Date -11-02-21

* xs (for phones - screens less than 768px wide)
* sm (for tablets - screens equal to or greater than 768px wide)
* md (for small laptops - screens equal to or greater than 992px wide)
* lg (for laptops and desktops - screens equal to or greater than 1200px wide)

-----------------------------------------------Bootstrap grid example---------------------------------------------

<!DOCTYPE html>

<html lang="en">

<head>

<title>Bootstrap Example</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>

</head>

<body>

<div class="container">

<h1>Hello World!</h1>

<div class="row">

<div class="col-sm-4 col-lg-12 col-md-6 col-xs-12" style="background-color:lavender;">.col 1</div>

<div class="col-sm-4 col-lg-12 col-md-6 col-xs-12" style="background-color:lavenderblush;">.col 2</div>

<div class="col-sm-4 col-lg-12 col-md-6 col-xs-12" style="background-color:lavender;">.col 3</div>

</div>

</div>

</body>

</html>

---------------------------------------------------flexbox model------------------------------------------------

flexbox as being one dimensional means that flexbox deals with layout in one dimension at a time — either as a row or as a column.

---------------------flex container property----------------------

1. Flex-direction: row/column/row-reverse/column-reverse

1. flex-container

Items display in a row (the flex-direction property's default is row).

* The items start from the start edge of the main axis.
* The items do not stretch on the main dimension, but can shrink.
* The items will stretch to fill the size of the cross axis.
* The [flex-basis](https://developer.mozilla.org/en-US/docs/Web/CSS/flex-basis) property is set to auto.
* The [flex-wrap](https://developer.mozilla.org/en-US/docs/Web/CSS/flex-wrap) property is set to nowrap.

The result of this is that your items will all line up in a row, using the size of the content as their size in the main axis. If there are more items than can fit in the container, they will not wrap but will instead overflow. If some items are taller than others, all items will stretch along the cross axis to fill its full size.

1. Flex-wrap: wrap/nowrap/wrap-reverse
2. Justify-content: center/flex-start/flex-end/space-between

The *center* value aligns the flex items at the center of the container:

The *flex-start* value aligns the flex items at the beginning of the container(by default)

The *flex-end* value aligns the flex items at the end of the container:

The *space-around* value displays the flex items with space before, between, and after the lines(each box have its own space in left and right direction).

The *space-between* value displays the flex items with space between the lines only.(no space in start and end).

Space-evenly - all space between the lines,start,end all same.

1. Align-items: center/flex-start/flex-end/stretch

---------------------flex-child property-----------------

1. The order property specifies the order of the flex items.

The first flex item in the code does not have to appear as the first item in the layout.

The order value must be a number, default value is 0.

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

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

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

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

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

If the items don’t have a size then the content's size is used as the flex-basis. This is why when we just declare display: flex on the parent to create flex items, the items all move into a row and take only as much space as they need to display their contents

5) 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.

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