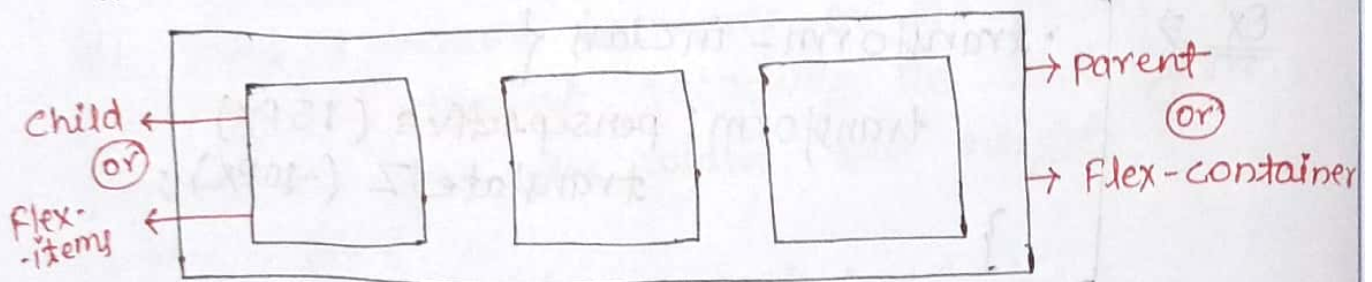


* CSS FlexBox

Flexbox is a great way to get more flexibility in your layouts and to simplify responsive layout design.

The first step is to set `display: flex` on a 'container' element. The "children" to the flex container becomes flex item.

A set of properties can be applied to flex container, and have an effect to all the items as a whole and a different set of properties can be applied to flex items and have their effect on the targeted items.



* Flex - Container Properties

Flex - container is a parent container. The flex container becomes flexible by setting the display property to flex.

Properties

- i) flex-direction
- ii) flex-wrap
- iii) flex-flow
- iv) justify-content
- v) align-items

vi) align-content

Ex

```
flex-container {
  display: flex;
}
```

#i) Flex - direction

The flex-direction property defines in which direction the container wants to stack the flex items.

```
• flex-container {  
    display : flex ;  
    flex-direction : row / column / row-reverse /  
    column-reverse ;  
}
```

#ii) Flex - Wrap

The flex-wrap property specifies whether the flex items should "wrap" or "not"

```
• flex-container {  
    display : flex ;  
    flex-wrap : Wrap / no wrap / wrap-reverse  
    }  
    ↳ default
```

#iii) Flex - flow

The flex-flow property is a shorthand property for setting both the flex-direction and flex-wrap property.

```
• flex-container {  
    display : flex ;  
    flex-flow : row wrap ;  
    }  
    ↓           ↓  
    flex-direction flex-wrap
```


#iv) Justify - Content

Justify - content property is used to align the flex items

NOTE ⇒ It is responsible for the alignment in horizontal - Axis

property :-

i) center

ii) flex-start

iii) flex-end

iv) space-around

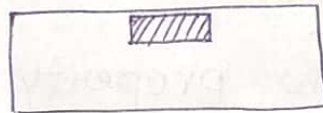
v) space-between

vi) space-evenly

① Center

The center value align the flex items at the center of the container.

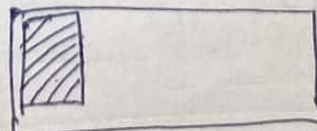
```
• flex-container {  
  display: flex;  
  justify-content: center;  
}
```



② Flex-Start (Default)

The flex-start value aligns the flex items at the beginning of the container.

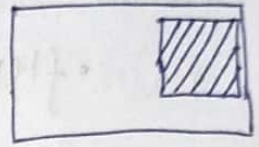
```
• flex-container {  
  display: flex;  
  justify-content: flex-start;  
}
```



③ Flex - End

The flex-end value aligns the flex items at the end of the container.

justify-content : flex-end;



④ Space - around

⇒ The space-around value displays the flex items with space before, between, and after the items.

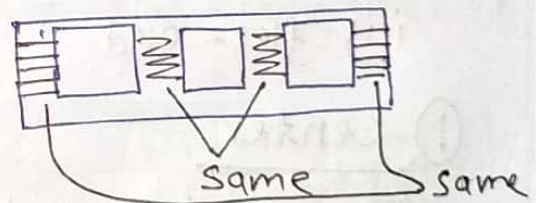
⇒ The space between the boxes will be the same, the space at the start and end will be the same, but the space between boxes and at the start or end will not be the same.

• flex-container {

display : flex;

justify-content : space-around;

}



⑤ Space - between

⇒ The space-between value displays the flex items with space between the boxes.

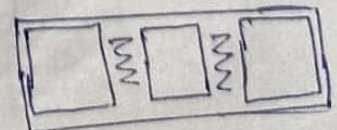
⇒ Equal space distribution between the boxes will happen, and the start/end boxes will move to their extreme positions.

• flex-container {

display : flex;

justify-content : space-between;

}



⑥ Space - Evenly

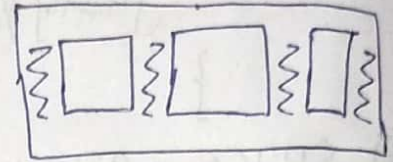
The Space - distribution at the start, between and end will be equal

```
• flex-container {
```

```
  display: flex;
```

```
  justify-content: space-around;
```

```
}
```



#⑤ Align - items

The align-items property is used to align the flex items. Used for single row only

NOTE: It is responsible for the alignment of the flex items in the Vertical-Axis

Property:-

i) center

ii) flex-start

iii) flex-end

iv) stretch

v) baseline

① center

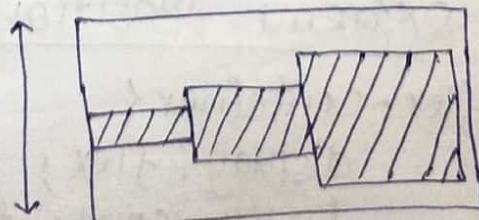
The center value align the flex items in the middle of the container.

```
• flex-container {
```

```
  display: flex;
```

```
  align-items: center;
```

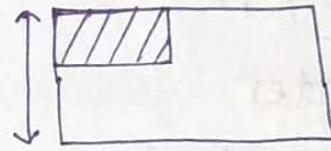
```
}
```



② Flex - Start

The flex-start value aligns the flex items at the top of the container.

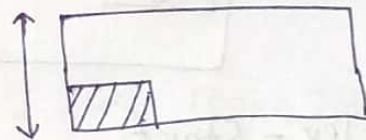
```
align-items: flex-start;  
}
```



③ Flex - End

The flex-end value aligns the flex items at the bottom of the container.

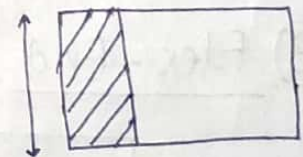
```
align-items: flex-end;  
}
```



④ Stretch (default)

The Stretch value stretches the flex items to fill the container.

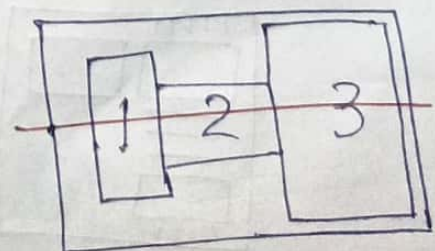
```
align-items: stretch;  
}
```



⑤ Baseline

The baseline value aligns the flex items such as their baseline align.

```
align-items: baseline;  
}
```



⑥ Align - Content

It is same as align-items but it is used when our content is of more than one rows.

Property :-

i) center

ii) flex-start

iii) flex-end

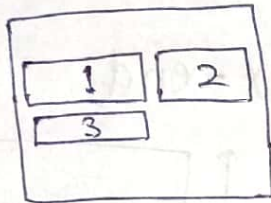
iv) stretch (default)

v) space-around

vi) space-between

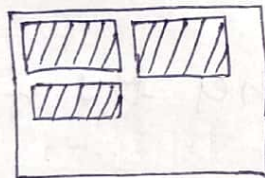
vii) space-evenly

① center



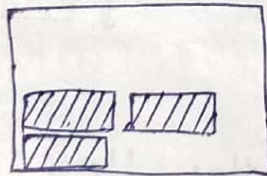
```
align-content: center;  
}
```

② Flex-start



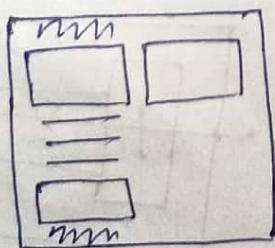
```
align-contentcontent: flex-start;  
}
```

③ Flex-end



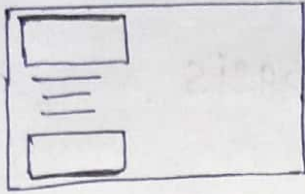
```
align-contentcontent: flex-end;  
}
```

④ space-around



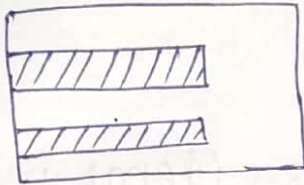
```
align-content: space-around;  
}
```

⑤ Space-between



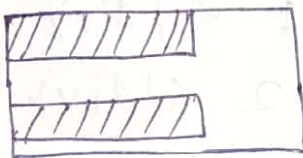
align-content: space-between;
}

⑥ Space-evenly



align-content: space-evenly;
}

⑦ Stretch



align-content: stretch;
}

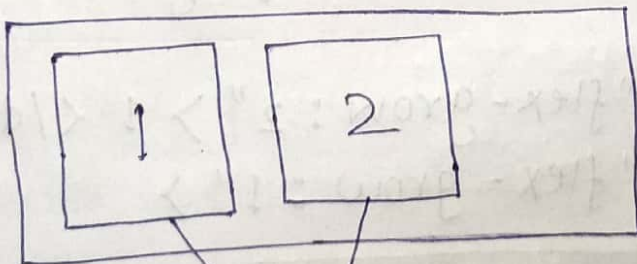
Q: How to perfectly center a flex item/div?

justify-content: center;

align-items: center;

* Flex Items Property (child Element)

A direct child element of a flex container automatically becomes flexible (flex) items.



Flex-items (child)

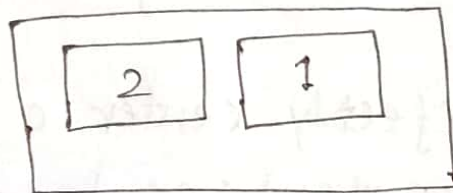
Property

- i) Order
- ii) flex-grow
- iii) flex-shrink
- iv) flex-basis
- * v) flex
- vi) align-self

#(i) Order

Specifies the order of the flex items inside the same container.

```
<div style="order: 2" > 1 </div>  
<div style="order: 1" > 2 </div>
```

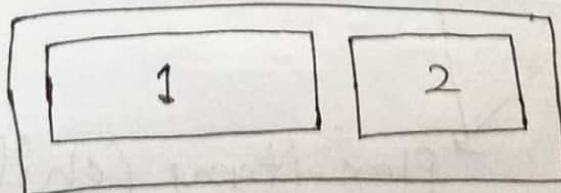


#(ii) flex-grow

The flex-grow property specifies how much a flex item will grow relative to the rest of the flex items.

The value must be number, default value is '0'.

```
<div style="flex-grow: 2" > 1 </div>  
<div style="flex-grow: 1" >
```

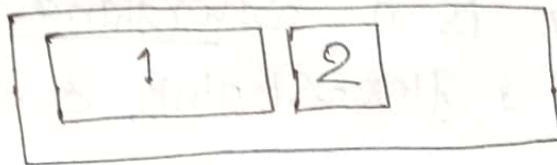


iii) flex-shrink

The flex-shrink property specifies how much a flex will shrink relative to the rest of the flex items.

The value must be a number, default value is '1'

```
<div style="flex-grow: 0" > 1 </div>  
<div > 2 </div>
```

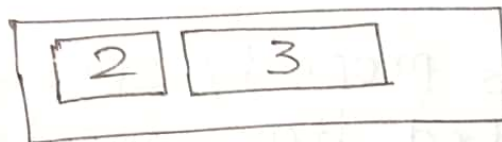


#iv) flex-basis

The flex-basis property specifies the initial (customized) length of a flex item.

```
<div > 2 </div>
```

```
<div style="flex-basis: 300px" > 3 </div>
```



Q) Width Vs flex-basis

Width ⇒ Visually you won't notice any difference when the content is less.

⇒ when the content is more than the width of the flex, then the additional content will be trimmed out.

⇒ The overflow will be hidden.

⇒ That particular flex will be no longer responsive.

flex-basis \Rightarrow Visually looks same when content is less.

is less.

⇒ If the content is more than the flex width, then the width will be increased

⇒ content will not be trimmed out

\Rightarrow That particular flex will be responsive

#(V) Flex

The flex property is a Shorthand property for the flex-grow, flex-shrink, & flex-basis property.

Ex ⇒ flex : 0 0 300px;
 ↓ ↓ ↓
 flex-grow flex-shrink flex-basis

#(vi) Align-Self

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

⇒ The align-self property overrides the default alignment set by the container's align-items property.

Property:-

Syntax

- i) Stretch
- ii) Center
- iii) flex-start
- iv) flex-end
- v) baseline

align-self : center / flex-start
flex-end / baseline