

# CSS Notes For Students By Geekster



Geekster

[www.geekster.in](http://www.geekster.in)

# flex

## Defintion of Flex :

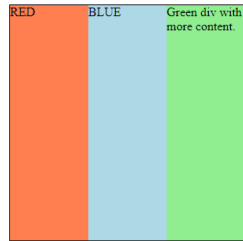
The flex property sets flexible length on flexible items.

Input:

```
flex.html X
html > flex.html > html > body
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  #main {
6    width: 300px;
7    height: 300px;
8    border: 1px solid black;
9    display: flex;
10 }
11
12 #main div {
13   -ms-flex: 1; /* IE 10 */
14   flex: 1;
15 }
16 </style>
17 </head>
18 <body>
19
20 <h1>The Flex Property</h1>
21
22 <div id="main">
23   <div style="background-color: coral;">RED</div>
24   <div style="background-color: lightblue;">BLUE</div>
25   <div style="background-color: lightgreen;">Green div with more content.</div>
26 </div>
27
28
29
30 </body>
31 </html>
32
```

Output :

## The flex Property



The flex property is shorthand property for:

Flex-grow  
Flex-shrink  
Flex-basis

How to use Flex :

Syntax:

Flex: flex-grow flex-shrink flex-basis|auto|initial|inherit;

Definition of flex-grow:

A number specifying what properties item will grow relative to the remainder of the flexible items .

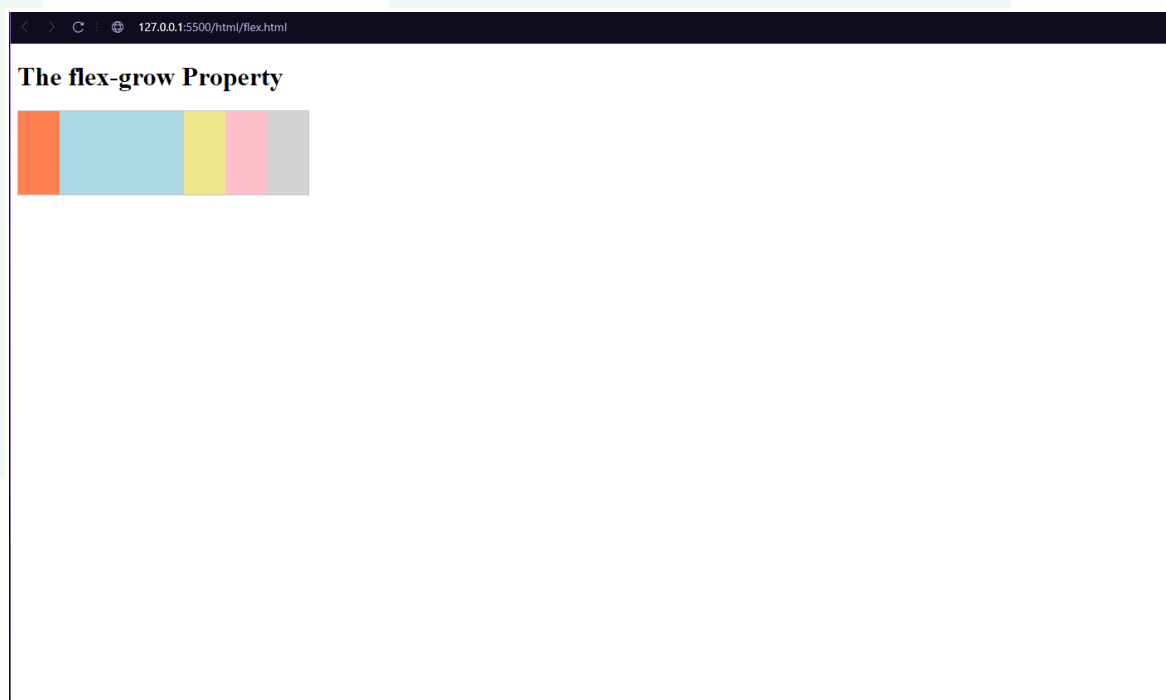
Input:

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 350px;
7   height: 100px;
8   border: 1px solid #c3c3c3;
9   display: flex;
10 }
11
12 #main div:nth-of-type(1) {flex-grow: 1;}
13 #main div:nth-of-type(2) {flex-grow: 3;}
14 #main div:nth-of-type(3) {flex-grow: 1;}
15 #main div:nth-of-type(4) {flex-grow: 1;}
16 #main div:nth-of-type(5) {flex-grow: 1;}
17 </style>
18 </head>
19 <body>
20
21 <h1>The flex-grow Property</h1>
22
23 <div id="main">
24   <div style="background-color: coral;"></div>
25   <div style="background-color: lightblue;"></div>
26   <div style="background-color: khaki;"></div>
27   <div style="background-color: pink;"></div>
28   <div style="background-color: lightgrey;"></div>
29 </div>
30
31
32
33 </body>
34 </html>
35

```

Output:



Definition of flex-shrink:

A number specifying what proportion item will shrink relative to the rest of the flexible items

Input:

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 350px;
7   height: 100px;
8   border: 1px solid #c3c3c3;
9   display: flex;
10 }
11
12 #main div {
13   flex-grow: 1;
14   flex-shrink: 1;
15   flex-basis: 100px;
16 }
17
18 #main div:nth-of-type(2) {
19   flex-shrink: 3;
20 }
21 </style>
22 </head>
23 <body>
24
25 <h1>The flex-shrink Property</h1>
26
27 <div id="main">
28   <div style="background-color: coral;"></div>
29   <div style="background-color: lightblue;"></div>
30   <div style="background-color: yellow;"></div>
31   <div style="background-color: pink;"></div>
32   <div style="background-color: lightgrey;"></div>
33 </div>
34
35 </body>
36 </html>

```

Output:



Definition of flex-basis:

The length of the item. Legal values: "auto", "inherit", or variety followed by "%", "px", "em" or the opposite length unit.

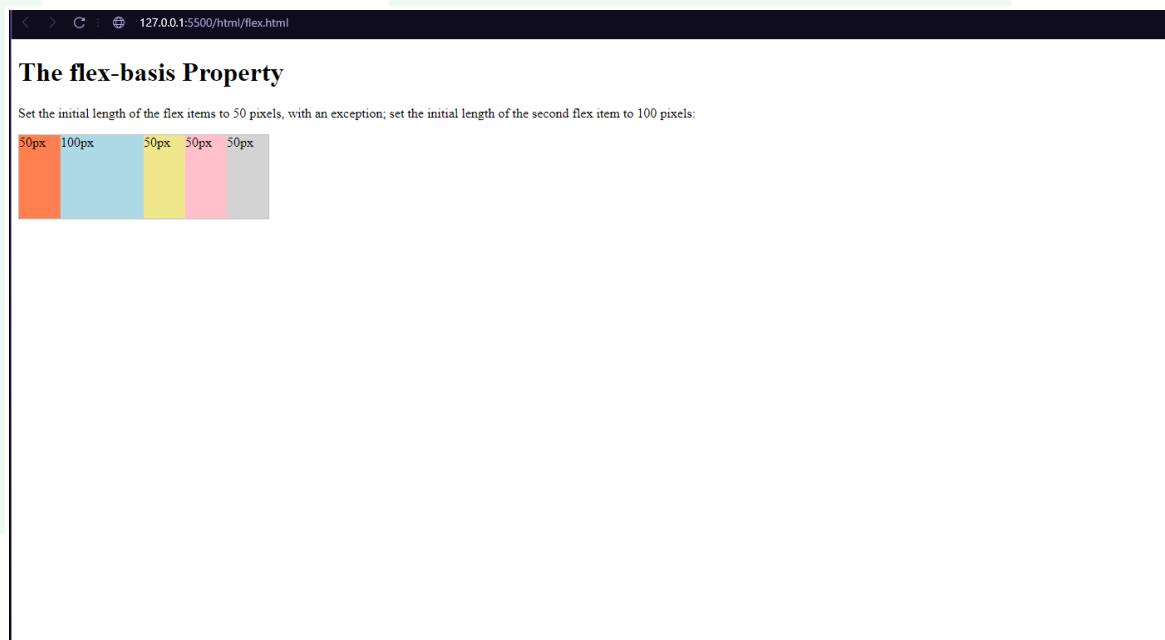
Input:

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 300px;
7   height: 100px;
8   border: 1px solid #c3c3c3;
9   display: flex;
10 }
11
12 #main div {
13   flex-grow: 0;
14   flex-shrink: 0;
15   flex-basis: 50px;
16 }
17
18 #main div:nth-of-type(2) {
19   flex-basis: 100px;
20 }
21 </style>
22 </head>
23 <body>
24
25 <h1>The flex-basis Property</h1>
26
27 <p>Set the initial length of the flex items to 50 pixels, with an exception; set the initial length of the second flex item to 100 pixels:</p>
28 <div id="main">
29   <div style="background-color: coral;">50px</div>
30   <div style="background-color: lightblue;">100px</div>
31   <div style="background-color: khaki;">50px</div>
32   <div style="background-color: pink;">50px</div>
33   <div style="background-color: lightgrey;">50px</div>
34 </div>
35
36
37 </body>
38 </html>
39

```

Output:



What is flex-direction:

Set of the direction of flexible-items inside the <div> element in reverse order.

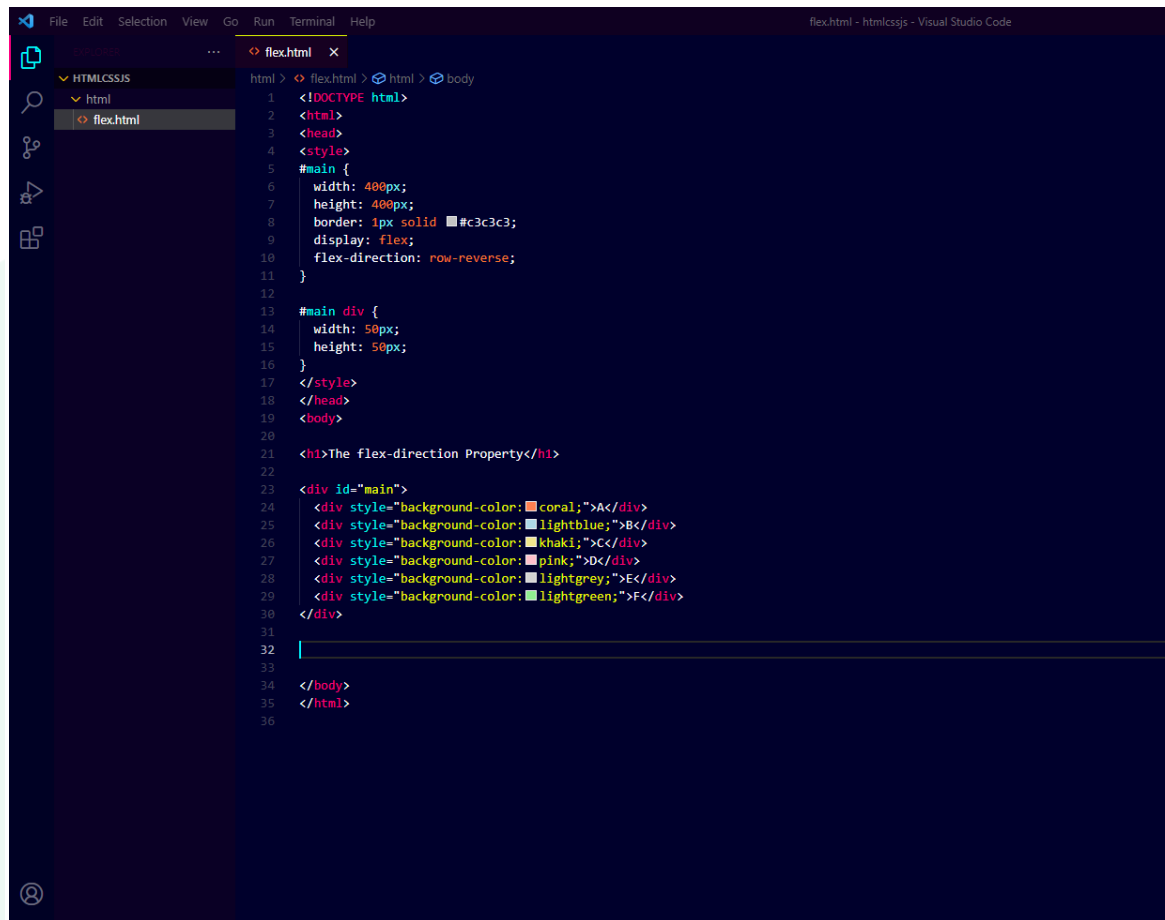
For example:

```

div{
display: flex;
flex-direction: row-reverse;
}

```

Input:

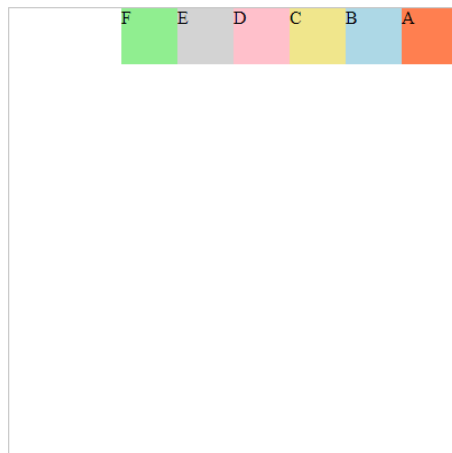


```

1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  #main {
6    width: 400px;
7    height: 400px;
8    border: 1px solid #c3c3c3;
9    display: flex;
10   flex-direction: row-reverse;
11 }
12
13 #main div {
14   width: 50px;
15   height: 50px;
16 }
17 </style>
18 </head>
19 <body>
20
21 <h1>The flex-direction Property</h1>
22
23 <div id="main">
24   <div style="background-color: coral;">A</div>
25   <div style="background-color: lightblue;">B</div>
26   <div style="background-color: khaki;">C</div>
27   <div style="background-color: pink;">D</div>
28   <div style="background-color: lightgrey;">E</div>
29   <div style="background-color: lightgreen;">F</div>
30 </div>
31
32
33
34 </body>
35 </html>
36
  
```

Output:

## The flex-direction Property



Align-items:

Definition:

The align-items property specifies the default alignment for items inside flexible container.

Examples:

```
div
{
display:flex;
align-items:center;
}
```

Input:



```

File Edit Selection View Go Run Terminal Help
flex.html - htmlcssjs - Visual Studio Code

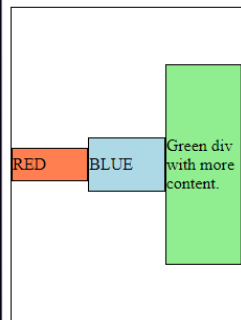
flex.html x
HTMLCSSJS
  html
    flex.html
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 220px;
7   height: 300px;
8   border: 1px solid black;
9   display: flex;
10  align-items: center;
11 }
12
13 #main div {
14   flex: 1;
15   border: 1px solid black;
16   display: flex;
17   align-items: center;
18 }
19 </style>
20 </head>
21 <body>
22
23 <h2>align-items: center</h2>
24
25 <div id="main">
26   <div style="background-color: coral; min-height: 30px;">RED</div>
27   <div style="background-color: lightblue; min-height: 50px;">BLUE</div>
28   <div style="background-color: lightgreen; min-height: 190px;">Green div with more content.</div>
29 </div>
30
31
32
33 </body>
34 </html>
35

```

Output:

127.0.0.1:5500/html/flex.html

**align-items: center**



Items are positioned at the begin to the container:

```
div
{
display:flex;
align-items:flex-start;
}
```

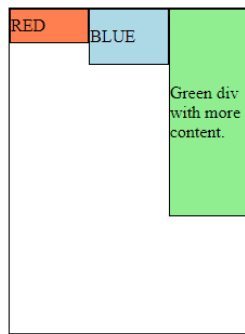
Input:

```

flex.html - htmlcssjs - Visual Studio Code
flex.html x
HTMLCSSJS
html
flex.html
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 220px;
7   height: 300px;
8   border: 1px solid black;
9   display: flex;
10  align-items: flex-start;
11 }
12
13 #main div {
14   flex: 1;
15   border: 1px solid black;
16   display: flex;
17   align-items: center;
18 }
19 </style>
20 </head>
21 <body>
22
23 <h2>align-items: flex-start</h2>
24
25 <div id="main">
26   <div style="background-color: coral;min-height:30px;">RED</div>
27   <div style="background-color: lightblue;min-height:50px;">BLUE</div>
28   <div style="background-color: lightgreen;min-height:190px;">Green div with more content.</div>
29 </div>
30
31
32 </body>
33 </html>
34

```

Output:

**align-items: flex-start**

Items are positioned at the tip of the container:

```
div
{
display:flex;
align-items:flex-end;
}
```

Input:

```

File Edit Selection View Go Run Terminal Help
flex.html - htmlcssjs - Visual Studio Code

HTMLCSSJS
  html
    flex.html

flex.html
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 220px;
7   height: 300px;
8   border: 1px solid black;
9   display: flex;
10  align-items: flex-end;
11 }
12
13 #main div {
14   flex: 1;
15   border: 1px solid black;
16   display: flex;
17   align-items: center;
18 }
19 </style>
20 </head>
21 <body>
22
23 <h2>align-items: flex-end</h2>
24
25 <div id="main">
26   <div style="background-color: coral; min-height: 30px;">RED</div>
27   <div style="background-color: lightblue; min-height: 50px;">BLUE</div>
28   <div style="background-color: lightgreen; min-height: 190px;">Green div with more content.</div>
29 </div>
30
31 </body>
32 </html>
33

```

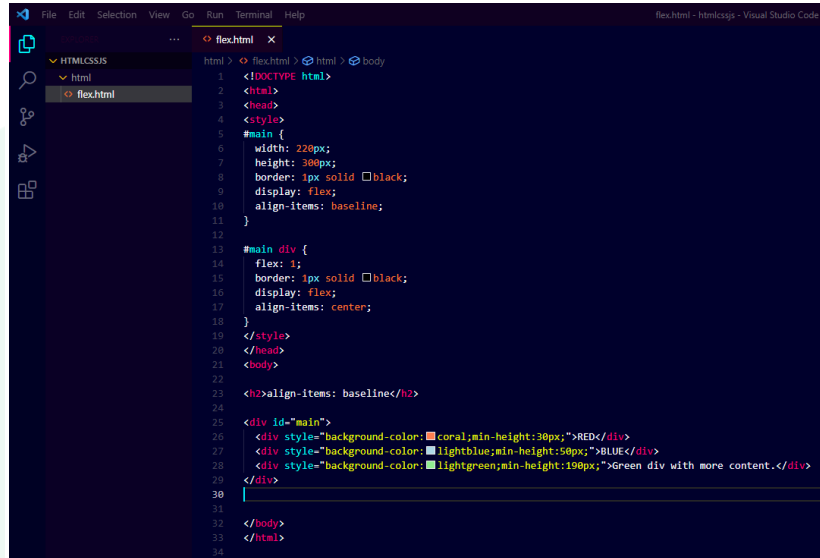
Output:



Items are positioned at the baseline of containers :  
div

```
{  
display: flex;  
align-items: baseline;  
}
```

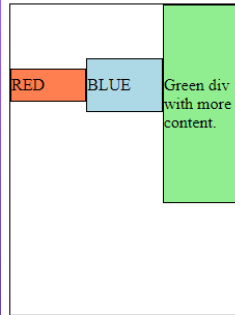
Input:



```
1 <!DOCTYPE html>  
2 <html>  
3 <head>  
4 <style>  
5 #main {  
6   width: 220px;  
7   height: 300px;  
8   border: 1px solid black;  
9   display: flex;  
10  align-items: baseline;  
11 }  
12  
13 #main div {  
14   flex: 1;  
15   border: 1px solid black;  
16   display: flex;  
17   align-items: center;  
18 }  
19 </style>  
20 </head>  
21 <body>  
22  
23 <h2>align-items: baseline</h2>  
24  
25 <div id="main">  
26   <div style="background-color: coral; min-height: 30px;">RED</div>  
27   <div style="background-color: lightblue; min-height: 50px;">BLUE</div>  
28   <div style="background-color: lightgreen; min-height: 190px;">Green div with more content.</div>  
29 </div>  
30  
31  
32 </body>  
33 </html>  
34
```

Output:

127.0.0.1:5500/html/flex.html

**align-items: baseline****Justify-Content Property:**

The justify-content property aligns the flexible container's items when the items don't use all available space on the main-axis.

**Example**

Align the flex items at middle of the container:

```
div{  
  display:flex;  
  justify-content:center;  
}
```

Input:

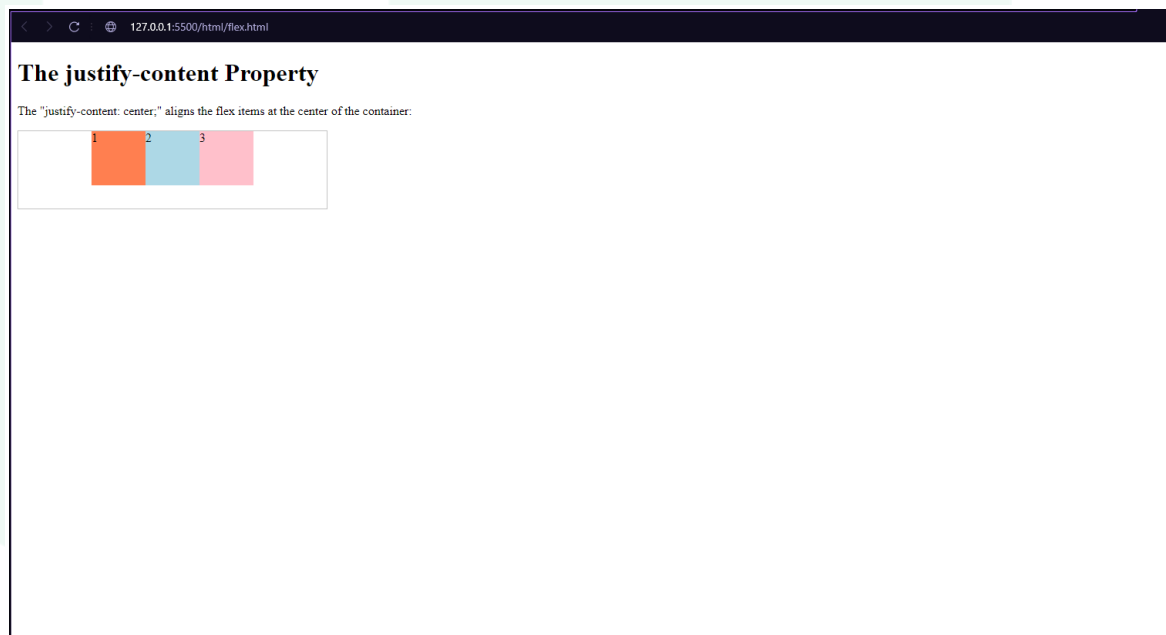
```

File Edit Selection View Go Run Terminal Help
flex.html - htmlcssjs - Visual Studio Code

flex.html
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 400px;
7   height: 100px;
8   border: 1px solid #c3c3c3;
9   display: flex;
10  justify-content: center;
11 }
12
13 #main div {
14   width: 70px;
15   height: 70px;
16 }
17 </style>
18 </head>
19 <body>
20
21 <h1>The justify-content Property</h1>
22
23 <p>The "justify-content: center;" aligns the flex items at the center of the container.</p>
24
25 <div id="main">
26   <div style="background-color: coral;">1</div>
27   <div style="background-color: lightblue;">2</div>
28   <div style="background-color: pink;">3</div>
29 </div>
30
31 </body>
32 </html>
33

```

Output:



## Example

Align the flex items at beginning of the container (this is default):

div

{

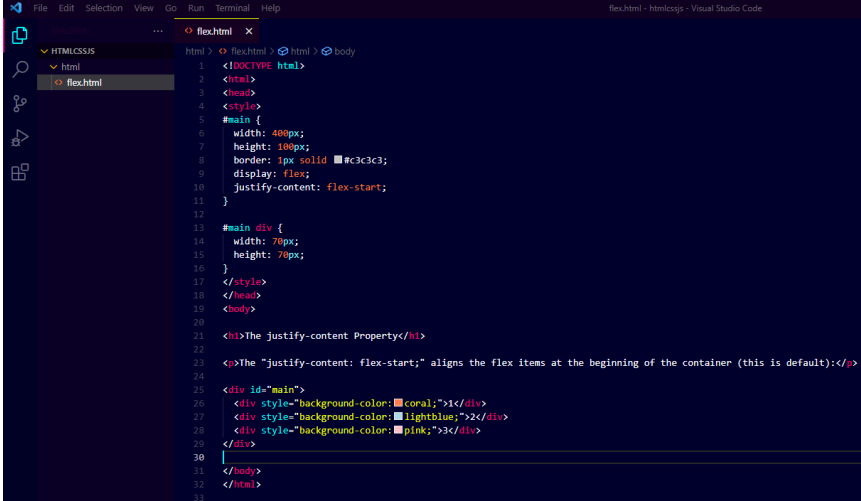


display:flex;

justify-content:flex-start;

}

Input:



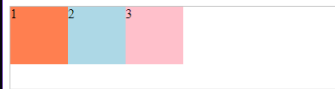
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5   #main {
6     width: 400px;
7     height: 100px;
8     border: 1px solid #cc3333;
9     display: flex;
10    justify-content: flex-start;
11  }
12
13  #main div {
14    width: 70px;
15    height: 70px;
16  }
17 </style>
18 </head>
19 <body>
20
21 <h1>The justify-content Property</h1>
22
23 <p>The "justify-content: flex-start;" aligns the flex items at the beginning of the container (this is default):</p>
24
25 <div id="main">
26   <div style="background-color: #coral;">1</div>
27   <div style="background-color: #lightblue;">2</div>
28   <div style="background-color: #pink;">3</div>
29 </div>
30
31 </body>
32 </html>
```

Output:

127.0.0.1:5500/html/flex.html

## The justify-content Property

The "justify-content: flex-start;" aligns the flex items at the beginning of the container (this is default):



Align the flex items at tip of the container:

```
div{  
  display:flex;  
  justify-content:flex-end;  
}
```

Input:

```
File Edit Selection View Go Run Terminal Help
flex.html - htmlcssjs - Visual Studio Code

flex.html x
html > flex.html > html > body
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 400px;
7   height: 100px;
8   border: 1px solid #c3c3c3;
9   display: flex;
10  justify-content: flex-end;
11 }
12
13 #main div {
14   width: 70px;
15   height: 70px;
16 }
17 </style>
18 </head>
19 <body>
20
21 <h1>The justify-content Property</h1>
22
23 <p>The "justify-content: flex-end;" aligns the flex items at the end of the container:</p>
24
25 <div id="main">
26   <div style="background-color: coral;">1</div>
27   <div style="background-color: lightblue;">2</div>
28   <div style="background-color: pink;">3</div>
29 </div>
30
31 </body>
32 </html>
33
```

Output:

127.0.0.1:5500/html/flex.html

## The justify-content Property

The "justify-content: flex-end;" aligns the flex items at the end of the container:



Display flex items with space between the lines:

```
div{
display: flex;
justify-content: space-between;}
```

Input:

```
File Edit Selection View Go Run Terminal Help
flex.html - htmxup - Visual Studio Code

flex.html
HTMLCSSJS
  html
    flex.html

flex.html
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 400px;
7   height: 100px;
8   border: 1px solid #c3c3c3;
9   display: flex;
10  justify-content: space-between;
11 }
12
13 #main div {
14   width: 70px;
15   height: 70px;
16 }
17 </style>
18 </head>
19 <body>
20
21 <h1>The justify-content Property</h1>
22
23 <p>The "justify-content: space-between;" displays the flex items with space between the lines:</p>
24
25 <div id="main">
26   <div style="background-color: coral;">1</div>
27   <div style="background-color: lightblue;">2</div>
28   <div style="background-color: pink;">3</div>
29 </div>
30
31 </body>
32 </html>
33
```

Output:

127.0.0.1:5500/html/flex.html

## The justify-content Property

The "justify-content: space-between;" displays the flex items with space between the lines:

1

2

3

Flex - Wrap definition:

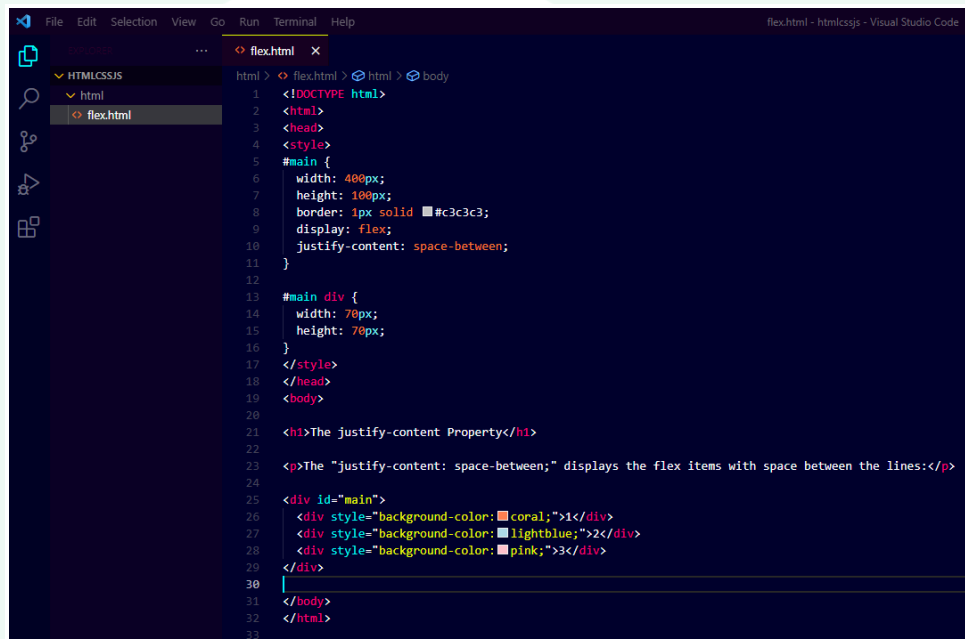
The flex-wrap property specifies whether flexible items should wrap or not.

Example:

Make the flexible items wrap if is specified :

```
div{  
display:flex;  
flex-wrap:wrap;  
}
```

Input:



```
1 <!DOCTYPE html>  
2 <html>  
3 <head>  
4 <style>  
5 #main {  
6   width: 400px;  
7   height: 100px;  
8   border: 1px solid #c3c3c3;  
9   display: flex;  
10  justify-content: space-between;  
11 }  
12  
13 #main div {  
14   width: 70px;  
15   height: 70px;  
16 }  
17 </style>  
18 </head>  
19 <body>  
20  
21 <h1>The justify-content Property</h1>  
22  
23 <p>The "justify-content: space-between;" displays the flex items with space between the lines:</p>  
24  
25 <div id="main">  
26   <div style="background-color: coral;">1</div>  
27   <div style="background-color: lightblue;">2</div>  
28   <div style="background-color: pink;">3</div>  
29 </div>  
30  
31 </body>  
32 </html>  
33
```

Output:

127.0.0.1:5500/html/flex.html

## The justify-content Property

The "justify-content: space-between;" displays the flex items with space between the lines:



### Align-self:

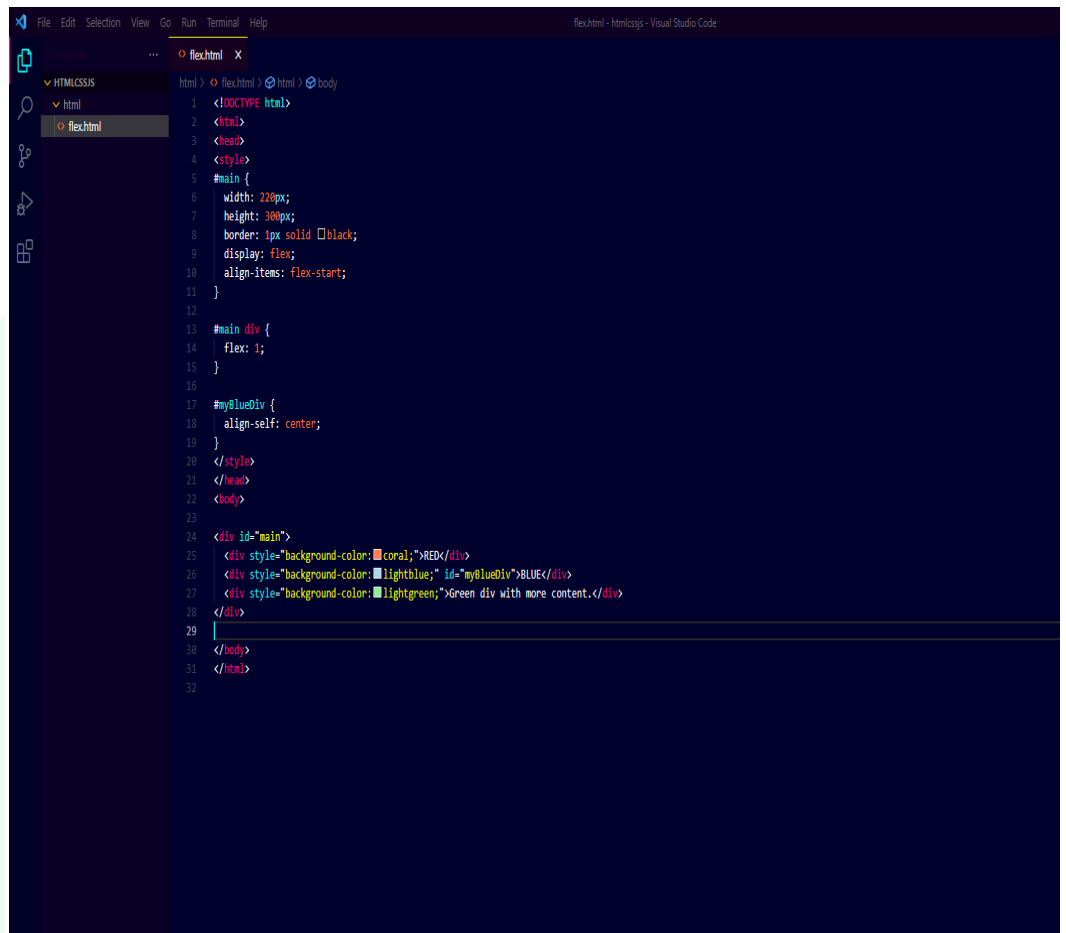
The align-self property specifies the alignment for selected item inside the flexible container.

## Example

Center the alignments for one amongst items a flexible element:

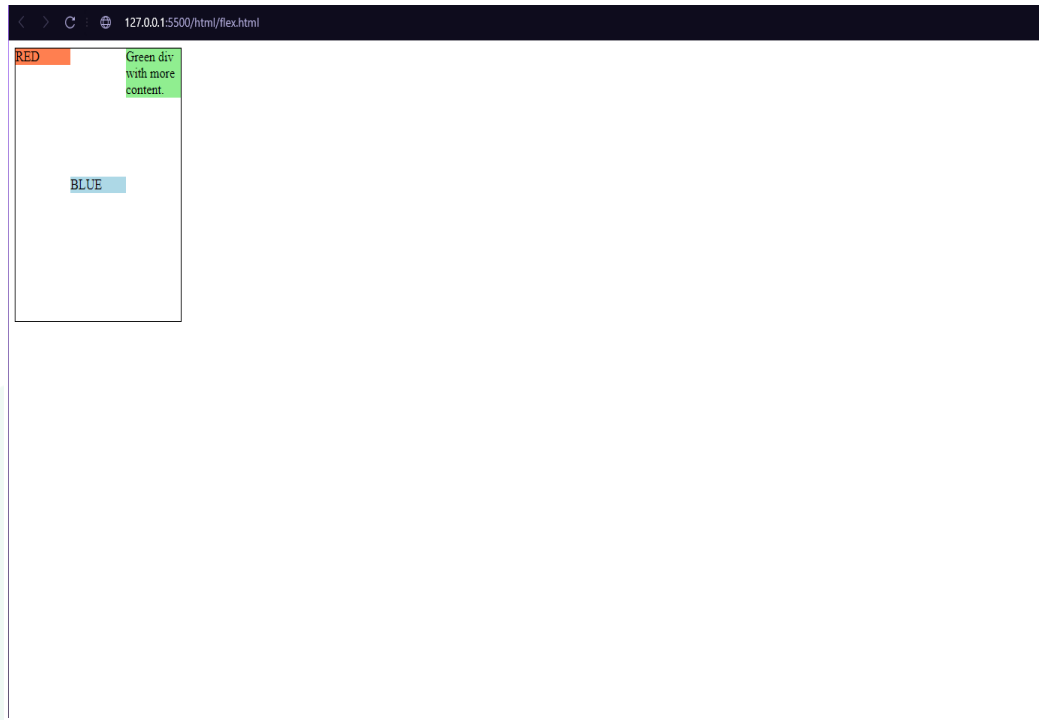
```
#myBlueDiv{  
  align-self:center;  
}
```

Input:



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 #main {
6   width: 220px;
7   height: 300px;
8   border: 1px solid black;
9   display: flex;
10  align-items: flex-start;
11 }
12
13 #main div {
14   flex: 1;
15 }
16
17 #myBlueDiv {
18   align-self: center;
19 }
20 </style>
21 </head>
22 <body>
23
24 <div id="main">
25   <div style="background-color: coral;">RED</div>
26   <div style="background-color: lightblue;" id="myBlueDiv">BLUE</div>
27   <div style="background-color: lightgreen;">Green div with more content.</div>
28 </div>
29
30 </body>
31 </html>
32
```

Output:



#### References:

1. [https://www.w3schools.com/css/css3\\_flexbox.asp](https://www.w3schools.com/css/css3_flexbox.asp)
2. [https://www.w3schools.com/css/css3\\_flexbox\\_container.asp](https://www.w3schools.com/css/css3_flexbox_container.asp)
3. [https://www.w3schools.com/css/css3\\_flexbox\\_items.asp](https://www.w3schools.com/css/css3_flexbox_items.asp)
4. [https://www.w3schools.com/css/css3\\_flexbox\\_responsive.asp](https://www.w3schools.com/css/css3_flexbox_responsive.asp)
5. <https://www.youtube.com/watch?v=5qVuORLniwM>
6. <https://www.youtube.com/watch?v=JJSoEo8JSnc>