# Bootstrap 4 (Part 1)

ITS290F

# What is Bootstrap?

- Bootstrap is a free **front-end framework** for faster and easier web development
- Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins
- Bootstrap also gives you the ability to easily create **responsive designs**

# What is Responsive Web Design?

Responsive web design is about creating web sites which **automatically adjust** themselves to look good on all devices, from small phones to large desktops.

# Responsive Design

Responsive design was introduced to help designers build one site on one domain that responds to a users viewport.  There are 2 necessary elements:

1. A **meta viewport tag** to disable scaling and
2. **Media queries** to alter the design as the page gets smaller.

A big challenge with responsive design is **finding a balance between the content needs for both mobile and desktop**. A desktop site has a lot of visual real estate that is often filled with carousels, videos, large parallax background images, and large blocks of text.

If you load a feature-rich website on a mobile device you often increase the page load for mobile visitors.

# Did You Know?

There are many frameworks for responsive web design.

- Foundation by Zurb: used by Adobe, Amazon, HP, eBay, etc.
- Bulma: open-source CSS framework (No jQuery or JavaScript library)
- W3.CSS: standard CSS framework (No jQuery or JavaScript library)
- Pure.css, Bootstrap, and more…

# Why Use Bootstrap?

- **Easy to use**: Anybody with just basic knowledge of HTML and CSS can start using Bootstrap
- **Responsive features**: Bootstrap's responsive CSS adjusts to phones, tablets, and desktops
- **Mobile-first approach**: In Bootstrap, mobile-first styles are part of the core framework
- **Browser compatibility**: Bootstrap 4 is compatible with all modern browsers (Chrome, Firefox, Internet Explorer 10+, Edge, Safari, and Opera)

# Where to Get Bootstrap 4?

There are two ways to start using Bootstrap 4 on your own website

You can:

- Include Bootstrap 4 from a **CDN (Content Delivery Network)**
- Download Bootstrap 4 from `getbootstrap.com`

# Bundle of Bootstrap 4

Bootstrap components require the use of JavaScript plugins and Popper to function. For example:

- Alerts for dismissing
- Buttons for toggling states and checkbox/radio functionality
- Collapse for toggling visibility of content
- Dropdowns for displaying and positioning
- Tooltips and popovers for displaying and positioning

# Create a Web Page with Bootstrap 4

1. Add the HTML5 `doctype`
2. Bootstrap 4 is mobile-first
3. Containers

# Add the HTML5 doctype

Bootstrap 4 uses HTML elements and CSS properties that require the HTML5 doctype.

Always include the HTML5 doctype at the beginning of the page, along with the lang attribute and the correct character set.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
  </head>
</html>
```

# Bootstrap 4 is Mobile-first

Bootstrap 4 is designed to be responsive to mobile devices. Mobile-first styles are part of the core framework.

To ensure proper rendering and touch zooming, add the following <meta> tag inside the <head> element:

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

The `width=device-width` part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).
The `initial-scale=1` part sets the initial zoom level when the page is first loaded by the browser

# Bootstrap 4 Containers

# Container

A Bootstrap **container** is the **root** of the Bootstrap 4 **grid system** and it is used to control the width of the layout.

The Bootstrap 4 **container contains all the elements in a page**. This means your page should have the following structure: first the body of the HTML page, inside of it you should add the container and all the other elements inside the container.

```
<body>
   <div class="container">
    ...
   </div>
</body>
```

# Container

Bootstrap 4 also requires a containing element to wrap site contents.
There are two container classes to choose from:

1. The `.container` class provides a responsive fixed width container. The width (max-width) of the browser window will change at different breakpoints.
2. The `.container-fluid` class provides a full width container, spanning the entire width of the viewport.


# Basic Bootstrap 4 Pages

1. `Container` Example

   https://www.w3schools.com/bootstrap4/tryit.asp?filename=trybs_gs_container&stacked=h

2. `Container Fluid` Example

   https://www.w3schools.com/bootstrap4/tryit.asp?filename=trybs_gs_container-fluid&stacked=h

# Bootstrap 4 Rows

# Row

Bootstrap 4 **rows** are **horizontal slices** of the screen. They are used only as **wrappers for columns**.

```
<div class="row">
  ...
</div>
```

# About Rows

- They are **only used for containing columns**. If you place other elements inside the row along with columns you will not get the expected result.
- They **have to be placed in containers**. If you don't do this, you will get a horizontal scroll on your page.
- The **columns have to be children of the row**. Otherwise they will not align. The rows and columns are created to work together in this strict hierarchy.

# Bootstrap 4 Columns

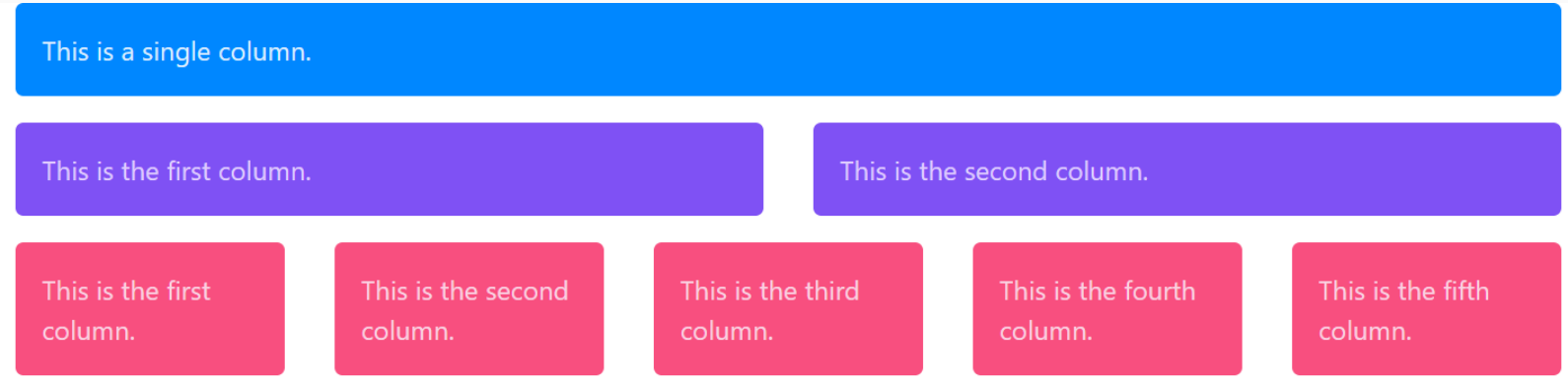
# Column

**Columns** help you divide the screen **horizontally**.

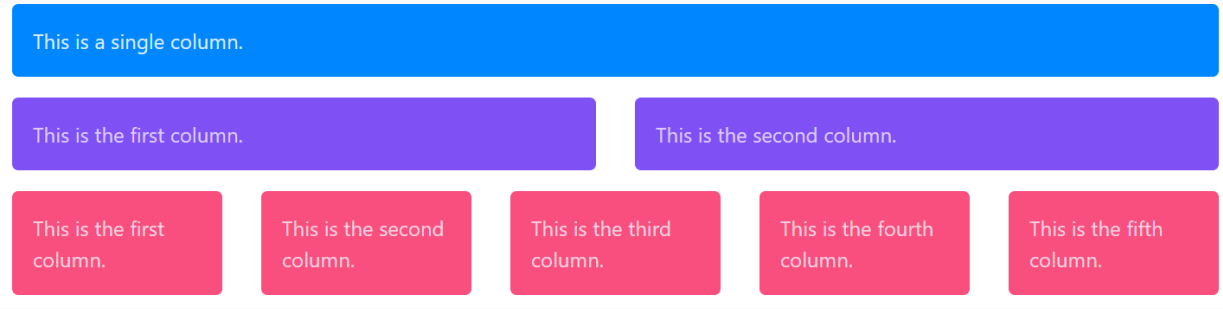
If you place a single column in your row, it will take up all the width.

If you add two columns, they will each take 1/2 from the width.

And so it goes for any number of columns.



# Column




```
<div class="container">
  <div class="row">
    <div class="col">
      ...
    </div>
  </div>
  <div class="row">
    <div class="col">
      ...
    </div>
    <div class="col">
      ...
    </div>
  </div>
  <div class="row">
    <div class="col">
      ...
    </div>
    <div class="col">
      ...
    </div>
    <div class="col">
      ...
    </div>
    <div class="col">
      ...
    </div>
    <div class="col">
      ...
    </div>
  </div>
</div>
```

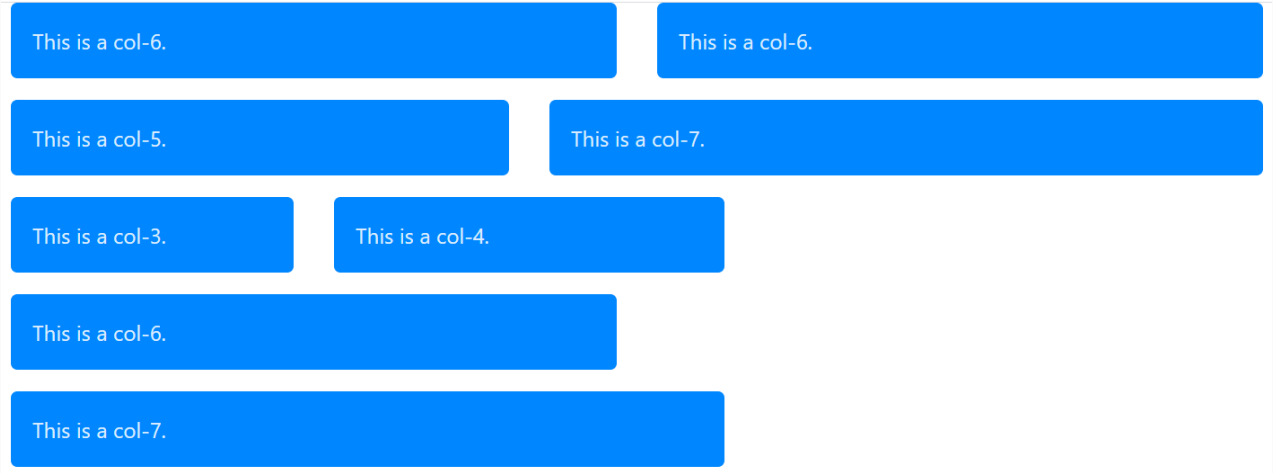
Example04.01

# Setting Sizes for Columns

Width for the column is set dynamically. That means that depending on the number of columns in a row, the width of a column will be the width of the container divided by the number of columns.

But there is another way to define columns.

The **Bootstrap grid** consists of **12 columns**. You can select any size from 1 to 12 for your column. If you want 3 equal columns, you can use `col-4` for each one (because 3*4 cols each = 12). Or you can set different sizes for them. Here are some examples:




Example04.02

# Bootstrap 4 Grids

# Bootstrap 4 Grid System

Bootstrap's grid system is built with flexbox and allows up to 12 columns across the page.

If you do not want to use all 12 columns individually, you can group the columns together to create wider columns:

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 |
| span 4 | | | span 4 | | | span 4 | | | | |
| span 4 | | | span 8 | | | | | | | |
| span 6 | | | | | | span 6 | | | | |
| span 12 | | | | | | | | | | |

The grid system is responsive, and the columns will re-arrange automatically depending on the screen size.

Make sure that the sum adds up to 12 or fewer (it is not required that you use all 12 available columns).

# Example

**Put One, Two, Three, … into 12 columns**


**Put Six Buttons into 6 columns**

# Setting **Breakpoints** for Columns

This is a col-6.

This is a col-6.

This is a col-5.

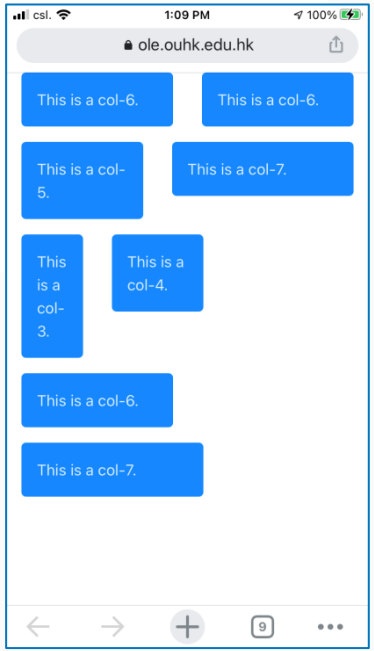
This is a col-7.

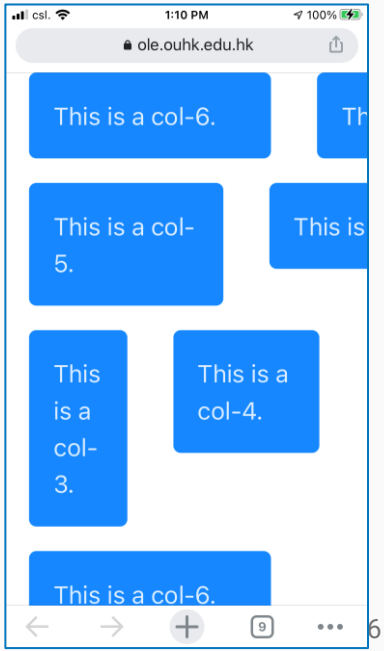
This is a col-3.

This is a col-4.

This is a col-6.

This is a col-7.




If you take the example above and want to **display it on mobile**, you may run into some problems.

Example04.02

# Breakpoints

A breakpoint is a Bootstrap 4 **variable** that stands for a **screen resolution**.

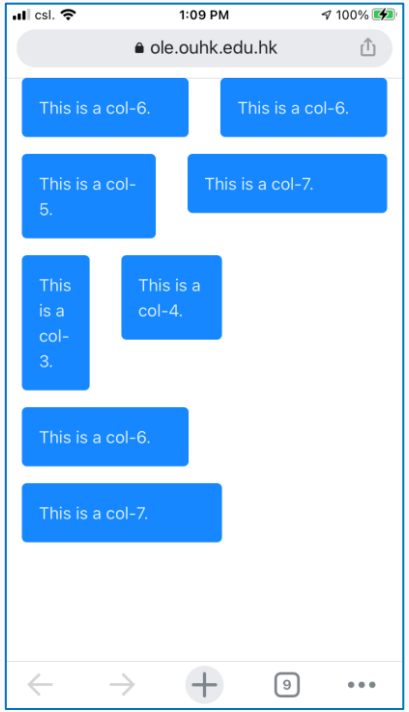
When you are specifying **a breakpoint for a class**, you are telling ***the class to be active only for resolutions that are at least as big as the number that the breakpoint holds***.

The Bootstrap 4 grid system has five classes:

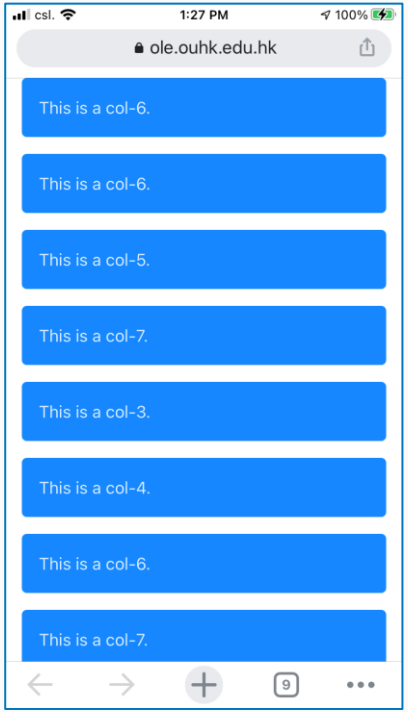
- `.col-` (extra small devices - screen width less than 576px)
- `.col-sm-` (small devices - screen width equal to or greater than 576px)
- `.col-md-` (medium devices - screen width equal to or greater than 768px)
- `.col-lg-` (large devices - screen width equal to or greater than 992px)
- `.col-xl-` (xlarge devices - screen width equal to or greater than 1200px)

# Breakpoint Example: Large vs. small screens

Let's say you want to display two columns on the same line on bigger screens (such as laptops & desktops) and one after another vertically on small screens.




col



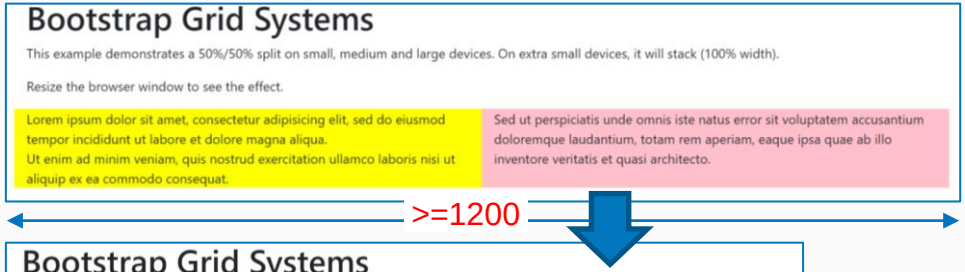

col-lg

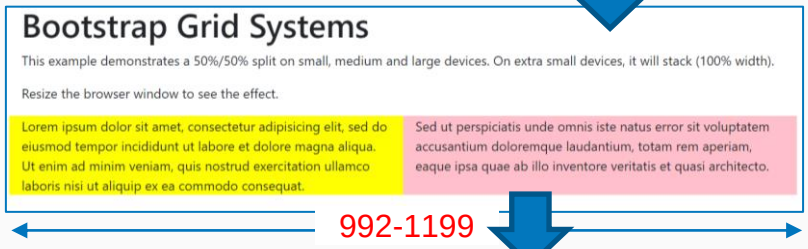
```
<div class="container">
  <div class="row">
    <div class="col-lg-6">
      <div class="panel panel-blue">This is a col-6.</div>
    </div>
    <div class="col-lg-6">
      <div class="panel panel-blue">This is a col-6.</div>
    </div>
...
```

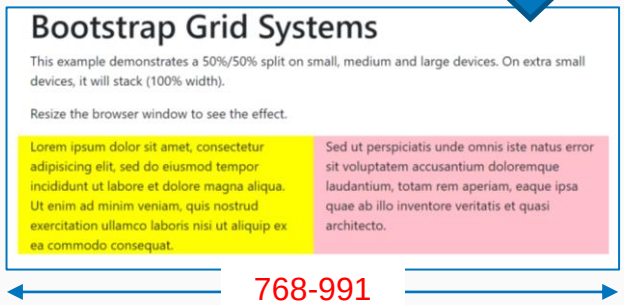
Example04.05

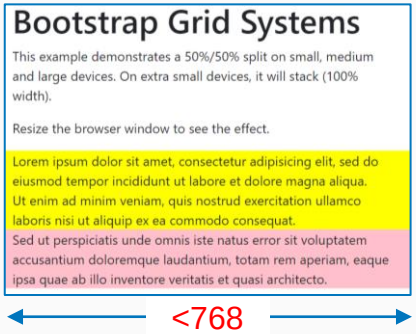
# Breakpoint Example

This example demonstrates a 50%/50% split on small, medium and large devices. On extra small devices, it will stack (100% width).




```
<div class="container">
  <h1>Bootstrap Grid Systems</h1>
  <p>This example demonstrates a 50%/50% split on small, medium and
      large devices. On extra small devices, it will stack
      (100% width).</p>
  <p>Resize the browser window to see the effect.</p>
  <div class="row">
    <div class="col-sm-6" style="background-color:yellow;"> </div>
    <div class="col-sm-6" style="background-color:pink;"></div>
  </div>
</div>
```

- `.col-` (extra small devices - screen width less than 576px)
- `.col-sm-` (small devices - screen width equal to or greater than 576px)
- `.col-md-` (medium devices - screen width equal to or greater than 768px)
- `.col-lg-` (large devices - screen width equal to or greater than 992px)
- `.col-xl-` (xlarge devices - screen width equal to or greater than 1200px)

Example04.06

# Exericse

Create a page that layouts 4 pictures using Bootstrap Grid. The layout requirements are based on the size of display:

1. For notebooks and desktop computers: 1 row x 4 pictures.
2. For tablets: 2 rows x 2 pictures.
3. For mobile phones: 4 rows x 1 picture.

We need to replace the `col` classes in order to achieve the goal.

```
<div class="container-fluid">
  <div class="row">
    <div class="col">
      <img src="http://www.ouhk.edu.hk/images/2014/main/logo_main.svg">
    </div>
    <div class="col">
      <img src="http://www.ouhk.edu.hk/images/2014/main/logo_main.svg">
    </div>
    <div class="col">
      <img src="http://www.ouhk.edu.hk/images/2014/main/logo_main.svg">
    </div>
    <div class="col">
      <img src="http://www.ouhk.edu.hk/images/2014/main/logo_main.svg">
    </div>
  </div>
</div>
```

# Exericse

Create a page that layouts 4 pictures using Bootstrap Grid. The layout requirements are based on the size of display:

1. For notebooks and desktop computers: 1 row x 4 pictures.
2. For tablets: 2 rows x 2 pictures.
3. For mobile phones: 4 rows x 1 picture.

**Which `col` classes should we choose for each column?**

Consideration 1: We need to cater three types of devices

- For notebooks and desktop, we need `col-lg` classes
- For tablets, we need `col-md` classes
- For mobile phones, it will stack to 100% width, so no need to specify any class.

# Exericse

Create a page that layouts 4 pictures using Bootstrap Grid. The layout requirements are based on the size of display:

1. For notebooks and desktop computers: 1 row x 4 pictures.
2. For tablets: 2 rows x 2 pictures.
3. For mobile phones: 4 rows x 1 picture.

**Which `col` classes should we choose for each column?**

Consideration 2: We need to know the column width of each device type.

- In Bootstrap grid system, each row can have 12 columns
- For notebooks and desktop, we need to display 4 pictures in a row
  - Therefore, the width of each column span should be **3**
- For tablets, we need to display 2 pictures in a row
  - Therefore, the width of each column span should be **6**

# Exericse

## Which `col` classes should we choose for each column?

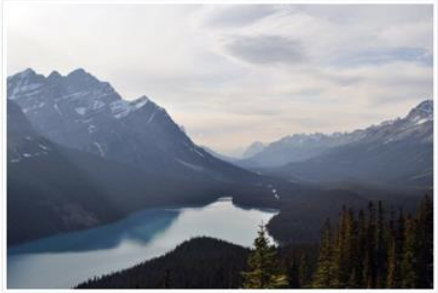
- `col-lg-3` and `col-md-6`

```
<div class="container-fluid">
  <div class="row">
    <div class="col-lg-3 col-md-6">
      <img src="http://www.ouhk.edu.hk/images/2014/main/logo_main.svg">
    </div>
    <div class="col-lg-3 col-md-6">
      <img src="http://www.ouhk.edu.hk/images/2014/main/logo_main.svg">
    </div>
    <div class="col-lg-3 col-md-6">
      <img src="http://www.ouhk.edu.hk/images/2014/main/logo_main.svg">
    </div>
    <div class="col-lg-3 col-md-6">
      <img src="http://www.ouhk.edu.hk/images/2014/main/logo_main.svg">
    </div>
  </div>
</div>
```

Example04.07

# Example: Layout for Tablets (md), Desktops (lg) and Large Desktops (xl)

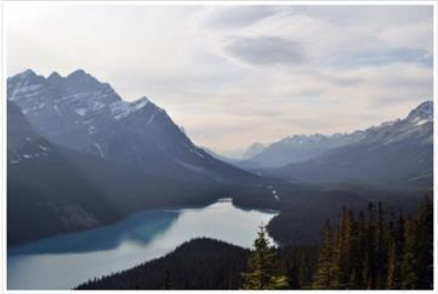




Example04.08

# Example: Hide Preview on **Small** (sm) Devices?






Hide this preview!

# BootStrap 4: Hiding Elements

- To **hide elements** use the `.d-none` class or one of the `.d-{sm,md,lg,xl}-none`classes for any responsive screen variation.
- To **show an element only on a given interval** of screen sizes you can combine one `.d-*-none` class with a `.d-*-*` class, for example `.d-none .d-md-block .d-xl-none` will hide the element for all screen sizes except on medium and large devices.

| Screen Size | Class |
|---|---|
| Hidden on all | `.d-none` |
| Hidden only on xs | `.d-none .d-sm-block` |
| Hidden only on sm | `.d-sm-none .d-md-block` |
| Hidden only on md | `.d-md-none .d-lg-block` |
| Hidden only on lg | `.d-lg-none .d-xl-block` |
| Hidden only on xl | `.d-xl-none` |
| Visible on all | `.d-block` |
| Visible only on xs | `.d-block .d-sm-none` |
| Visible only on sm | `.d-none .d-sm-block .d-md-none` |
| Visible only on md | `.d-none .d-md-block .d-lg-none` |
| Visible only on lg | `.d-none .d-lg-block .d-xl-none` |
| Visible only on xl | `.d-none .d-xl-block` |

Example: Visible only on sm

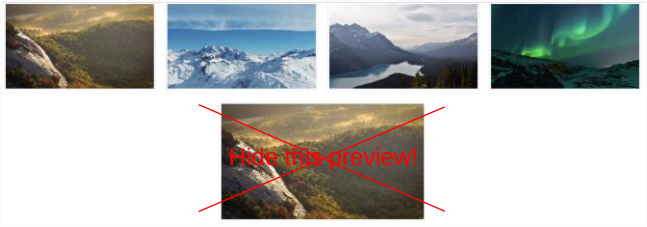
1 `.d-none`
2 `.d-sm-block`
3 `.d-md-none`


https://getbootstrap.com/docs/4.1/utilities/display/

# Example: Hide Preview on **Small** (sm) Devices?

Hide on *all* devices

Show on md (lg and xl) devices




```
<div class="row d-none d-md-block">
      <div class="col-md-12">
            <div class="text-center">
                  <img id="preview" class="img-fluid"
          src="https://www.w3schools.com/howto/img_lights.jpg">
              </div>
      </div>
</div>
```

Example04.08

# More Bootstrap Grid Examples

https://getbootstrap.com/docs/4.1/examples/grid/

# References

https://www.w3schools.com/bootstrap4/default.asp

https://medium.freecodecamp.org/learn-the-bootstrap-4-grid-system-in-10-minutes-e83bfae115da