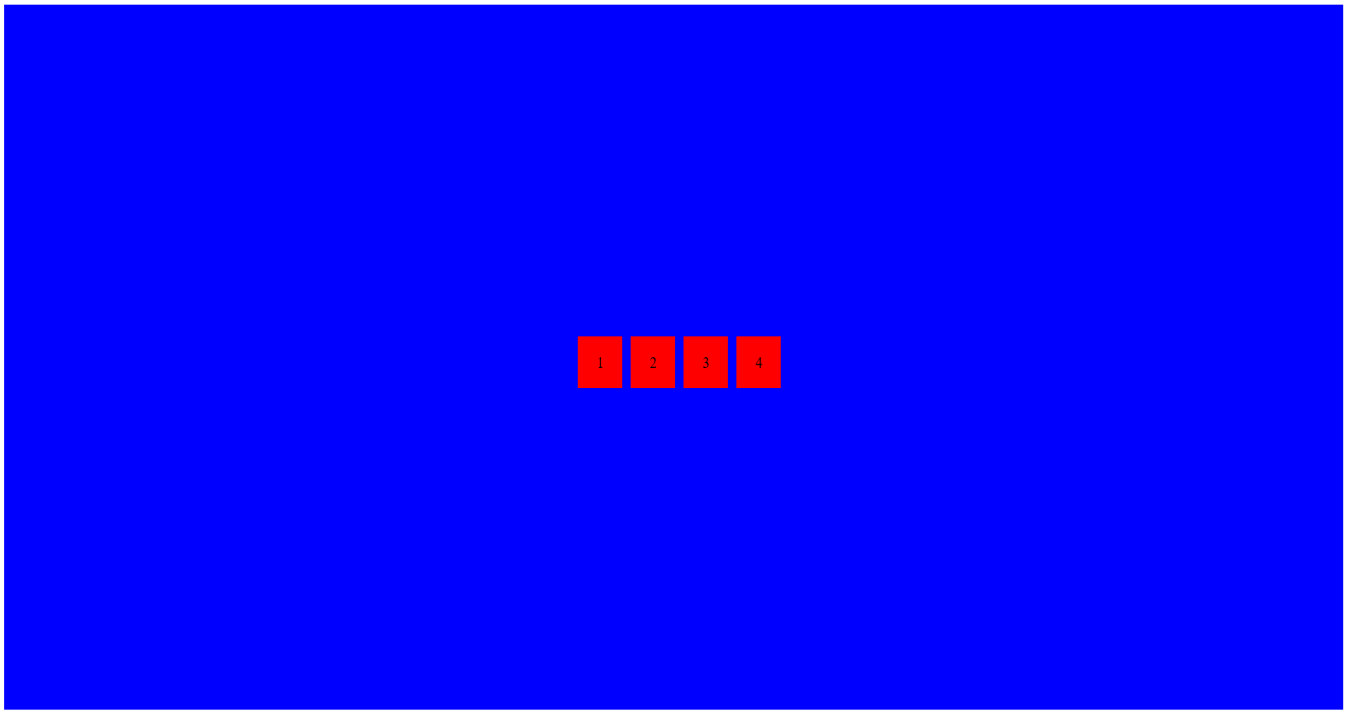
- Main Axis: **Vertical** (top to bottom).
- Cross Axis: **Horizontal** (left to right).

## Key Points:

- **Main axis** is controlled by `flex-direction`.
- **Cross axis** is always perpendicular to the main axis.
- Use `justify-content` to align items along the **main axis**.
- Use `align-items` to align items along the **cross axis**.



```
  <style>
    .mainbox {
      background-color: blue;
      height: 90vh;
      padding-top: 10px;
      padding-bottom: 10px;
      padding-left: 3px;
      display: flex;
      flex-direction: row;
      justify-content: center;
      align-items: center;
    }
    .box {
      background-color: red;
      height: 50px;
      width: 50px;
      margin-top: 10px;
      margin-left: 10px;
      display: flex;
      justify-content: center;
      align-items: center;
    }
  </style>
</head>
<body>
  <div class="mainbox">
    <div class="box">1</div>
    <div class="box">2</div>
    <div class="box">3</div>
    <div class="box">4</div>
  </div>
</body>
</html>
```


1
2
3
4

```
  <style>
    .mainbox {
      background-color: blue;
      height: 90vh;
      padding-top: 10px;
      padding-bottom: 10px;
      padding-left: 3px;
      display: flex;
      flex-direction: row;
      justify-content: end;
      align-items: center;
    }
    .box {
      background-color: red;
      height: 50px;
      width: 50px;
      margin-top: 10px;
      margin-left: 10px;
      display: flex;
      justify-content: center;
      align-items: center;
    }
  </style>
</head>
<body>
  <div class="mainbox">
    <div class="box">1</div>
    <div class="box">2</div>
    <div class="box">3</div>
    <div class="box">4</div>
  </div>
</body>
```

1
2
3
4

```
  .mainbox {
    background-color: blue;
    height: 90vh;
    padding-top: 10px;
    padding-bottom: 10px;
    padding-left: 3px;
    display: flex;
    flex-direction: row;
    justify-content: end;
    align-items: end;
  }
  .box {
    background-color: red;
    height: 50px;
    width: 50px;
    margin-top: 10px;
    margin-left: 10px;
    display: flex;
    justify-content: center;
    align-items: center;
  }
</style>
head>
ody>
<div class="mainbox">
  <div class="box">1</div>
  <div class="box">2</div>
  <div class="box">3</div>
  <div class="box">4</div>
</div>
body>
```

1
2
3
4