

## Introduction

Friday, March 17, 2017 3:35 AM

### B What is Bootstrap?

- A framework
- Mobile first
- Responsive design
- Battle tested
- Uses modern web technologies

### B What's a Framework?

- A way to organize your code
- If you follow a certain structure...
- A framework gives you many abilities.
  - *Pre-built CSS styles*
  - *JavaScript functionality*

## **B** When Should I Use Bootstrap?

- Excellent for small, quick projects
- Prototyping and scaffolding sites
- Also great for larger projects
  - *Customize your install*

## Files and Folders

Friday, March 17, 2017 3:49 AM

- Bootstrap folder has 3 folders
  - Css - contain all css files
  - Font - contain fonts file
  - JS - contain javascript file for supporting corosel and other functionalities.

▼	css	
	bootstrap-theme.css	Today, 6:56 PM
	bootstrap-theme.css.map	Jun 16, 2015, 4:12 PM
	bootstrap-theme.min.css	Jun 16, 2015, 4:12 PM
	bootstrap.css	Jun 16, 2015, 4:12 PM
	bootstrap.css.map	Jun 16, 2015, 4:12 PM
▼	fonts	
	glyphicon-halflings-regular.eot	Today, 6:51 PM
	glyphicon-halflings-regular.svg	Jun 16, 2015, 3:56 PM
	glyphicon-halflings-regular.ttf	Jun 16, 2015, 3:56 PM
	glyphicon-halflings-regular.woff	Jun 16, 2015, 3:56 PM
	glyphicon-halflings-regular.woff2	Jun 16, 2015, 3:56 PM
▼	js	
	bootstrap.js	Today, 6:51 PM
	bootstrap.min.js	Jun 16, 2015, 4:12 PM
	npm.js	Jun 16, 2015, 4:12 PM

- Bootstrap(**bootstrap.js** or **bootstrap.min.js**) uses Jquery for working, so we have to download Jquery and put that in same folder as bootstrap.js or bootstrap.min.js

# Containers

Friday, March 17, 2017 3:53 AM

## Understanding the Bootstrap Grid

- Making Responsive Website is little bit challenging due to math involved in calculating width and height of components, according to the size of the window.

### B Grid System

- 12-column grid
- Accessed through CSS classes
- Three key concepts
  - *Containers*
  - *Rows*
  - *Columns*

### B Bootstrap Containers

- Control layout
- Add padding
- Container class automatically add 15px padding to the element, so if we want to make an element full width of browser do not keep it inside **container**.
- Two types
  - *Fluid*
  - *Fixed-Width*

# Fluid-Width

- Fluid-width container, class - **container-fluid**
- These container are always wanted to change themselves according to size, so there are no breakpoints, and as we change size the size of these container changes.
- 15px padding is done to both containers

## B Fixed-Width

- 15 pixel padding on each side
- Adjusts to media query breakpoints

	Extra Small <code>&lt;768px</code>	Small <code>≥768px</code>	Medium <code>≥992px</code>	Large <code>≥1200px</code>
Width	Auto	750px	970px	1170px

- So here we have some breakpoint, according to which size of the element will be and size will not be fluid completely.( so between 992px and 1200px size will remain 970 px, for <768px it will become like FLUID WIDTH )
- Class - **container**

# Rows and Columns

Friday, March 17, 2017 4:20 AM

## B Rows

- Horizontal groups of columns
- Place within container
- Should always include columns
- Gets rid of container padding
- When we use **ROW** inside **CONTAINER** it removes PADDING using NEGATIVE MARGIN(-15px),  
So ROWS can accommodate COLUMNS properly.

## B Columns

- 30px gutters
  - 15px on each side
- GUTTER is space between two columns, by default 30px, we can also change.
- Use col-**SIZE**-SPAN
  - Size: When it stacks
  - SPAN: How many columns
    - SIZE - size type extra-small, small, medium or large according to size
    - SPAN - no. of column it will span out of 12
- >12 columns wrap
  - If more than 12 columns are given a new row will be created by all the remaining after 12.

## B Column Metrics

	Extra small <768px	Small ≥768px	Medium ≥992px	Large ≥1200px
Container	Auto	750px	970px	1170px
Size	.col-xs-	.col-sm-	.col-md-	.col-lg-
Column width	Auto	~62px	~81px	~97px

### RULE OF THUMB -

- If a screen size styling is not given it will go to its **nearest lower styling**, if not given then default(size = 12 cols) will be taken.

### Example 1 - only one is used

When we specify the class `col-xs-12`, it means the element should span all 12 of the available Bootstrap columns on extra small screens. But what about larger displays? In the above code, we haven't specified how the div should behave on larger screen types. Fortunately, Bootstrap will automatically follow the layout specified for the smallest screen size. Hence, our div will span 12 columns in all types of displays in this code.

- Here every style will come to nearest lower so all will come to `.col-xs`

### Example 2 - more than 1 is used

```
<div class="row">
  <div class="col-xs-12 col-sm-6 col1">
```

In this code we have used the class `col-xs-12` for an extra small display and class `col-sm-6` for a smaller sized display. Hence, each column in an extra small-sized display will occupy all the 12 available Bootstrap columns, which will appear as a stack of columns. Yet on a smaller display, they will occupy only six Bootstrap columns each to achieve a two-column layout as shown in Figure 2.4. .

- Here `col-lg` and `col-md` will come to `col-sm` as it is nearest lower.

# Resetting columns and Offsetting columns

Friday, March 17, 2017 4:24 PM

- Resetting, watch video( something related to breaking in layout.

## Offsetting Columns

Offsetting is another great feature of Bootstrap's grid system. It is generally used to increase the left margin of a column. For example, if you have a column that should appear after a gap of three Bootstrap columns, you can use the offsetting feature.

Classes that are available for offsetting are:

- col-xs-offset-\*
- col-sm-offset-\*
- col-md-offset-\*
- col-lg-offset-\*

- We can tell the element to start at some column out of given columns.
- Given by **col-SIZE-SPAN** syntax.

```
<div class="row">
  <section class="col-xs-offset-3 col-xs-6 ">
```

- Here in extra-small screen 6 cols has been given to SECTION, out of which the section will start from 3rd col, (by default it has started with 1st column), so 3 empty ,6 having section, 3 empty.(3 chhadkar 6 diye hai)

NOTE - as we know the undefined go to nearest lower style, so this offset will stick with all other offset , so we have to **reset** for other using **col-sm-offset-0**

```
<section class="col-xs-offset-3 col-xs-6 col-sm-offset-0 ">
```



# Nested Columns and Reordering Columns

Friday, March 17, 2017 4:36 PM

- When we wanted to create pretty complicated design we create row inside a row, which will further devide row in to 12 columns(parts), and we can use it.

## Reordering Grids Manually

We can also reorder the columns irrespective of the order in which they're written in the code. If we have written a `col-md-9` column first and then a `col-md-3` column, we can easily swap their position when the HTML page is rendered by a browser. This is done using Bootstrap's `pull` and `push` classes.

Classes such as `col-xs-pull-*` and `col-xs-push-*` are used to move a column towards the left and right respectively by a certain number of columns. For example:

```
<div class="row">
  <div class="col-xs-9 col-xs-push-3">
    <h1>Pushed Column</h1>
  </div>
  <div class="col-xs-3 col-xs-pull-9">
    <h1>Pulled Column</h1>
  </div>
</div>
```

- It will push the column 3 toward right in extra small screen(which will work for every screen if no other is not given)
  - `col-xs-pull-*` and `col-xs-push-*` for extra smaller screens
  - `col-sm-pull-*` and `col-sm-push-*` for smaller screens
  - `col-md-pull-*` and `col-md-push-*` for medium screens
  - `col-lg-pull-*` and `col-lg-push-*` for larger screens

# Headings and other classes

Friday, March 17, 2017 4:55 PM

## HEADINGS

- We can use bootstrap heading classes, to style like headings.
- We can style anything to have font and style like heading using following classes.

.h1 - .h6

Makes an element look like a heading of the chosen class (h1-h6)

Example -

```
<div class="h1">h1 Bootstrap heading</div>
<h1>h1 Bootstrap heading </h1>
-----
```

h1 Bootstrap heading

h1 Bootstrap heading

## Small Class

- Similarly we also have class for **<small>** tag which makes text smaller with current context.

.small

Creates a lighter, secondary text in any heading

```
<h2>Typography <small>course</small></h2>
<h2>Typography <span class="small">course</span></h2>
```

Typography course

Typography course

## Lead Class

.lead

Increase the font size and line height of a paragraph

- This class can be used to specially signify some text between whole paragraphs.

```
<h2>Typography</h2>
<p>Use the .lead class to make a paragraph "stand out":</p>
<p class="lead">This paragraph stands out.</p>
```

# Typography

Use the `.lead` class to make a paragraph "stand out":

This paragraph stands out.

# Typography style Tags

Friday, March 17, 2017 5:09 PM

- Bootstrap automatically adds some styling to tags which are generally used , we can see them in developer tools.  
Some of these tags are following.

## TAGS

- <s> or <del> - strike through the text
- <ins> or <u> - shows inserted text, underline but not link.
- <strong> or <b>- bold
- <em> or <i> - italic

### <blockquote>

Bootstrap will style the HTML <blockquote> element in the following way:

#### Example

For 50 years, WWF has been protecting the future of nature. The world's leading conservation organization, WWF works in 100 countries and is supported by 1.2 million members in the United States and close to 5 million globally.

— From WWF's website

- The name comes from <footer> tag inside <blockquote> tag.

#### Example

To show the quote on the right, use the .blockquote-reverse class:

```
<blockquote class="blockquote-reverse">
```

For 50 years, WWF has been protecting the future of nature. The world's leading conservation organization, WWF works in 100 countries and is supported by 1.2 million members in the United States and close to 5 million globally.

From WWF's website —

### <code>

Bootstrap will style the HTML <code> element in the following way:

## Example

The following HTML elements: `span`, `section`, and `div` defines a section in a document.

## <kbd>

Bootstrap will style the HTML `<kbd>` element in the following way:

- `<kbd>` keyboard tag can be used to represent key combinations.

## Example

Use `ctrl + p` to open the Print dialog box.

## <pre>

Bootstrap will style the HTML `<pre>` element in the following way:

- This is what used in wordpress.

## Example

Text in a `pre` element  
is displayed in a fixed-width  
font, and it preserves  
both spaces and  
line breaks.

## .pre-scrollable class

- Pre-scrollable class can be used with long scripts, it will add scrollbar if the script become more than 300 px.

```
<h2>Long Script</h2>
<pre class="pre-scrollable">
//replace IMG inside carousels with a background image
$('#featured .item img').each(function() {
  var imgSrc = $(this).attr('src');
  $(this).parent().css({'background-image':
    'url('+imgSrc+')'});
  $(this).remove();
});
```



# Contextual Colors and Backgrounds

Friday, March 17, 2017 9:39 PM

## Contextual Colors and Backgrounds

Bootstrap also has some contextual classes that can be used to provide "meaning through colors".

The classes for text colors are: `.text-muted`, `.text-primary`, `.text-success`, `.text-info`, `.text-warning`, and `.text-danger`:

### Example

```
This text is muted.  
This text is important.  
This text indicates success.  
This text represents some information.  
This text represents a warning.  
This text represents danger.
```

The classes for background colors are: `.bg-primary`, `.bg-success`, `bg-info`, `bg-warning`, and `.bg-danger`:

### Example

```
This text is important.  
This text indicates success.  
This text represents some information.  
This text represents a warning.  
This text represents danger.
```

# Alignment, Transformation and Abbreviation

Friday, March 17, 2017 5:23 PM

## Text Alignment

- They are same as `text-align` property

<code>.text-left</code>	Indicates left-aligned text
-------------------------	-----------------------------

<code>.text-center</code>	Indicates center-aligned text
---------------------------	-------------------------------

<code>.text-right</code>	Indicates right-aligned text
--------------------------	------------------------------

<code>.text-justify</code>	Indicates justified text
----------------------------	--------------------------

- `.text nowrap` can be used to tell browser not to wrap text and make a single line(ofcourse it makes screen size bigger)

<code>.text nowrap</code>	Indicates no wrap text
---------------------------	------------------------

## Text Transformation

- Done work same as `text-transform` property

<code>.text-lowercase</code>	Indicates lowercased text
------------------------------	---------------------------

<code>.text-uppercase</code>	Indicates uppercased text
------------------------------	---------------------------

<code>.text-capitalize</code>	Indicates capitalized text
-------------------------------	----------------------------

# Typography class Reference

Friday, March 17, 2017 9:42 PM

## Typography Classes

Class	Description
.lead	Makes a paragraph stand out
.small	Indicates smaller text (set to 85% of the size of the parent)
.text-left	Indicates left-aligned text
.text-center	Indicates center-aligned text
.text-right	Indicates right-aligned text
.text-justify	Indicates justified text
.text nowrap	Indicates no wrap text
.text-lowercase	Indicates lowercased text
.text-uppercase	Indicates uppercased text
.text-capitalize	Indicates capitalized text
.initialism	Displays the text inside an <code>&lt;abbr&gt;</code> element in a slightly smaller font size
.list-unstyled	Removes the default list-style and left margin on list items (works on both <code>&lt;ul&gt;</code> and <code>&lt;ol&gt;</code> ). This class only applies to immediate children list items (to remove the default list-style from any nested lists, apply this class to any nested lists as well)
.list-inline	Places all list items on a single line
.dl-horizontal	Lines up the terms ( <code>&lt;dt&gt;</code> ) and descriptions ( <code>&lt;dd&gt;</code> ) in <code>&lt;dl&gt;</code> elements side-by-side. Starts off like default <code>&lt;dl&gt;</code> s, but when the browser window expands, it will line up side-by-side
.pre-scrollable	Makes a <code>&lt;pre&gt;</code> element scrollable



# Styling Lists

Saturday, March 18, 2017 12:27 AM

## Working with List Styles

### .list-unstyled

#### .list-unstyled

Removes the default list-style and left margin on list items (works on both `<ul>` and `<ol>`). This class only applies to immediate children list items (to remove the default list-style from any nested lists, apply this class to any nested lists as well)

- .list-unstyled removes all styles from list(bullets, number and padding) and makes it simple bunch of elements.
- It only affect current level and not recursively to all internal lists(if any)

```
<ol class="list-unstyled">
  <li>Lorem ipsum</li>
  <li>Velit
    <ul>
```

Before adding .list-unstyled

### List

1. Lorem ipsum
2. Velit
  - Eligendi modi soluta
  - Architecto tempora
3. Obcaecati molestiae

After adding .list-unstyled

### List

- Lore ipsum  
Velit
  - Eligendi modi soluta
  - Architecto temporaObcaecati molestiae

### .list-inline

#### .list-inline

Places all list items on a single line

- .list-inline removes all styling and puts all element side by side horizontally.

Before adding .list-inline

## Another List

- quam
- porro
- quasi

After adding .list-unstyled

## Another List

quam porro quasi

## .dl-horizontal

### .dl-horizontal

Lines up the terms (`<dt>`) and descriptions (`<dd>`) in `<dl>` elements side-by-side. Starts off like default `<dl>`s, but when the browser window expands, it will line up side-by-side

- 
- .dl-horizontal lines title and description side by side after some breakpoint

Before adding .dl-horizontal

## Definitions

### **Lorem**

Ipsum dolor sit amet, consectetur adipisicing elit. Esse quo ducimus dolorum dolore ipsam unde, facere explicabo quae natus alias deserunt quibusdam voluptatem, officiis itaque rerum magni eius odio mollitia!

### **Sed**

Doloremque quam porro ab ratione facilis, magni quo, velit impedit facere, error, rem obcaecati asperiores veniam vitae. Culpa vel dolorem fugiat odit numquam voluptates ut, quasi minima est unde? Lorem\*5

### **Quia**

Corporis ut, quisquam sed optio debitis eveniet ad provident soluta, aspernatur ducimus placeat illum. Similique quas odio repudiandae perferendis facilis quos, consectetur suscipit, ad velit error, enim, minus quidem!

After adding .dl-horizontal

## Definitions

<b>  Lorem</b>	Ipsum dolor sit amet, consectetur adipisicing elit. Esse quo ducimus dolorum dolore ipsam unde, facere explicabo quae natus alias deserunt quibusdam voluptatem, officiis itaque rerum magni eius odio mollitia!
<b>  Sed</b>	Doloremque quam porro ab ratione facilis, magni quo, velit impedit facere, error, rem obcaecati asperiores veniam vitae. Culpa vel dolorem fugiat odit numquam voluptates ut, quasi minima est unde? Lorem*5
<b>  Quia</b>	Corporis ut, quisquam sed optio debitis eveniet ad provident soluta, aspernatur ducimus placeat illum. Similique quas odio repudiandae perferendis facilis quos, consectetur suscipit, ad velit error, enim, minus quidem!



# Buttons

Saturday, March 18, 2017 12:50 AM

## Bootstrap Buttons

- Three kind of things can be styled like buttons
  - <a> - links
  - <button> - buttons
  - <input type='submit'> - input fields.

The button classes can be used on an `<a>`, `<button>`, or `<input>` element:

## Button Styles

Bootstrap provides seven styles of buttons:



To achieve the button styles above, Bootstrap has the following classes:

- `.btn-default`
- `.btn-primary`
- `.btn-success`
- `.btn-info`
- `.btn-warning`
- `.btn-danger`
- `.btn-link`

### Example

```
<button type="button" class="btn btn-default">Default</button>
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-success">Success</button>
```

- 'btn btn-success', dono part dena jaroori first give button style and second give color.

## Button Styles



# Button Sizes

Bootstrap provides four button sizes:

Large    Medium    Small    XSmall

The classes that define the different sizes are:

- `.btn-lg`
- `.btn-md`
- `.btn-sm`
- `.btn-xs`

## Example

```
<button type="button" class="btn btn-primary btn-lg">Large</button>
<button type="button" class="btn btn-primary btn-md">Medium</button>
<button type="button" class="btn btn-primary btn-sm">Small</button>
<button type="button" class="btn btn-primary btn-xs">XSmall</button>
```

## Button Sizes

Large    Medium    Small    XSmall

## Block Level Buttons

A block level button spans the entire width of the parent element.

Button 1

Button 2

Add class `.btn-block` to create a block level button:

## Example

```
<button type="button" class="btn btn-primary btn-block">Button
1</button>
```

# Active/Disabled Buttons

A button can be set to an active (appear pressed) or a disabled (unclickable) state:

Active Primary

Disabled Primary

The class `.active` makes a button appear pressed, and the class `.disabled` makes a button unclickable:

## Example

```
<button type="button" class="btn btn-primary active">Active  
Primary</button>  
<button type="button" class="btn btn-primary disabled">Disabled  
Primary</button>
```

# Button Groups

Saturday, March 18, 2017 7:18 PM

## Button Groups

Bootstrap allows you to group a series of buttons together (on a single line) in a button group:

```
Apple Samsung Sony
```

Use a `<div>` element with class `.btn-group` to create a button group:

### Example

```
<div class="btn-group">
  <button type="button" class="btn btn-primary">Apple</button>
  <button type="button" class="btn btn-primary">Samsung</button>
  <button type="button" class="btn btn-primary">Sony</button>
</div>
```

**Tip:** Instead of applying button sizes to every button in a group, use class `.btn-group-lg|sm|xs` to size all buttons in the group:

### Example

```
<div class="btn-group btn-group-lg">
  <button type="button" class="btn btn-primary">Apple</button>
  <button type="button" class="btn btn-primary">Samsung</button>
  <button type="button" class="btn btn-primary">Sony</button>
</div>
```

Large Buttons:

```
Apple Samsung Sony
```

# Vertical Button Groups

Bootstrap also supports vertical button groups:



Use the class `.btn-group-vertical` to create a vertical button group:

## Example

```
<div class="btn-group-vertical">
  <button type="button" class="btn btn-primary">Apple</button>
  <button type="button" class="btn btn-primary">Samsung</button>
  <button type="button" class="btn btn-primary">Sony</button>
</div>
```

# Justified Button Groups

To span the entire width of the screen, use the `.btn-group-justified` class:

Apple      Samsung      Sony

Example with `<a>` elements:

## Example

```
<div class="btn-group btn-group-justified">
  <a href="#" class="btn btn-primary">Apple</a>
  <a href="#" class="btn btn-primary">Samsung</a>
  <a href="#" class="btn btn-primary">Sony</a>
</div>
```

- Justified means adjust so that they occupy full line.
- Here, no need to use 'btn-group', as 'btn-group-justified' is enough.

**Note:** For `<button>` elements, you must wrap each button in a `.btn-group` class:

## Example

```
<div class="btn-group btn-group-justified">
  <div class="btn-group">
    <button type="button" class="btn btn-primary">Apple</button>
  </div>
  <div class="btn-group">
    <button type="button" class="btn btn-primary">Samsung</button>
  </div>
  <div class="btn-group">
    <button type="button" class="btn btn-primary">Sony</button>
  </div>
</div>
```

- Alag se .button-group me dalna padega.

# ListGroups

Saturday, March 18, 2017 11:07 PM

## Basic List Groups

The most basic list group is an unordered list with list items:

First item

Second item

Third item

To create a basic list group, use an `<ul>` element with class `.list-group`, and `<li>` elements with class `.list-group-item`:

- For creating a basic List Group just add these two classes.

### Example

```
<ul class="list-group">
  <li class="list-group-item">First item</li>
  <li class="list-group-item">Second item</li>
  <li class="list-group-item">Third item</li>
</ul>
```

## List Group With Badges

You can also add badges to a list group. The badges will automatically be positioned on the right:

New	12
Deleted	5
Warnings	3

To create a badge, create a `<span>` element with class `.badge` inside the list item:

## Example

```
<ul class="list-group">
  <li class="list-group-item">New <span class="badge">12</span></li>
  <li class="list-group-item">Deleted <span class="badge">5</span>
  </li>
  <li class="list-group-item">Warnings <span class="badge">3</span>
  </li>
</ul>
```

## List Group With Linked Items

The items in a list group can also be hyperlinks. This will add a grey background color on hover:

First item
Second item
Third item

To create a list group with linked items, use `<div>` instead of `<ul>` and `<a>` instead of `<li>`:

## Example

```
<div class="list-group">
  <a href="#" class="list-group-item">First item</a>
  <a href="#" class="list-group-item">Second item</a>
  <a href="#" class="list-group-item">Third item</a>
</div>
```

## Active State

First item
Second item
Third item

Use the `.active` class to highlight the current item:

### Example

```
<div class="list-group">
  <a href="#" class="list-group-item active">First item</a>
  <a href="#" class="list-group-item">Second item</a>
  <a href="#" class="list-group-item">Third item</a>
</div>
```

## Disabled Item

The following list group has a disabled item:

First item
Second item
Third item

To disable an item, add the `.disabled` class:

### Example

```
<div class="list-group">
  <a href="#" class="list-group-item disabled">First item</a>
  <a href="#" class="list-group-item">Second item</a>
  <a href="#" class="list-group-item">Third item</a>
</div>
```

- We always use '`.disabled`' class for disabling an item.

## Contextual Classes

Contextual classes can be used to color list items:

First item
Second item

# Contextual Classes

Contextual classes can be used to color list items:

First item
Second item
Third item
Fourth item

The classes for coloring list-items are: `.list-group-item-success`, `list-group-item-info`, `list-group-item-warning`, and `.list-group-item-danger`:

## Example

```
<ul class="list-group">
  <li class="list-group-item list-group-item-success">First item</li>
  <li class="list-group-item list-group-item-info">Second item</li>
  <li class="list-group-item list-group-item-warning">Third item</li>
  <li class="list-group-item list-group-item-danger">Fourth item</li>
</ul>
```

# Custom Content

You can add nearly any HTML inside a list group item.

Bootstrap provides the classes `.list-group-item-heading` and `.list-group-item-text` which can be used as follows:

## Example

```
<div class="list-group">
  <a href="#" class="list-group-item active">
    <h4 class="list-group-item-heading">First List Group Item Heading</h4>
    <p class="list-group-item-text">List Group Item Text</p>
  </a>
  <a href="#" class="list-group-item">
    <h4 class="list-group-item-heading">Second List Group Item Heading</h4>
    <p class="list-group-item-text">List Group Item Text</p>
  </a>
  <a href="#" class="list-group-item">
    <h4 class="list-group-item-heading">Third List Group Item Heading</h4>
    <p class="list-group-item-text">List Group Item Text</p>
  </a>
</div>
```

# List Group With Custom Content

First List Group Item Heading

List Group Item Text

Second List Group Item Heading

List Group Item Text

Third List Group Item Heading

List Group Item Text

# Panels

Sunday, March 19, 2017 8:16 PM

## Panels

A panel in bootstrap is a bordered box with some padding around its content:

A Basic Panel

Panels are created with the `.panel` class, and content inside the panel has a `.panel-body` class:

### Example

```
<div class="panel panel-default">
  <div class="panel-body">A Basic Panel</div>
</div>
```

- As there are contextual type of panels , which decides color of panels(like other components).

## Panel Heading

Panel Heading

Panel Content

The `.panel-heading` class adds a heading to the panel:

### Example

```
<div class="panel panel-default">
  <div class="panel-heading">Panel Heading</div>
  <div class="panel-body">Panel Content</div>
</div>
```

- A `.panel-title` class will make `.panel-heading` more beautiful.

```
<div class="panel panel-default">  
  <div class="panel-heading">  
    <h3 class="panel-title">Dr. Winthrop</h3>  
  </div>  
  
  <div class="panel-body">  
    <p>Dr. Winthrop is the guardian of Missy, a three-year old Llaso  
    mix, who he adopted at the shelter. Dr. Winthrop is passionate about  
    spay and neuter and pet adoption, and works tireless hours outside  
    the clinic, performing free spay and neuter surgeries for the  
    shelter.</p>  
  </div>
```

### Dr. Winthrop

Dr. Winthrop is the guardian of Missy, a three-year old Llaso mix, who he adopted at the shelter. Dr. Winthrop is passionate about spay and neuter and pet adoption, and works tireless hours outside the clinic, performing free spay and neuter surgeries for the shelter.

## Panel Footer

Panel Content

Panel Footer

The `.panel-footer` class adds a footer to the panel:

### Example

```
<div class="panel panel-default">  
  <div class="panel-body">Panel Content</div>  
  <div class="panel-footer">Panel Footer</div>  
</div>
```

- So we can add both HEADER and FOOTER to the panel and place content in to PANEL-BODY

# Panel Group

To group many panels together, wrap a `<div>` with class `.panel-group` around them.

The `.panel-group` class clears the bottom-margin of each panel:

## Example

```
<div class="panel-group">
  <div class="panel panel-default">
    <div class="panel-body">Panel Content</div>
  </div>
  <div class="panel panel-default">
    <div class="panel-body">Panel Content</div>
  </div>
</div>
```

- In panel-group margin between panels decreases than it will be with out `.panel-group`.

# Panel Group

The panel-group class clears the bottom-margin. Try to remove the class and see what happens.

Panel Header

Panel Content

Panel Header

Panel Content

Panel Header

Panel Content

# Panels with Contextual Classes

To color the panel, use contextual classes (`.panel-default`, `.panel-primary`, `.panel-success`, `.panel-info`, `.panel-warning`, or `.panel-danger`):

# Panels with Contextual Classes

To color the panel, use contextual classes (`.panel-default`, `.panel-primary`, `.panel-success`, `.panel-info`, `.panel-warning`, or `.panel-danger`):

## Example

Panel with panel-default class

Panel Content

Panel with panel-primary class

Panel Content

Panel with panel-success class

Panel Content

Panel with panel-info class

Panel Content

Panel with panel-warning class

Panel Content

Panel with panel-danger class

Panel Content

# Tables

Saturday, March 18, 2017 1:05 AM

## Bootstrap Basic Table

A basic Bootstrap table has a light padding and only horizontal dividers.

The `.table` class adds basic styling to a table:

### Example

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

- We just need to add `.table` class to `<table>` tag and it will do a lot of styling.

```
<table class="table">
```

```
  <tr>
```

## Striped Rows

The `.table-striped` class adds zebra-stripes to a table:

### Example

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

- Alternate tables will have different stylings.

## Bordered Table

The `.table-bordered` class adds borders on all sides of the table and cells:

### Example

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

## Hover Rows

The `.table-hover` class adds a hover effect (grey background color) on table rows:

### Example

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

## Condensed Table

The `.table-condensed` class makes a table more compact by cutting cell padding in half:

### Example

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

```
<table class="table table-hover table-striped table-condensed">
```

## Contextual Classes

Contextual classes can be used to color table rows (`<tr>`) or table cells (`<td>`):

### Example

Firstname	Lastname	Email
Default	Defaultson	def@somemail.com
Success	Doe	john@example.com
Danger	Moe	mary@example.com
Info	Dooley	july@example.com
Warning	Refs	bo@example.com
Active	Activeson	act@example.com

The contextual classes that can be used are:

Class	Description
.active	Applies the hover color to the table row or table cell
.success	Indicates a successful or positive action
.info	Indicates a neutral informative change or action
.warning	Indicates a warning that might need attention
.danger	Indicates a dangerous or potentially negative action

NOTE - contextual classes can be applied to `<tr>` and `<td>` both but if both used then `<tr>`'s affect will sustain.

```
<tr class="danger">
  <td class="success">Default</td>
  <td>Defaultson</td>
  <td>def@somemail.com</td>
</tr>
```

Default

Defaultson

def@someemail.com

- Here we have used '.success' for first column but due to its parent <tr>'s '.danger' it becomes red.

## Responsive Tables

The `.table-responsive` class creates a responsive table. The table will then scroll horizontally on small devices (under 768px). When viewing on anything larger than 768px wide, there is no difference:

### Example

```
<div class="table-responsive">
  <table class="table">
    ...
  </table>
</div>
```

- A scroll bar will appear, so it is always good to make **table-repsonsive**.

## Table

The `.table-responsive` class creates a responsive table which will scroll horizontally on small devices (under 768px). When viewing on anything larger than 768px wide, there is no difference:

#	Firstname	Lastname	Age	City	Country
1	Anna	Pitt	35	New York	USA

# Image Styling

Saturday, March 18, 2017 1:57 AM

## Bootstrap Image Shapes

Rounded Corners:



Circle:



Thumbnail:



### Rounded Corners

The `.img-rounded` class adds rounded corners to an image (IE8 does not support rounded corners):

#### Example

```

```

### Circle

The `.img-circle` class shapes the image to a circle (IE8 does not support rounded corners):

#### Example

```

```

### Thumbnail

The `.img-thumbnail` class shapes the image to a thumbnail:

#### Example

```

```

# Responsive Images

Images come in all sizes. So do screens. Responsive images automatically adjust to fit the size of the screen.

Create responsive images by adding an `.img-responsive` class to the `<img>` tag. The image will then scale nicely to the parent element.

The `.img-responsive` class applies `display: block;` and `max-width: 100%;` and `height: auto;` to the image:

## Example

```

```

- If `.img-responsive` is applied to an image then it will automatically resize to all sizes according to the screen size(cols given).

# Centering Image

- We can use `center-block` to center any element inside a layout.

<code>.center-block</code>	Centers any element (Sets an element to <code>display:block</code> with <code>margin-right:auto</code> and <code>margin-left:auto</code> )
----------------------------	---

# Image Photogallery

Sunday, March 19, 2017 2:47 AM

## Image Gallery

You can also use Bootstrap's grid system in conjunction with the `.thumbnail` class to create an image gallery.



`Lorem ipsum donec id  
elit non mi porta  
gravida at eget metus.`



`Lorem ipsum donec id  
elit non mi porta  
gravida at eget metus.`



`Lorem ipsum donec id  
elit non mi porta  
gravida at eget metus.`

- Just add `.thumbnail` on `<div>` containing the image
- Add caption if wanted to.

Example - for first image.

```
<div class="row">
  <div class="col-md-4">
    <div class="thumbnail">
      <a href="/w3images/lights.jpg">
        
        <div class="caption">
          <p>Lorem ipsum...</p>
        </div>
      </a>
    </div>
  </div>
```

Similary....

- As we can see image has been given `.col-md-4`, and `.thumbnail` class wrapped around it.
- Caption will be given in `.caption` class.

# Helper Classes

Saturday, March 18, 2017 2:19 AM

For more: [www.w3schools.com/bootstrap/bootstrap\\_ref\\_css\\_helpers.html](http://www.w3schools.com/bootstrap/bootstrap_ref_css_helpers.html)

- We have already learnt some helper classes.

## Text

Add meaning through text-colors with the classes below. Links will darken on hover:

Class	Description
.text-muted	Text styled with class "text-muted"
.text-primary	Text styled with class "text-primary"
.text-success	Text styled with class "text-success"
.text-info	Text styled with class "text-info"
.text-warning	Text styled with class "text-warning"
.text-danger	Text styled with class "text-danger"

## Background

Add meaning through background-colors with the classes below. Links will darken on hover just like text classes:

Class	Description	Example
.bg-primary	Table cell is styled with class "bg-primary"	<a href="#">Try it</a>
.bg-success	Table cell is styled with class "bg-success"	<a href="#">Try it</a>
.bg-info	Table cell is styled with class "bg-info"	<a href="#">Try it</a>
.bg-warning	Table cell is styled with class "bg-warning"	<a href="#">Try it</a>
.bg-danger	Table cell is styled with class "bg-danger"	<a href="#">Try it</a>
.hidden	Forces an element to be hidden (display:none)	
.invisible	Forces an element to be invisible (visibility:hidden). Will take up space on page even though it is invisible	

- Invisible is not visible but its place left empty, but in case of hidden ,space is occupied by others.

Class	Description
.pull-left	Floats an element to the left
.pull-right	Floats an element to the right
.center-block	Sets an element to display:block with margin-right:auto and margin-left:auto

FOR RESPONSIVE UTILITIES(to control visibility at different breakpoints see video)

# Badges and Labels

Sunday, March 19, 2017 12:37 AM

## Badges

Badges are numerical indicators of how many items are associated with a link:

News 5  
Comments 10  
Updates 2

The numbers (5, 10, and 2) are the badges.

Use the `.badge` class within `<span>` elements to create badges:

### Example

```
<a href="#">News <span class="badge">5</span></a><br>
<a href="#">Comments <span class="badge">10</span></a><br>
<a href="#">Updates <span class="badge">2</span></a>
```

Badges can also be used inside other elements, such as buttons:

Primary 7

The following example shows how to add badges to buttons:

### Example

```
<button type="button" class="btn btn-primary">Primary <span
class="badge">7</span></button>
```

# Labels

Labels are used to provide additional information about something:

Example New

Example New

Example New

Example New

Example New

Example New

## Example

```
<h1>Example <span class="label label-default">New</span></h1>
<h2>Example <span class="label label-default">New</span></h2>
<h3>Example <span class="label label-default">New</span></h3>
<h4>Example <span class="label label-default">New</span></h4>
<h5>Example <span class="label label-default">New</span></h5>
<h6>Example <span class="label label-default">New</span></h6>
```

Use the `.label` class, followed by one of the six contextual classes `.label-default`, `.label-primary`, `.label-success`, `.label-info`, `.label-warning` or `.label-danger`, within a `<span>` element to create a label:

- Note here that size of `label` is same as the size of element in which it present.

The following example shows all contextual label classes:

Default Label Primary Label Success Label Info Label Warning Label Danger Label

## Example

```
<span class="label label-default">Default Label</span>
<span class="label label-primary">Primary Label</span>
<span class="label label-success">Success Label</span>
<span class="label label-info">Info Label</span>
<span class="label label-warning">Warning Label</span>
<span class="label label-danger">Danger Label</span>
```

# Input Sizing

Saturday, March 18, 2017 6:00 PM

## Input Sizing in Forms

Set the heights of input elements using classes like `.input-lg` and `.input-sm`.

Set the widths of elements using grid column classes like `.col-lg-*` and `.col-sm-*`.

## Height Sizing

### Small input

---

---

### Default input

---

---

### Large input

---

---

## Example

```
<form>
  <div class="form-group">
    <label for="inputdefault">Default input</label>
    <input class="form-control" id="inputdefault" type="text">
  </div>
  <div class="form-group">
    <label for="inputlg">input-lg</label>
    <input class="form-control input-lg" id="inputlg" type="text">
  </div>
  <div class="form-group">
    <label for="inputsml">input-sm</label>
    <input class="form-control input-sm" id="inputsml" type="text">
  </div>
</form>
```

You can quickly size labels and form controls within a **Horizontal form** by adding `.form-group-*` to the `<div class="form-group">` element:

## Example

```
<div class="form-group form-group-lg">
```

- We can make every input field larger or smaller by adding `.form-group-*` on a `.form-group`, similarly for `input-group`, we can add `input-group-*`.

You can also quickly size all inputs and other elements inside an `.input-group` with the `.input-group-sm` or `.input-group-lg` classes:

## Example

```
<div class="input-group input-group-lg">
```

# Wells

Sunday, March 19, 2017 8:27 PM

- There are super simple variation of panels.
- It do not have any heading or footer etc,it just contain only content.

## Wells

The `.well` class adds a rounded border around an element with a gray background color and some padding:

Basic Well

### Example

```
<div class="well">Basic Well</div>
```

## Well Size

Small Well

Normal Well

Large Well

Change the size of the well by adding the `.well-sm` class for small wells or `.well-lg` class for large wells:

### Example

```
<div class="well well-sm">Small Well</div>
<div class="well well-lg">Large Well</div>
```

By default, wells are medium in size.

# Bootstrap Forms

Saturday, March 18, 2017 12:19 PM

## Bootstrap Form Layouts

Bootstrap provides three types of form layouts:

- Vertical form (this is default)
- Horizontal form
- Inline form

Standard rules for all three form layouts:

- Wrap labels and form controls in `<div class="form-group">` (needed for optimum spacing)
- Add class `.form-control` to all textual `<input>`, `<textarea>`, and `<select>` elements
- `.form-control` gives 100% width so it will occupy full screen.
- Har group(label+input) ko `.form-group` me dalo taki optimum spacing achieve kar sako(2 groups ke beech me).

## Bootstrap Vertical Form (default)

**Email:**

Enter email

**Password:**

Enter password

Remember me

The following example creates a vertical form with two input fields, one checkbox, and a submit button:

### Example

```
<form>
  <div class="form-group">
    <label for="email">Email address:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="checkbox">
    <label><input type="checkbox"> Remember me</label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

# Bootstrap Inline Form

Email:  Password:   Remember me

In an inline form, all of the elements are inline, left-aligned, and the labels are alongside.

**Note: This only applies to forms within viewports that are at least 768px wide!**

- So if the viewport <768px then **inline-form** become **vertical form**.

Additional rule for an inline form:

- Add class **.form-inline** to the **<form>** element
- Form me add kar do **.form-inline** class, (by default vertical hai)

## Example

```
<form class="form-inline">
  <div class="form-group">
    <label for="email">Email address:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="checkbox">
    <label><input type="checkbox"> Remember me</label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

## Hiding Labels

**Tip:** If you don't include a label for every input, screen readers will have trouble with your forms. You can hide the labels for all devices, except screen readers, by using the **.sr-only** class:

## Example

```
<form class="form-inline">
  <div class="form-group">
    <label class="sr-only" for="email">Email address:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label class="sr-only" for="pwd">Password:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="checkbox">
    <label><input type="checkbox"> Remember me</label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

## Inline form with .sr-only class

Make the viewport larger than 768px wide to see that all of the form elements are inline, left aligned, and the labels are alongside.

The screenshot shows a horizontal form with the following structure:

- An input field labeled "Enter email".
- An input field labeled "Enter password".
- A checkbox labeled "Remember me".
- A button labeled "Submit".

- Here label will be hidden for us, but can be read by screen readers.

## Bootstrap Horizontal Form

The screenshot shows a horizontal form with the following structure:

- A label "Email:" followed by an input field labeled "Enter email".
- A label "Password:" followed by an input field labeled "Enter password".
- A checkbox labeled "Remember me".
- A button labeled "Submit".

A horizontal form stands apart from the other forms both in the amount of markup, and in the presentation of the form.

Additional rules for a horizontal form:

- Add class `.form-horizontal` to the `<form>` element
- Add class `.control-label` to all `<label>` elements

**Tip:** Use Bootstrap's predefined grid classes to align labels and groups of form controls in a horizontal layout.

- Make form `.form-horizontal`, also make `<label>.control-label`, also we have to align the cols using grid of Bootstrap.(kisko kitne col milenge)

## Example

```
<form class="form-horizontal">
  <div class="form-group">
    <label class="control-label col-sm-2" for="email">Email:</label>
    <div class="col-sm-10">
      <input type="email" class="form-control" id="email" placeholder="Enter email">
    </div>
  </div>
  <div class="form-group">
    <label class="control-label col-sm-2" for="pwd">Password:</label>
    <div class="col-sm-10">
      <input type="password" class="form-control" id="pwd" placeholder="Enter password">
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
      <div class="checkbox">
        <label><input type="checkbox"> Remember me</label>
      </div>
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
      <button type="submit" class="btn btn-default">Submit</button>
    </div>
  </div>
</form>
```

## Horizontal form

Email:

Password:

Remember me

# BootStrap Inputs

Saturday, March 18, 2017 1:55 PM

## Supported Form Controls

Bootstrap supports the following form controls:

- input
- textarea
- checkbox
- radio
- select

## Bootstrap Input

Bootstrap supports all the HTML5 input types: text, password, datetime, datetime-local, date, month, time, week, number, email, url, search, tel, and color.

**Note:** Inputs will NOT be fully styled if their type is not properly declared!

**Name:**

**Password:**

The following example contains two input elements; one of type text and one of type password:

### Example

```
<div class="form-group">
  <label for="usr">Name:</label>
  <input type="text" class="form-control" id="usr">
</div>
<div class="form-group">
  <label for="pwd">Password:</label>
  <input type="password" class="form-control" id="pwd">
</div>
```

# Bootstrap Textarea

Comment:

The following example contains a textarea:

## Example

```
<div class="form-group">
  <label for="comment">Comment:</label>
  <textarea class="form-control" rows="5" id="comment"></textarea>
</div>
```

# Bootstrap Checkboxes

- Checkboxes and radionbuttons are different and can't be styled using `.form-control`, rather we have to use classes for them.
- Wrap them inside their class rather than `.form-group`.

- Option 1
- Option 2
- Option 3

Checkboxes are used if you want the user to select any number of options from a list of preset options.

The following example contains three checkboxes. The last option is disabled:

## Example

```
<div class="checkbox">
  <label><input type="checkbox" value="">Option 1</label>
</div>
<div class="checkbox">
  <label><input type="checkbox" value="">Option 2</label>
</div>
<div class="checkbox disabled">
  <label><input type="checkbox" value="" disabled>Option 3</label>
</div>
```

- Here we have wrapped the options inside '`.checkbox`' class.

Use the `.checkbox-inline` class if you want the checkboxes to appear on the same line:

- Option 1  Option 2  Option 3

## Example

```
<label class="checkbox-inline"><input type="checkbox" value="">Option 1</label>
<label class="checkbox-inline"><input type="checkbox" value="">Option 2</label>
<label class="checkbox-inline"><input type="checkbox" value="">Option 3</label>
```

- Radio-buttons are exactly same as checkbox.

## Bootstrap Radio Buttons

- Option 1  
 Option 2  
 Option 3

Radio buttons are used if you want to limit the user to just one selection from a list of preset options.

The following example contains three radio buttons. The last option is disabled:

## Example

```
<div class="radio">
  <label><input type="radio" name="optradio">Option 1</label>
</div>
<div class="radio">
  <label><input type="radio" name="optradio">Option 2</label>
</div>
<div class="radio disabled">
  <label><input type="radio" name="optradio" disabled>Option 3</label>
</div>
```

Use the `.radio-inline` class if you want the radio buttons to appear on the same line:

- Option 1
- Option 2
- Option 3

## Example

```
<label class="radio-inline"><input type="radio" name="optradio">Option 1</label>
<label class="radio-inline"><input type="radio" name="optradio">Option 2</label>
<label class="radio-inline"><input type="radio" name="optradio">Option 3</label>
```

## Bootstrap Select List

- Dropdown list can be styled using `.form-control` class like others.

### Select list (select one):

4

### Multiple select list (hold ctrl or shift (or drag with the mouse) to select more than one):

1  
2  
3  
4  
5

Select lists are used if you want to allow the user to pick from multiple options.

The following example contains a dropdown list (select list):

## Example

```
<div class="form-group">
  <label for="sel1">Select list:</label>
  <select class="form-control" id="sel1">
    <option>1</option>
    <option>2</option>
    <option>3</option>
    <option>4</option>
  </select>
</div>
```

## Static Control

If you need to insert **plain text** next to a form label within a horizontal form, use the `.form-control-static` class on a `<p>` element:

```
Email: someone@example.com
```

### Example

```
<form class="form-horizontal">
  <div class="form-group">
    <label class="control-label col-sm-2" for="email">Email:</label>
    <div class="col-sm-10">
      <p class="form-control-static">someone@example.com</p>
    </div>
  </div>
</form>
```

- Can be used if we wanted to insert plain text in place of input field in horizontal form

## Bootstrap Input Groups

The `.input-group` class is a container to enhance an input by adding an icon, text or a button in front or behind it as a "help text".

The `.input-group-addon` class attaches an icon or help text next to the input field.

<input type="text"/>	Email
<input type="text"/>	Password

[Text](#) [Additional Info](#)

- Input-groups allows us to combine input fields together to form good inputs.
- `.input-group` banao (instead of `form-group`), if `span` is used after `input` then we get icon at right side.
- For better spacing between inputs also add `.form-group`
- `.input-group-addon` used to add icon.(here not showing because of internet failure)
- We can use either `<i>` tag or `<span>` tag to add glyphicon icon.

## Example

```
<form>
  <div class="input-group">
    <span class="input-group-addon"><i class="glyphicon glyphicon-user"></i></span>
    <input id="email" type="text" class="form-control" name="email"
placeholder="Email">
  </div>
  <div class="input-group">
    <span class="input-group-addon"><i class="glyphicon glyphicon-lock"></i></span>
    <input id="password" type="password" class="form-control" name="password"
placeholder="Password">
  </div>
  <div class="input-group">
    <span class="input-group-addon">Text</span>
    <input id="msg" type="text" class="form-control" name="msg"
placeholder="Additional Info">
  </div>
</form>
```

- Form moving icons at right hand use `<span>` after `<input>` like following.

`<p>It can also be used on the right side of the input:</p>`

```
<form>
  <div class="input-group">
    <input id="email" type="text" class="form-control" name="email" placeholder="Email">
    <span class="input-group-addon"><i class="glyphicon glyphicon-user"></i></span>
  </div>
  <div class="input-group">
    <input id="password" type="password" class="form-control" name="password"
placeholder="Password">
    <span class="input-group-addon"><i class="glyphicon glyphicon-lock"></i></span>
  </div>
</form>
```

It can also be used on the right side of the input:

Email	<input type="text"/>
Password	<input type="text"/>

The `.input-group-btn` attaches a button next to an input. This is often used together with a search bar:

Search	<input type="button" value=""/>
--------	---------------------------------

- We can click this button to do what we wanted.

## Example

```
<form>
  <div class="input-group">
    <input type="text" class="form-control" placeholder="Search">
    <div class="input-group-btn">
      <button class="btn btn-default" type="submit">
        <i class="glyphicon glyphicon-search"></i>
      </button>
    </div>
  </div>
</form>
```

## Bootstrap Form Control States

- We can control the status of form using these control states.
- **INPUT FOCUS** - The outline of the input is removed and a box-shadow is applied on focus
- **DISABLED INPUTS** - Add a `disabled` attribute to disable an input field
- **DISABLED FIELDSETS** - Add a `disabled` attribute to a fieldset to disable all controls within
- **READONLY INPUTS** - Add a `readonly` attribute to an input to prevent user input
- **VALIDATION STATES** - Bootstrap includes validation styles for error, warning, and success messages. To use, add `.has-warning`, `.has-error`, or `.has-success` to the parent element
- **ICONS** - You can add feedback icons with the `.has-feedback` class and an icon
- **HIDDEN LABELS** - Add a `.sr-only` class on non-visible labels

Focused	<input type="text" value="Click to focus"/>
Disabled	<input disabled="disabled" type="text"/>
Fieldset disabled	<input disabled="disabled" type="text"/>
	<input type="text" value="Disabled select"/>
Input with success and icon	<input checked="checked" type="text"/>
Input with warning and icon	<input type="text"/>
Input with error and icon	<input type="text"/>

- As we can see , for disabling an input we just add attribute **disabled** or we can add **.disabled** class (use full when we want to get disable icon on checkbox text also(do not work on labels) ,bootstrap provide really nice disable icon ).
- For disabling some input fields we can use <fieldset> tag to group them and then add **disabled** inside it.

## Example

```

<form class="form-horizontal">
  <div class="form-group">
    <label class="col-sm-2 control-label">Focused</label>
    <div class="col-sm-10">
      <input class="form-control" id="focusedInput" type="text" value="Click
to focus">
    </div>
  </div>
  <div class="form-group">
    <label for="inputPassword" class="col-sm-2 control-
label">Disabled</label>
    <div class="col-sm-10">
      <input class="form-control" id="disabledInput" type="text" disabled>
    </div>
  </div>
  <fieldset disabled>
    <div class="form-group">
      <label for="disabledTextInput" class="col-sm-2 control-label">Fieldset
disabled</label>
    </div>
  </fieldset>
</form>

```

```

        <div class="col-sm-10">
            <input type="text" id="disabledTextInput" class="form-control">
        </div>
    </div>
    <div class="form-group">
        <label for="disabledSelect" class="col-sm-2 control-label"></label>
        <div class="col-sm-10">
            <select id="disabledSelect" class="form-control">
                <option>Disabled select</option>
            </select>
        </div>
    </div>
</fieldset>
<div class="form-group has-success has-feedback">
    <label class="col-sm-2 control-label" for="inputSuccess">
        Input with success and icon</label>
    <div class="col-sm-10">
        <input type="text" class="form-control" id="inputSuccess">
        <span class="glyphicon glyphicon-ok form-control-feedback"></span>
    </div>
</div>
<div class="form-group has-warning has-feedback">
    <label class="col-sm-2 control-label" for="inputWarning">
        Input with warning and icon</label>
    <div class="col-sm-10">
        <input type="text" class="form-control" id="inputWarning">
        <span class="glyphicon glyphicon-warning-sign form-control-feedback"></span>
    </div>
</div>
<div class="form-group has-error has-feedback">
    <label class="col-sm-2 control-label" for="inputError">
        Input with error and icon</label>
    <div class="col-sm-10">
        <input type="text" class="form-control" id="inputError">
        <span class="glyphicon glyphicon-remove form-control-feedback"></span>
    </div>
</div>
</form>

```

## Adding Color and Feedback to Inputs

- For adding color to our input field(after validating using php/JS) , we can add following classes to control colors
  - .has-warning , .has-error and .has-success give yellow, red and green color .
- For adding feedback in form of icons, we have to add **.has-feedback** class on that form-group and then inside it, we add icon, giving **form-control-feedback** class.

EXAMPLE -

```
<div class="form-group has-success has-feedback">
  <label class="control-label"
    for="inputName">Name</label>
  <input class="form-control" type="text"
    id="inputName" placeholder="Name">
  <span class="glyphicon glyphicon-ok form-control-
  feedback" aria-hidden="true"></span>
```

- Here **.has-success** add green color, while **has-feedback** lets it to give feedback properly using icon **glyphicon-ok**

## Help Text

Use the **.help-block** class to add a block level help text in forms:

### Example

```
<div class="form-group">
  <label for="pwd">Password:</label>
  <input type="password" class="form-control" id="pwd" placeholder="Enter password">
  <span class="help-block">This is some help text...</span>
</div>
```

**Password:**

This is some help text that breaks onto a new line and may extend more than one line.

# Dropdowns

Saturday, March 18, 2017 6:09 PM

## Basic Dropdown

A dropdown menu is a toggleable menu that allows the user to choose one value from a predefined list:

### Dropdown Example ▾

- Dropdown consist of two things
  - Button + caret(for showing that it is dropdown list)
  - List of links in dropdown.
- Making Dropdown
  - Step1 - make div containing dropdown **.dropdown**
  - Step2 - make button **.dropdown-toggle** , so when it clicked list show, and add a custom attribute **data-toggle="dropdown"** (**data/list ko kaise dikhana hai**) used by Bootstrap JS to show list.
  - Step3 - make list(any kind) **.dropdown-menu** , telling that it is menu list of dropdown.

## Example

```
<div class="dropdown">
  <button class="btn btn-primary dropdown-toggle" type="button" data-
  toggle="dropdown">Dropdown Example
  <span class="caret"></span></button>
  <ul class="dropdown-menu">
    <li><a href="#">HTML</a></li>
    <li><a href="#">CSS</a></li>
    <li><a href="#">JavaScript</a></li>
  </ul>
</div>
```

## Example Explained

The **.dropdown** class indicates a dropdown menu.

To open the dropdown menu, use a button or a link with a class of **.dropdown-toggle** and the **data-toggle="dropdown"** attribute.

The **.caret** class creates a caret arrow icon (▼), which indicates that the button is a dropdown.

Add the **.dropdown-menu** class to a **<ul>** element to actually build the dropdown menu.

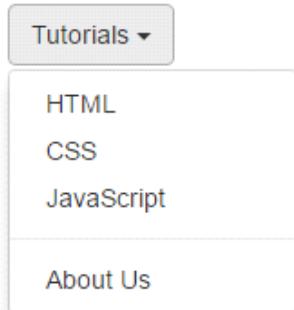
# Dropdown Divider

The `.divider` class is used to separate links inside the dropdown menu with a thin horizontal border:

## Example

```
<li class="divider"></li>
```

- It is just like 'separator' in javafx.



- These are just common 'li' given some special class.

# Dropdown Header

## Dropdown Header Example ▾

The `.dropdown-header` class is used to add headers inside the dropdown menu:

## Example

```
<li class="dropdown-header">Dropdown header 1</li>
```



# Disable and Active items

# Disable and Active items

## Dropdown Disabled Example ▾

Highlight a specific dropdown item with the `.active` class (adds a blue background color).

To disable an item in the dropdown menu, use the `.disabled` class (gets a light-grey text color and a "no-parking-sign" icon on hover):

### Example

```
<li class="disabled"><a href="#">CSS</a></li>
<li class="active"><a href="#">HTML</a></li>
```

# Dropdown Position

## Dropdown Right Example ▾

To right-align the dropdown, add the `.dropdown-menu-right` class to the element `.dropdown-menu`:

- HTML
- CSS
- JavaScript

---

- About Us

### Example

```
<ul class="dropdown-menu dropdown-menu-right">
```

- Add `.dropdown-menu-right` to list.

# Nesting Button Groups & Dropdown Menus

Apple   Samsung   Sony ▾

Nest button groups to create dropdown menus:

- `.button-group` ke andar ek `.button-group` dal do, usko banao dropdown.(in tabs we put dropdown inside `<li>`)

```
<div class="container">
  <h2>Nesting Button Groups</h2>
  <p>Nest button groups to create drop down menus:</p>
  <div class="btn-group">
    <button type="button" class="btn btn-primary">Apple</button>
    <button type="button" class="btn btn-primary">Samsung</button>
    <div class="btn-group dropdown">
      <button type="button" class="btn btn-primary dropdown-toggle" data-
      toggle="dropdown">
        Sony <span class="caret"></span></button>
      <ul class="dropdown-menu" style="list-style-type: none;">
        <li>HTML</li>
        <li>CSS</li>
        <li>JavaScript</li>
        <li>About Us</li>
      </ul>
    </div>
  </div>
</div>
```

```
<div class="container">
  <h2>Nesting Button Groups</h2>
  <p>Nest button groups to create drop down menus:</p>
  <div class="btn-group">
    <button type="button" class="btn btn-primary">Apple</button>
    <button type="button" class="btn btn-primary">Samsung</button>
    <div class="btn-group dropdown">
      <button type="button" class="btn btn-primary dropdown-toggle" data-
toggle="dropdown">
        Sony <span class="caret"></span></button>
        <ul class="dropdown-menu" role="menu">
          <li><a href="#">Tablet</a></li>
          <li><a href="#">Smartphone</a></li>
        </ul>
    </div>
  </div>
</div>
```

# Understanding nav component

Saturday, March 18, 2017 10:48 PM

- BS navigation is very complex and consist of family of component which all share `.nav` component,
- Three different type of nav
  - Tabs
  - Pills
  - Navbar

## Menus

Most web pages have some kind of a menu.

In HTML, a menu is often defined in an unordered list `<ul>` (and styled afterwards), like this:

```
<ul>
  <li><a href="#">Home</a></li>
  <li><a href="#">Menu 1</a></li>
  <li><a href="#">Menu 2</a></li>
  <li><a href="#">Menu 3</a></li>
</ul>
```

- We want to make simple menu with link we can add `.list-inline` class, which is used to show list inline.

If you want to create a horizontal menu of the list above, add the `.list-inline` class to `<ul>`:

```
<ul class="list-inline">
```

### Inline List

Home Menu 1 Menu 2 Menu 3

Or you can display the menu above with Bootstraps' Tabs and Pills (see below).

*On next page.*

# Tabs

Saturday, March 18, 2017 11:15 PM

## Tabs



Tabs are created with `<ul class="nav nav-tabs">`:

**Tip:** Also mark the current page with `<li class="active">`.

The following example creates navigation tabs:

### Example

```
<ul class="nav nav-tabs">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">Menu 1</a></li>
  <li><a href="#">Menu 2</a></li>
  <li><a href="#">Menu 3</a></li>
</ul>
```

- `.nav` ke ange `.nav-tabs` laga do.
- Kisi ko active karne ke liye use `.active`(same as list-groups).

## Tabs With Dropdown Menu



Tabs can also hold dropdown menus.

- Dropdown dal do ek `<li>` me.

## Example

```
<ul class="nav nav-tabs">
  <li class="active"><a href="#">Home</a></li>
  <li class="dropdown">
    <a class="dropdown-toggle" data-toggle="dropdown" href="#">Menu 1
    <span class="caret"></span></a>
    <ul class="dropdown-menu">
      <li><a href="#">Submenu 1-1</a></li>
      <li><a href="#">Submenu 1-2</a></li>
      <li><a href="#">Submenu 1-3</a></li>
    </ul>
  </li>
  <li><a href="#">Menu 2</a></li>
  <li><a href="#">Menu 3</a></li>
</ul>
```

## Centered Tabs and Pills

Home

Menu 1

Menu 2

Menu 3

To center/justify the tabs and pills, use the `.nav-justified` class.

## Example

```
<!-- Centered Tabs -->
<ul class="nav nav-tabs nav-justified">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">Menu 1</a></li>
  <li><a href="#">Menu 2</a></li>
  <li><a href="#">Menu 3</a></li>
</ul>

<!-- Centered Pills -->
<ul class="nav nav-pills nav-justified">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">Menu 1</a></li>
  <li><a href="#">Menu 2</a></li>
  <li><a href="#">Menu 3</a></li>
</ul>
```

## Toggleable / Dynamic Tabs

- Tabs we created so far are not dynamic , so we can't go to other tab by clicking over it.
- Making dynamic tabs and pills are exactly same there is only difference to **data-toggle="tab"** and **data-toggle="pill"**.
- This can be done with some steps.

### Step 1

To make the tabs toggleable, add the **data-toggle="tab"** attribute to each link.

- Waha in dropdown , we do data-toggle='dropdown' to button.

### Step 2

add a **.tab-pane** class with a unique ID for every tab and wrap them inside a **<div>** element with class **.tab-content** .

- Add a new **<div>** having all content for all tabs, so having **.tab-content** class.
- Inside this **<div>** created, add **<div>** for each tab/pill and , also give them an 'id' which referenced by **<li>**(this is same as same page links)

## Example

```
<ul class="nav nav-tabs">
  <li class="active"><a data-toggle="tab" href="#home">Home</a></li>
  <li><a data-toggle="tab" href="#menu1">Menu 1</a></li>
  <li><a data-toggle="tab" href="#menu2">Menu 2</a></li>
</ul>

<div class="tab-content">
  <div id="home" class="tab-pane fade in active">
```

FOR JUMPING TO  
TAB PANE

```
<div class="tab-content">
  <div id="home" class="tab-pane fade in active">
    <h3>HOME</h3>
    <p>Some content.</p>
  </div>
  <div id="menu1" class="tab-pane fade">
    <h3>Menu 1</h3>
    <p>Some content in menu 1.</p>
  </div>
  <div id="menu2" class="tab-pane fade">
    <h3>Menu 2</h3>
    <p>Some content in menu 2.</p>
  </div>
</div>
```

two pane

If you want the tabs to fade in and out when clicking on them, add the `.fade` class to `.tab-pane`:

# Pills

Saturday, March 18, 2017 11:34 PM

## Pills

Home    Menu 1    Menu 2    Menu 3

Pills are created with `<ul class="nav nav-pills">`. Also mark the current page with `<li class="active">`:

### Example

```
<ul class="nav nav-pills">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">Menu 1</a></li>
  <li><a href="#">Menu 2</a></li>
  <li><a href="#">Menu 3</a></li>
</ul>
```

- Yaha '.nav' ke bad '.nav-pills' daal do.

## Vertical Pills

Home

Menu 1

Menu 2

Menu 3

Pills can also be displayed vertically. Just add the `.nav-stacked` class:

### Example

```
<ul class="nav nav-pills nav-stacked">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">Menu 1</a></li>
  <li><a href="#">Menu 2</a></li>
  <li><a href="#">Menu 3</a></li>
</ul>
```

- For more types see [www.w3schools.com/bootstrap/bootstrap\\_tabs\\_pills.html](http://www.w3schools.com/bootstrap/bootstrap_tabs_pills.html)

## Toggleable / Dynamic Pills

The same code applies to pills; only change the data-toggle attribute to `data-toggle="pill"`:

### Example

```
<ul class="nav nav-pills">
  <li class="active"><a data-toggle="pill" href="#home">Home</a></li>
  <li><a data-toggle="pill" href="#menu1">Menu 1</a></li>
  <li><a data-toggle="pill" href="#menu2">Menu 2</a></li>
</ul>

<div class="tab-content">
  <div id="home" class="tab-pane fade in active">
    <h3>HOME</h3>
    <p>Some content.</p>
  </div>
```

```
<div><h1>HOME</h1>
  <p>Some content.</p>
</div>
<div id="menu1" class="tab-pane fade">
  <h3>Menu 1</h3>
  <p>Some content in menu 1.</p>
</div>
<div id="menu2" class="tab-pane fade">
  <h3>Menu 2</h3>
  <p>Some content in menu 2.</p>
</div>
</div>
```

## Navbar

Sunday, March 19, 2017 12:05 AM

- We usually do not create "navbar" like this because navbar are better suited with <nav> tab.

```
<ul class="nav navbar-nav">
  <li role="presentation"><a href="#">Home</a></li>
  <li role="presentation" class="active"><a href="#">Mission</a></li>
  <li role="presentation"><a href="#">Services</a></li>
  <li role="presentation"><a href="#">Staff</a></li>
  <li role="presentation"><a href="#">Testimonials</a></li>
</ul>
```

OUTPUT -

Home Mission Services Staff Testimonials

## Our Mission

Wisdom Pet Medicine strives to blend the best in traditional and alternative medicine in the diagnosis and treatment of companion animals including dogs, cats, birds, reptiles, rodents, and fish. We apply the wisdom garnered in the centuries old tradition of veterinary medicine, to find the safest treatments and cures.

We strive to be your pet's medical experts from youth through the senior years. We build preventative health care plans for each and every one of our patients, based on breed, age, and sex, so that your pet receives the most appropriate care at crucial milestones. We want to give your pet a long and healthy life.

## Navigation Bars

- So we wrap <ul class="nav navbar-nav"> around <nav> like this.

```
<nav class="navbar navbar-default">
  <ul class="nav navbar-nav">
    <li><a href="#">Home</a></li>
    <li class="active"><a href="#">Mission</a></li>
    <li><a href="#">Services</a></li>
    <li><a href="#">Staff</a></li>
    <li><a href="#">Testimonials</a></li>
  </ul>
</nav>
```

- We have to add classe .navbar to tell this is <nav> and then type of navbar
  - .navbar-default

Home Mission Services Staff Testimonials

## Our Mission

- .navbar-inverse ( black background,white links)

Home Mission Services Staff Testimonials

## Our Mission

## Fixed Navbar

- By default , navbar disappears as we scroll down ,but we can make it fixed on top or bottom using
  - `.navbar-fixed-top`
  - `.navbar-fixed-bottom`

## Collapsing The Navigation Bar

The navigation bar takes up too much space on a small screen.

We should hide the navigation bar; and only show it when it is needed.

- We can add anything to navbar according to need  
see [www.w3schools.com/bootstrap/bootstrap\\_navbar.html](http://www.w3schools.com/bootstrap/bootstrap_navbar.html) for more. And see video.

# Breadcrumb, Pager and Pagination

Sunday, March 19, 2017 12:36 AM

- Breadcrumb tell where user are in heirarchy.
- We create a list of links and add class '.breadcrumb' to make it breadcrumb.

## Breadcrumbs

Another form for pagination, is breadcrumbs:

Home / Private / Pictures / Vacation

The `.breadcrumb` class indicates the current page's location within a navigational hierarchy:

### Example

```
<ul class="breadcrumb">
  <li><a href="#">Home</a></li>
  <li><a href="#">Private</a></li>
  <li><a href="#">Pictures</a></li>
  <li class="active">Vacation</li>
</ul>
```

# What is Pager?

Pager is also a form of pagination (as described in the previous chapter).

Pager provides previous and next buttons (links).

To create previous/next buttons, add the `.pager` class to an `<ul>` element:

## Example

```
<ul class="pager">
  <li><a href="#">Previous</a></li>
  <li><a href="#">Next</a></li>
</ul>
```

## Result:

Previous Next

# Align Buttons

Use the `.previous` and `.next` classes to align each button to the sides of the page:

## Example

```
<ul class="pager">
  <li class="previous"><a href="#">Previous</a></li>
  <li class="next"><a href="#">Next</a></li>
</ul>
```

## Result:

Previous

Next

- We can have more than 2 `<li>` but it doesn't make sense.
- We can add '`.disabled`' to disable if we are at first or last page.

- We can have more than 2 <li> but it doesn't make sense.
- We can add '.disabled' to disable if we are at first or last page.

## Basic Pagination

If you have a web site with lots of pages, you may wish to add some sort of pagination to each page.

A basic pagination in Bootstrap looks like this:

1	2	3	4	5
---	---	---	---	---

To create a basic pagination, add the `.pagination` class to an `<ul>` element:

### Example

```
<ul class="pagination">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

# Active State

The active state shows what is the current page:

1	2	3	4	5
---	---	---	---	---

Add class `.active` to let the user know which page he/she is on:

## Example

```
<ul class="pagination">
  <li><a href="#">1</a></li>
  <li class="active"><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

# Disabled State

A disabled link cannot be clicked:

1	2	3	4	5
---	---	---	---	---

Add class `.disabled` if a link for some reason is disabled:

## Example

```
<ul class="pagination">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li class="disabled"><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

# Pagination Sizing

Pagination blocks can also be sized to a larger size or a smaller size:



Add class `.pagination-lg` for larger blocks or `.pagination-sm` for smaller blocks:

## Example

```
<ul class="pagination pagination-lg">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
</ul>
```

## Jumbotron and Page header

Sunday, March 19, 2017 12:50 AM

### Creating a Jumbotron

A jumbotron indicates a big box for calling extra attention to some special content or information.

A jumbotron is displayed as a grey box with rounded corners. It also enlarges the font sizes of the text inside it.

**Tip:** Inside a jumbotron you can put nearly any valid HTML, including other Bootstrap elements/classes.

Use a `<div>` element with class `.jumbotron` to create a jumbotron:

# Bootstrap Tutorial

Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile-first projects on the web.

## Jumbotron Inside Container

Place the jumbotron inside the `<div class="container">` if you want the jumbotron to NOT extend to the edge of the screen:

### Example

```
<div class="container">
  <div class="jumbotron">
    <h1>Bootstrap Tutorial</h1>
    <p>Bootstrap is the most popular HTML, CSS, and JS framework
for developing
    responsive, mobile-first projects on the web.</p>
  </div>
  <p>This is some text.</p>
  <p>This is another text.</p>
</div>
```

## Jumbotron Outside Container

Place the jumbotron outside the `<div class="container">` if you want the jumbotron to extend to the screen edges:

### Example

```
<div class="jumbotron">
  <h1>Bootstrap Tutorial</h1>
  <p>Bootstrap is the most popular HTML, CSS, and JS framework
for developing responsive,
  mobile-first projects on the web.</p>
</div>
<div class="container">
  <p>This is some text.</p>
  <p>This is another text.</p>
</div>
```

- Isme jumbotron poore screen ke corners tak jayeg.

## Creating a Page Header

A page header is like a section divider.

# Creating a Page Header

A page header is like a section divider.

The `.page-header` class adds a horizontal line under the heading (+ adds some extra space around the element):

## Example Page Header

Use a `<div>` element with class `.page-header` to create a page header:

### Example

```
<div class="page-header">
  <h1>Example Page Header</h1>
</div>
```

## Example Page Header

This is some text.

This is another text.

- Here we can see after page-header we get small line to separate it from other content.

## Media Objects

Sunday, March 19, 2017 7:18 PM

- Bootstrap has pretty good classes which used to show images/videos with other content like images with text, as we use in comments.

## Media Objects

Bootstrap provides an easy way to align media objects (like images or videos) to the left or to the right of some content. This can be used to display blog comments, tweets and so on:



John Doe *Posted on February 19, 2016*  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.



John Doe *Posted on February 20, 2016*  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Alicia Keyes *Posted on February 25, 2016*  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Nulla vel metus scelerisque ante sollicitudin commodo.



## Basic Media Object



John Doe

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

John Doe

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.



- For placing media and content together we need to wrap both media and content inside `.media` class, inside which we align '`.media-left/right`' and place content in '`.media-body`'.
- In case of right-aligned media comes after content-body.

## Example

```
<!-- Left-aligned -->
<div class="media">
  <div class="media-left">
    
  </div>
  <div class="media-body">
    <h4 class="media-heading">John Doe</h4>
    <p>Lorem ipsum...</p>
  </div>
</div>

<!-- Right-aligned -->
<div class="media">
  <div class="media-body">
    <h4 class="media-heading">John Doe</h4>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
    incididunt ut labore et dolore magna aliqua.</p>
  </div>
  <div class="media-right">
    
  </div>
</div>
```

## Example explained

Use a <div> element with the `.media` class to create a container for media objects.

Use the `.media-left` class to align the media object (image) to the left, or the `.media-right` class to align it to the right.

Text that should appear next to the image, is placed inside a container with `class="media-body"`.

Additionally, you can use `.media-heading` for headings.

## Top, Middle or Bottom Alignment

The media object can also be top, middle or bottom aligned with the `media-top`, `media-middle` or `media-bottom` class:



### Media Top

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

### Media Middle

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

### Media Bottom

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

 Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

- Along with left/right also add position top,middle or bottom.

## Example

```
<!-- Media top -->
<div class="media">
  <div class="media-left media-top">
    
  </div>
  <div class="media-body">
    <h4 class="media-heading">Media Top</h4>
    <p>Lorem ipsum...</p>
  </div>
</div>

<!-- Media middle -->
<div class="media">
  <div class="media-left media-middle">
    
  </div>
  <div class="media-body">
    <h4 class="media-heading">Media Middle</h4>
    <p>Lorem ipsum...</p>
  </div>
</div>
```

```

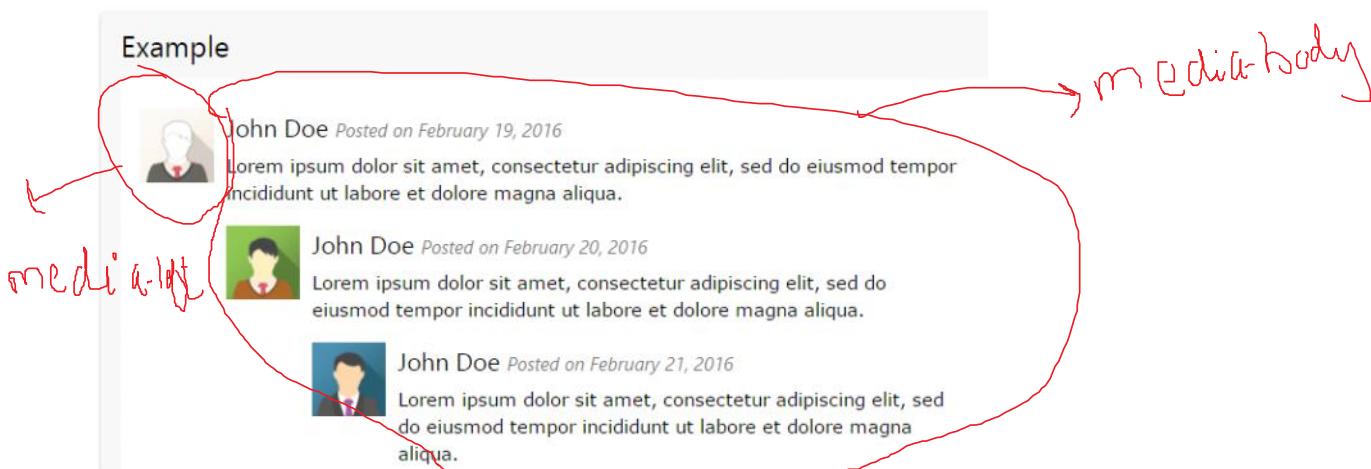
<!-- Media bottom -->
<div class="media">
  <div class="media-left media-bottom">
    
  </div>
  <div class="media-body">
    <h4 class="media-heading">Media Bottom</h4>
    <p>Lorem ipsum...</p>
  </div>
</div>

```

- See w3school for following nestings. [www.w3schools.com/bootstrap/bootstrap\\_media\\_objects.html](http://www.w3schools.com/bootstrap/bootstrap_media_objects.html)

## Nesting Media Objects

Media objects can also be nested (a media object inside a media object):



- Image will be at left or right, but inside the `.media-body` nest another `.media` and so on.

```

<div class="media">
  <div class="media-left">
    
  </div>
  <div class="media-body">
    <h4 class="media-heading">John Doe <small><i>Posted on February 19, 2016</i></small></h4>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
    incididunt ut labore et dolore magna aliqua.</p>

    <!-- Nested media object -->
    <div class="media">
      <div class="media-left">
        
      </div>
      <div class="media-body">
        <h4 class="media-heading">John Doe <small><i>Posted on February 19, 2016</i></small></h4>
        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
        incididunt ut labore et dolore magna aliqua.</p>

        <!-- Nested media object -->
        <div class="media">
          <div class="media-left">
            
          </div>
          <div class="media-body">
            <h4 class="media-heading">John Doe <small><i>Posted on February 19, 2016</i></small></h4>
            <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
            incididunt ut labore et dolore magna aliqua.</p>
          </div>
        </div>
      </div>
    </div>
  </div>

```

- Assign `.media-heading` for better heading.

## Another Example of Nesting

### Example

## Another Example of Nesting

### Example



John Doe *Posted on February 19, 2016*

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.



John Doe *Posted on February 20, 2016*

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.



John Doe *Posted on February 21, 2016*

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.



Jane Doe *Posted on February 20, 2016*

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

# Audio and Video

Sunday, March 19, 2017 8:00 PM

## Responsive Embeds

Also let videos or slideshows scale properly on any device.

Classes can be applied directly to `<iframe>`, `<embed>`, `<video>`, and `<object>` elements.

The following example creates a responsive video by adding an `.embed-responsive-item` class to an `<iframe>` tag (the video will then scale nicely to the parent element). The containing `<div>` defines the aspect ratio of the video:

### Example

```
<div class="embed-responsive embed-responsive-16by9">
  <iframe class="embed-responsive-item" src="..."></iframe>
</div>
```

- Just add `.embed-responsive` and `embed-responsive-*` to parent and `embed-responsive-item` to that item(`iframe,video` etc)

### EXAMPLE -

```
<div class="col-xs-6 col-sm-4">
  <div class="embed-responsive embed-responsive-4by3">
    <video class="embed-responsive-item" src="video/brushing.mp4"
      autoplay controls muted></video>
  </div>
  <h3>Grooming</h3>
  <p>Our therapeutic grooming treatments help battle fleas,
    allergic dermatitis, and other challenging skin conditions.</p>
</div>
```

## Services



### Exotic Pets

We offer specialized care for reptiles, rodents, birds, and other exotic pets.

### Grooming

Our therapeutic grooming treatments help battle fleas, allergic dermatitis, and other challenging skin conditions.

- Now, the video is responsive and give same responsiveness as the images.

# Carousel

Sunday, March 19, 2017 8:30 PM

## Bootstrap Carousel Plugin

### The Carousel Plugin

The Carousel plugin is a component for cycling through elements, like a carousel (slideshow).

**Tip:** Plugins can be included individually (using Bootstrap's individual "carousel.js" file), or all at once (using "bootstrap.js" or "bootstrap.min.js").

### Making a Carousel

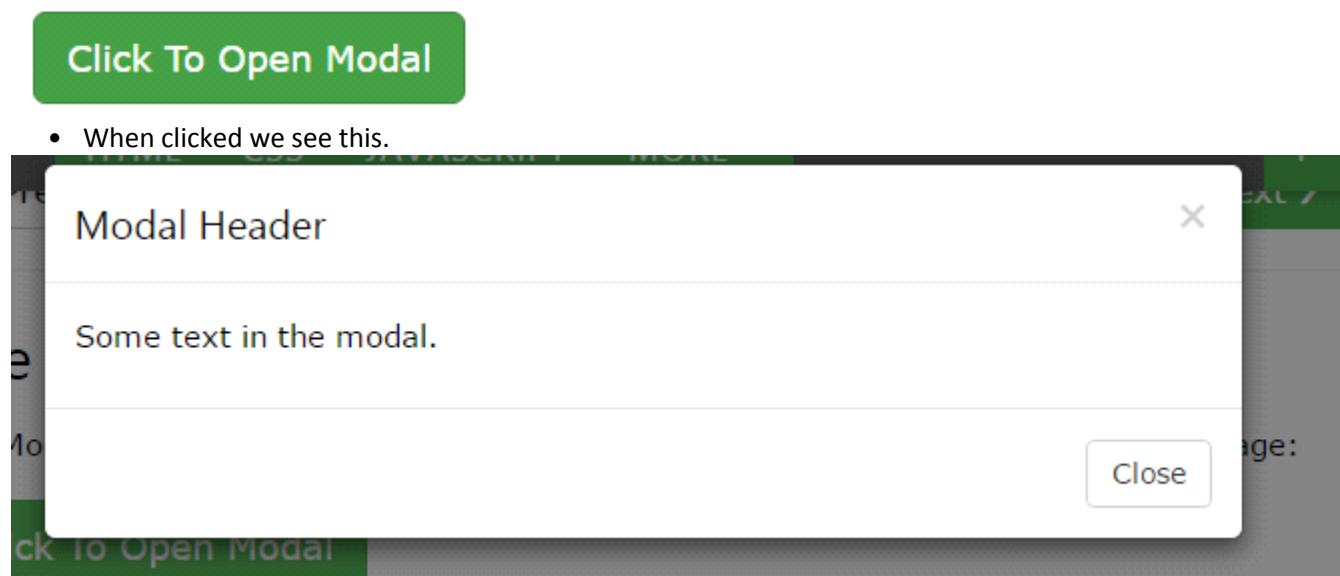
- Step1 - carousel created using class **.carousel** , so create a **div.carousel**
- Step2 - all images go into **div.carousel-inner**
- step3- each image is wrapped **into div.item**
- step4 - make any <img> active by adding class **.active**
- Step5- for a image we can add caption , by wraption caption inside **.div-carousel-caption**
- Step6- we can add prev and next navigation using **.carousel-control** class.  
Watch Video for more.

# Modal

Sunday, March 19, 2017 8:41 PM

## The Modal Plugin

The Modal plugin is a dialog box/popup window that is displayed on top of the current page:



## How To Create a Modal

- Step1 - create button ( trigge )

```
<!-- Trigger the modal with a button -->
<button type="button" class="btn btn-info btn-lg" data-toggle="modal" data-target="#myModal">Open Modal</button>
```

First create a button, by giving it `.data-toggle='modal'` and target as id of modal.

- Step2 - create model  
see video.

# Collapse and Accordion

Sunday, March 19, 2017 8:49 PM

## Basic Collapsible

Collapsibles are useful when you want to hide and show large amount of content:

**Click Me**

- When button clicked

**Click Me**

  Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

- We have to create
  - Trigger(button) ( add `data-toggle='collapse'` and `data-target='#id_ofCollapse'` )
  - Content

### Example

```
<button data-toggle="collapse" data-target="#demo">Collapsible</button>

<div id="demo" class="collapse">
  Lorem ipsum dolor text....
</div>
```

### Example Explained

The `.collapse` class indicates a collapsible element (a `<div>` in our example); this is the content that will be shown or hidden with a click of a button.

To control (show/hide) the collapsible content, add the `data-toggle="collapse"` attribute to an `<a>` or a `<button>` element. Then add the `data-target="#id"` attribute to connect the button with the collapsible content (`<div id="demo">`).

**Note:** For `<a>` elements, you can use the `href` attribute instead of the `data-target` attribute:

## Example

```
<a href="#demo" data-toggle="collapse">Collapsible</a>

<div id="demo" class="collapse">
Lorem ipsum dolor text....
</div>
```

By default, the collapsible content is hidden. However, you can add the `.in` class to show the content by default:

## Example

```
<div id="demo" class="collapse in">
Lorem ipsum dolor text....
</div>
```

## Collapsible Panel

Collapsible panel

- When clicked

Collapsible panel

Panel Body

Panel Footer

- `.panel-title` ko trigger bana do `panel-body` aur `panel-footer` ko `collapse` bana do

## Example

```
<div class="panel-group">
<div class="panel panel-default">
<div class="panel-heading">
<h4 class="panel-title">
<a data-toggle="collapse" href="#collapse1">Collapsible panel</a>
</h4>
</div>
<div id="collapse1" class="panel-collapse collapse">
```

## Example

```
<div class="panel-group">
  <div class="panel panel-default">
    <div class="panel-heading">
      <h4 class="panel-title">
        <a data-toggle="collapse" href="#collapse1">Collapsible panel</a>
      </h4>
    </div>
    <div id="collapse1" class="panel-collapse collapse">
      <div class="panel-body">Panel Body</div>
      <div class="panel-footer">Panel Footer</div>
    </div>
  </div>
</div>
```

## Collapsible List Group

Collapsible list group

WHEN CLICKED

Collapsible list group

One

Two

Three

Footer

```
<div class="panel-group">
  <div class="panel panel-default">
    <div class="panel-heading">
      <h4 class="panel-title">
        <a data-toggle="collapse" href="#collapse1">Collapsible list
group</a>
      </h4>
    </div>
    <div id="collapse1" class="panel-collapse collapse">
      <ul class="list-group">
        <li class="list-group-item">One</li>
        <li class="list-group-item">Two</li>
        <li class="list-group-item">Three</li>
      </ul>
      <div class="panel-footer">Footer</div>
    </div>
  </div>
</div>
```

- Accordion is a group of Collapse Panels, in which only 1 can be seen at a time.

## Accordion

Collapsible Group 1

Collapsible Group 2

Collapsible Group 3

- When any panel clicked.

Collapsible Group 1

Collapsible Group 2

  Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt  
  ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation  
  ullamco laboris nisi ut aliquip ex ea commodo consequat.

Collapsible Group 3

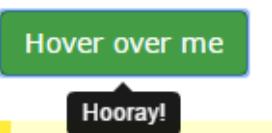
- When a panel is clicked other automatically hides and that opens up.
- FOR MORE SEE VIDEO or W3Schools.com

# ToolTip

Sunday, March 19, 2017 9:06 PM

## The Tooltip Plugin

The Tooltip plugin is small pop-up box that appears when the user moves the mouse pointer over an element:



## How To Create a Tooltip

To create a tooltip, add the `data-toggle="tooltip"` attribute to an element.

Use the `title` attribute to specify the text that should be displayed inside the tooltip:

```
<a href="#" data-toggle="tooltip" title="Hooray!">Hover over me</a>
```

- We already add 'title' attribute to any <tag> to get tooltip, but it is not fancy, so we can add just `data-toggle='tooltip'` to make it better.
- But we have to do little work here, we need to add following code as it is.

**Note:** Tooltips must be initialized with jQuery: select the specified element and call the `tooltip()` method.

The following code will enable all tooltips in the document:

```
<script>
$(document).ready(function(){
    $('[data-toggle="tooltip"]').tooltip();
});
</script>
```

- Telling Jquery to look for any element having `data-toggle="tooltip"` and add tooltip to it.

# Positioning Tooltips

By default, the tooltip will appear on top of the element.

Use the `data-placement` attribute to set the position of the tooltip on top, bottom, left or the right side of the element:

## Example

```
<a href="#" data-toggle="tooltip" data-placement="top"
title="Hooray!">Hover</a>
<a href="#" data-toggle="tooltip" data-placement="bottom"
title="Hooray!">Hover</a>
<a href="#" data-toggle="tooltip" data-placement="left"
title="Hooray!">Hover</a>
<a href="#" data-toggle="tooltip" data-placement="right"
title="Hooray!">Hover</a>
```

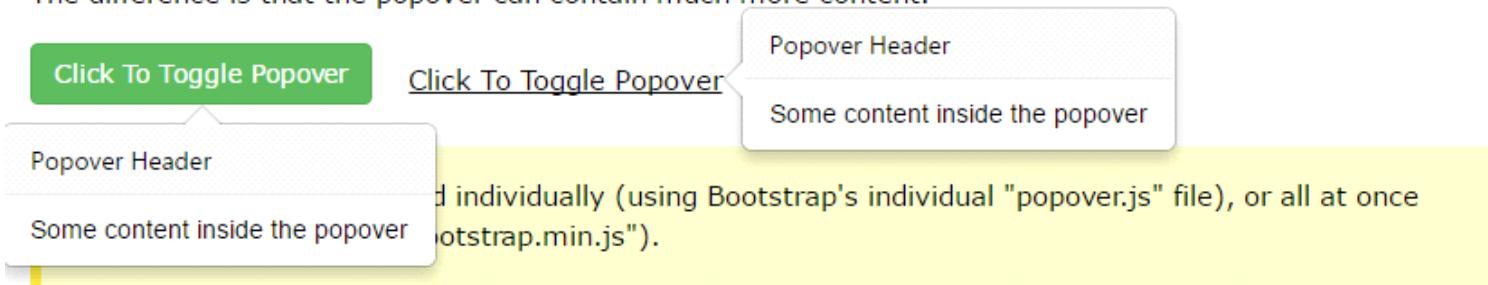
- Add `data-placement` to specify position.

# Popovers

Sunday, March 19, 2017 9:14 PM

## The Popover Plugin

The Popover plugin is similar to tooltips; it is a pop-up box that appears when the user clicks on an element. The difference is that the popover can contain much more content.



## How To Create a Popover

To create a popover, add the `data-toggle="popover"` attribute to an element.

Use the `title` attribute to specify the header text of the popover, and use the `data-content` attribute to specify the text that should be displayed inside the popover's body:

```
<a href="#" data-toggle="popover" title="Popover Header" data-content="Some content inside the popover">Toggle popover</a>
```

- Just as Tooltip we have to manually initialize with jQuery.

**Note:** Popovers must be initialized with jQuery: select the specified element and call the `popover()` method.

The following code will enable all popovers in the document:

### Example

```
<script>
$(document).ready(function(){
    $('[data-toggle="popover"]').popover();
});
</script>
```

# Positioning Popovers

By default, the popover will appear on the right side of the element.

Use the `data-placement` attribute to set the position of the popover on top, bottom, left or the right side of the element:

## Example

```
<a href="#" title="Header" data-toggle="popover" data-placement="top" data-content="Content">Click</a>
<a href="#" title="Header" data-toggle="popover" data-placement="bottom" data-content="Content">Click</a>
<a href="#" title="Header" data-toggle="popover" data-placement="left" data-content="Content">Click</a>
<a href="#" title="Header" data-toggle="popover" data-placement="right" data-content="Content">Click</a>
```

- Exactly same as Tooltip.

# Closing Popovers

By default, the popover is closed when you click on the element again. However, you can use the `data-trigger="focus"` attribute which will close the popover when clicking outside the element:

## Example

```
<a href="#" title="Dismissible popover" data-toggle="popover" data-trigger="focus" data-content="Click anywhere in the document to close this popover">Click me</a>
```

**Tip:** If you want the popover to be displayed when you move the mouse pointer over the element, use the `data-trigger` attribute with a value of "hover":

## Example

```
<a href="#" title="Header" data-toggle="popover" data-trigger="hover" data-content="Some content">Hover over me</a>
```

# Affix and Scrollpy

Sunday, March 19, 2017 9:19 PM

## The Affix Plugin

The Affix plugin allows an element to become affixed (locked) to an area on the page. This is often used with navigation menus or social icon buttons, to make them "stick" at a specific area while scrolling up and down the page.

The plugin **toggles** this behavior on and off (changes the value of `CSS position` from `static` to `fixed`), depending on scroll position.

- This prevents us to not go to up for navbar after reaching at the middle of page.

**Example 1)** An affixed navbar:

A screenshot showing a dark-themed browser window. At the top, there is a navigation bar with three items: "Page 1" (highlighted in white), "Page 2", and "Page 3". Below the navigation bar, the main content area contains the text "Some text to enable scrolling" repeated four times. A vertical scrollbar is visible on the right side of the content area.

**Example 2)** An affixed sidebar:

A screenshot showing a dark-themed browser window. On the left, there is a sidebar with three sections: "Section 1" (highlighted in blue), "Section 2", and "Section 3". The main content area contains the text "Some text to enable scrolling" repeated six times. A vertical scrollbar is visible on the right side of the content area.

# The Scrollspy Plugin

The Scrollspy plugin is used to automatically update links in a navigation list based on scroll position.

WebSiteName Section 1 Section 2 Section 3 Section 4 ▾

## Section 2

Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!

Try to scroll this section and look at the navigation bar while scrolling! Try to scroll this section and look at the navigation bar while scrolling!