

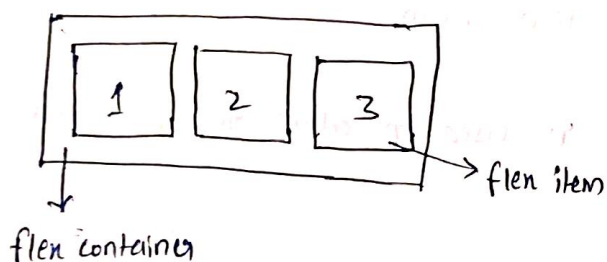
CSS Flexbox:- one dimensional layout model.

* Before the Flexbox Layout module, there were four layout modes:

- Block, for sections in a webpage.
- Inline, for text
- Table, for two-dimensional table data.
- Positioned, for explicit position of an element.

* The flexible box layout module, makes it easier to design flexible responsive layout structure without using float or positioning.

Flexbox Elements:-



HTML:-

```
<div class="flex-container">
  <div>1</div>
  <div>2</div>
  <div>3</div>
</div>
```

Parent Element (Container):-

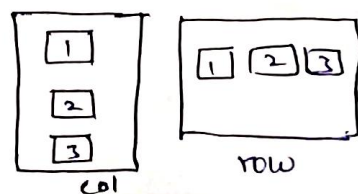
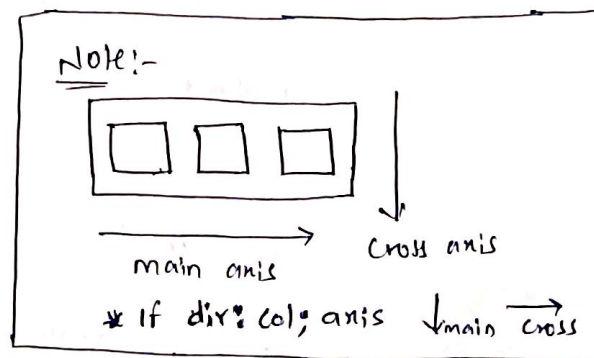
* The flex container becomes flexible by setting the display property to flex:

```
flex-container {
  display: flex;      (or) inline-flex
}
```

Properties of flex container:-

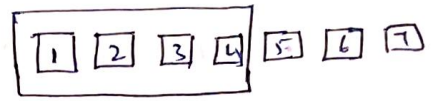
1.) Flex-direction : column (or) column-reverse (or) row (or)

row-reverse;

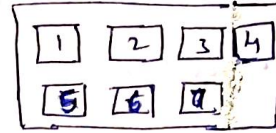


2) flex-wrap : wrap (or) nowrap (or) wrap-reverse ;

En:- If nowrap is value then if items are more than screen size then they go out of screen (or) container (default)



En:- If it is wrap then they will be adjusted within the width and enter one's comes to down rows.



3) flex-flow is a shorthand property of setting both flex-direction and flex-wrap.

En:- flex-flow : row wrap ;

4) justify-content property is used to align the flex items: values (direction)
(main axis)

* Center → aligns the flex items at the center of the container.

* flex-start → aligns the flex items at the beginning of the container (default)

* flex-end → aligns the flex items at the end of the container.

* space-around → displays the flex items with space before, between and after the lines.

* space-between → displays the flex items with space between the lines.

* space-evenly

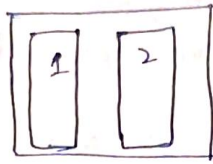
5) align-items:- used to align the flex item in cross axis direction (cross axis)

values

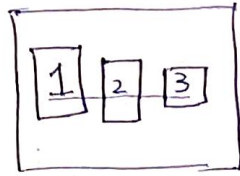
center (or) flex-start (or) flex-end (or) stretch (default)

(or) baseline.

stretch →



baseline →

items ~~get~~ aligned by the text baseline

⑥ Align-content :- aligns lines of content along the cross-axis

* Multiple rows must be existed to get this property.

Values :-

stretch

flex-start

flex-end

center

space-between

space-around

* It's like applying both justify-content and align-items.

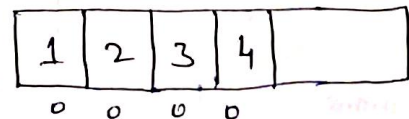
Flex item Properties.

1) Order

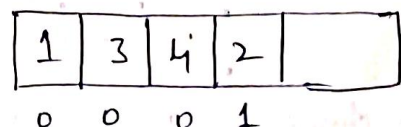
By default order of every item is '0'. If order is same then they display as same structure in HTML. (top to bottom)

ex:- { order : 1 /* Whole number, can be also integer */ }

order:-



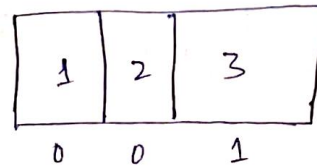
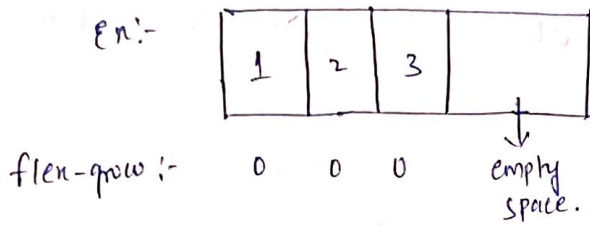
order:-



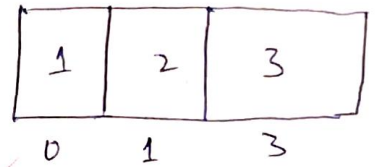
2) flex-grow:- It is used to occupy the extra space by an item.

* By default flex-grow value is 0 for all items.

Ex:-

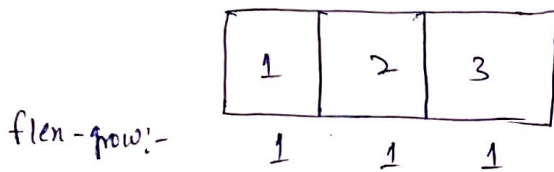


↓
takes all remained space.



↓
takes 1 part of remained space

↓
takes 3 parts of remained space.



↓
equally took remained space.

Ex:-

flex-grow: 1 /* fraction value (or) whole numbers */
only (+ve) no }

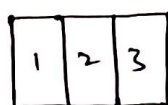
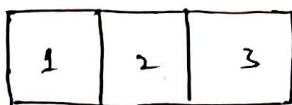
3) flex-shrink:-

* It's default value is 1 for flex-items

* shrinking possible upto a certain width (or point) after which the items simply overflow.

* if shrink is '0' then items don't shrink, if width is decreased.

Ex:-



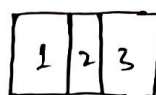
→ when width of browser is decreased



further decreased after a certain point

flex-shrink:-

1	1	1
---	---	---



→ when width of browser is decreased

flex-shrink:-

1	2	1
---	---	---

ex:- {

flex-shrink : 4 /* whole number (or) positive fraction */
}

4) flex-basis:-

- * It is used to fix the initial width of an flex-item (Don't use width for setting initial width for an item use only flex-basis).

flex-basis : auto (default) (or) length in px, rem, %

5) flex:- shorthand property for flex grow, flex-shrink and flex basis

{

flex : 2 5 200px;

}

↓ ↓ ↓
grow shrink basis

flex : 2; // flex-grow

flex : 2 2; // flex-grow & shrink

flex : 100px; // flex-basis

flex : 2 100px // flex-grow & basis

flex : none initial auto // default.

6) Align-self:- used to control alignment of individual flex items.

- * values are same as align-items property on flex-container.
- * so, its reflection will be along cross-axis.

{

Align-self : auto (default) (or) stretch (or) flex-start (or) flex-end

}

Note
* auto is, ^{it} inherits from align-items from the container.

Uses:-

- Flex box is famous & best in centering an element (or) item.
- Variable heights

Holy Grail Layout:-

