Bootstrap Grid

In this week's lab, you will create HTML pages that include bootstrap code which will create a responsive page.

Bootstrap

Exercise 1 – Create a basic bootstrap page

1. Copy the starter template (https://getbootstrap.com/docs/5.3/getting-started/introduction/#quick-start) for "2. Include Bootstrap's CSS and JS" into a new HTML document. Run your code to see if it works.
   Add the code below to your page

```html
<body>
  <div class="container">
    <div class="row">
      <div class="col-md-6">
        Box Six columns wide
      </div>
    </div>
  </div>
```

   Run your code to see if it works.

2. Add a new class to your document that will make the background colour grey, with a red border (so that we can see the effects of the bootstrap grid).

   Apply this style to the "col-md-6" div. Run your code to see if it works. You should see something similar to this:


3. Use all 12 columns by adding a second div that is also 6 columns wide. This DIV can also use the class you created in exercise 2.

   Add the class **col-sm-12** to each DIV and test your code. When the page is smaller, the boxes should fill the size of the page. Open the webpage in responsive mode within your browser and adjust the size of the page between small and medium. Remember medium is for any viewport from 768 to 991 pixels in width while small is for any view port less than 768 pixels in width. See screenshot as follows:
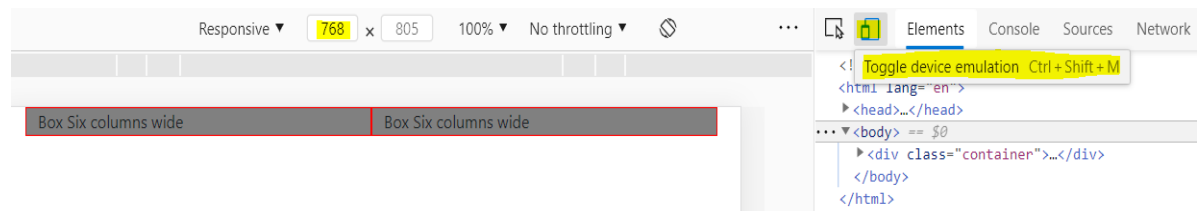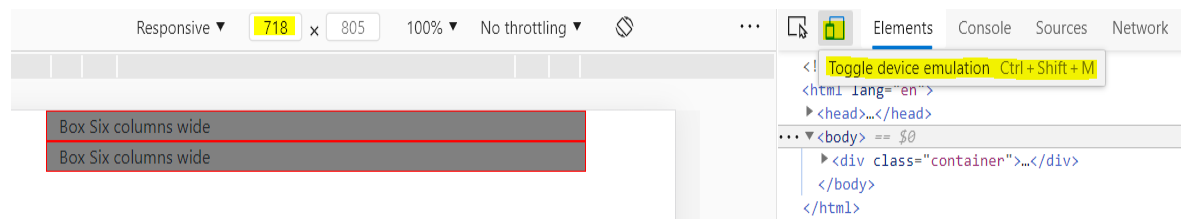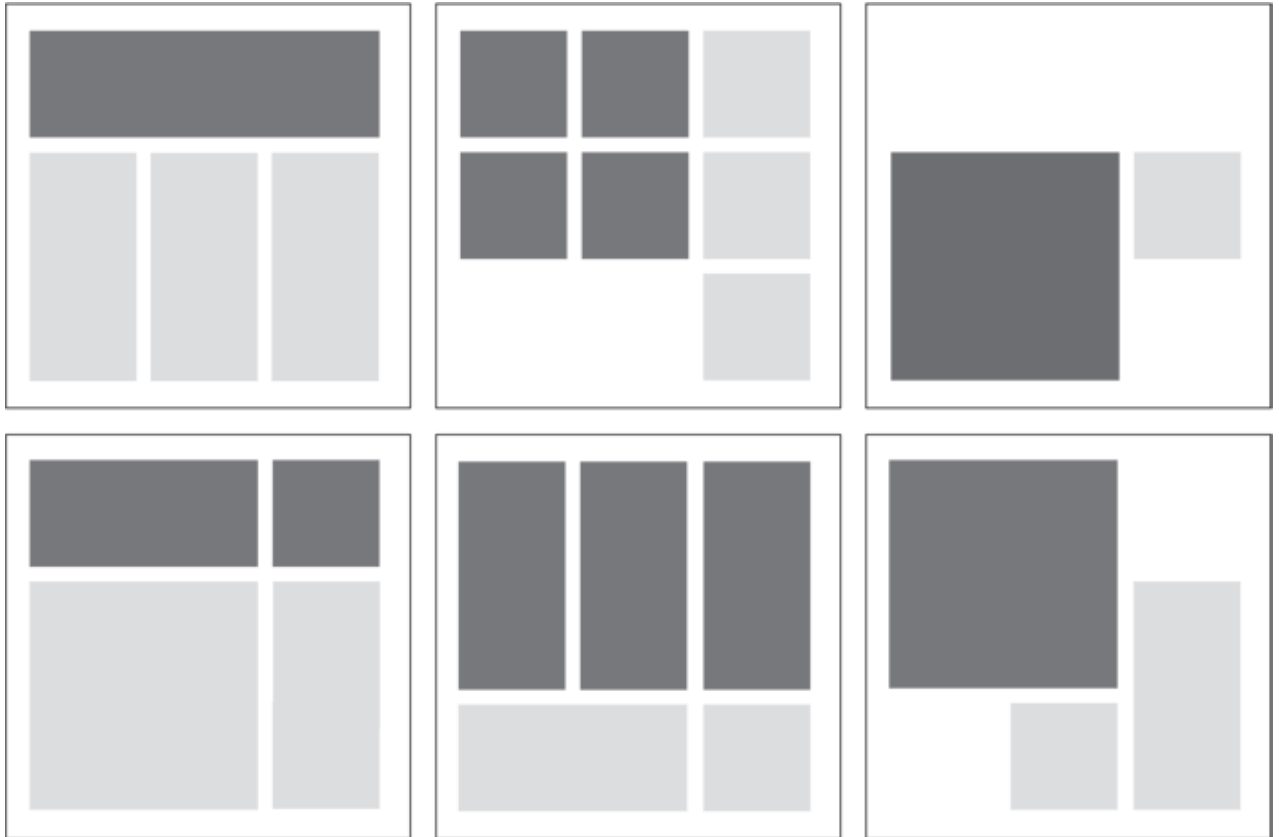
4. Recreate the following styles using the bootstrap grid:
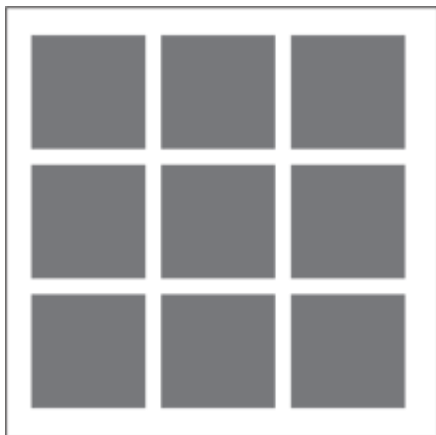
The page should display as shown above until the screen is made small, then each column is full size.

Hint: https://getbootstrap.com/docs/5.3/layout/gutters/

5. Recreate the following styles using the bootstrap grid:

Each page should display as shown until the screen is small, then each block should take up the full screen width.

Exercise 2 – Create the following page using bootstrap

1. Recreate the page shown below using the bootstrap grid system. Your page should look as shown below when viewed on a large screen (3,6,3) (Text provided in Moodle – see "Lab 21 - exercise2.txt" which is included on Moodle along with this lab)

Welcome to Bootstrap

This is the bootstrap grid

The boostrap grid is a mobile-first grid used to build layouts of all shapes and sizes thanks to a twelve column system, five default responsive tiers, and dozens of predefined classes.

History of Bootstrap

Bootstrap, originally named Twitter Blueprint, was developed by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden.

According to Twitter developer Mark Otto:

"A super small group of developers and I got together to design and build a new internal tool and saw an opportunity to do something more. Through that process, we saw ourselves build something much more substantial than another internal tool. Months later, we ended up with an early version of Bootstrap as a way to document and share common design patterns and assets within the company."

Once a container is in place, other Bootstrap layout components implement a CSS grid layout through defining rows and columns.

An open-source framework

Bootstrap is also known as a Front-end-framework.

Bootstrap requires the use of the HTML5 doctype. Without it, you will see some funky incomplete styling, but including it should not cause any considerable hiccups. Be sure to have your pages set up with the latest design and development standards. That means using an HTML5 doctype and including a viewport meta tag for proper responsive behaviors. Originally released on Friday, August 19, 2011, Bootstrap has since had over twenty releases, including two major rewrites with v2 and v3. Bootstrap 2 added responsive functionality to the entire framework as an optional stylesheet. Building on that, Bootstrap 3 rewrote the library once more to make it responsive by default with a mobile first approach.

2. Amend your code so that when the page is scaled to a medium size screen, it is displayed as shown below(12, 6, 6)

Welcome to Bootstrap

This is the bootstrap grid

The boostrap grid is a mobile-first grid used to build layouts of all shapes and sizes thanks to a twelve column system, five default responsive tiers, and dozens of predefined classes.

History of Bootstrap

Bootstrap, originally named Twitter Blueprint, was developed by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden.

According to Twitter developer Mark Otto:

"A super small group of developers and I got together to design and build a new internal tool and saw an opportunity to do something more. Through that process, we saw ourselves build something much more substantial than another internal tool. Months later, we ended up with an early version of Bootstrap as a way to document and share common design patterns and assets within the company."

Once a container is in place, other Bootstrap layout components implement a CSS grid layout through defining rows and columns.

An open-source framework

Bootstrap is also known as a Front-end-framework.

Bootstrap requires the use of the HTML5 doctype. Without it, you will see some funky incomplete styling, but including it should not cause any considerable hiccups. Be sure to have your pages set up with the latest design and development standards. That means using an HTML5 doctype and including a viewport meta tag for proper responsive behaviors. Originally released on Friday, August 19, 2011,

3. Amend your code so that when the page is scaled to a small size screen, it is displayed as shown on the next page - (12, 12, 12)

Welcome to Bootstrap

This is the bootstrap grid

The boostrap grid is a mobile-first grid used to build layouts of all shapes and sizes thanks to a twelve column system, five default responsive tiers, and dozens of predefined classes.

History of Bootstrap

Bootstrap, originally named Twitter Blueprint, was developed by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden.

According to Twitter developer Mark Otto:

"A super small group of developers and I got together to design and build a new internal tool and saw an opportunity to do something more. Through that process, we saw ourselves build something much more substantial than another internal tool. Months later, we ended up with an early version of Bootstrap as a way to document and share common design patterns and assets within the company."

Once a container is in place, other Bootstrap layout components implement a CSS grid layout through defining rows and columns.

An open-source framework

Bootstrap is also known as a Front-end-framework.

Bootstrap requires the use of the HTML5 doctype. Without it, you will see some funky incomplete styling, but including it should not cause any considerable hiccups. Be sure to have your pages set up with the latest design and development standards. That means using an HTML5 doctype and including a viewport meta tag for proper responsive behaviors. Originally released on Friday, August 19, 2011, Bootstrap has

Exercise 3 – Create a website using bootstrap.

In this exercise, you are required to create a website with 4 pages:

- index.html
- cart.html
- checkout.html
- opening.html

The website should consist of the following file / folder structure:
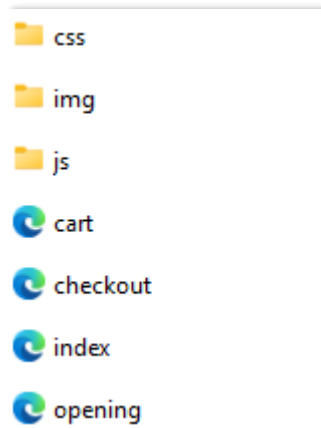
Download the images which are available along with this lab on Moodle – see lab_images.zip. Unzip this file and copy the images to the img folder.

In the js folder, create the following two files:

- checkout.js
- opening.js

In the css folder, create a file named index.css.

All web pages:

Take a copy of the bootstrap template - https://getbootstrap.com/docs/5.3/getting-started/introduction/#quick-start, option 2 - **Include Bootstrap's CSS and JS**.

Add this template to each you're your webpages.

Include a container on each of your web pages - https://getbootstrap.com/docs/5.3/layout/containers/#default-container

Add a navbar component to each web page and a padded div – refer to last week's lab for guidance with this.

See the following screenshot for example of this and how these components should be styled:
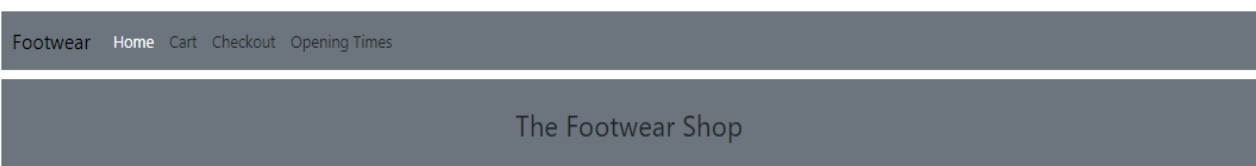
Please ensure all links are configured correctly so navigation between each of the pages is possible.

Requirements for each web page are included on the following pages:

index.html

Using the bootstrap grid system (https://getbootstrap.com/docs/5.3/layout/grid/), images in the img folder and card components (https://getbootstrap.com/docs/5.3/components/card/#example), create the following layout for a large screen:

For any smaller screen, the following layout should apply:

All images should be set to the same height, so create a rule in index.css to ensure this. Recommended value is 350px. A link to index.css should be included in index.html.

cart.html

On this webpage, add a list group component (https://getbootstrap.com/docs/5.3/components/list-group/). See screenshot below:

Footwear   Home   **Cart**   Checkout   Opening Times

The Footwear Shop

Socks
2.50

Shoes
75.00

Runners
50.00

Total
127.50

checkout.html

On this webpage, add a form (https://getbootstrap.com/docs/5.3/forms/overview/), with username input, password input and a submit button. See screenshot below:

Footwear   Home   Cart   **Checkout**   Opening Times

The Footwear Shop

Username

Password

Submit

On clicking the submit button, a check should be done to ensure username and password are valid. It should be possible to provide at least 3 valid usernames and passwords. Display an alert message indicating if username / password is valid / invalid.

All JavaScript for this page should be included in checkout.js. Please ensure this file is linked in checkout.html.

opening.html

Create a list group(https://getbootstrap.com/docs/5.3/components/list-group/) which includes the opening times of the shop as per screenshot below:

Just below the list group, include a paragraph - <p>.

Include JavaScript in opening.js which updates the text content of this <p> tag with a message indicating if the store is currently open or not, based on the times listed in the list group. If the store is currently open, set the colour of the relevant <li> tag to red. For example, if it is 5pm on Friday then the entry highlighted in red on the screenshot above is updated.

Please ensure opening.js file is linked in opening.html