

# Flex Direction in CSS

## Overview

The **flex-direction** property defines in which **direction** the container lays out the flex-items.

It may take 4 values:

- flex-direction : row (**Default**)
- flex-direction : column
- flex-direction : row-reverse
- flex-direction : column-reverse

## row

The row value stacks the flex items horizontally, from left to right.

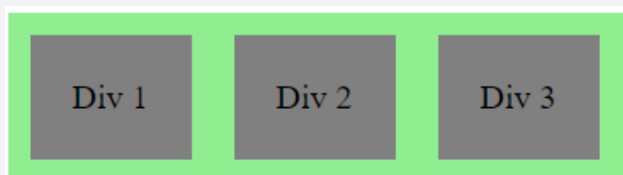
For **forward direction**, add the `flex-direction: row ;` to the flex-container class.

It will not change anything as this is the **default value** and the layout will still look the same.

**CSS :**

```
.parent {
    display: flex;
    flex-direction : row ;
    background-color: lightgreen;
}
```

**Browser :**



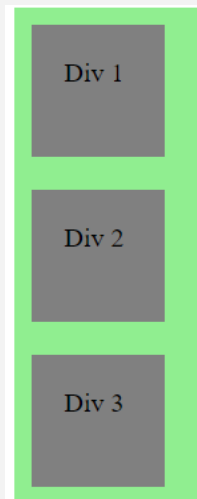
## column

The column value stacks the flex items vertically, from **top to bottom**.

For downward direction, add the `flex-direction: column ;` to the flex-container class.

**CSS :**

```
.parent {
    display: flex;
    flex-direction : column ;
    background-color: lightgreen;
}
```



**Browser :**

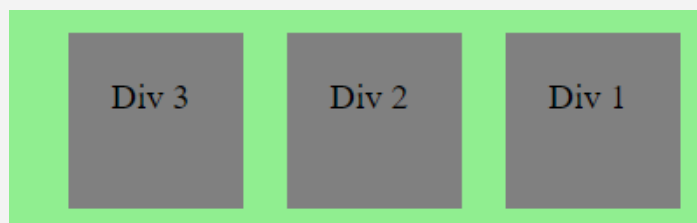
## row-reverse

The row-reverse value stacks the flex items horizontally, from **right to left**.

For backward direction, add the `flex-direction: row-reverse ;` to the flex container class.

**CSS :**

```
.parent {
    display: flex;
    flex-direction : row-reverse ;
    background-color: lightgreen;
}
```



**Browser :**

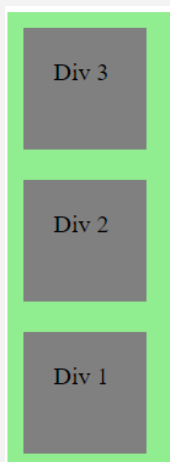
## column-reverse

The column-reverse value stacks the flex items vertically, from **bottom to top**.

For upward direction, add the `flex-direction: column-reverse ;` to the flex container class.

**CSS :**

```
.parent {
    display: flex;
    flex-direction : column-reverse ;
    background-color: lightgreen;
}
```



**Browser :**

## ORDER

The order property is used to **specify the order of the flex items** in the flex container.

This property value must be a whole number. By default, the number is 0(zero).

The higher the number the latter would the flex item appear in the flex container.

The flex items are displayed in the order like:

- First, the items **not having order property** or **order : 0 ;** property is displayed in sequence in which they appear in the source order.
- Then the items are displayed in **ascending order of the value of the order property**. The items having the same order value are displayed in the sequence in which they appear in the source order.

In the below example order is : Div2 , Div 1 , Div 3.

**HTML :**

```
<div class="parent">
  <div style="order: 1"> Div 1</div>
  <div style="order: 0"> Div 2</div>
  <div style="order: 2"> Div 3</div>
</div>
```

**CSS :**

```
.parent {
  display: flex;
  background-color: lightgreen;
}

.parent > div {
  background-color: grey ;
  margin: 10px;
  padding: 20px;
}
```



**Browser :**