

Twitter Bootstrap

Tutorial 2

Sukwon Oh

soh@cs.toronto.edu

Designed for everyone, everywhere.

Bootstrap makes front-end web development faster and easier. It's made for folks of all skill levels, devices of all shapes, and projects of all sizes.



Preprocessors

In addition to vanilla CSS, Bootstrap includes support for the two most popular CSS preprocessors, [Less](#) and [Sass](#).



One framework, every device.

Bootstrap easily and efficiently scales your project with one code base, from phones to tablets to desktops.



Comprehensive docs

With Bootstrap, you get extensive and beautiful documentation with hundreds of live examples, code snippets, and more.

Objective

- Learn basic features supported in Bootstrap.
- Simple demo using some of features learned.

What is Bootstrap?

Global CSS Settings

- Grid system
- Typography
- Forms
- Buttons
- Responsive utilities
- ...

Components

- Glyphicons
- Navbar
- Pagination
- Thumbnails
- ...

Using Bootstrap

Download

Bootstrap has a few easy ways to quickly get started, each one appealing to a different skill level and use case. Read through to see what suits your particular needs.

Bootstrap

Compiled and minified CSS, JavaScript, and fonts. No docs or original source files are included.

Download Bootstrap

Currently v3.1.1.

Source code

Source Less, JavaScript, and font files, along with our docs. **Requires a Less compiler and some setup.**

Download source

Sass

[Bootstrap ported from Less to Sass](#) for easy inclusion in Rails, Compass, or Sass-only projects.

Download Sass

Using Bootstrap

Or you could use CDN links below

```
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap.min.css">

<!-- Optional theme -->
<link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap-theme.min.css">

<!-- Latest compiled and minified JavaScript -->
<script src="//netdna.bootstrapcdn.com/bootstrap/3.1.1/js/bootstrap.min.js"></script>
```

- put

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
```

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Bootstrap 101 Template</title>

    <!-- Bootstrap -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
    <!--[if lt IE 9]>
      <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
      <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
    <![endif]-->
  </head>
  <body>
    <h1>Hello, world!</h1>

    <!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
    <!-- Include all compiled plugins (below), or include individual files as needed -->
    <script src="js/bootstrap.min.js"></script>
  </body>
</html>
```

Grid System

- Used for creating page layouts through a series of rows and columns

.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1
.col-md-8								.col-md-4			
.col-md-4				.col-md-4				.col-md-4			
.col-md-6						.col-md-6					

Grid System - Basics

- Place columns inside `<div class="row"></div>` for each row.
- Set each column width using class prefixes:
 - `.col-xs-*`, `.col-sm-*`, `.col-md-*`, `.col-lg-*`

	Extra small devices Phones (<768px)	Small devices Tablets (≥768px)	Medium devices Desktops (≥992px)	Large devices Desktops (≥1200px)
Grid behavior	Horizontal at all times	Collapsed to start, horizontal above breakpoints		
Container width	None (auto)	750px	970px	1170px
Class prefix	<code>.col-xs-</code>	<code>.col-sm-</code>	<code>.col-md-</code>	<code>.col-lg-</code>
# of columns	12			
Column width	Auto	60px	78px	95px
Gutter width	30px (15px on each side of a column)			

Example

Create a row of columns of width 3, 4, and 5.

```
<div class="bs-docs-grid">  
  <div class="row show-grid">  
    <div class="col-md-3">.col-md-3</div>  
    <div class="col-md-4">.col-md-4</div>  
    <div class="col-md-5">.col-md-5</div>  
  </div>  
</div>
```

.col-md-3	.col-md-4	.col-md-5
-----------	-----------	-----------

Grid System - Multiple Devices

- Sometimes columns are stacked vertically on smaller screens
- Use `col-xs-*`, `col-sm-*`, `col-md-*` together to set different column widths for different screen sizes

Example

Allow 2 columns in a row for mobile, 3 for desktop

```
<div class="row show-grid">  
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>  
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>  
  <div class="col-xs-6 col-md-4">.col-xs-6 .col-md-4</div>  
</div>
```

On Desktop:

.col-xs-6 .col-md-4	.col-xs-6 .col-md-4	.col-xs-6 .col-md-4
---------------------	---------------------	---------------------

On Mobile:

.col-xs-6 .col-md-4	.col-xs-6 .col-md-4
.col-xs-6 .col-md-4	

Grid System – Column Resets

Sometimes at breakpoints, columns may be aligned in wrong places.

```
<div class="row show-grid">
  <div class="col-xs-6 col-sm-3">
    .col-xs-6 .col-sm-3
  <br>
  Resize your viewport or check it out on your phone for an example.
</div>
<div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>
<div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>
<div class="col-xs-6 col-sm-3">.col-xs-6 .col-sm-3</div>
</div>
```

Problem:

.col-xs-6 .col-sm-3 Resize your viewport or check it out on your phone for an example.	.col-xs-6 .col-sm-3
.col-xs-6 .col-sm-3	.col-xs-6 .col-sm-3

Correct Output:

.col-xs-6 .col-sm-3 Resize your viewport or check it out on your phone for an example.	.col-xs-6 .col-sm-3
.col-xs-6 .col-sm-3	.col-xs-6 .col-sm-3

Grid System – Column Resets

Fix

```
<!-- Add the extra clearfix for only the required viewport -->  
<div class="clearfix visible-xs"></div>
```

- `.clearfix` clears `float` CSS attribute
- `visible-xs` is one of many `responsive utilities`
 - `clearfix` is applied to only mobile devices

Responsive utilities

Available classes

Use a single or combination of the available classes for toggling content across viewport breakpoints.

	Extra small devices Phones (<768px)	Small devices Tablets (≥768px)	Medium devices Desktops (≥992px)	Large devices Desktops (≥1200px)
<code>.visible-xs</code>	Visible	Hidden	Hidden	Hidden
<code>.visible-sm</code>	Hidden	Visible	Hidden	Hidden
<code>.visible-md</code>	Hidden	Hidden	Visible	Hidden
<code>.visible-lg</code>	Hidden	Hidden	Hidden	Visible
<code>.hidden-xs</code>	Hidden	Visible	Visible	Visible
<code>.hidden-sm</code>	Visible	Hidden	Visible	Visible
<code>.hidden-md</code>	Visible	Visible	Hidden	Visible
<code>.hidden-lg</code>	Visible	Visible	Visible	Hidden

Grid System - Offsetting

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

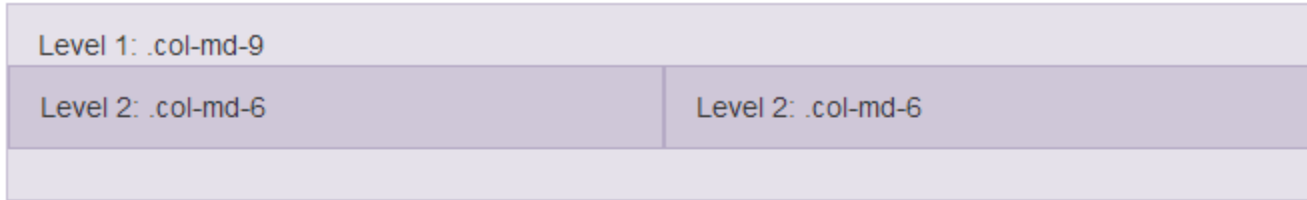
Example

```
<div class="row show-grid">  
  <div class="col-md-3 col-md-offset-3">.col-md-3 .col-md-offset-3</div>  
  <div class="col-md-3 col-md-offset-3">.col-md-3 .col-md-offset-3</div>  
</div>
```

.col-md-3 .col-md-offset-3

.col-md-3 .col-md-offset-3

Grid System – Nested columns



```
<div class="row">  
  <div class="col-md-9">  
    Level 1: .col-md-9  
    <div class="row">  
      <div class="col-md-6">  
        Level 2: .col-md-6  
      </div>  
      <div class="col-md-6">  
        Level 2: .col-md-6  
      </div>  
    </div>  
  </div>  
</div>
```

Nested columns add up to its parent or less

Typography - Headings

Supports `<h1>` to `<h6>`, same as HTML

EXAMPLE

h1. Bootstrap heading

Semibold 36px

h2. Bootstrap heading

Semibold 30px

h3. Bootstrap heading

Semibold 24px

h4. Bootstrap heading

Semibold 18px

h5. Bootstrap heading

Semibold 14px

h6. Bootstrap heading

Semibold 12px

Typography - Alignment

EXAMPLE

Left aligned text.

Center aligned text.

Right aligned text.

Justified text.

```
<p class="text-left">Left aligned text.</p>  
<p class="text-center">Center aligned text.</p>  
<p class="text-right">Right aligned text.</p>  
<p class="text-justify">Justified text.</p>
```

Typography – Addresses

EXAMPLE

Twitter, Inc.

795 Folsom Ave, Suite 600

San Francisco, CA 94107

P: (123) 456-7890

Full Name

first.last@example.com

```
<address>
  <strong>Twitter, Inc.</strong><br>
  795 Folsom Ave, Suite 600<br>
  San Francisco, CA 94107<br>
  <abbr title="Phone">P:</abbr> (123) 456-7890
</address>

<address>
  <strong>Full Name</strong><br>
  <a href="mailto:#">first.last@example.com</a>
</address>
```

append **
** to preserve formatting

Typography - Description

EXAMPLE

Description lists

A description list is perfect for defining terms.

Euismod

Vestibulum id ligula porta felis euismod semper eget lacinia odio sem nec elit.

Donec id elit non mi porta gravida at eget metus.

Malesuada porta

Etiam porta sem malesuada magna mollis euismod.

```
<dl>
  <dt>...</dt>
  <dd>...</dd>
</dl>
```

Typography – Horizontal Description

EXAMPLE

Description lists	A description list is perfect for defining terms.
Euismod	Vestibulum id ligula porta felis euismod semper eget lacinia odio sem nec elit. Donec id elit non mi porta gravida at eget metus.
Malesuada porta	Etiam porta sem malesuada magna mollis euismod.
Felis euismod semp...	Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus.

```
<dl class="dl-horizontal">  
  <dt>...</dt>  
  <dd>...</dd>  
</dl>
```

Notice truncation for “Felis euismod semp...”

Forms

Specify class `.form-control` for `<input>`, `<textarea>` and `<select>` to set its width to 100%

Wrap labels and controls in `.form-group` for optimum spacing

Forms - Basics

Using Bootstrap

EXAMPLE

Email address

Password

File input

No file chosen

Example block-level help text here.

☐ Check me out

Source

```
<form role="form">
  <div class="form-group">
    <label for="exampleInputEmail1">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail1" placeholder="Enter email">
  </div>
  <div class="form-group">
    <label for="exampleInputPassword1">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword1" placeholder="Password">
  </div>
  <div class="form-group">
    <label for="exampleInputFile">File input</label>
    <input type="file" id="exampleInputFile">
    <p class="help-block">Example block-level help text here.</p>
  </div>
  <div class="checkbox">
    <label>
      <input type="checkbox"> Check me out
    </label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

Forms - Controls

Supports HTML form elements

- `<input>`
- `<textarea>`
- `<input type="checkbox">`
- `<input type="radio">`
- `<select>`
- ...

Forms - inlining

add **form-inline** to left align your form elements

EXAMPLE



Enter email Password ☐ Remember me Sign in

```
<form class="form-inline" role="form">
  <div class="form-group">
    <label class="sr-only" for="exampleInputEmail2">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail2" placeholder="Enter email">
  </div>
  <div class="form-group">
    <label class="sr-only" for="exampleInputPassword2">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword2" placeholder="Password">
  </div>
  <div class="checkbox">
    <label>
      <input type="checkbox"> Remember me
    </label>
  </div>
  <button type="submit" class="btn btn-default">Sign in</button>
</form>
```

Forms – Validation states

3 types of states

- Warning: `.has-warning`
- Error: `.has-error`
- Success: `.has-success`

To use validation states:

1. Set one of has-success, has-error, has-warning to parent
2. Set control-label for label
3. Set form-control to other form controls

Example

```
<form role="form">
  <div class="form-group has-success">
    <label class="control-label" for="inputSuccess1">Input with success</label>
    <input type="text" class="form-control" id="inputSuccess1">
  </div>
  <div class="form-group has-warning">
    <label class="control-label" for="inputWarning1">Input with warning</label>
    <input type="text" class="form-control" id="inputWarning1">
  </div>
  <div class="form-group has-error">
    <label class="control-label" for="inputError1">Input with error</label>
    <input type="text" class="form-control" id="inputError1">
  </div>
</form>
```

EXAMPLE

Input with success

Input with warning

Input with error

Forms – Validation states with icons


To add feedback icons

1. add **has-feedback** class to parent
2. insert `` specifying glyphicon for the control's feedback


```
<span class="glyphicon glyphicon-ok form-control-feedback"></span>
```

EXAMPLE

Input with success



Input with warning



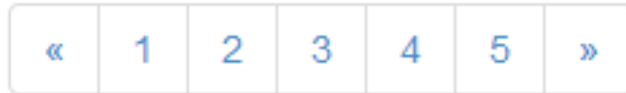
Input with error



Pagination

Great for apps supporting search functionalities.

EXAMPLE



Simply add **pagination** class to ul element. `<ul class="pagination">`

Each page is a list of href element. e.g. `1`

Use **pagination-lg** or **pagination-sm** to size page button sizes

Pager

Simpler version with “**Previous**” or “**Next**” buttons

EXAMPLE

[Previous](#)

[Next](#)

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

You can also align buttons

EXAMPLE

[← Older](#)

[Newer →](#)

```
<ul class="pager">
  <li class="previous"><a href="#">&larr; Older</a></li>
  <li class="next"><a href="#">Newer &rarr;</a></li>
</ul>
```


Navbar

Class options for navbar

- .navbar-inverse
- .navbar-form
- .navbar-left
- .navbar-right
- .navbar-fixed-top
- .navbar-fixed-bottom
- .navbar-static-top

Navbar – Search bar

```
<form class="navbar-form navbar-right" role="search">  
  <div class="form-group">  
    <input type="text" class="form-control" placeholder="Search">  
  </div>  
  <button type="submit" class="btn btn-default">Submit</button>  
</form>
```

Navbar – inversed color + fixed to top

```
<div class="navbar navbar-inverse navbar-fixed-top">
  <div class="container">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="#">Brand Button</a>
    </div>
    <div class="collapse navbar-collapse">
      <ul class="nav navbar-nav">
        <li class="active"><a href="#button1">Button1</a></li>
        <li><a href="#button2">Button2</a></li>
        <li><a href="#button3">Button3</a></li>
      </ul>
    </div><!--/.nav-collapse -->
  </div>
</div>
```

Brand Button

Button1

Button2

Button3

Dropdown ▾

Search

Submit

There are still many more to learn!

- Go over <http://getbootstrap.com/getting-started/#examples>
- Playground for bootstrap <http://www.bootply.com/>