



## Chapter 2

# Bootstrap

---

CT313H – WEB TECHNOLOGY

# Objective

---

Introduce **Bootstrap** and  
use Bootstrap to create **web layouts**

# Content

---

- Bootstrap introduction
- Bootstrap grid system
- Bootstrap components
- Bootstrap customization

# Bootstrap Introduction

# What is Bootstrap?

---



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

[Download Bootstrap](#)

Currently v3.3.5

# Bootstrap Advantages

---

- ✓ **Easy to use:** bases on HTML, CSS, JavaScript  
=> anybody with basic knowledge can approach easily
- ✓ **Responsive:** adjusts the webpage layout accordingly to the screen resolution (PC, tablet,...)
- ✓ **Browser compatibility:** compatible with almost all the latest version browsers
- ✓ **Tools:** provides various tools to create web layouts flexibly and easily

CSS requires carefulnesss and various skills  
Design responsive webpages with CSS is a big chalenge

# How to use Bootstrap?

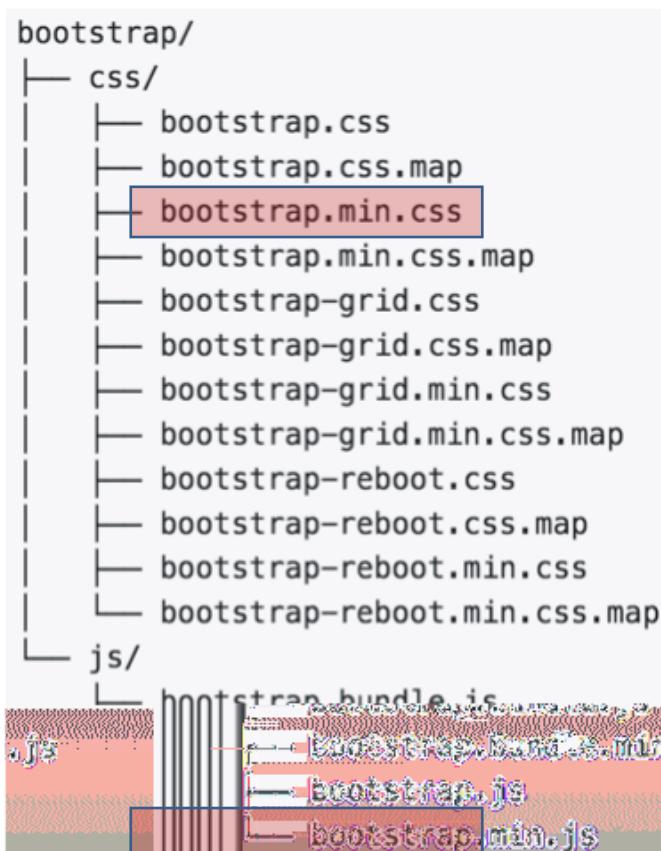
---

## 1) Download Bootstrap (<http://getbootstrap.com>):

- **Precompiled Bootstrap:** for using only without modifications
- **Bootstrap source code:** including compiled CSS, JS, less source codes (CSS pre-processors) and documentations  
⇒ customizability
- Cons:
  - Can work in offline mode
- Pros:
  - May be slower than using CDNs (Content Delivery Networks) as CDNs is optimized for sharing

# How to use Bootstrap?

- Structure of download files:



CSS files	Layout	Content	Components	Utilities
bootstrap.css bootstrap.min.css	Included	Included	Included	Included
bootstrap-grid.css bootstrap-grid.min.css	Only grid system	Not included	Not included	Only flex utilities
bootstrap-reboot.css bootstrap-reboot.min.css	Not included	Only Reboot	Not included	Not included

JS files	Popper	jQuery
bootstrap.bundle.js bootstrap.bundle.min.js	Included	Not included
bootstrap.js bootstrap.min.js	Not included	Not included

❖ 7..46489: &C-48. HG+4C. -& ↴%. 3&4. &G6)&7..46489: ;

% . 3&4. &56)&7..46489: ;

---

<!-- Latest compiled and minified CSS -->

<**link** 8)/=>641/)6, ))4>  
, 8)?=>\*..46489:@+66@\*..46489:ABC-A+66>D&

<!-- jQuery library -->

<**script** 68+=>[FG)81@[FG)81ABC-AE6>D&<@**script**D&

<!-- Latest compiled JavaScript -->

<**script** 68+=>\*..46489:@E6@\*..46489:ABC-AE6>D&<@**script**D

# How to use Bootstrap?

---

## 2) Use a Bootstrap CDN (Content Delivery Network):

- Create links to Bootstrap framework
- Pros:
  - Fast: Bootstrap framework will be downloaded from the nearest CDN
  - Bootstrap framework can be cached
- Cons:
  - Cannot work in offline mode
  - Can be restricted by the CDN policies

# MaxCDN

---

<!-- Latest compiled and minified CSS -->

```
<link rel="stylesheet"  
      href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
```

<!-- jQuery library -->

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js">  
</script>
```

<!-- Latest compiled JavaScript -->

```
<script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js">  
</script>
```

# Create Webpage with Bootstrap

---

## 1) Add the HTML5 doctype:

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

## 2) Set mobile-first setting (in <head>):

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

# Create Webpage with Bootstrap

## 3) Create links to Bootstrap framework (in <head>)

Use the downloaded files or CDN

## 4) Create containers (in <body>):

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

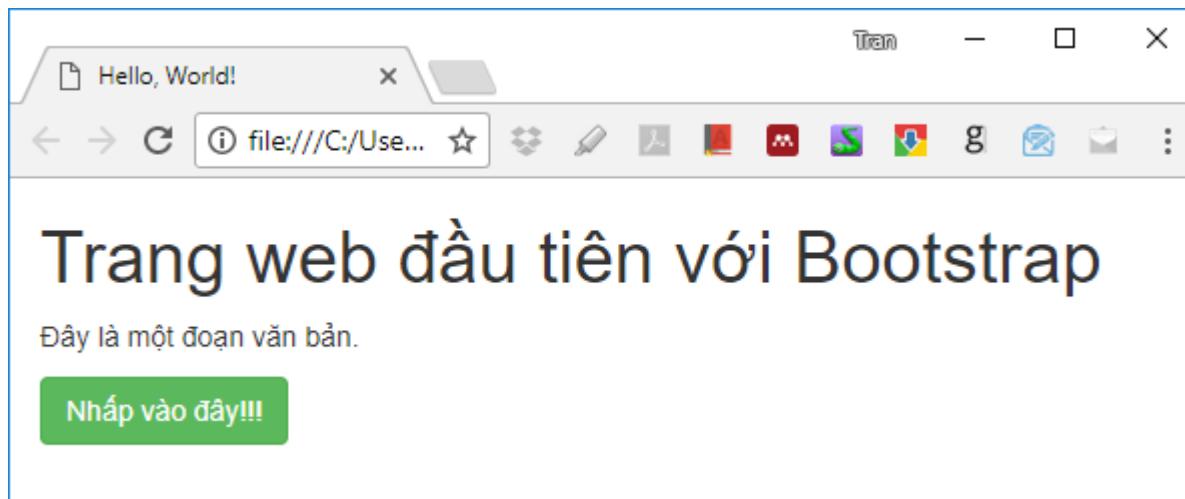


# Hello World – Bootstrap!

---

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Hello, World!</title>
  <link rel="stylesheet"
        href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
</head>
<body>
  <div class="container">
    <h1>Trang web đầu tiên với Bootstrap</h1>
    <p>Đây là một đoạn văn bản.</p>
    <button class="btn btn-success">Nhấp vào đây!!!</button>
  </div>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
  <script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
</body>
</html>
```

# Hello Wo ld Boo ap



# Bootstrap Features

---

- Theme
- Responsive layout
- Grid system
- UI components



# Bootstrap Theme

# Bootstrap Theme

- Theme:

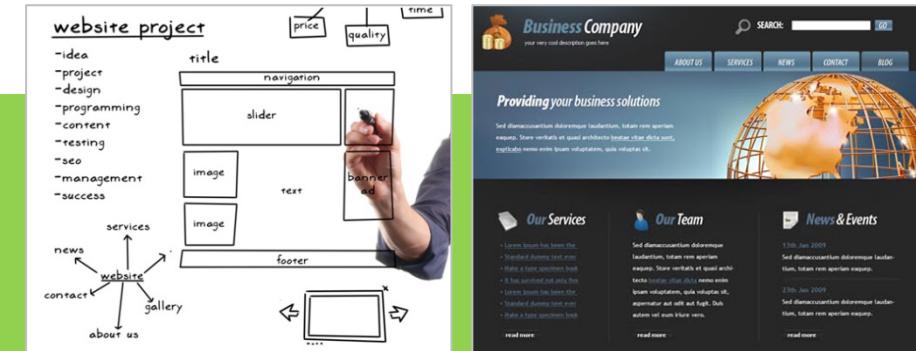
- UI definitions, separated from HTML code
- Helps to change UI elements' look easily
- A trends in web UI design
- Bootstrap theme: <https://bootswatch.com>,  
<http://www.free-css.com/template-categories/bootstrap> , etc.

## Layout vs. Template vs. Theme?

**Layout:** arrangement of elements

**Template:**

- Pre-design webpages
- Users can use to “plug-in” their own contents (texts, images, etc.)
- Save time and efford in designing a website



# Bootstrap Theme

The screenshot shows a web browser window with the title "Bootstrap Case". The address bar displays the URL "file:///E:/Dropbox/Bo%20mon/CT275%20-%20Cong%20nghe%20Web/examples/bootstrap/demo-themes.html". The page content is a dark-themed template from Bootstrap's documentation. It features a large, bold, white font header "My first Bootstrap website!". Below it is a subtitle "This page will grow as we add more and more components from Bootstrap...". A blue search bar with a magnifying glass icon and the word "Search" is positioned below the subtitle. At the bottom of the page, there are three columns of text and a navigation menu. The left column contains placeholder text: "Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.". The middle column contains: "Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.". The right column contains: "Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.". To the right of the right column, a vertical navigation menu is shown with four items: "Home" (highlighted in orange), "Menu 1", "Menu 2", and "Menu 3".

Bootstrap Website Home Page 1 Page 2 Page 3

# My first Bootstrap website!

This page will grow as we add more and more components from Bootstrap...

Search

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.

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.

Home

Menu 1

Menu 2

Menu 3

# Bootstrap Theme

The screenshot shows a web browser window with the title "Bootstrap Case". The address bar displays the URL "file:///E/Dropbox/Bo%20mon/CT275%20-%20Cong%20nghe%20Web/examples/bootstrap/demo-themes.html". The page content is a Bootstrap template. At the top, there's a navigation bar with tabs: "BOOTSTRAP WEBSITE", "Home", "Page 1", "Page 2", and "Page 3". Below the navigation bar is a large, bold heading "My first Bootstrap website!". A subtext below it reads "This page will grow as we add more and more components from Bootstrap...". There is a blue search button labeled "Search". In the footer area, there are three columns of placeholder text (Lorem ipsum dolor sit amet...) and a red navigation bar with links: "Home", "Menu 1", and "Menu 2".

Bootstrap Website

file:///E/Dropbox/Bo%20mon/CT275%20-%20Cong%20nghe%20Web/examples/bootstrap/demo-themes.html

BOOTSTRAP WEBSITE Home Page 1 Page 2 Page 3

# My first Bootstrap website!

This page will grow as we add more and more components from Bootstrap...

Search

Home

Menu 1

Menu 2

Footer content (repeated):

1. Lorem ipsum dolor sit amet, do eiusmod tempor incididunt ut labore et dolore magna aliqua.

2. Ut enim ad minim veniam, quis laboris nisi ut aliquip ex ea commodo consequat.

3. Sed ut perspiciatis unde omnis

# Bootstrap Theme

---

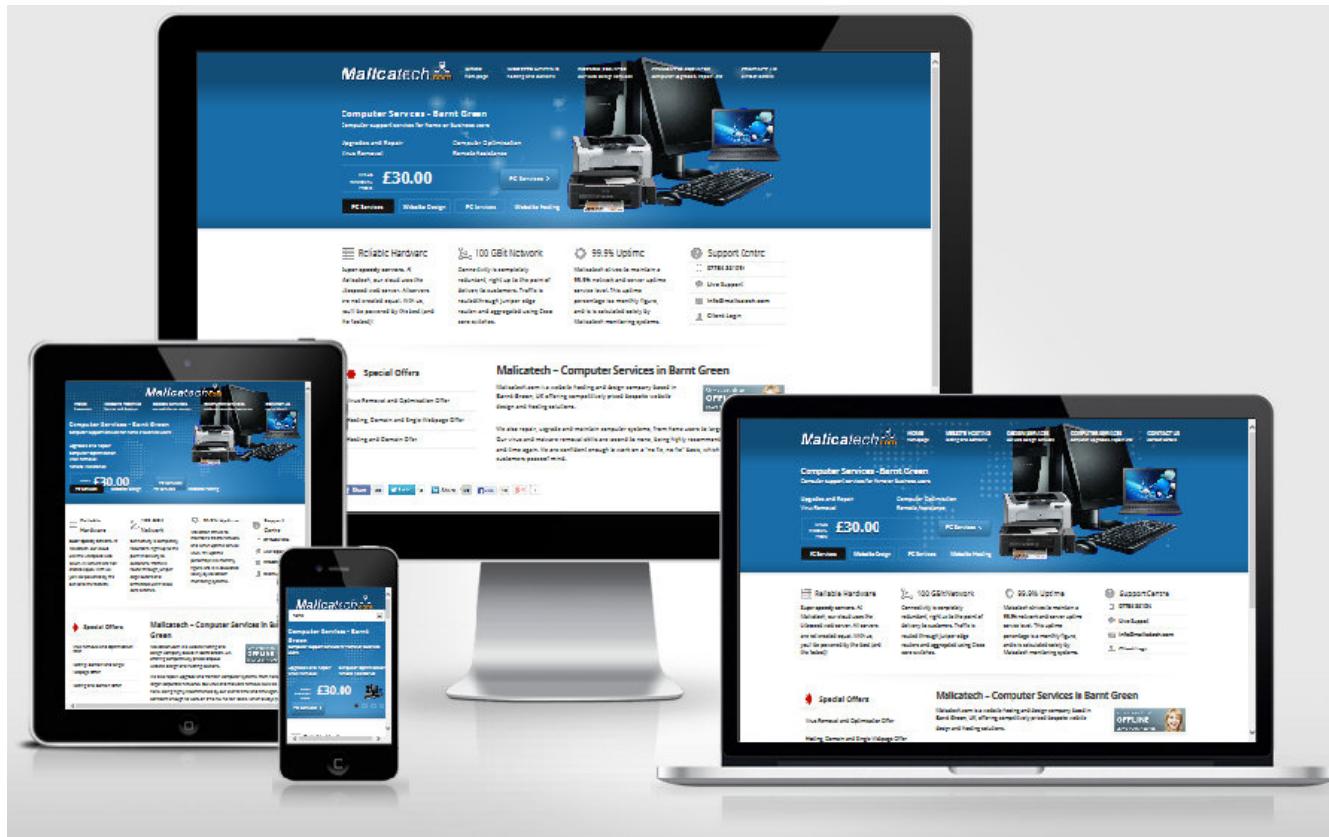
```
<!DOCTYPE html>
<html>
  <head>
    <title>Bootstrap Theme</title>
    <meta charset="utf-8">
    <link rel="stylesheet" href="themes/journal/bootstrap.min.css">
  </head>
  <body>
    <!--nội dung-->
  </body>
</html>
```

```
<!DOCTYPE html>
<html>
  <head>
    <title>Bootstrap Theme</title>
    <meta charset="utf-8">
    <link rel="stylesheet" href="themes/superhero/bootstrap.min.css">
  </head>
  <body>
    <!-- nội dung -->
  </body>
</html>
```

# Responsive Layout

# What is responsive layout?

- A layout that can adapt to the device resolution



# Mobile-first design

---

- New approach of design strategy “**responsive design**”
- Start sketching and prototyping on small screen first
- Then, work your way up to larger screens  
*(mobile devices is prioritized over desktop)*



Source: [knowledgecentre.ubiquity.co.nz](http://knowledgecentre.ubiquity.co.nz)

# Grid System

# Grid layout

---

- Bootstrap's grid system uses a series of **containers**, **rows**, and **columns** to layout and align content
- Bootstrap's grid system allows up to **12 columns** across the page (if you do not want to use all 12 columns individually, you can group the columns together to create wider columns)
- The Bootstrap grid system has **4 classes**:
  - : for laptops and desktops - screens  $\geq 1200\text{px}$
  - : for small laptops - screens  $\in [992\text{px}, 1200\text{px})$
  - : for tablets - screens  $\in [768\text{px}, 991\text{px})$
  - : for phones - screens  $< 768\text{px}$

# Grid system

---

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

# Row and columns

---

- Rows are created using class **row**
- Column definition includes three parts: **col-\*-\***
  - **col**: CSS class for columns
  - Grid size: **lg** (large), **md** (medium), **sm** (small), **xs** (extra small)
  - Number of columns: an integer number, total number of columns in a row must be **12** columns
- Example:
  - **col-lg-6**: applied for large grid ( $\geq 1200\text{px}$ ), column width is 6 columns
  - **col-md-4**: applied for medium grid [992 – 1200px), columns width is 4 columns

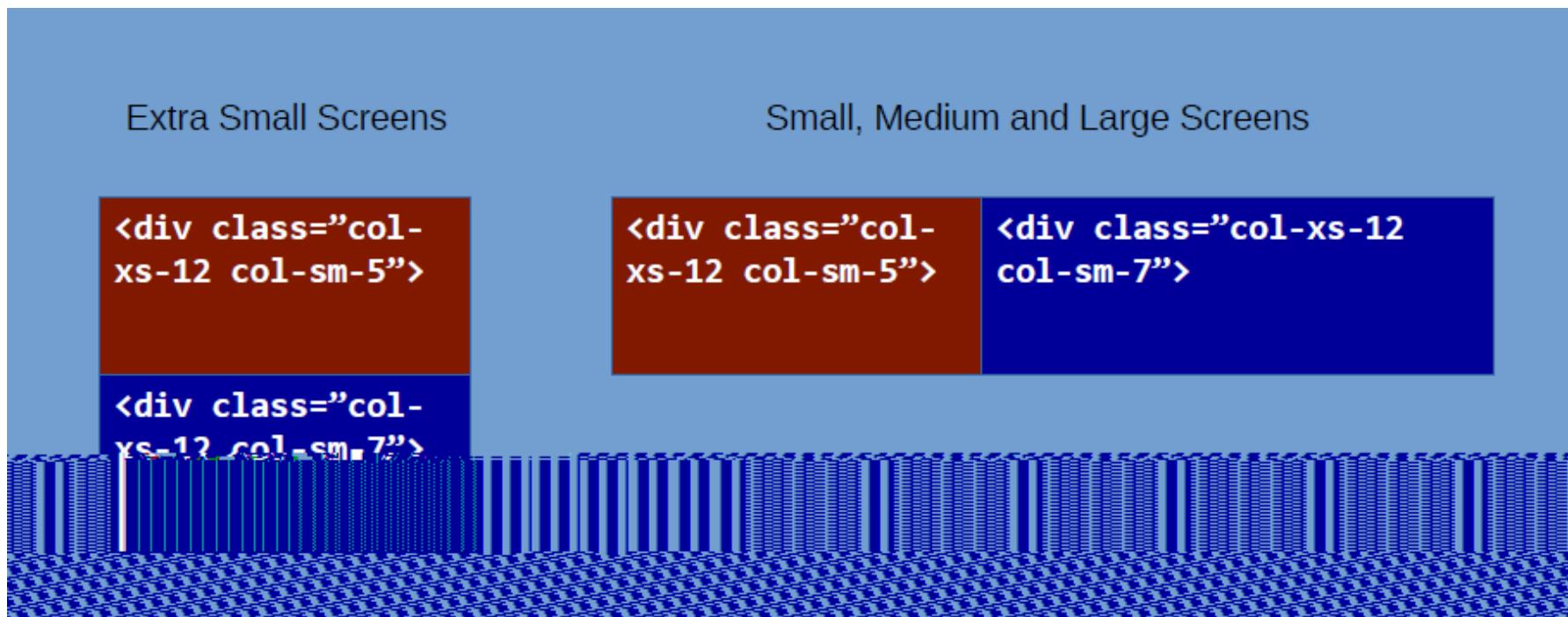
# Rows and columns

---

```
<div class="container">  
  
<div class="row">  
  <div class="col-sm-5">  
    1  2  3  4  5  
  <div class="col-sm-7">  
    6  7  8  9  10  11  12  
</div>
```

# Multiple column definitions

- There may be several definitions for a columns  
⇒ allows to create responsive webpages



# Multiple column definitions

The screenshot shows a browser window titled "Bootstrap Grid Demo". The address bar displays the local file path: "file:///C:/Users/Tran%20Cong%20An/C...". The main content area features a large "Hello World!" heading. Below it, a message encourages users to "Resize the browser window to see the effect.". The page contains several rows of colored boxes demonstrating responsive grid behavior:

- A red row containing ".col-xs-7 .col-sm-6 .col-lg-8" and ".col-xs-5 .col-sm-6 .col-lg-4".
- A pink row containing ".col-xs-6 .col-sm-8 .col-lg-10" and ".col-xs-6 .col-sm-4 .col-lg-2".
- A light blue row containing ".col-xs-6" and ".col-xs-6".

# Multiple column definitions

---

```
<div class="container-fluid">
<h1>Hello World!</h1>
<p>Resize the browser window to see the effect.</p>
<div class="row" style="margin-top:5px;">
    <div class="col-xs-7 col-sm-6 col-lg-8"
        style="background-color:red;">.col-xs-7 .col-sm-6 .col-lg-8</div>
    <div class="col-xs-5 col-sm-6 col-lg-4"
        style="background-color:lavender;">.col-xs-5 .col-sm-6 .col-lg-4</div>
</div>
<div class="row" style="margin-top:5px;">
    <div class="col-xs-6 col-sm-8 col-lg-10"
        style="background-color:lavenderblush;">.col-xs-6 .col-sm-8 .col-lg-10</div>
    <div class="col-xs-6 col-sm-4 col-lg-2"
        style="background-color:lightgrey;">.col-xs-6 .col-sm-4 .col-lg-2</div>
</div>
<div class="row" style="background-color:lightcyan; margin-top:5px;">
    <div class="col-xs-6">.col-xs-6</div>
    <div class="col-xs-6">.col-xs-6</div>
</div>
</div>
```

Web browser will adjust the webpage layout  
accordingly to the screen resolution

# Exercise

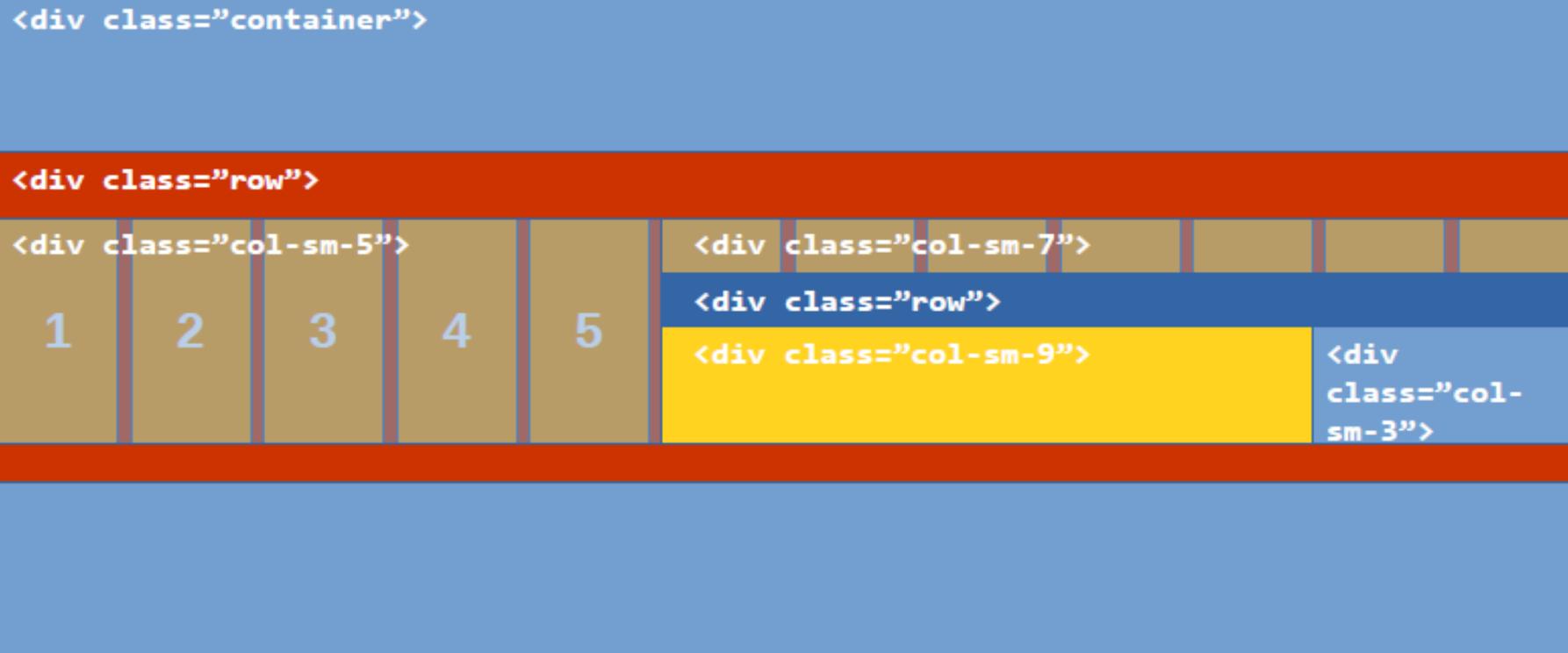
---

- How to design a webpage with the following layout:
  - 1 column on phones
  - 2 columns on tablets and laptops
  - 4 columns on desktop PCs
- Suggested solution:



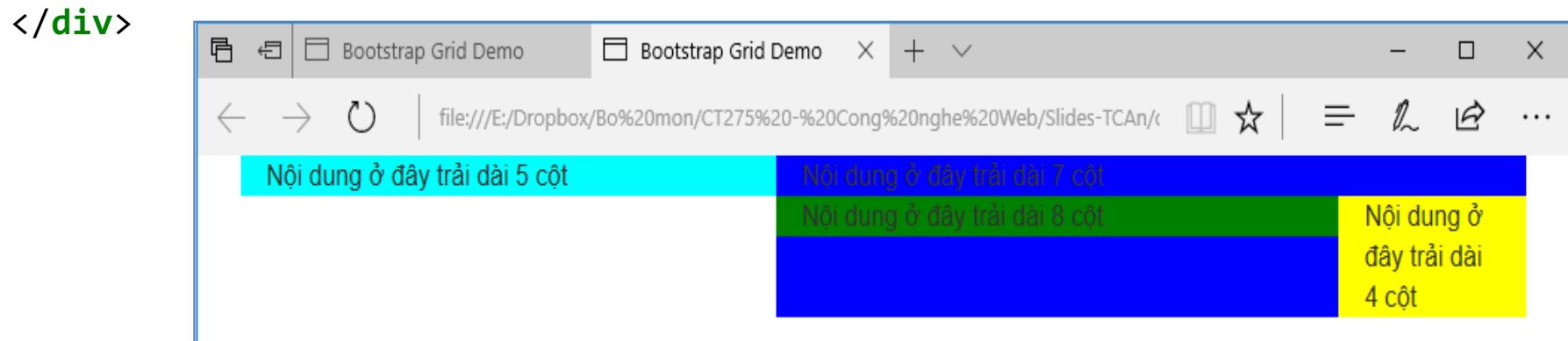
# Nested columns

- Columns in columns – create rows inside a column
- Each nested row has 12 columns





```
<div class="container">
  <div class="row">
    <div class="col-sm-5">Nội dung ở đây trải dài 5 cột</div>
    <div class="col-sm-7">Nội dung ở đây trải dài 7 cột
      <div class="row">
        <div class="col-sm-9">Nội dung ở đây trải dài 8 cột</div>
        <div class="col-sm-3">Nội dung ở đây trải dài 4 cột</div>
      </div>
    </div>
  </div>
</div>
```



# Column positioning

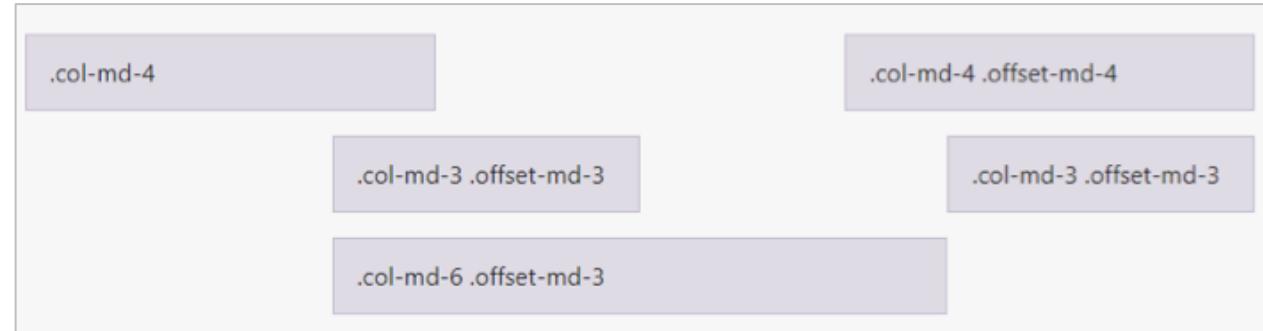
---

- Offset (`col-*-offset-*`):
  - Left alignment the column
  - Example: `col-md-offset-2, col-sm-offset-4,...`
- Pull (`col-* pull-*`):
  - Pull the column to the left
  - Example: `col-lg-pull-3, col-lg-pull-9,...`
- Push (`col-* push-*`):
  - Push the column to the right
  - Example: `col-xs-push-8, col-xs-push-7,...`

# Column positioning

- Coulumn offset example:

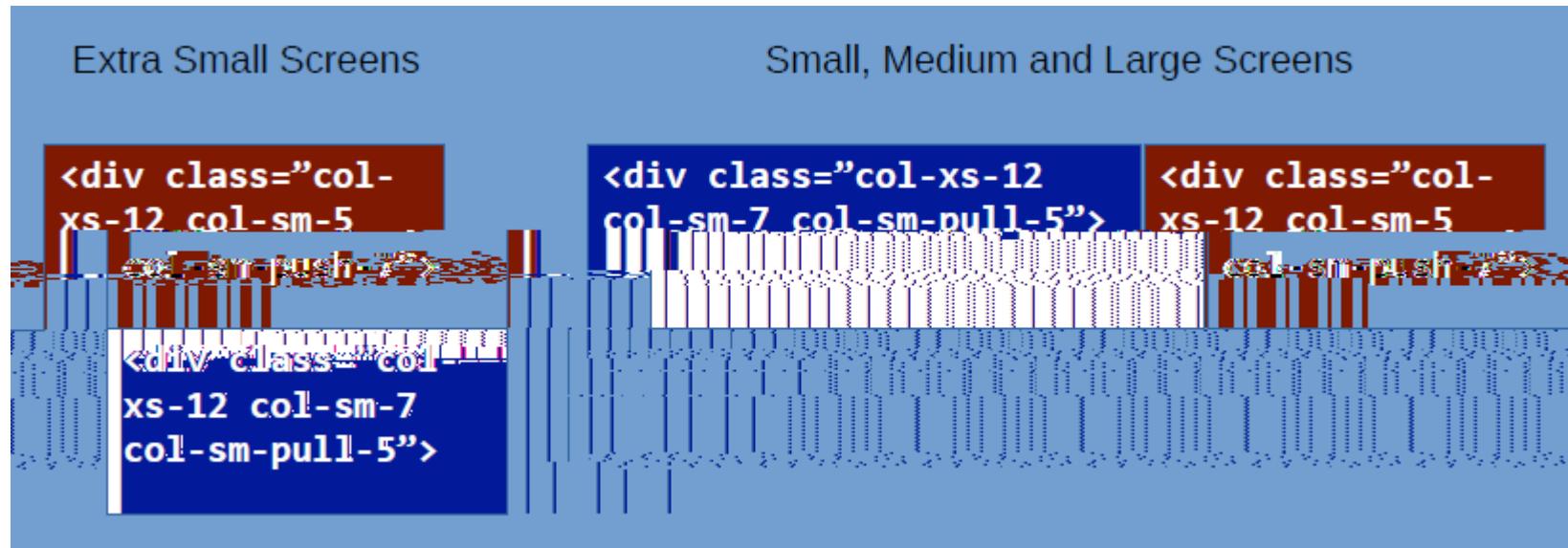
```
<div class="row">
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4 offset-md-4">.col-md-4 .offset-md-4</div>
</div>
<div class="row">
  <div class="col-md-3 offset-md-3">.col-md-3 .offset-md-3</div>
  <div class="col-md-3 offset-md-3">.col-md-3 .offset-md-3</div>
</div>
<div class="row">
  <div class="col-md-6 offset-md-3">.col-md-6 .offset-md-3</div>
</div>
```



# Column positioning

- Column pull and push example:

```
<div class="row">
  <div class="col-xs-12 col-sm-5 col-sm-push-7">...</div>
  <div class="col-xs-12 col-sm-7 col-sm-pull-5">...</div>
</div>
```





- 
- - 
  - - (hidden-{lg, md, sm, xs})
  - - (visible-{lg, md, sm, xs})

# Hide/Show columns

---

<h4>Resize to see how the text changes:</h4>

<h3 class="visible-sm bg-info">Shown only on a SMALL screen.</h3>

<h3 class="visible-md bg-warning">Shown only on a MEDIUM screen.</h3>

<h3 class="visible-lg bg-success">Shown only on a LARGE screen.</h3>

<h3 class="hidden-sm bg-info">Hidden on a SMALL screen.</h3>

<h3 class="hidden-md bg-warning">Hidden on a MEDIUM screen.</h3>

<h3 class="hidden-lg bg-success">Hidden on a LARGE screen.</h3>

Resize to see how the text changes:

Shown only on a LARGE screen.

Hidden on a SMALL screen.

Hidden on a MEDIUM screen.

Resize to see how the text changes:

Shown only on a MEDIUM screen.

Hidden on a SMALL screen.

Hidden on a LARGE screen.

Resize to see how the text changes:

Shown only on a SMALL screen.

Hidden on a MEDIUM screen.

Hidden on a LARGE screen.

# Bootstrap GUI Elements

# Bootstrap GUI elements

---

- Bootstrap provides template for many HTML GUI elements (using HTML + CSS + JS)
  - Modern and beautiful
  - Support theme for easy customization
- To use Bootstrap GUI template:
  - Use the build-in templates provided
  - Customize the provided templates make them suited to the user requirement

# Bootstrap GUI elements

The image displays a variety of Bootstrap GUI elements:

- Buttons:** A row of six buttons labeled "Default", "Primary", "Success", "Info", "Warning", and "Danger".
- Pagination:** A set of numbered buttons from 1 to 5, with arrows for navigation.
- Navigation:** Examples of horizontal navigation bars with items like "Home", "Profile", and "Messages".
- Modals:** Three cards showing "Thumbnail label" placeholder text and two green "Button" buttons.
- Alerts:** A series of colored boxes with alert messages:
  - Green box: "Well done! You successfully read this important alert message."
  - Light blue box: "Heads up! This alert needs your attention, but it's not super important."
  - Yellow box: "Warning! Better check yourself, you're not looking too good."
  - Pink box: "Oh snap! Change a few things up and try submitting again."
- Collapsible Groups:** Three examples of collapsible groups with titles "Collapsible Group Item #1", "Collapsible Group Item #2", and "Collapsible Group Item #3".

# Buttons

---

- Tags: <**a**> , <**button**>, <**input**>
- Create a button:

btn + {  
    btn-default  
    btn-primary  
    btn-success  
    btn-info  
    btn-warning  
    btn-danger  
    btn-link  
} + {  
    btn-lg  
    btn-md  
    btn-sm  
    btn-xs  
} + btn-block

Button type <i>(color)</i>	Size	Block style
-------------------------------	------	-------------

# Labels

---

- Tags: <**span**> or <**div**> and class **label**
- Label style (background and text color) classes:  
`label-default`, `label-primary`, `label-success`,  
`label-info`, `label-warning`, `label-danger`.

<**p**

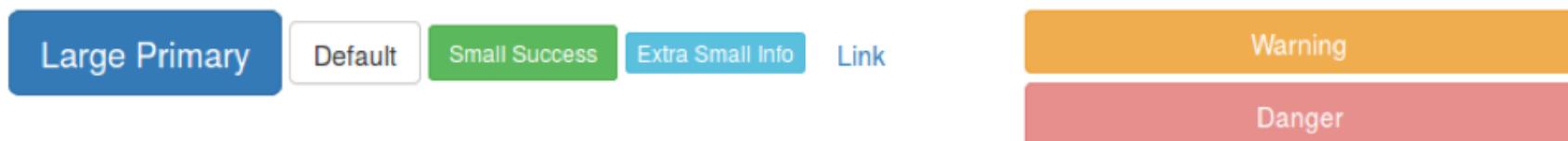
```
<span class="label label-success">Hooray!</span>
</p>
```

Example. **Hooray!**

# Buttons

---

```
<div class="row">
  <div class="col-md-5 col-md-offset-1">
    <button type="button" class="btn btn-primary btn-lg">Large Primary</button>
    <button type="button" class="btn btn-default">Default</button>
    <button type="button" class="btn btn-success btn-sm">Small Success</button>
    <button type="button" class="btn btn-info btn-xs">Extra Small Info</button>
    <button type="button" class="btn btn-link">Link</button>
  </div>
  <div class="col-md-3">
    <button type="button" class="btn btn-warning btn-block">Warning</button>
    <button type="button" class="btn btn-danger btn-block"
           disabled="disabled">Danger</button>
  </div>
</div>
```



# Button Groups

---

- Used to group buttons (no space between buttons)
- Create button group: tag **div** with class **b g**

```
div c a   b   g
  button e b   c a   b   b   i a   A   e   button
  button e b   c a   b   b   i a   Sa   g   button
  button e b   c a   b   b   i a   S   button
div
```



# Forms

---

- There are three form types:
  - Vertical (default)
  - Horizontal
  - Inline
- Create a form:
  - Tag: `<form role="form">`
  - Grouping form elements:  
`<div class="form-group">`
  - Add class `form-control` to all `<input>`, `<textarea>` and `<select>`

# Form – Vertical Form

```
<form role="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>
```

Email:  
Enter email

Password:  
Enter password

Remember me

Submit

# Form – Horizontal Form

---

```
<form class="form-horizontal" role="form">
  <div class="form-group">
    <label for="firstname" class="col-sm-2 control-label">First Name</label>
    <div class="col-sm-10">
      <input type="text" name="firstname" id="firstname"
             class="form-control" placeholder="Enter First Name">
    </div>
  </div>
</form>
```

A screenshot of a web page displaying a horizontal form. The form consists of a single group of fields enclosed in a blue-bordered box. On the left, the label "First Name" is displayed in bold black font. To its right is a text input field with a placeholder "Enter First Name". The input field has a light gray border and a white background.

# Form – Inline Form

```
<form class="form-inline" role="form">
  <div class="form-group">
    <label class="sr-only" for="email">Email address</label>
    <input type="email" class="form-control" id="email" placeholder="Email">
  </div>
  <div class="form-group">
    <label class="sr-only" for="password">Password</label>
    <input type="password" class="form-control" id="password"
           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>
```

The screenshot shows a user interface for a sign-in form. It consists of two horizontal input fields: one for 'Email' and one for 'Password', both with placeholder text. To the right of these fields is a 'checkbox' labeled 'Remember me'. Below the inputs is a 'Sign in' button.

# Input Group

---

- Used to **extend form controls** by adding text, buttons, button groups to either sides of textual inputs, custom selects, and custom file inputs

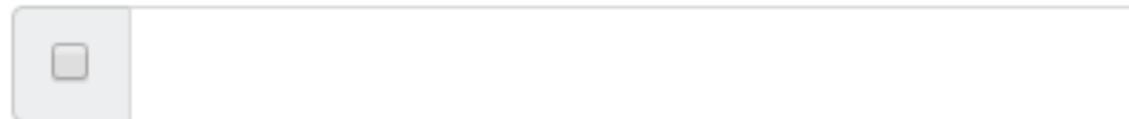
```
<div class="input-group">
  <div class="input-group-addon" style="background-color: #ccc;">(
  <div class="input-group-addon" style="background-color: #ccc;">)
```



# Input Group

---

```
<div class="row">
  <div class="col-lg-6">
    <div class="input-group">
      <span class="input-group-addon">
        <input type="checkbox"
              aria-label="Checkbox for following text input">
      </span>
      <input type="text" class="form-control"
              aria-label="Text input with checkbox">
    </div>
  </div>
</div>
```



# Glyphicon

---

- A collection of **260 icons** in font format (Glyphicons Halflings)  
⇒ can change their size, color,... w/o changing quality
- Usage: <**span** class="glyphicon glyphicon-name"></**span**>
- Example:

```
<button type="button" class="btn btn-info">
  <span class="glyphicon glyphicon-search"></span>Search
</button>
```

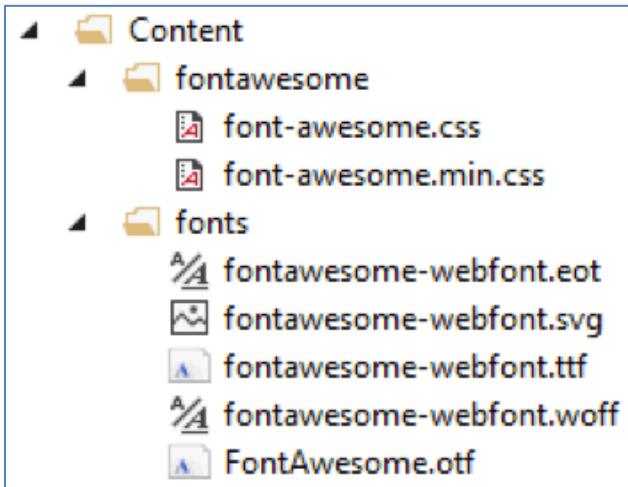


# Font Awesome

---

- A collection of **symbol fonts**
- Usage:
  - Download font files: <http://fontawesome.io/>
  - Use CDN:

```
<link rel="stylesheet"  
      href="https://maxcdn.bootstrapcdn.com/font-awesome/4.4.0/css/font-awesome.min.css" />
```



# Font Awesome

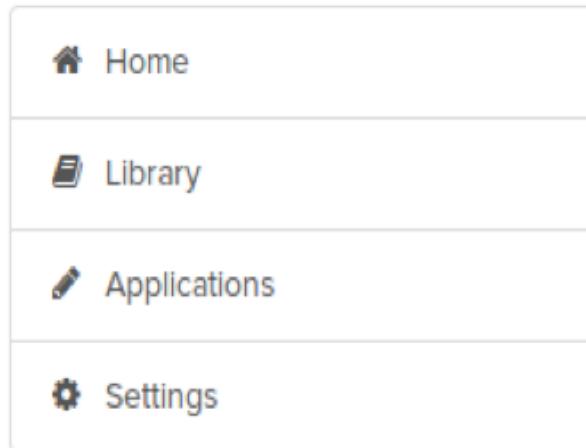
---

- Insert Font Awesome
  - 1) Create link to Font Awesome CSS
  - 2) Identify Font Awesome symbol used
  - 3) Use tag <**span**> or <**i**>:  
`<i class="fa symbol-name"></i>`

# Font Awesome

---

```
<div class="list-group">
  <a class="list-group-item" href="#">
    <i class="fa fa-home"></i>&ampnbsp Home</a>
  <a class="list-group-item" href="#">
    <i class="fa fa-book"></i>&ampnbsp Library</a>
  <a class="list-group-item" href="#">
    <i class="fa fa-pencil"></i>&ampnbsp Applications</a>
  <a class="list-group-item" href="#">
    <i class="fa fa-cog"></i>&ampnbsp Settings</a>
</div>
```



# Tables

---

- Create a table:

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

-

# Tables

---

- Contextual classes used with <td> and <tr>: active, success, info, warning, danger

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

# Tables

---

```
<table class="table">
  <thead>
    <tr> <th>Firstname</th> <th>Lastname</th> <th>Email</th> </tr>
  </thead>
  <tbody>
    <tr>
      <td>Default</td> <td>Defaultson</td> <td>def@somemail.com</td> </tr>
    <tr class="success">
      <td>Success</td> <td>Doe</td> <td>john@example.com</td> </tr>
    <tr class="danger">
      <td>Danger</td> <td>Moe</td> <td>mary@example.com</td> </tr>
    <tr class="info">
      <td>Info</td> <td>Dooley</td> <td>july@example.com</td> </tr>
    <tr class="warning">
      <td>Warning</td> <td>Refs</td> <td>bo@example.com</td> </tr>
    <tr class="active">
      <td>Active</td> <td>Activeson</td> <td>act@example.com</td> </tr>
  </tbody>
</table>
```

# Panels

---

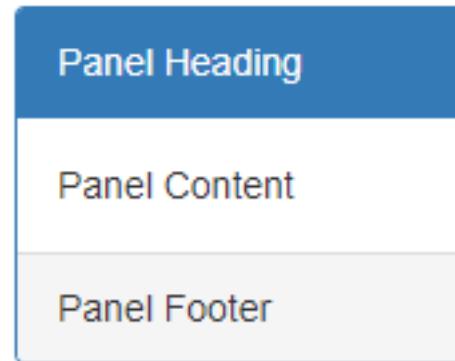
- Bordered box with some paddings around its content
- Tag: <**div** class="panel">
- Related classes:
  - **panel-heading**: create a panel header (highlighted background)
  - **panel-body**: create a panel body
  - **panel-footer**: create a panel footer
  - Contextual classes (change background and border color):  
**panel-default**, **panel-primary**, **panel-success**,  
**panel-info**, **panel-warning**, **panel-danger**



# Panels

---

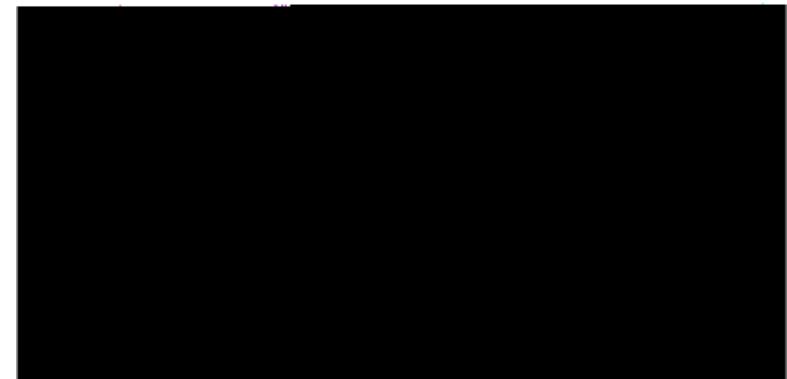
```
<div class="panel panel-primary">  
  <div class="panel-heading">Panel Heading</div>  
  <div class="panel-body">Panel Content</div>  
  <div class="panel-footer">Panel Footer</div>  
</div>
```



# Panels

---

```
<div class="panel panel-primary">
  <div class="panel-heading">
    <h3 class="panel-title">Facts At a Glance</h3>
  </div>
  <div class="panel-body">
    <dl class="dl-horizontal">
      <dt>Started</dt>
      <dd>3 Feb. 2013</dd>
      <dt>Major Stake Holder</dt>
      <dd>HK Fine Foods</dd>
      <dt>Last Year's Turnover</dt>
      <dd>$1,250,375</dd>
      <dt>Employees</dt>
      <dd>40</dd>
    </dl>
  </div>
</div>
```



# Wells

---

- Well class adds rounded border around an element with a gray background color and some padding

```
<div class="well">
  <blockquote>
    <p>You better cut the pizza in four pieces because
       I'm not hungry enough to eat six.</p>
    <footer><cite title="Source Title">The Wit and Wisdom of Yogi
          Berra, P. Pepe, Diversion Books, 2014</cite>
    </footer>
  </blockquote>
</div>
```

You better cut the pizza in four pieces because I'm not hungry enough to eat six.

— *The Wit and Wisdom of Yogi Berra, P. Pepe, Diversion Books, 2014*

# Images

---

- Bootstrap classes for <**img**> tag:
  - **img-responsive**: auto adjusts image size to fit the screen
  - **img-rounded**: adds rounded corners to an image
  - **img-circle**: shapes the image to a circle
  - **img-thumbnail**: shapes the image to a thumbnail

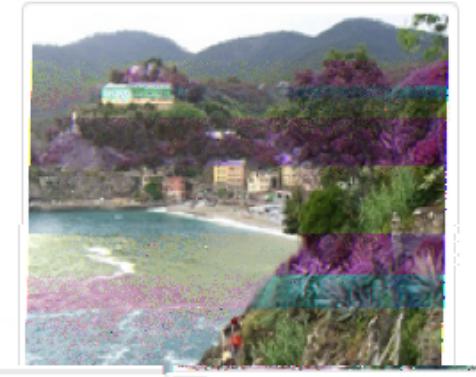
Rounded Corners:



Circle:

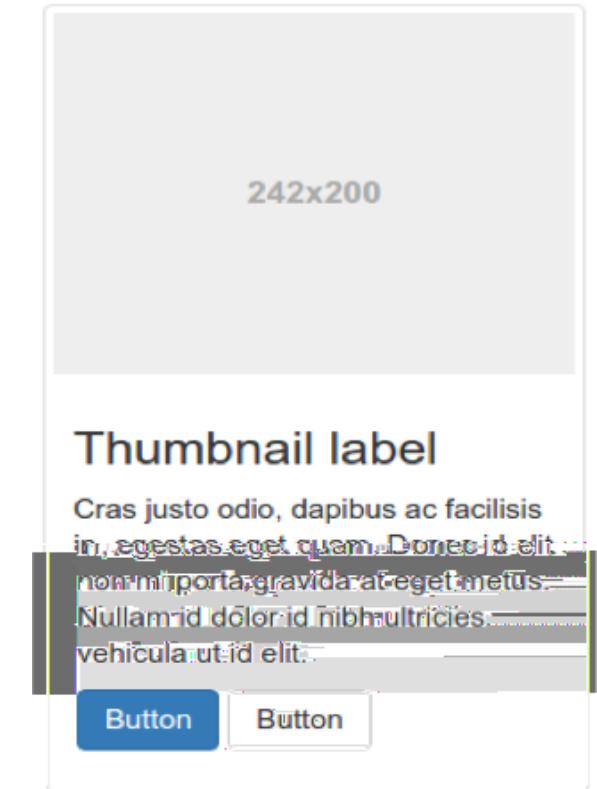


Thumbnail:



# Images

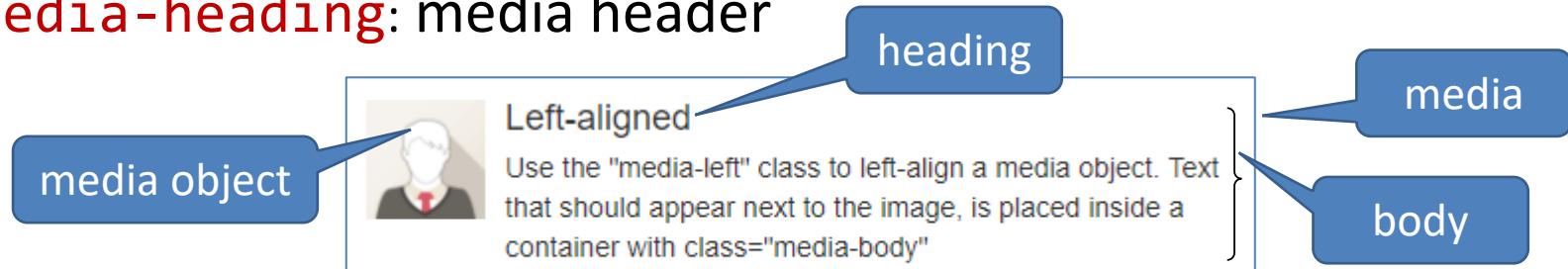
```
<div class="thumbnail">
  
  <div class="caption">
    <h3>Thumbnail label</h3>
    <p>...</p>
    <p>
      <a href="#" class="btn btn-primary">
        Button
      </a>
      <a href="#" class="btn btn-default">
        Button
      </a>
    </p>
  </div>
</div>
```



# Media Objects

---

- Align media objects (images or videos) to the left or to the right of some content
- Create a media object: <**div** class="media">
- Related classes:
  - **media-object**: tạo 1 media object
  - **media-left/right**: horizontal alignment
  - **media-top/bottom/middle**: vertical alignment
  - **media-body**: media content (usually text)
  - **media-heading**: media header



# Media Objects

---

```
<div class="media">
  <div class="media-left">
    
  </div>
  <div class="media-body">
    <h4 class="media-heading">Left-aligned</h4>
    <p>Use the "media-left" class to left-align a media object...</p>
  </div>
</div>
```



## Left-aligned

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

# Media Objects

```


a      a
  div a      a
    img      a a a
      div
      div a      a b
        h4 a      a a J D
          small i P F b a           i small h4
          p Media objects can also be nested (a media object inside a media object) p
          div a      a
            div a      a
              img      a a a
                div
                div a      a b
                  h4 a      a a J D
                    small i P F b a           i small h4
                    p>This is a nested media object. p
                    div
                    div
                    div


```



John Doe *Posted on February 19, 2016*

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



John Doe *Posted on February 19, 2016*

This is a nested media object.

# Responsive Embed (Videos/Slideshows)

```
<div class="embed-responsive embed-responsive-16by9">
  <iframe class="embed-responsive-item"
    src="https://www.youtube.com/embed/2P2mXutRdkE">
  </iframe>
</div>
```



# Jumbotrons

---

- Jumbotron: a big box for calling extra attention to some special content or information
- Create a Jumbotron: <**div** class="jumbotron">

```
<div class="jumbotron">
  <h1>Bootstrap Tutorial</h1>
  <p>B      a          a HTML, CSS, a d JS a      ...</p>
</div>
```

## Bootstrap Tutorial

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

# Alerts

---

- Used to create predefined alert messages
- Create an alert: `<div class="alert alert-style">`
- Alert styles: `alert-success`, `alert-info`,  
`alert-warning`, `alert-danger`
- Related classes:
  - `alert-link`: creates “matching colored links”
  - `alert-dismissible`: allows to close an alert
    - Add a link or button with `class="close"` and  
`data-dismiss="alert"` to create a closing button

# Alerts

---

```
<div class="alert alert-danger alert-dismissible">
  <a href="#" class="close" data-dismiss="alert"
    aria-label="close">x</a>
  <strong>Danger!</strong>
  This alert box could indicate a dangerous or
  potentially negative action.
</div>
```

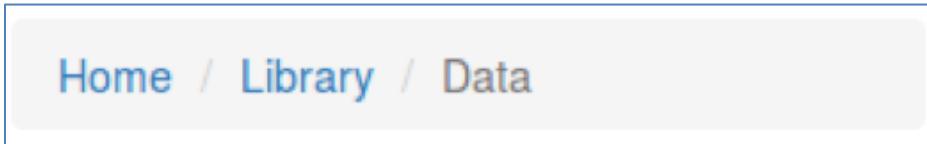
Danger! This alert box could indicate a dangerous or potentially negative action. 

# Breadcrumbs

---

- Used to create an indication of the current page's location within a navigational hierarchy
- Create a breadcrumb: <ol class="breadcrumb">

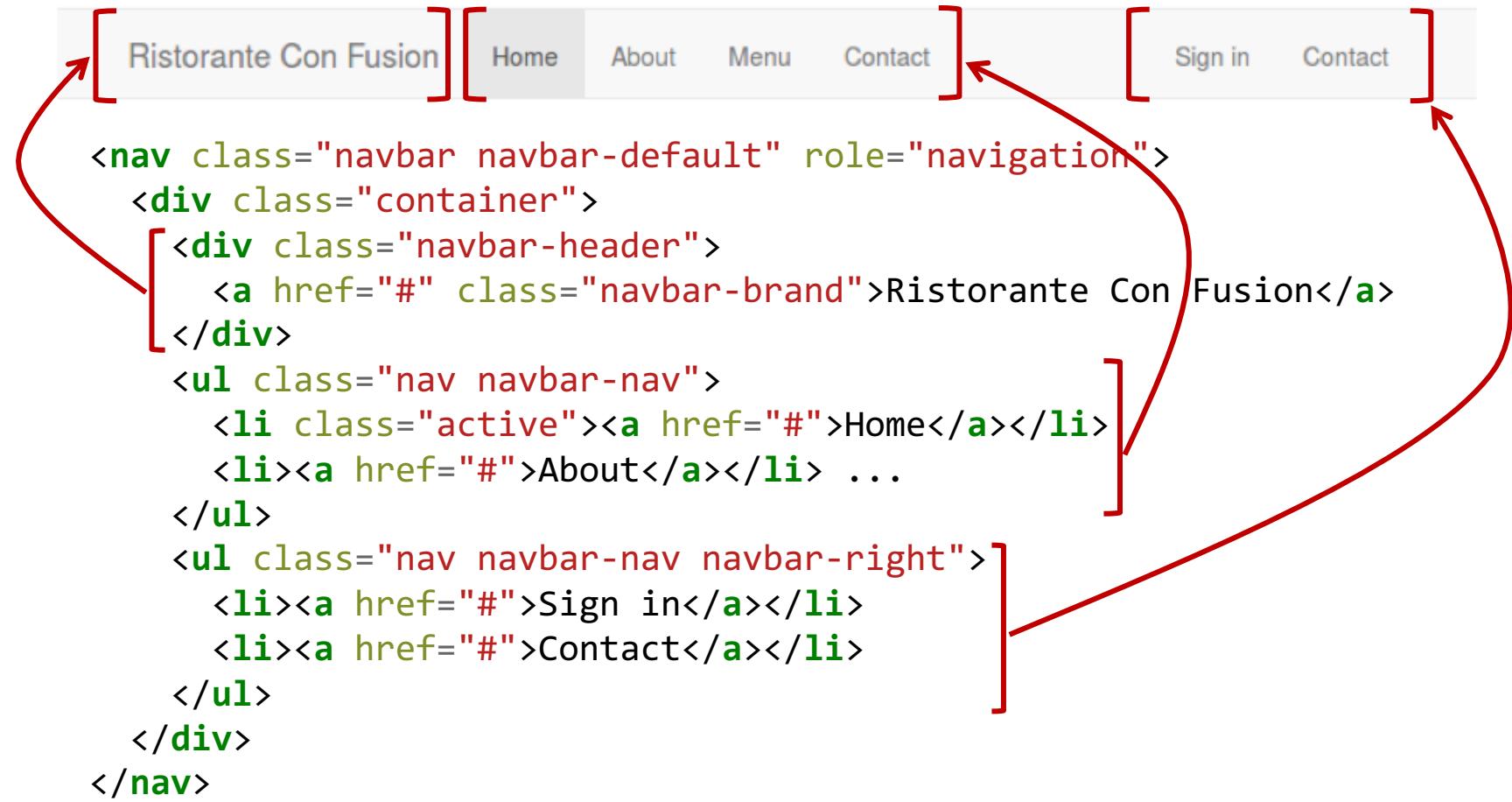
```
<ol class="breadcrumb">
  <li><a href="#">Home</a></li>
  <li><a href="#">Library</a></li>
  <li class="active">Data</li>
</ol>
```



Home / Library / Data

# Navbar

- A collapsible and toggleable navigation bar



# Navbar with Dropdown

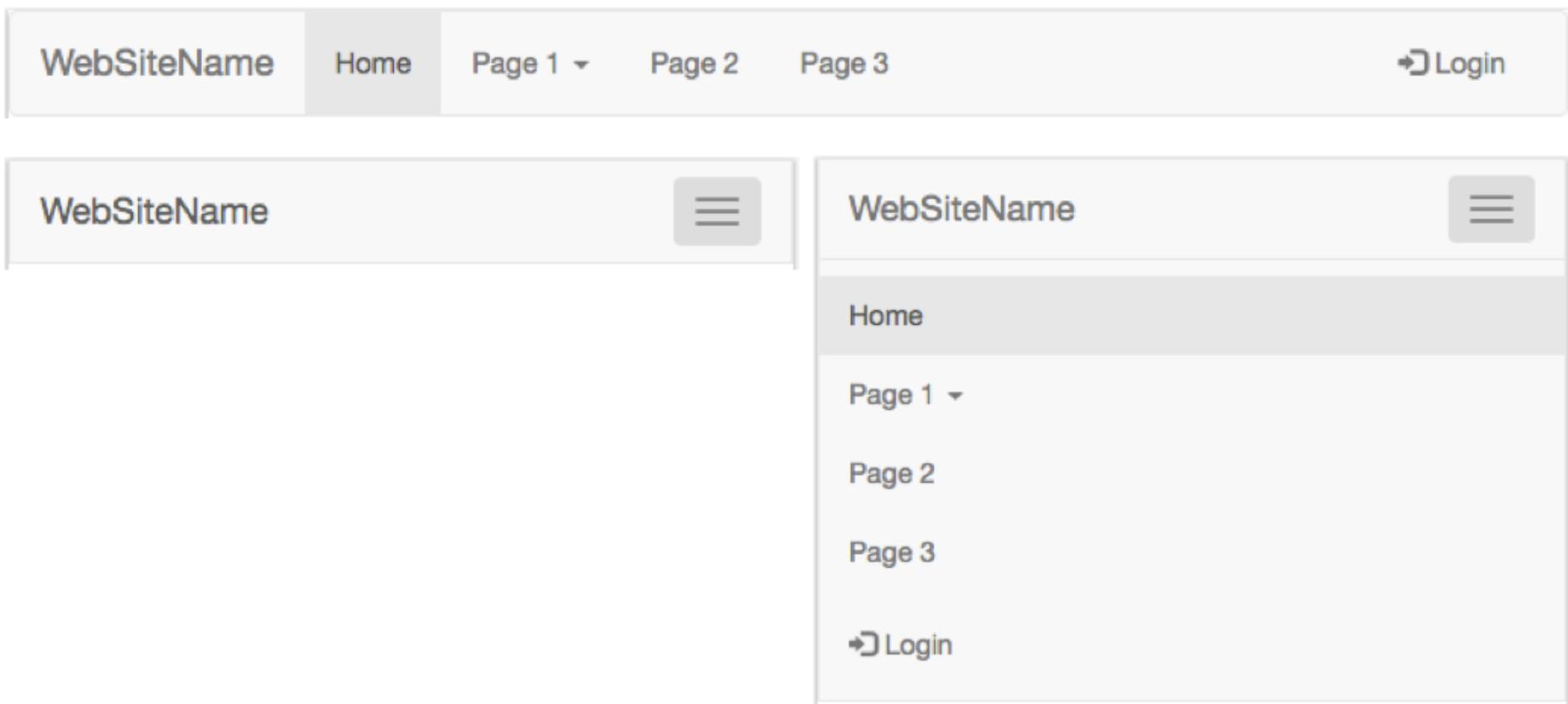
The diagram illustrates a Bootstrap Navbar with a dropdown menu. The navbar contains items: 'WebSiteName', 'Home', 'Page 1', 'Page 2', and 'Page 3'. The 'Page 1' item has a dropdown menu with three items: 'Page 1-1', 'Page 1-2', and 'Page 1-3'. A red bracket highlights the 'Page 1' item, and another red bracket highlights its dropdown menu. Red arrows point from the explanatory code boxes to the corresponding elements in the navbar.

```
<li class="dropdown">
  <a class="dropdown-toggle" data-toggle="dropdown" href="#">Page 1
    <span class="caret"></span>
  </a>
  ... (định nghĩa menu "con")
</li>
```

```
<ul class="dropdown-menu">
  <li><a href="#">Page 1-1</a></li>
  <li><a href="#">Page 1-2</a></li>
  <li><a href="#">Page 1-3</a></li>
</ul>
```

# Collapsed Navbar

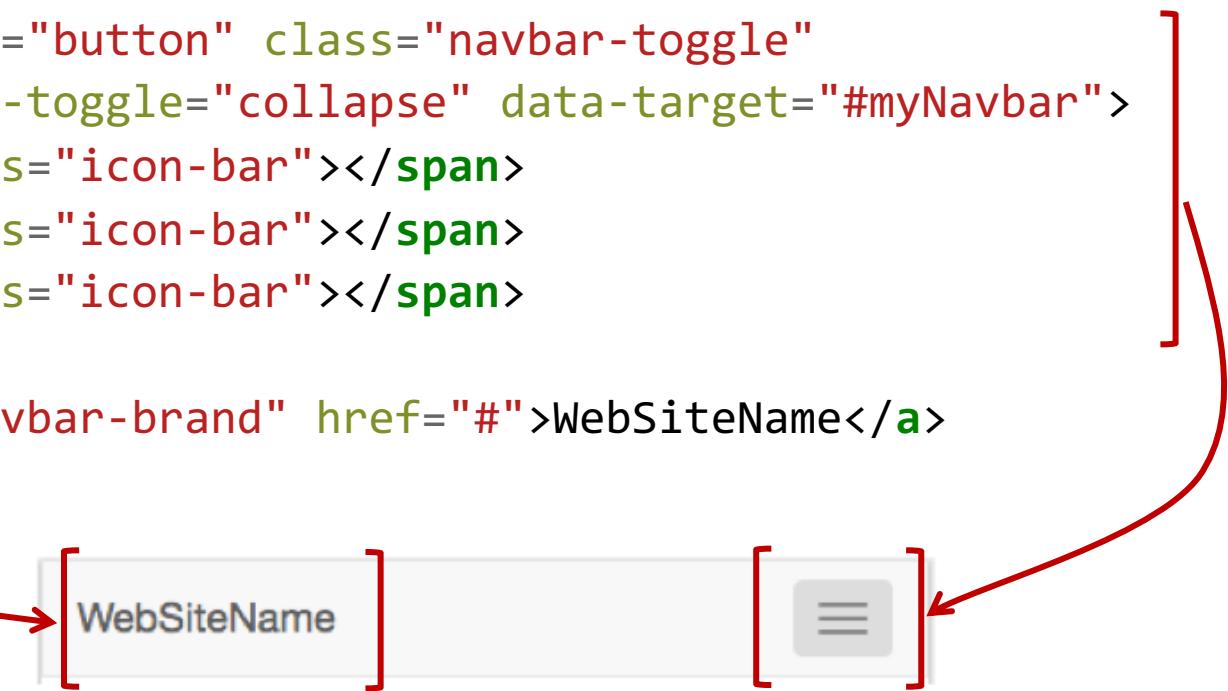
- Navbar that is collapsed (and toggleable) in mobile views



# Collapsed Navbar

```
<nav class="navbar navbar-default">
  <div class="container-fluid">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle"
             data-toggle="collapse" data-target="#myNavbar">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="#">WebSiteName</a>
    </div>
```

(to be cont.)





# Other navigation components

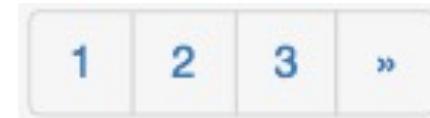
- Tabs: <http://getbootstrap.com/components/#nav-tabs>

```
<ul class="nav nav-tabs">
  <li class="active"><a href="#">Home</a></li>
  <li><a href="#">Profile</a></li>
  <li><a href="#">Messages</a></li>
</ul>
```



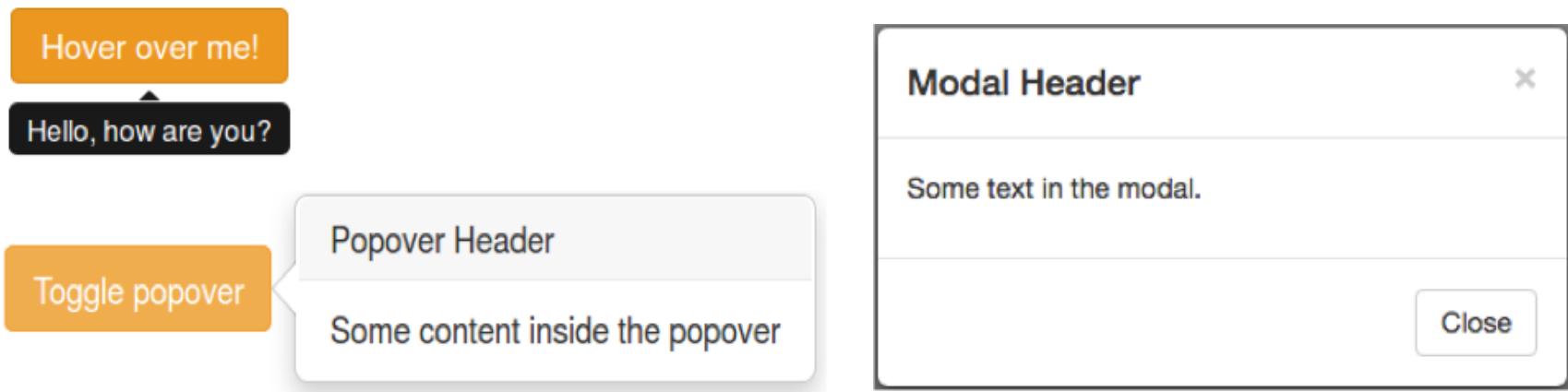
- Pagination: <http://getbootstrap.com/components/#nav-pagination>

```
<ul class="pagination">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#" aria-label="Next">
    <span aria-hidden="true">&raquo;</span>
  </a>
</li>
</ul>
```



# Tooltip, Popover and Modal

- Small pop-up boxes that appear (overlay) when the user moves the mouse pointer over an element
- Usually combined with a Bootstrap function call to be triggered
- Flexibility: Tooltip < Popover < Modal



# Tooltip, Popover and Modal

- **Tooltip:** a small pop-up displayed when the mouse pointer overs an element
  - 1) Add **data-toggle** and **data-placement** att. into the element

```
<a role="button" class="btn btn-warning" href="#" data-toggle="tooltip" data-placement="bottom" title="Hello, how are you?">Hover over me!</a>
```
  - 2) Add the following JS code into the webpage:

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

A diagram illustrating a tooltip. It consists of two overlapping rounded rectangles. The top rectangle is orange and contains the text "Hover over me!". A black arrow points from the bottom right corner of this rectangle towards the bottom rectangle. The bottom rectangle is black and contains the text "Hello, how are you?".

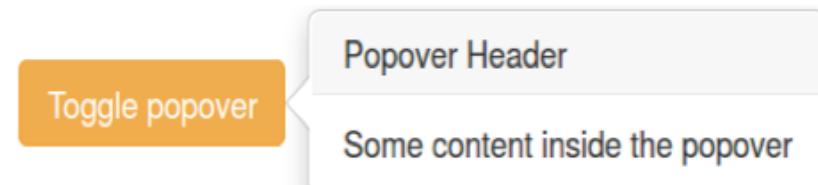
# Tooltip, Popover and Modal

---

- **Popover**: a small pop-up with **header** and **content** displayed when an element is clicked
  - 1) Add **data-toggle**, **title** and **data-content** attribute into the element:  

```
<a role="button" class="btn btn-warning" href="#" data-toggle="popover" title="Popover Header" data-content="Some content inside the popover">Toggle popover</a>
```
  - 2) Add the following JS code into the webpage:

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



# Tooltip, Popover and Modal

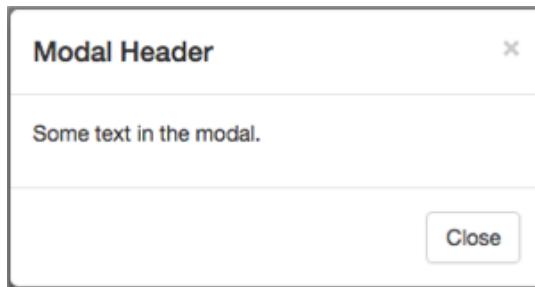
---

- **Modal:**

- Create model parts:
  - Model container: `modal`
  - Set proper model width and margins: `modal-dialog`
  - Model contents: `modal-header`, `modal-content`, `modal-footer`
- Close model: create a button with the attribute  
`data-dismiss="modal"`
- Trigger a model:
  - Use `JS` code (cf. next slide)
  - Use the attributes:  
`data-toggle="modal"` and `data-target="<#model-id>"`

# Tooltip, Popover and Modal

```
<div id="loginModal" class="modal fade" role="dialog">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close"
               data-dismiss="modal">&times;</button>
        <h4 class="modal-title">Login</h4>
      </div>
      <div class="modal-body">
        ...
      </div>
    </div>
  </div>
</div>
```



## Method 1

```
<script>
  $(document).ready(function () {
    $("#loginLink").click(function () {
      $("#loginModal").modal();
    });
  });
</script>
```

**Method 2** `<a data-toggle="modal" data-target="#loginModal">Login</a>`

# Bootstrap Customization

# Bootstrap customization

---

- Bootstrap is a CSS library
- Should use the Bootstrap standard library:
  - Consistency between pages
- Bootstrap customization methods:
  - Modify Bootstrap CSS ⇒ 
  - Create customization CSS ⇒ 
  - Create new Bootstrap build ⇒ 

# Create customization CSS

---

- Basic idea:
  - Override Bootstrap CSS pre-define classes
  - Create new selectors and classes
- Notes:
  - Include the customization files after the Bootstrap files
  - Selectors of the customization CSS must be more specific than the Bootstrap selector

(**Tips:** Use the “**Inspect Element**” of the browser to inspect the elements)

# Create customization CSS

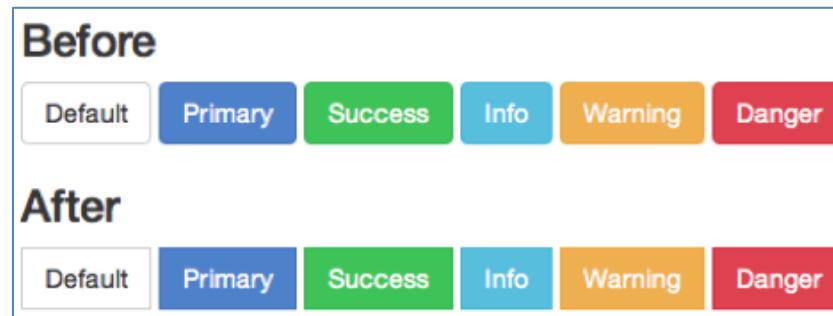
---

- Include the customization CSS:

```
<head>
  <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
  <link rel="stylesheet" type="text/css" href="css/custom.css">
</head>
```

- Override the Bootstrap class:

```
.btn {
  border-radius: 0px;
}
```





# Question?

CT313H – WEB TECHNOLOGY