**3. Containers**

Bootstrap also requires a containing element to wrap site contents.

There are two container classes to choose from:

1. The .container class provides a responsive **fixed width container**
2. The .container-fluid class provides a **full width container**, spanning the entire width of the viewport

Grid Classes

The Bootstrap grid system has four classes:

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

The classes above can be combined to create more dynamic and flexible layouts.

Basic Structure of a Bootstrap Grid

The following is a basic structure of a Bootstrap grid:

<div class="row">  
  <div class="col-\*-\*"></div>  
  <div class="col-\*-\*"></div>  
</div>  
<div class="row">  
  <div class="col-\*-\*"></div>  
  <div class="col-\*-\*"></div>  
  <div class="col-\*-\*"></div>  
</div>  
<div class="row">  
  ...  
</div>

First; create a row (<div class="row">). Then, add the desired number of columns (tags with appropriate .col-\*-\* classes). Note that numbers in .col-\*-\* should always add up to 12 for each row.

Below we have collected some examples of basic Bootstrap grid layouts.

## Two Unequal Columns

<div class="row">  
  <div class="col-sm-4">.col-sm-4</div>  
  <div class="col-sm-8">.col-sm-8</div>  
</div>

## Bootstrap's Default Settings

Bootstrap's global default font-size is 14px, with a line-height of 1.428.

This is applied to the <body> element and all paragraphs (<p>).

In addition, all <p> elements have a bottom margin that equals half their computed line-height (10px by default).

## Bootstrap vs. Browser Defaults

In this chapter, we will look at some HTML elements that will be styled a little bit differently by Bootstrap than browser defaults.

## <small>

In Bootstrap the HTML <small> element is used to create a lighter, secondary text in any heading.

## <blockquote>

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

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

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

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—

## <dl>

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

**Coffee**

- black hot drink

**Milk**

- white cold drink

## <code>

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

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:

Use ctrl + p to open the Print dialog box.

## 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:

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:

This text is important.

This text indicates success.

This text represents some information.

This text represents a warning.

This text represents danger.

The Bootstrap classes below can be added to style HTML elements further:

|  |  |
| --- | --- |
| **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 <abbr> element in a slightly smaller font size |
| .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-inline | Places all list items on a single line |
| .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 |
| .pre-scrollable | Makes a <pre> element scrollable |

## Bootstrap Basic Table

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

The .table class adds basic styling to a table:

## Striped Rows

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

## Bordered Table

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

## Hover Rows

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

## Condensed Table

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

## Contextual Classes

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

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 |

## 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

## Rounded Corners

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

<img src="cinqueterre.jpg" class="img-rounded" alt="Cinque Terre">

## Circle

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

## Thumbnail

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

## 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

## Image Gallery

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

## 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:

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

You can choose between two aspect ratio classes:

<!-- 16:9 aspect ratio -->  
<div class="embed-responsive embed-responsive-16by9">  
  <iframe class="embed-responsive-item" src="..."></iframe>  
</div>  
  
<!-- 4:3 aspect ratio -->  
<div class="embed-responsive embed-responsive-4by3">  
  <iframe class="embed-responsive-item" src="..."></iframe>  
</div>

## 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

## 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:

<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:

<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>

## 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)

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

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

## Wells

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

<div class="well">Basic Well</div>

## Well Size

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

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

By default, wells are medium in size.

## lerts

Bootstrap provides an easy way to create predefined alert messages

Alerts are created with the .alert class, followed by one of the four contextual classes .alert-success, .alert-info, .alert-warning or .alert-danger

## Alert Links

Add the alert-link class to any links inside the alert box to create "matching colored links":

<div class="alert alert-success">  
  <strong>Success!</strong> You should <a href="#" class="alert-link">read this message</a>.  
</div>

## Closing Alerts

To close the alert message, add a .alert-dismissible class to the alert container. Then add class="close" and data-dismiss="alert" to a link or a button element (when you click on this the alert box will disappear).

<div class="alert alert-success alert-dismissible">  
  <a href="#" class="close" data-dismiss="alert" aria-label="close">&times;</a>  
  <strong>Success!</strong> Indicates a successful or positive action.  
</div>

**The aria-\* attribute and &times; explanation**  
  
To help improve accessibility for people using screen readers, you should include the aria-label="close" attribute, when creating a close button.  
  
&times; (×) is an HTML entity that is the preferred icon for close buttons, rather than the letter "x".  
For a list of all HTML Entities, [visit our HTML Entities Reference](https://www.w3schools.com/charsets/ref_html_entities_4.asp).

## Animated Alerts

The .fade and .in classes adds a fading effect when closing the alert message:

<div class="alert alert-danger fade in">

## Button Styles

Bootstrap provides different styles of buttons:

* .btn
* .btn-default
* .btn-primary
* .btn-success
* .btn-info
* .btn-warning
* .btn-danger
* .btn-link
* <button type="button" class="btn">Basic</button>  
  <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>  
  <button type="button" class="btn btn-info">Info</button>  
  <button type="button" class="btn btn-warning">Warning</button>  
  <button type="button" class="btn btn-danger">Danger</button>  
  <button type="button" class="btn btn-link">Link</button>

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

<a href="#" class="btn btn-info" role="button">Link Button</a>  
<button type="button" class="btn btn-info">Button</button>  
<input type="button" class="btn btn-info" value="Input Button">  
<input type="submit" class="btn btn-info" value="Submit Button">

## Button Sizes

The classes that define the different sizes are:

* .btn-lg
* .btn-sm
* .btn-xs
* <button type="button" class="btn btn-primary btn-lg">Large</button>  
  <button type="button" class="btn btn-primary">Normal</button>  
  <button type="button" class="btn btn-primary btn-sm">Small</button>  
  <button type="button" class="btn btn-primary btn-xs">XSmall</button>

## Block Level Buttons

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

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

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

## Button Groups

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

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

<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:

<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>

## Vertical Button Groups

Bootstrap also supports vertical button groups:

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

<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:

<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>

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

<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>

## Nesting Button Groups & Dropdown Menus

<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">  
    <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>

## Split Button Dropdowns

<div class="btn-group">  
  <button type="button" class="btn btn-primary">Sony</button>  
  <button type="button" class="btn btn-primary dropdown-toggle" data-toggle="dropdown">  
    <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>

## Glyphicon Syntax

A glyphicon is inserted with the following syntax:

<span class="glyphicon glyphicon-*name*"></span>

## Badges

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

[News **5**](javascript:void(0);)  
[Comments **10**](javascript:void(0);)  
[Updates **2**](javascript:void(0);)

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

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

<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>

The following example shows how to add badges to buttons:

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

## Labels

Labels are used to provide additional information about something:

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:

<h1>Example <span class="label label-default">New</span></h1>  
<h2>Example <span class="label label-default">New</span></h2>

<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>

## Basic Progress Bar

A progress bar can be used to show a user how far along he/she is in a process.

To create a default progress bar, add a .progress class to a <div> element:

<div class="progress">  
  <div class="progress-bar" role="progressbar" aria-valuenow="70"  
  aria-valuemin="0" aria-valuemax="100" style="width:70%">  
    <span class="sr-only">70% Complete</span>  
  </div>  
</div>

**Note:** To help improve accessibility for people using screen readers, you should include the aria-\* attributes.

## Progress Bar With Label

Remove the .sr-only class from the progress bar to show a visible percentage:

<div class="progress">  
  <div class="progress-bar" role="progressbar" aria-valuenow="70"  
  aria-valuemin="0" aria-valuemax="100" style="width:70%">  
    70%  
  </div>  
</div>

Colored Progress Bars

Contextual classes are used to provide "meaning through colors".

The contextual classes that can be used with progress bars are:

* .progress-bar-success
* .progress-bar-info
* .progress-bar-warning
* .progress-bar-danger

<div class="progress">  
  <div class="progress-bar progress-bar-success" role="progressbar" aria-valuenow="40"  
  aria-valuemin="0" aria-valuemax="100" style="width:40%">  
    40% Complete (success)  
  </div>  
</div>  
  
<div class="progress">  
  <div class="progress-bar progress-bar-info" role="progressbar" aria-valuenow="50"  
  aria-valuemin="0" aria-valuemax="100" style="width:50%">  
    50% Complete (info)  
  </div>  
</div>  
  
<div class="progress">  
  <div class="progress-bar progress-bar-warning" role="progressbar" aria-valuenow="60"  
  aria-valuemin="0" aria-valuemax="100" style="width:60%">  
    60% Complete (warning)  
  </div>  
</div>  
  
<div class="progress">  
  <div class="progress-bar progress-bar-danger" role="progressbar" aria-valuenow="70"  
  aria-valuemin="0" aria-valuemax="100" style="width:70%">  
    70% Complete (danger)  
  </div>  
</div>

## Striped Progress Bars

Progress bars can also be striped:

Add class .progress-bar-striped to add stripes to the progress bars:

<div class="progress">  
  <div class="progress-bar progress-bar-success progress-bar-striped" role="progressbar"  
  aria-valuenow="40" aria-valuemin="0" aria-valuemax="100" style="width:40%">  
    40% Complete (success)  
  </div>  
</div>

## Animated Progress Bar

Add class .active to animate the progress bar:

<div class="progress">  
  <div class="progress-bar progress-bar-striped active" role="progressbar"  
  aria-valuenow="40" aria-valuemin="0" aria-valuemax="100" style="width:40%">  
    40%  
  </div>  
</div>

## Stacked Progress Bars

Create a stacked progress bar by placing multiple bars into the same <div class="progress">:

<div class="progress">  
  <div class="progress-bar progress-bar-success" role="progressbar" style="width:40%">  
    Free Space  
  </div>  
  <div class="progress-bar progress-bar-warning" role="progressbar" style="width:10%">  
    Warning  
  </div>  
  <div class="progress-bar progress-bar-danger" role="progressbar" style="width:20%">  
    Danger  
  </div>  
</div>

## Basic Pagination

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

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

<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:

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

<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:

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

<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

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

<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>  
  
<ul class="pagination pagination-sm">  
  <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>

## Breadcrumbs

Another form for pagination, is breadcrumbs:

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

<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>

# **Bootstrap 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:

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

## Align Buttons

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

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

## Basic List Groups

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

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

<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

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

<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

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

<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

Use the .active class to highlight the current item:

<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

To disable an item, add the .disabled class:

<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>

## Contextual Classes

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

<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:

<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>

## Panels

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

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

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

## Panel Heading

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

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

## 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:

<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>

## Basic Dropdown

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

<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:

<li class="divider"></li>

## Dropdown Header

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

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

## Disable and Active items

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

## Dropdown Position

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

## Dropup

<div class="dropup"

## Dropdown Accessibility

To help improve accessibility for people using screen readers, you should include the following role and aria-\* attributes, when creating a dropdown menu:

<div class="dropdown">  
  <button class="btn btn-default dropdown-toggle" type="button" id="menu1" data-toggle="dropdown">Tutorials  
  <span class="caret"></span></button>  
  <ul class="dropdown-menu" role="menu" aria-labelledby="menu1">  
    <li role="presentation"><a role="menuitem" href="#">HTML</a></li>  
    <li role="presentation"><a role="menuitem" href="#">CSS</a></li>  
    <li role="presentation"><a role="menuitem" href="#">JavaScript</a></li>  
    <li role="presentation" class="divider"></li>  
    <li role="presentation"><a role="menuitem" href="#">About Us</a></li>  
  </ul>  
</div>

## Basic Collapsible

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

<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

<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:

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

## Collapsible Panel

<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

<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

**Note:** Use the data-parent attribute to make sure that all collapsible elements under the specified parent will be closed when one of the collapsible item is shown.

<div class="panel-group" id="accordion">  
  <div class="panel panel-default">  
    <div class="panel-heading">  
      <h4 class="panel-title">  
        <a data-toggle="collapse" data-parent="#accordion" href="#collapse1">  
        Collapsible Group 1</a>  
      </h4>  
    </div>  
    <div id="collapse1" class="panel-collapse collapse in">  
      <div class="panel-body">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.</div>  
    </div>  
  </div>  
  <div class="panel panel-default">  
    <div class="panel-heading">  
      <h4 class="panel-title">  
        <a data-toggle="collapse" data-parent="#accordion" href="#collapse2">  
        Collapsible Group 2</a>  
      </h4>  
    </div>  
    <div id="collapse2" class="panel-collapse collapse">  
      <div class="panel-body">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.</div>  
    </div>  
  </div>  
  <div class="panel panel-default">  
    <div class="panel-heading">  
      <h4 class="panel-title">  
        <a data-toggle="collapse" data-parent="#accordion" href="#collapse3">  
        Collapsible Group 3</a>  
      </h4>  
    </div>  
    <div id="collapse3" class="panel-collapse collapse">  
      <div class="panel-body">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.</div>  
    </div>  
  </div>  
</div>

## 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>

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

<ul class="list-inline">

## Tabs

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

<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>

### **Tabs With Dropdown Menu**

Tabs can also hold dropdown menus.

<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>

## Pills

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

<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>

### **Vertical Pills**

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

<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>

### **Vertical Pills in a Row**

The following example places the vertical pill menu inside the last column. So, on a large screen the menu will be displayed to the right. But on a small screen, the content will automatically adjust itself into a single-column layout:

<div class="col-md-3">  
  <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>  
</div>

### **Pills With Dropdown Menu**

<ul class="nav nav-pills nav-stacked">  
  <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

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

Note that on screens that are smaller than 768px, the list items are stacked (content will remain centered):

<!-- 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

To make the tabs toggleable, add the data-toggle="tab" attribute to each link. Then add a .tab-pane class with a unique ID for every tab and wrap them inside a <div> element with class .tab-content.

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

<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">  
    <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>

## Toggleable / Dynamic Pills

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

<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 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>

## Navigation Bars

A navigation bar is a navigation header that is placed at the top of the page

With Bootstrap, a navigation bar can extend or collapse, depending on the screen size.

A standard navigation bar is created with <nav class="navbar navbar-default">.

<nav class="navbar navbar-default">  
  <div class="container-fluid">  
    <div class="navbar-header">  
      <a class="navbar-brand" href="#">WebSiteName</a>  
    </div>  
    <ul class="nav navbar-nav">  
      <li class="active"><a href="#">Home</a></li>  
      <li><a href="#">Page 1</a></li>  
      <li><a href="#">Page 2</a></li>  
      <li><a href="#">Page 3</a></li>  
    </ul>  
  </div>  
</nav>  
...

**Note:** All of the examples on this page will show a navigation bar that takes up too much space on small screens (however, the navigation bar will be on one single line on large screens - because Bootstrap is responsive). This problem (with the small screens) will be solved in the last example on this page.