

Introduction

WEB DEVELOPMENT FUNDAMENTALS - BOOTSTRAP



Overview

- Objectives
 - To explain the aims and objectives of the course
 - Contents
 - Course administration
 - Course objectives and assumptions
 - Introductions
 - Any questions?
 - Exercise
 - Locate the exercises
 - Locate the help files
-

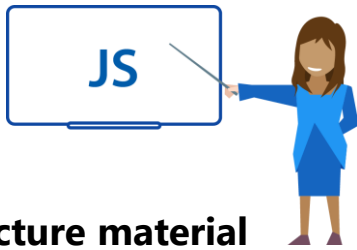
Administration

- Front door security
- Name card
- Chairs
- Fire exits
- Toilets
- Coffee Room
- Timing
- Breaks
- Lunch
- Downloads and viruses
- Admin. support
- Messages
- Taxis
- Trains/Coaches
- Hotels
- First Aid
- Telephones/Mobiles

early on that something is wrong, we have the chance to fix it. If you tell us after the course, it's too late! We ask you to fill in an evaluation form at the end of the course. If you alert us to a problem for the first time on the feedback form at the end of the course, we won't have had the opportunity to put it right.

If this course is being held at your company's site, much of this will not apply or will be outside our control.

Course delivery

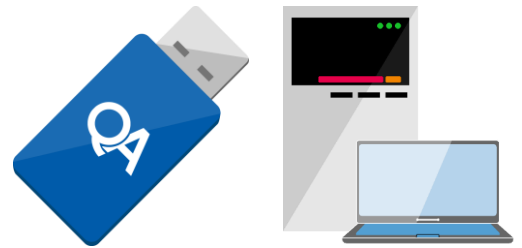


Lecture material

Hear and Forget
See and Remember
Do and Understand



Course workbooks



The course notebooks contain all the overhead foils that will be shown, so you do not need to copy them. In addition, there are extra textual comments (like these) below the foils, which are there to amplify the foils and provide further information. Hopefully, these notes mean you will not need to write too much and can listen and observe during the lectures. There is, however, space to make your own annotations too.

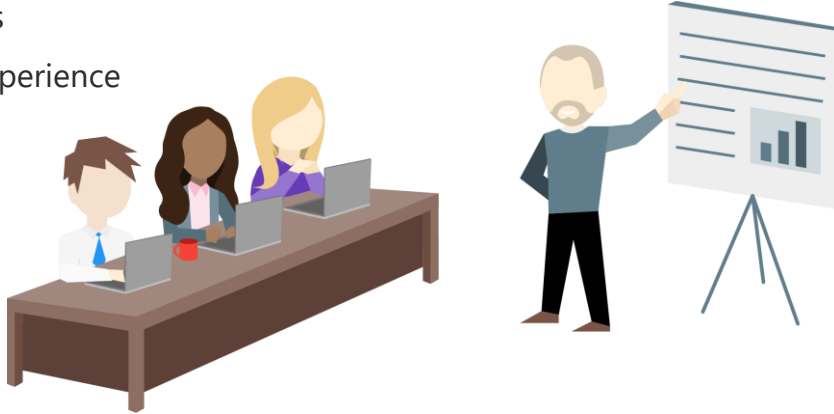
The appendices cover material that is beyond the scope of the course, together with some help and guidelines. There are also appendices on bibliography and Internet resources to help you find more information after the course.

In the practical exercise sessions, you will be given the opportunity to experiment and consolidate what has been taught during the lecture sessions. Please tell the instructor if you are having difficulty in these sessions. It is sometimes difficult to see that someone is struggling, so please be direct.

A training experience

A course should be

- A two-way process
- A group process
- An individual experience



Work with other people during practical exercise sessions. The person next to you may have the answer, or you may know the remedy for them. Obviously do not simply 'copy from' or 'jump-in on' your neighbour, but group collaboration can help with the enjoyment of a course.

We are also individuals. We work at different paces and may have special interests in particular topics. The aim of the course is to provide a broad picture for all. Do not be dismayed if you do not appear to complete exercises as fast as the next person. The practical exercises are there to give plenty of practical opportunities; they do not have to be finished and you may even choose to focus for a long period on the topic that most interests you. Indeed, there will be parts labelled 'if time allows' that you may wish to save until later to give yourself time to read and absorb the course notes. If you have finished early, there is a great deal to investigate. Such "'hacking' time is valuable. You may not get the opportunity to do it back in the office!

Course aims and objectives

By the end of the course, you will be able to:

- Explain what Bootstrap is and how it is applied
 - Use Bootstrap classes to produce a grid layout
 - Style text and images using Bootstrap
 - Use lists and tables with Bootstrap
 - Create navigation components using Bootstrap
 - Create and style forms using Bootstrap
 - Use different Bootstrap specific components to enhance websites
-

Assumptions

- This course assumes the following prerequisites
 - You have good skills, knowledge and understanding of HTML and CSS
- If there are any issues, please tell your instructor now

Introductions

Please say a few words about yourself

- What is your name and job?
- What is your current experience of
 - Computing?
 - Programming?
 - Web development?
- What is your main objective for attending the course?

It is useful for us all to be aware of levels of experience. It will help the instructor judge the level of depth to go into and the analogies to make to help you understand a topic. People in the group may have specialised experience that will be helpful to others.

It is worth highlighting particular interests, as we may be able to address them during the course. However, it is a general course that aims to cover a broad range of topics, so the instructor may have to deal with some areas during a coffee break or over lunch.

Any questions

- Golden Rule
 - "There is no such thing as a stupid question"
- First amendment to the Golden Rule
 - "... even when asked by an instructor"
- Corollary to the Golden Rule
 - "A question never resides in a single mind"

questions during the course. If so, ask them!

Bootstrap

RESPONSIVE WEB DESIGN TOOLS



Learning Objectives

- To understand what Bootstrap is and how it is applied
- To be able to use Bootstrap classes to produce a grid layout
- To be able to style text and images using Bootstrap
- To be able to use lists and tables with Bootstrap
- To be able to create navigation components using Bootstrap
- To be able to create and style forms using Bootstrap
- To be able to use different Bootstrap specific components to enhance websites



BOOTSTRAP BASICS

- What is Bootstrap?
- Why use Bootstrap?
- How to get Bootstrap
- Basic Bootstrap pages
- Grids
- Typography
- Text and Background Colours
- Images

Logo: https://commons.wikimedia.org/wiki/File:Bootstrap_logo.svg

By Bootstrap (<http://blog.getbootstrap.com>) [Public domain], via Wikimedia Commons

What is Bootstrap?

- Grid-based CSS framework designed to simplify:
 - Page layout
 - Cross-browser compatibility
 - Responsive development
- Provides reusable patterns and tools for:
 - Layout
 - Navigation
 - Typography
 - Buttons, tables, modals, image carousels...etc
- Bootstrap 4 released January 2018 with faster stylesheets and more responsiveness
 - Does not support <IE10 and Glyphicons

Why use Bootstrap?

- Easy to use
 - In many cases, its just about applying CSS classes to elements
- Responsive
 - Using the correct structure allows Bootstrap to take care of responsive aspects of web design
- Mobile-first
 - Mobile-first styles are part of the core framework
- Cross-browser compatibility
 - Compatible with all modern browsers
 - Need to use a lower version (e.g. Bootstrap 3) for compatibility with older browsers
 - Not recommended!

How to get Bootstrap

- Can be downloaded from <https://getbootstrap.com>
- Can be included on pages from a Content Delivery Network
 - Bootstrap CDN

```
<link
  rel="stylesheet"
  href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.
css"
  integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"
  crossorigin="anonymous"
>
```

- This is for pure CSS features of Bootstrap

15

The file can also be downloaded and put on your own server.

Bootstrap and JavaScript

- Some components require additional JavaScript for functionality
- These are:
 - Alerts for dismissing
 - Buttons for toggling states and checkbox/radio functionality
 - Carousel for all slide behaviours, controls, and indicators
 - Collapse for toggling visibility of content
 - Dropdowns for displaying and positioning
 - Modals for displaying, positioning, and scroll behaviour
 - Navbar for extending our Collapse plugin to implement responsive behaviour
 - Tooltips and popovers for displaying and positioning
 - Scrollspy for scroll behaviour and navigation updates

Bootstrap and JavaScript

- To use these components an additional Bootstrap file needs to be included with the page:
bootstrap.bundle.js (or **bootstrap.bundle.min.js** for production)
 - Available from <https://getbootstrap.com>
- Also requires jQuery to be included
 - From <http://jquery.com>
 - From a CDN, such as MaxCDN

Basic Bootstrap Pages

- Page must be written in HTML5
- The viewport meta data must be included
- A container must wrap the site contents
 - `.container` – provides responsive, fixed-width container

```
<main class="container">  
...  
</main>
```

- `.container-fluid` – provides full width container spanning entire width of viewport

```
<main class="container-fluid">  
...  
</main>
```

QuickLab 1 –Bootstrap Basics

- Add Bootstrap to a page and experiment with different containers

Bootstrap Grids

- Follows the 12-column grid pattern
- Built with flexbox
- Can group columns together but number of columns in a row must be 12
- Five classes can be used within a containing `.row`

Class	Description
<code>.col-x</code>	For extra small devices with screen width < 576px
<code>.col-sm-x</code>	For small devices with screen width >= 576px
<code>.col-md-x</code>	For medium devices with screen width >= 768px
<code>.col-lg-x</code>	For large devices with screen width >= 992px
<code>.col-xl-x</code>	For extra large devices with screen width >= 1200px

- `x` is replaced by the number of columns to span

20

Classes can be combined to make more flexible, dynamic layouts.

Each class scales up, meaning if you want let sizes be the same from a particular size, you only need to declare for that and smaller.

E.G. If the md, lg and xl screens should use the same layout, only the md needs to be specified.

Column Widths

- To create equal columns on all devices and screen widths `.col` can be used
- To create responsive designs, specify the *size of device* and the *number of columns* to be taken by each element
 - If the screen is too small to display then the elements automatically stack

```
<main class="container">
  <div class="row">
    <div class="col-sm-6">Col 1</div>
    <div class="col-sm-6">Col 2</div>
  </div>
</main>
```

- Multiple classes can be used to create a 'stepped' layout

```
<div class="col-12 col-sm-6 col-md-3 col-lg-2 col-xl-1">Col 1</div>
```

QuickLab 2 – Bootstrap Grids

- Experiment with different classes used to create layouts on differently sized devices

Typography

- By default, fonts in Bootstrap are set to:
 - `font-size : 16px`
 - `line-height: 1.5`
 - `font-family: "Helvetica Neue", Helvetica, Arial, sans-serif`
- `p` elements have a `margin-top` set to 0 and `margin-bottom` set to `1rem` (16px)
- `h2-h6` are scaled from `2.0rem` (32px) for `h2` to `1rem` (16px) for `h6`
 - `h1` is set at `2.5rem` (40px)
 - `.display-x` classes can be used to make headings stand out more than normal headings
 - `<small>` is used to create lighter, secondary text within a `hx` element

Typography

- There are a number of other tags that have defined styles in Bootstrap

Tag	Effect
<mark>	Styled with a yellow background and some padding
<abbr>	Styled with a dotted bottom border
<blockquote class="blockquote"> <footer class="blockquote-footer">	Used to present content from another source and the source's name
<dl>	The child <dt> elements are styled in bold font
<code>	Uses mono-space font style to highlight code snippets or words
<kbd>	Key combinations in text are put in a round-bordered black background with a white, mono-space font style
<pre>	Used around multi-line code snippets to preserve white-space and line breaks, using a mono-space font style

Typography

- A number of CSS classes can also be added to elements to help style text within them

Class	Description	Class	Description
<code>.font-weight-bold</code>	Bold text	<code>.text-center</code>	Centre-aligned text
<code>.font-italic</code>	Italic text	<code>.text-*-center</code>	Centre alignment on sm, etc
<code>.font-weight-light</code>	Light weight text	<code>.text-justify</code>	Justified text
<code>.font-weight-normal</code>	Normal text	<code>.text-left</code>	Left-aligned text
<code>.initialism</code>	Text inside <abbr> smaller	<code>.text-*-left</code>	Left alignment on sm, etc
<code>.lead</code>	Make paragraph stand out	<code>.text-lowercase</code>	Lowercase text
<code>.list-inline</code>	List items on single line	<code>.text-monospace</code>	Monospaced text
<code>.list-unstyled</code>	Remove default list style	<code>.text-nowrap</code>	No wrap text
<code>.pre-scrollable</code>	Makes <pre> scrollable	<code>.text-right</code>	Right-aligned text
<code>.small</code>	Make text 85% of parent	<code>.text-*-right</code>	Right alignment on sm, etc
<code>.text-capitalize</code>	Capitalised text	<code>.text-uppercase</code>	Uppercase text

Text and Background Colours

- Contextual classes help provide meaning through colours
- Text and backgrounds can be styled using the following classes:

Text Class	Effect	Background Class	Effect
.text-muted	This text is muted.	.bg-primary	This text is important.
.text-primary	This text is important.	.bg-success	This text indicates success.
.text-success	This text indicates success.	.bg-info	This text represents some information.
.text-info	This text represents some information.	.bg-warning	This text represents a warning.
.text-warning	This text represents a warning.	.bg-danger	This text represents danger.
.text-danger	This text represents danger.	.bg-secondary	Secondary background color.
.text-secondary	Secondary text.	.bg-dark	Dark grey background color.
.text-white		.bg-light	Light grey background color.
.text-dark	Dark grey text.		
.text-body	Body text.		
.text-light	Light grey text.		

Images

- Images can be:
 - Shaped by adding the classes: **rounded**, **circle** and **img-thumbnail**



Rounded border image



Rounded-circle border image



Thumbnail styled image

- Aligned by adding the classes: **float-left**, **float-right** and to centre use **mx-auto** and **d-block**
- Made responsive by adding **img-fluid** class
 - Applies **max-width: 100%** and **height: auto** to the image

Image from: <https://openclipart.org/detail/267783/surreal-beach-sunset>

QuickLab 3 – Text, Colours and Images

- Use Bootstrap classes to style some text and images



Logo: https://commons.wikimedia.org/wiki/File:Bootstrap_logo.svg

By Bootstrap (<http://blog.getbootstrap.com>) [Public domain], via Wikimedia Commons

Lists and List Groups

- Created by adding class **list-group** to `` elements and **list-group-item** to `` elements
- Can create links by making `<div>` class **list-group** and then adding `<a>` elements with class **list-group-item** (and **list-group-item-active** if interactivity on hover is required)
 - Adding attribute `disabled` will add a lighter colour and disable the link and its effects
- List groups appear in a rounded border box with separating lines by default
 - Outer borders can be removed by adding **list-group-flush** class to ``
- **-success**, **-secondary**, **-info**, **-warning**, **-danger**, **-primary**, **-dark** and **-light** can be appended to the `` class to make contextual items in a list
- Badges can be added by using **span** with classes **badge**, **badge-pill** and **badge-XX** (as above) and including content
 - Alignment can be achieved by using **flex**, **justify** and **align** classes on the ``

Tables

- Tables have several classes that can be applied to the <table> element for styling

Class	Effect
table	Adds light padding ad horizontal dividers to rows and table spans parent width
table-striped	Adds alternate grey and white backgrounds to rows
table-bordered	Adds borders on all sides of the table and cells
table-hover	Default table with mouse-over greying of row
table-dark	Adds black background to table and makes text white
table-borderless	Removes all borders from table
table-XX	Use any of the contextual colours to style the table
thead-dark/thead-light	Changes the appearance of the table headers (applied to <thead> elements)
table-sm	Makes the table smaller by halving the cell padding
table-responsive-XX	Use sm md lg xl to make the table responsive dependent on screen size



Logo: https://commons.wikimedia.org/wiki/File:Bootstrap_logo.svg

By Bootstrap (<http://blog.getbootstrap.com>) [Public domain], via Wikimedia Commons

Buttons

- All buttons use the class `btn` (can be used on `<a>`, `<button>` or `<input>`)
 - `btn-XX` can then be used to add colour using the 8 `bg-XX` options plus `btn-link` for links

```
<button type="button" class="btn btn-success">Success</button>
```



- Outline/bordered buttons
 - Use `btn-outline-XX` to style the border and text of the button

```
<button type="button" class="btn btn-outline-Danger">Danger</button>
```



All buttons also include interactivity in the form of colour changes on hover

Buttons

- **Sizing Buttons**
 - `btn-lg` and `btn-sm` classes provided to size buttons different from the default
- **Active/Disabled Buttons**
 - Adding *class* `active` to a button makes it appear pressed
 - Adding *attribute* `disabled` to button makes it 'greyed' out and unclickable
- **Block Level Buttons**
 - Adding `btn-block` class to button makes it span the entire width of its parent element

Button Groups

- Buttons can be placed together in a series using **btn-group** as class on **<div>** that wraps the buttons to group
 - Inline by default – means side-by-side button groups are possible
 - Sizing can be applied using **btn-group-lg** and **button-group-sm**
- Button groups can be made vertical using **btn-group-vertical**
- Dropdowns can be added by:
 - Adding class **dropdown-toggle** to the button
 - Adding attribute **data-toggle** set to **dropdown** to button
 - Adding a **<div>** with a class of **dropdown-menu** to **btn-group <div>**
 - Adding **<a>** elements with a class of **dropdown-item**

Navigation Menus

- Simple menus can be created using:
 - **nav** class on `` elements
 - **nav-list** class on their `` items
 - **nav-link** class to their `<a>`
- Links can be aligned by adding class to ``
 - To centre add **justify-content-center**
 - To right-align add **justify-content-end**
- Vertical menus can be created by adding class **flex-column** to ``

```
<ul class="nav">
  <li class="nav-item">
    <a class="nav-link" href="#">
      Link Text
    </a>
  </li>
  ...
</ul>
```

Tabs and Pills

- Menu can be converted to Tabs by adding the **nav-tabs** class to ``
 - For Pills add **nav-pills** instead
- To give equal width (justified) to Tabs/Pills also add **nav-justified** to ``
- Active Tab/Pill can be identified by adding **active** class to current/active link
- Links can be disabled by adding the **disabled** class to it

```
<ul class="nav nav-tabs">
  <li class="nav-item">
    <a class="nav-link active"
      href="#">Link Text</a>
  </li>
  <li class="nav-item">
    <a class="nav-link disabled"
      href="#">Link Text</a>
  </li>
  ...
</ul>
```

Tabs and Pills with Dropdowns

- To make a Tab/Pill have a dropdown:
 - Add **dropdown** to **** class that contains link to have dropdown options
 - To **<a>** to contain dropdown:
 - Add **dropdown-toggle** to **<a>** classes
 - Add **data-toggle** attribute set to **dropdown** to **<a>** classes
 - Add **<div>** with class **dropdown-menu** to ****
 - Add **<a>** elements with class **dropdown-item** to **<div>**

```
<ul class="nav nav-pills">
  <li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle"
      data-toggle="dropdown" href="#">
      Dropdown Pill</a>
    <div class="dropdown-menu">
      <a class="dropdown-item" href="#">
        Dropdown Item 1</a>
      <a class="dropdown-item" href="#">
        Dropdown Item 2</a>
    </div>
  </li>
  ...
</ul>
```

Dynamic/Togglable Tabs and Pills

- Display different content on same page
- Each link needs attribute **data-toggle** set to **tab** or **pill**
- Page needs **<div>** with **tab-content** class (for both Tabs and Pills)
 - Populate with **<div>** elements for content with **tab-pane** class with **id** related to **href** of associated link
 - First should have class **active** to display as default
 - Can be **container**
 - Can add **fade** to make content fade in and out on transition

```
<ul class="nav nav-pills">
  <li class="nav-item">
    <a class="nav-link dropdown-toggle"
      data-toggle="pill" href="#home">
      Home</a>
    <a class="nav-link dropdown-toggle"
      data-toggle="pill" href="#pill1">
      Pill 1</a>
  </li>
</ul>
<div class="tab-content">
  <div class="tab-pane container active"
    id="home">...<div>
  <div class="tab-pane container active"
    id="pill1">...<div>
</div>
```

Basic Navbars

- Can extend or collapse dependent on screen size
- Created using navbar **class** on **<nav>** element followed by responsive collapsing class (which stacks navbar vertically for specified screen(s))

navbar-expand-xl | lg | md | sm

- **** needs to have class **navbar-nav**
 - **** elements need to have **nav-item** class
 - **<a>** elements need to have **nav-link** class

```
<nav class=navbar navbar-expand-sm>
  <ul class=navbar-nav>
    <li class=nav-item>
      <a class=nav-link href=#>
        Link 1</a>
      </li>
    ...
  </ul>
</nav>
```


Navbar Formatting Options

- **Vertical Navbars**
 - Created by omitting the `navbar-expand-xx` class from `<nav>`
- **Centred Navbars**
 - Add class `justify-content-center` to `<nav>`
- **Coloured Navbars**
 - Use any of the `bg-xx` classes on the `<nav>`
 - Use `navbar-dark` to create white text links
 - User `navbar-light` to create black text links
- **Brand/Logo on Navbar**
 - Adding a link/image with class of `navbar-brand` as the immediate child of `<nav>`

Responsive Navbars

- Often desirable to hide navigation links and replace with 'burger' button on small screens
 - Links revealed when button is clicked
- Add a `<button>` to `<nav>`
 - Use class `navbar-toggler`
 - Add attribute `data-toggle` set to `collapse`
 - Add attribute `data-target` set to the `id` of the element that contains the `` for the navbar
 - Add a `` with a class of `navbar-toggler-icon`
- Wrap `` in a `<div>` with classes `collapse` and `navbar-collapse` and `id` that is same value as `data-target` of `<button>`
- NOTE: jQuery, PopperJS and BootstrapJS need to be included on pages for this to work

42

CDN links:

```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"> </script>
```

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js" integrity="sha384-ZMP7rVo3mlykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnjAK/l8WvCWPIpM49"
crossorigin="anonymous"> </script>
```

```
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js" integrity="sha384-ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwEULTy"
crossorigin="anonymous"> </script>
```

Dropdowns on Navbars

- To add a dropdown to a link on a navbar process is same as for Tabs/Pills
 - Add **dropdown** to **** class that contains link to have dropdown options
 - To **<a>** to contain dropdown:
 - Add **dropdown-toggle** to **<a>** classes
 - Add **data-toggle** attribute set to **dropdown** to **<a>** classes
 - Add **<div>** with class **dropdown-menu** to ****
 - Add **<a>** elements with class **dropdown-item** to **<div>**

Navbar with Form Elements

- Useful for adding Search Bars to a navbar
- Add a `<form>` with `form-inline` class and an appropriate `action` to `<nav>`
- Add `input` with class of `form-control` and appropriate spacing
- Add `input` with class `btn`, `type` of `submit` and `value` for text on the button



```
<nav>

  <form action="#" class="form-
    inline">

    <input type="text"
      class="form-control mr-sm-2"
      placeholder="Search">

    <input type="submit"
      class="btn btn-success"
      value="Search">

  </form>
</nav>
```

Navbar Positioning

- Appear in normal flow of HTML unless:
 - It is fixed to the top of the page using the class **fixed-top** on **<nav>**
 - It is fixed to the bottom of the using the class **fixed-bottom** on **<nav>**
 - It is made 'sticky', sticking to the top of the page even when scrolled past using **sticky-top**

QuickLab 5 - Navbars

- Create a responsive navbar with a dropdown option and a search bar



Logo: https://commons.wikimedia.org/wiki/File:Bootstrap_logo.svg

By Bootstrap (<http://blog.getbootstrap.com>) [Public domain], via Wikimedia Commons

Default Form Settings and Layouts

- Form controls automatically inherit global styling
- Text inputs – i.e. `<input>`, `<textarea>` and `<select>` - that have the `form-control` class applied have a width of 100% of the parent
- There are 2 layout options for Bootstrap forms
 - Stacked – a full width form – no class needed for `<form>`
 - Inline form - requires `form-inline` class to be added to `<form>`
 - Only applies to forms on screens at least 576px wide – stacks horizontally otherwise
- Usual to surround each input on a form in a `<div>` with the class `form-group`

Inputs

- The following form controls are supported by Bootstrap:
 - **input**
 - **textarea**
 - **checkbox**
 - **radio**
 - **select**
- All HTML5 input types are supported
- Styling depends on the type being declared correctly

<input> and <textarea>

- <input> and <textarea> needs to have **form-control** class applied
- Usually accompanied by a <label> (no class needed)

```
<div class="form-group">
  <label for="name">Name:</label>
  <input type="text" class="form-control"
    name="name">
</div>

<div class="form-group">
  <label for="comment">Comment:</label>
  <textarea class="form-control" rows="3"
    name="comment"></textarea>
</div>
```

Checkboxes/Radio Buttons

- Can be stacked or inline
 - Use **form-check** class on wrapping **<div>** for stacked
 - Use **form-check-inline** for inline
- Surround **<input>** with **<label>** that has class **form-check-label**
- For radio buttons, ensure that each **<input>** has same **name** attribute
- Input can be disabled by adding **disabled** attribute

```
<div class="form-check-inline">  
  <label class="form-check-label">  
    <input  
      class="form-check-input"  
      type="checkbox"  
      value="1"  
    >  
    Option 1  
  </label>  
</div>
```

Select

- Add the **form-control** class to the **<select>** element itself
 - No need to add anything to any of the **<option>** elements
- Multiple selected options can be enabled by adding the **multiple** attribute to **<select>**

```
<div class="form-check-inline">
  <label for="selectList">Select:
  <select
    class="form-control"
    name="selectList"
  >
    <option value="1">Option 1</option>
    <option value="2">Option 2</option>
    <option value="3">Option 3</option>
  </select>
</div>
```

Form sizing and other controls

- **Sizing**

- Forms can use the additional `form-control-sm` and `form-control-lg` classes to change the size of the inputs on the form

- **Making inputs plain text**

- Use the class `form-control-plaintext` instead of the `form-control` on the `<input>` element

- **Ranges**




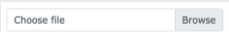

- If using a range as an input, use the `form-control-range` class to add Bootstrap styling

- **File Selecting**

- If using an input type of file, use `form-control-file` to add Bootstrap styling

Custom Forms

- Customised form elements to replace the browser defaults – use classes as shown

Element	Class	Effect
Wrapping <div> <input type="checkbox" id="cb1"> <label for="cb1">	custom-control custom-checkbox custom-control-input custom-control-label	 Custom checkbox
Wrapping <div> <input type="radio" id="rg1"> <label for="rg1">	custom-control custom-radio custom-control-input custom-control-label	 Custom radio
<select>	custom-select custom-select-sm lg	
Wrapping <div> <input type="file" id="cf1"> <label for="cf1">	custom-file custom-file-input custom-file-label	
<input type="range">	custom-range	

QuickLab 6 - Forms

- Use Bootstrap classes to style a form



Logo: https://commons.wikimedia.org/wiki/File:Bootstrap_logo.svg

By Bootstrap (<http://blog.getbootstrap.com>) [Public domain], via Wikimedia Commons

Jumbotron

- A grey box, usually at the top of a page, that highlights special content or information
- Add class **jumbttron** to a **<div>** element
 - Can be made full width by adding class **jumbotron-fluid** to the **<div>** removing the rounded borders and a Bootstrap **container** (or **container-fluid**) inside it

I am a Jumbotron

I am a full width Jumbotron with no rounded borders

```
<main class="container">
  <div class="jumbotron">
    <h1>I am a Jumbotron</h1>
  </div>
  <div class="jumbotron jumbotron-fluid">
    <div class="container">
      <h1>I am a full width Jumbotron with
        no rounded borders</h1>
    </div>
  </div>
</main>
```

57

For more information on Jumbotrons see:

<https://getbootstrap.com/docs/4.0/components/jumbotron/>

Carousels

- A slideshow for cycling through elements (not necessarily images!)
- Has a number of classes that can be applied to the wrapping element and its children

Bootstrap Class	Description
<code>carousel</code>	Makes an element apply Bootstrap's carousel styling
<code>carousel-indicators</code>	Adds dots at the bottom of each slide to indicate current and how many
<code>carousel-inner</code>	Adds slides to the carousel
<code>carousel-item</code>	Defines the content of each slide in the carousel
<code>carousel-control-prev</code> , <code>carousel-control-next</code>	Adds previous and next buttons to allow navigation between slides
<code>carousel-control-prev-icon</code> , <code>carousel-control-next-icon</code>	Used with the classes above to place the button on the screen
<code>carousel-caption</code>	Used inside a <code>carousel-item</code> to add a caption to the slide
<code>slide</code>	Adds CSS transition and animation – does not have to be included

For more information on Carousels see:
<https://getbootstrap.com/docs/4.0/components/carousel/>

Carousels

- The wrapping `<div>` must have an `id` and a `data-ride` attribute set to `carousel`
- The `` elements inside the `<ul class=carousel-indicators>` must have attributes:
 - `data-target` set to the `id` of the wrapping `<div>`
 - `data-slide-to` set to the slide number that the indicator is for
 - Class `active` is added to the initial slide indicator to be displayed
- Actual content is a `<div>` of class `carousel-inner` for each slide has:
 - A `<div>` with a class of `carousel-item` that wraps the content for the slide
 - Class `active` is added to the initial slide to be displayed
- Control buttons are placed in an `<a>` element with a `href` of the wrapping div `id`
 - Has a `data-slide` attribute set to `prev` or `next`
 - Uses a `` with icon class to display button

59

For more information on Carousels see:

<https://getbootstrap.com/docs/4.0/components/carousel/>

```

<div id="carouselDemo" class="carousel slide" data-ride="carousel">
  <ul class="carousel-indicators">
    <li data-target="#carouselDemo" data-slide-to="0" class="active"></li>
    <li data-target="#carouselDemo" data-slide-to="1"></li>
    <li data-target="#carouselDemo" data-slide-to="2"></li>
  </ul>

  <div class="carousel-inner">
    <div class="carousel-item active">
      
      <div class="carousel-caption">
        <h3 class="bg-info">A Carousel</h3>
        <p class="bg-info">A carousel in Philadelphia</p>
      </div>
    </div>
    ...

    <a href="#carouselDemo" class="carousel-control-prev" data-slide="prev">
      <span class="carousel-control-prev-icon"></span>
    </a>

    <a href="#carouselDemo" class="carousel-control-next" data-slide="next">
      <span class="carousel-control-next-icon"></span>
    </a>
  </div>

```

Extract from code to
produce a Bootstrap
carousel

60

For more information on Carousels see:

<https://getbootstrap.com/docs/4.0/components/carousel/>

Modals

- Elements that are displayed over the current page – a dialog box or popup window
- Usually requires some sort of button or link to open
 - Set `<button>` to have `data-toggle` attribute of `modal` and `data-target` of `id` of element to display
- Actual modal requires wrapping `<div>` with class of `modal` and `id` as defined in `<button>`
 - Two further enclosing `<div>` elements added with classes of `modal-dialog` and `modal-content` respectively
 - Modal can have header by using `modal-header` class on a `<div>`
 - `Modal-title` class used for heading and button usually supplied with class `close` and `data-dismiss` attribute set to `modal`
 - Modal body enclosed in a `<div>` with class `modal-body`
 - Modal footer enclosed in `<div>` with class `modal-footer` – could also contain close `<button>`

61

For more information on Modals see:

<https://getbootstrap.com/docs/4.1/components/modal/>

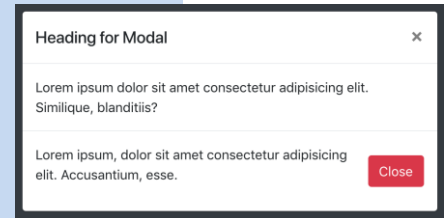
```

<button class="btn btn-info" type="button"
  data-toggle="modal" data-target="#modalDemo">
  Open Modal
</button>
<div class="modal bg-dark" id="modalDemo">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title">Heading for Modal</h5>
        <button class="close" type="button"
          data-dismiss="modal">&times</button>
      </div>

      <div class="modal-body text-white">
        Lorem ipsum dolor sit amet...
      </div>

      <div class="modal-footer text-white">
        <p>Lorem ipsum, dolor sit...</p>
        <button class="btn btn-danger" type="button"
          data-dismiss="modal">Close</button>
      </div>
    </div>
  </div>
</div>

```



Extract from code to
produce a Bootstrap
modal

62

For more information on Modals see:

<https://getbootstrap.com/docs/4.1/components/modal/>

Cards

- Cards are flexible and extensible content containers
 - Includes options for headers, footers, wide variety of content, contextual background colours and other display options
 - Bordered box with some padding around the content
- Many classes provided – created with **card** with separate **card-header**, **card-content** and **card-footer** containing elements
- Text can be formatted using **card-title**, **card-text**, **card-link**
- Images can be placed at the top or bottom of the card using **card-img-top** and **card-img-bottom**
 - Can also be placed behind text using **card-img-overlay**

63

For more information on cards see:

<https://getbootstrap.com/docs/4.1/components/card/>

Cards - Layout

- Columns
 - **card-columns** added to a `<div>` that wraps the cards to be put in columns
 - Automatically adjusts to the number of cards – displayed vertically on small screens
- Deck
 - **card-deck** added to a `<div>` that wraps the cards to be put in the deck
 - Automatically adjusts to the number of cards – displayed vertically on small screens
- Group
 - **card-group** added to a `<div>` that wraps the cards to be grouped
 - Same as **card-deck** without margins
 - Automatically adjusts to the number of cards – displayed vertically on small screens (WITH top and bottom margins)

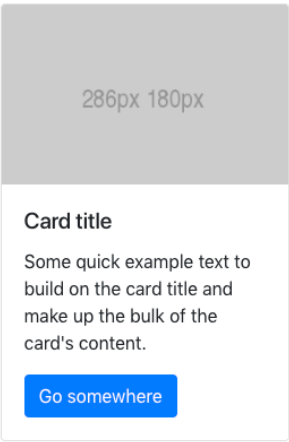
64

For more information on cards see:

<https://getbootstrap.com/docs/4.1/components/card/>

Cards

```
<div class="card" style="width: 18rem;">
  
  <div class="card-body">
    <h5 class="card-title">
      Card title
    </h5>
    <p class="card-text">
      Some quick example text to build
      on the card title and make up the
      bulk of the card's content.
    </p>
    <a href="#" class="btn btn-primary">
      Go somewhere
    </a>
  </div>
</div>
```



For more information on cards see:
<https://getbootstrap.com/docs/4.1/components/card/>

Badges

- Used to add extra information to content
- Uses the **badge class** with a `badge-XX` contextual class to create rectangular badges
- Scales to match the size of its parent element
- Can be used alone or as a pill, button or a link

66

For more information on badges see:

<https://getbootstrap.com/docs/4.1/components/badge/>

Badges

Basic Badge

```
<h1>Example heading <span
class="badge badge-
info">New</span></h1>
```



Pill Badge

```
<span class="badge badge-pill
badge-info">Info</span>
```



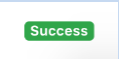
Badge in a Button

```
<button
  type="button"
  class="btn btn-primary">
  Notifications
  <span class="badge badge-light">
    5
  </span>
</button>
```



Badge as a Link

```
<a href="#"
class="badge badge-success">
Success</a>
```



For more information on badges see:
<https://getbootstrap.com/docs/4.1/components/badge/>

Pagination

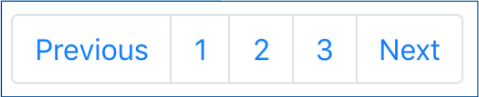
- Used to separate out large sets of content into more manageable chunks for the user
- **pagination** class used on a `` element
- **page-link** class used on each `` contained in the list (including previous and next)
- **active** class used to identify which is the current page
- Has **pagination-sm** and **pagination-lg** classes for sizing

68

For more information on pagination see:

<https://getbootstrap.com/docs/4.1/components/pagination/>

```
<nav aria-label="Page navigation example">
  <ul class="pagination">
    <li class="page-item">
      <a class="page-link" href="#">
        Previous
      </a>
    </li>
    <li class="page-item">
      <a class="page-link" href="#">
        1
      </a>
    </li>
    ...
    <li class="page-item">
      <a class="page-link" href="#">
        Next
      </a>
    </li>
  </ul>
</nav>
```



Excerpt of code
required to create
pagination

As pages likely have more than one such navigation section, it’s advisable to provide a descriptive aria-label for the <nav> to reflect its purpose. For example, if the pagination component is used to navigate between a set of search results, an appropriate label could be aria-label="Search results pages".

For more information in Pagination see:

<https://getbootstrap.com/docs/4.1/components/pagination/>

Collapse

- A plugin used to show and hide content
 - Uses `<button>` or `<a>` elements to produce the effect
 - `collapse` class added to elements that are to be collapsed
 - `show` class can be added to initially show an element
 - `data-toggle` attribute set to `collapse` needs to be added to the anchor/button
 - `data-target` or `href` needs to be set to the `id` of the element to collapse or show
 - Collapsing element animates the `height` from current to 0
 - Cannot use `padding` on collapse element

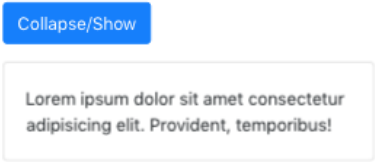
70

For more on collapse, including an accordion example using cards see:

<https://getbootstrap.com/docs/4.1/components/collapse/>

Collapse

```
<p>
  <a class="btn btn-primary"
    data-toggle="collapse"
    href="#collapseDemo"
    role=button>
    Collapse/Show
  </a>
</p>
<div class="collapse"
  id="collapseDemo">
  <div class="card card-body">
    Lorem ipsum...
  </div>
</div>
```



For more on collapse, including an accordion example using cards see:
<https://getbootstrap.com/docs/4.1/components/collapse/>

QuickLab 7

- Use different Bootstrap components to help format a page

Conclusion

WEB DEVELOPMENT FUNDAMENTALS - BOOTSTRAP



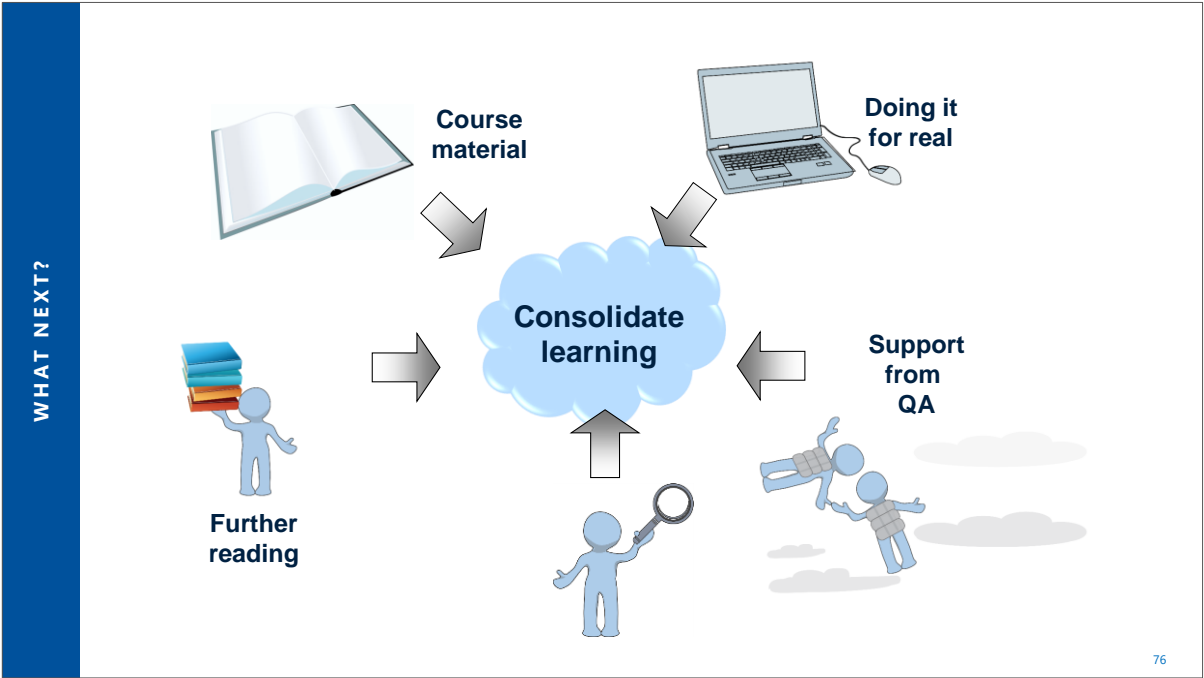
Overview

- Checking objectives
- References
- What next?

Review aims and objectives

By the end of the course, you will be able to:

- Explain what Bootstrap is and how it is applied
- Use Bootstrap classes to produce a grid layout
- Style text and images using Bootstrap
- Use lists and tables with Bootstrap
- Create navigation components using Bootstrap
- Create and style forms using Bootstrap
- Use different Bootstrap specific components to enhance websites



Help from QA

- Ever increasing range of courses
 - Vendor neutral skills
 - JavaScript Frameworks
 - PHP
 - Agile
 - Application development
 - C#
 - Python
 - Java
 - DevOps
- Web site
 - www.qa.com
 - Social networking
 - Blogs <http://www.qa.com/blogs>



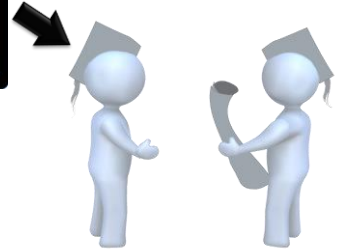
USB + Evaluation

- In closing...



Remember to make
a copy of your work
to take away

Please complete your
course evaluation



<http://evaluation.qa.com>
Course Code: QABOOTSTR
PIN: