



# Bootstrap

IMY 220 • Lecture 3

# Contents

What is Bootstrap?

Features of Bootstrap

How to use

Examples

# CSS

Fluid layouts can be tricky

End up doing a lot of hack-and-slash

Responsiveness complicates process a lot

For example, you cannot build responsive layouts using tables

Have to use divs that can slide underneath each other etc.

Cross-browser compatibility is also a problem

# CSS



# CSS



# CSS

Solution: Bootstrap



# What is Bootstrap?

Front end toolkit for rapidly developing web applications

HTML, CSS and JS framework for developing responsive, mobile first projects on the web

Originally made by Twitter to improve internal consistency, but was made public and Open Source on 19 August 2011

# What is Bootstrap?

Consists of:

- CSS file(s)
- JS file(s)
- ~~Some fonts/glyphs~~ (not anymore, as of v4)

Idea is to have one front-end framework to support every platform (desktop, tablet, mobile)

(One of the most popular projects on GitHub up to date)



# Features of Bootstrap

**12-column grid:** Uses a grid-system to easily create column-based layouts

**Responsive:** Caters for a range of screen resolutions and device sizes

**Cross-Browser:** Support for all modern browsers (down until IE7), including tablet and mobile

# How to use

Step 1: Download Bootstrap <https://getbootstrap.com/docs/5.3/getting-started/download/>

This includes:

- All the **CSS** files: bootstrap & bootstrap-theme, including minified versions
- **JavaScript** files for additional functionality, including minified versions

# How to use

Or link to the CDN version (<https://getbootstrap.com/docs/5.3/getting-started/download/>)

```
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.c
ss" integrity="sha384-
9ndCyUaIbzAi2FUVXJi0CjmCapSmO7SnpJef0486qhLnuZ2cdeRhO02iuK6FUUVM"
crossorigin="anonymous">
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.
min.js" integrity="sha384-
geWF76RCwLtnZ8qwWowPQNguL3RmwHVBC9FhGdlKrxdiJJigb/j/68SIy3Te4Bkz"
crossorigin="anonymous"></script>
```

# Quick aside: crossorigin and integrity

```
crossorigin="anonymous"
```

Specifies that a request for cross-origin resource sharing (CORS) is made without sending user credentials (as opposed to requests that may require user credentials to access content, such as scripts or images)

# Quick aside: crossorigin and integrity

```
integrity="sha384-  
geWF76RCwLtnZ8qwWowPQNguL3RmwHVBC9FhGdlKrxdiJJigb/j/68SIy3Te4Bkz"
```

?

# Quick aside: crossorigin and integrity

```
integrity="sha384-  
geWF76RCwLtnZ8qwWowPQNguL3RmwHVBC9FhGdlKrxdiJJigb/j/68SIy3Te4Bkz"
```

An SHA-hash that is used to validate the **integrity** of the resource, in other words, to check that it hasn't been tampered with.

(For example, if someone gained access to the CDN and injected extra content into the script)

[https://developer.mozilla.org/en-US/docs/Web/HTML/CORS\\_settings\\_attributes](https://developer.mozilla.org/en-US/docs/Web/HTML/CORS_settings_attributes)

[https://developer.mozilla.org/en-US/docs/Web/Security/Subresource\\_Integrity](https://developer.mozilla.org/en-US/docs/Web/Security/Subresource_Integrity)

# Including Bootstrap

Which is better: Local vs. hosted (CDN)?

Why?

# Including Bootstrap

Answer: CDN

Many users may have downloaded jQuery from the Google CDN which means it is **cached**.

It might be located **physically** closer (geolocation).

Requires less **bandwidth** from your site (since less files are accessed from your site)

Caveat: CDN version might be down



# Including Bootstrap

Best of both: link to CDN version by default and to local version as a fallback:

<https://stackoverflow.com/questions/26192897/should-i-use-bootstrap-from-cdn-or-make-a-copy-on-my-server>

```
<head>
  <!-- Bootstrap CSS CDN -->
  <link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.0.3/css/bootstrap.min.css" />
</head>
<body>
  <!-- APP CONTENT -->

  <!-- jQuery CDN -->
  <script src="//ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js"></script>
  <!-- jQuery local fallback -->
  <script>window.jQuery || document.write('<script src="/local/jquery.min.js"></script>')</script>
  <!-- Bootstrap JS CDN -->
  <script src="//netdna.bootstrapcdn.com/bootstrap/3.0.3/js/bootstrap.min.js"></script>
  <!-- Bootstrap JS local fallback -->
  <script>if(typeof($.fn.modal) === 'undefined') {document.write('<script src="/local/bootstrap.min.js"></script>')}</script>
  <!-- Bootstrap CSS local fallback -->
  <div id="bootstrapCssTest" class="hidden"></div>
  <script>
    $(function() {
      if ($('#bootstrapCssTest').is(':visible')) {
        $("head").prepend('<link rel="stylesheet" href="/local/bootstrap.min.css">');
      }
    });
  </script>
</body>
```

# Pracs + assignments + project

When including CDN versions (of anything), refer to the website of what you're including

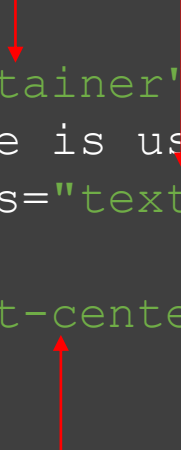
The website will usually have a section with CDN links to the **newest version** of the files

# How to use

Step 3: Use Bootstrap's styling/formatting by using their predefined **element structures** and **class names**

For example:

```
<body>
  <div class="container">
    <h1> This page is using Bootstrap!
      <small class="text-muted"> It looks better already! </small>
    </h1>
    <p class="text-center"> This text is centered! </p>
  </div>
</body>
```



# How to use

Side note: remember to always include the following tag for correct responsive behaviour

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

# How to use

Step 3 (continued): All of bootstrap's predefined mobile first, cross-browser styling is attached to specific **class names**

So, to use the CSS for specific stylings/formattings/elements, you must use the correct class name as coded in the bootstrap.css file

# How to use

And that's basically it!

It's quite a huge library (so we're not going to cover everything)

Go to <http://getbootstrap.com/> to look up specific styling & formatting tools

- **CSS** (global CSS, grid system, etc.)
- **Components** (cards, dropdowns, input groups, etc.)
- **JavaScript** (Extra functionality with jQuery plugins)

# Container

Bootstrap **requires** that you nest all your content in a container for certain functionality, like the grid system.

Bootstrap has two types of containers

Responsive, fixed width:

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

Full width (fills up whole screen width):

```
<div class="fluid-container">  
</div>
```

# Grid System

As previously mentioned, Bootstrap uses a 12-column grid system that scales easily for different devices

To use this, first create a row-element to nest all your content in:

```
<div class="row">  
  
</div>
```

Only columns may be the direct children of rows



# Grid System

Then, create columns for content

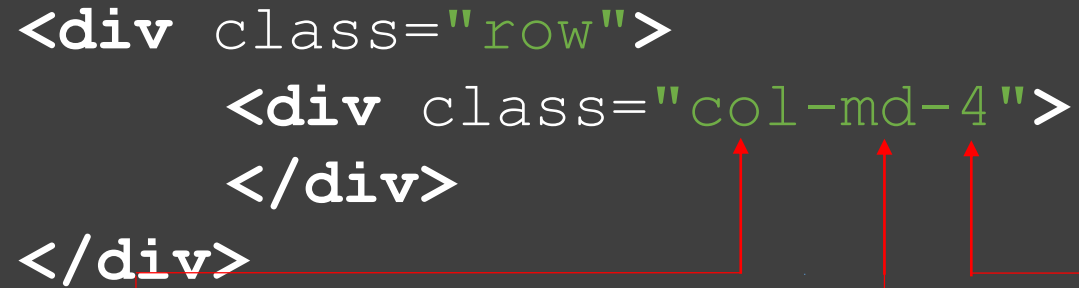
Columns will typically be divs and must always have class names starting with *col-*

for example, `col-md-4 col-sm-6, col-xl-8`

Let's look at a specific column-div and what the class name means

# Grid System

```
<div class="row">  
  <div class="col-md-4">  
  </div>  
</div>
```



div is a column

Applies to a  
desktop screen  
size

Width will be  
4/12 of the  
container

# Grid System

In the example, *md* refers to the viewport size (the screen size)

Bootstrap has **6 grid classes** for screen widths:

- xxl: extra extra large ( $\geq 1400\text{px}$ )
- xl: extra large ( $\geq 1200\text{px}$ )
- lg: large ( $\geq 992\text{px}$ )
- md: medium ( $\geq 768\text{px}$ )
- sm: small ( $\geq 576\text{px}$ )
- xs: extra small ( $< 576\text{px}$ )

So, if the screen width becomes smaller than 768px, bootstrap will look for a class name that starts with *col-sm*

# Grid System

The number refers to the width of the column

The grid system logically divides the row into 12 columns

The number is number of 12ths of the row that the column takes up

# Grid System

[illegible]

# Grid System

A class name of “**col-md-4 col-sm-6**” means that the width of the column will be **4/12** of the row on desktop screen sizes and **6/12** on md screen sizes

Take note that grid classes apply to screen sizes greater than or equal their breakpoint sizes.

That means if you define a *col-sm* size, but no *col-md*, *col-lg* or *col-xl* size, the width defined by the *col-sm* class name will be used for *col-sm*, *col-md*, *col-lg* and *col-xl*

# Grid System

Column sizes that add up to more than 12 will simply stack underneath the other columns

```
<div class="row">  
  <div class="col-md-6">Column 1</div>  
  <div class="col-md-6">Column 2</div>  
  <div class="col-md-6">Column 3</div>  
</div>
```

So, if you have three *col-md-6* columns, the result will be something like this:

Column 1	Column 2
Column 3	

# Grid System

## Columns can be nested

```
<div class="row">  
  <div class="col-sm-9">  
    Level 1: .col-sm-9  
    <div class="row">  
      <div class="col-8 col-sm-6">  
        Level 2: .col-8 .col-sm-6  
      </div>  
      <div class="col-4 col-sm-6">  
        Level 2: .col-4 .col-sm-6  
      </div>  
    </div>  
  </div>  
</div>
```

Level 1: .col-sm-9	
Level 2: .col-8 .col-sm-6	Level 2: .col-4 .col-sm-6



# Grid System

If the sizes is not set, equal column sizes are automatically created to fill the width of the row

```
<div class="row">  
  <div class="col">Column 1</div>  
  <div class="col">Column 2</div>  
  <div class="col">Column 3</div>  
</div>
```

Column 1	Column 2	Column 3
----------	----------	----------

# Grid System

If the sizes are only set for some columns within a row, the other (unset) columns will be evenly spaced to fill the width of the row

```
<div class="row">  
  <div class="col-5">Col 1</div>  
  <div class="col">Col 2</div>  
  <div class="col">Col 3</div>  
</div>
```

Col 1	Col 2	Col 3
-------	-------	-------

# Grid System

In the example below, Col 2 and Col 4 are automatically given the same width

```
<div class="row">  
  <div class="col-5">Col 1</div>  
  <div class="col">Col 2</div>  
  <div class="col-4">Col 3</div>  
  <div class="col">Col 4</div>  
</div>
```

Col 1	Col 2	Col 3	Col 4
-------	-------	-------	-------

# Grid System

You can also use this auto-sizing capability on rows with columns that have variable widths, using `col-{breakpoint}-auto`, where `{breakpoint}` is the name of the grid class for example:

```
<div class="row">  
  <div class="col">Col 1</div>  
  <div class="col-xs-auto">  
    The width of this column determines the width of the other two  
  </div>  
  <div class="col">Col 3</div>  
</div>
```

Col 1	The width of this column determines the width of the other two	Col 3
-------	--	-------

# Grid System

More information about the grid system can be found here:  
<https://getbootstrap.com/docs/5.3/layout/grid>

# Grid System Alignment

The alignment of columns within the grid can be adjusted with justify-content-\* classes

For example, if we only have two columns and we want to centre them:

```
<div class="row justify-content-center">  
  <div class="col-3">Column 1</div>  
  <div class="col-3">Column 2s</div>  
</div>
```



# Grid System Alignment

More information about vertical and horizontal alignment can be found here: <https://getbootstrap.com/docs/5.3/layout/columns/>

# Spacing

As of v4, Bootstrap includes some shorthand classes for adding margins and padding to elements

These classes have a notation of `{property}{side(s)}-{breakpoint}-{size}`

- property can be `m` (margin) or `p` (padding)
- side(s) can be `t` (top), `b` (bottom), `s` (start/left), `e` (end/right), `x` (left & right), or `y` (top & bottom)
- breakpoint works the same as the grid system: `xs`, `sm`, `md`, `lg`, `xl`, `xxl`
- size can be a value from `0` – `5` or `auto`. Numerical values set the `rems` to one of a series of preset values and `auto` sets it to `auto`



# Spacing

rems sets the size (usually the font size) relative to the root (HTML) size

- 0 - for classes that eliminate the `margin` or `padding` by setting it to 0
- 1 - (by default) for classes that set the `margin` or `padding` to `$spacer * .25`
- 2 - (by default) for classes that set the `margin` or `padding` to `$spacer * .5`
- 3 - (by default) for classes that set the `margin` or `padding` to `$spacer`
- 4 - (by default) for classes that set the `margin` or `padding` to `$spacer * 1.5`
- 5 - (by default) for classes that set the `margin` or `padding` to `$spacer * 3`
- `auto` - for classes that set the `margin` to `auto`

# Spacing

For example

```
<!-- set the top margin to 1.5 rems on all sizes -->
<div class="mt-xs-4"></div>

<!-- set the left padding to .5 rems on sm and higher -->
<span class="ps-sm-2"></span>

<!-- set the left and right margin to auto on md and higher -->
<!-- also set the bottom padding to 1 rem on lg and higher -->
<img class="mx-md-auto pb-lg-3" />
```

# Spacing

More information about spacing classes can be found here:

<https://getbootstrap.com/docs/5.3/utilities/spacing/>

# Typography

Bootstrap includes support and CSS for some basic typography uses:

```
<p class="text-decoration-line-through">This text has a line going through</p>  
<p class="text-decoration-underline">This text is underlined</p>  
<p class="fw-bold">This line rendered as bold text.</p>  
<p class="fst-italic">This line rendered as bold text.</p>
```

~~This text has a line going through~~

This text is underlined

**This line is rendered as bold text.**

*This line is rendered as italic text.*

# Typography

Bootstrap also has classes for alignment, transformations (e.g., upper- or lowercase), and font size, which you can find here:

<https://getbootstrap.com/docs/5.3/utilities/text/>

# Tables

Bootstrap also has different table styles, such as:

- Basic
- Striped
- Bordered
- Condensed

To use Bootstrap's table styles, you **must** give your table a class of "table"

# Tables

## Example: basic

```
<table class="table">
<tr>
  <th> Name </th>
  <th> Surname </th>
</tr>
<tr>
  <td> Troy </td>
  <td> Abed </td>
</tr>
<tr>
  <td> Barnes </td>
  <td> Nadir </td>
</tr>
</table>
```

Name	Surname
Troy	Abed
Barnes	Nadir

# Tables

## Example: bordered

```
<table class="table table-bordered">
<tr>
  <th> Name </th>
  <th> Surname </th>
</tr>
<tr>
  <td> Troy </td>
  <td> Abed </td>
</tr>
<tr>
  <td> Barnes </td>
  <td> Nadir </td>
</tr>
</table>
```

Name	Surname
Troy	Abed
Barnes	Nadir



# Tables

You can also colour the table rows and columns using contextual classes

```
<table class="table table-bordered">
<tr>
  <th> Name </th>
  <th> Surname </th>
</tr>
<tr class="table-warning">
  <td> Troy </td>
  <td> Abed </td>
</tr>
<tr>
  <td> Barnes </td>
  <td class="table-danger"> Nadir </td>
</tr>
</table>
```

Name	Surname
Troy	Abed
Barnes	Nadir

# Tables

More information about tables can be found here:

<https://getbootstrap.com/docs/5.3/content/tables/>

# Forms

Forms are styled using the appropriate “form-\*” classes, for example

```
<form>
  <div>
    <label for="email1" class="form-label">Email</label>
    <input type="email" name="email" id="email1" class="form-control" />
    <div id="emailHelp" class="form-text">We'll never share your details</div>
  </div>
  <div>
    <button type="submit" class="btn btn-primary"> Submit </button>
  </div>
</form>
```

Email

We'll never share your details

Submit

# Forms

More information about forms can be found here:

<https://getbootstrap.com/docs/5.3/forms/>

# Responsive Utilities

The display mode for content on different screen-sizes can easily be changed using **utility classes**

The notation for utility classes is as follows:

- *.d-{breakpoint}-{value}*
- *Where {breakpoint} is a viewport size (xs/sm/md/lg/xl/xxl) and {value} is a display type (block, inline, inline-block, none)*

For example, to make an element hidden on xs-size, use the class name “d-xs-none”

Example 2, to make an element display as an inline-block on sm-size, use the class name “d-sm-inline-block”

# Responsive Utilities

More information about responsive utilities can be found here:  
<https://getbootstrap.com/docs/5.3/utilities/display/>

# Components

Bootstrap includes some useful web design components, such as:

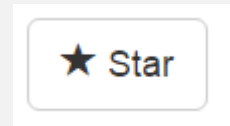
- Dropdowns
- Navbars
- Breadcrumbs
- Progress bars
- Etc.

They can be found here:

<https://getbootstrap.com/docs/5.3/components>

# Glyphicons

As mentioned before, Bootstrap used to include a bunch of Glyphicons for common website functionality, such as:



This has been removed as of v4

So, for this module we will be using another icon set for the same purpose: Linelcons



# Line Icons

To use Line Icons, first include the stylesheet...

```
<link rel="stylesheet" href="https://cdn.lineicons.com/4.0/lineicons.css" />
```

...then add an icon by creating an i-element and adding two classes:

- Ini (the official Line Icons prefix)
- Ini-{icon name} where {icon name} is the name of an official LI icon

# Line Icons

For example...

```
<i class="lni lni-pizza"></i>
```

...creates:

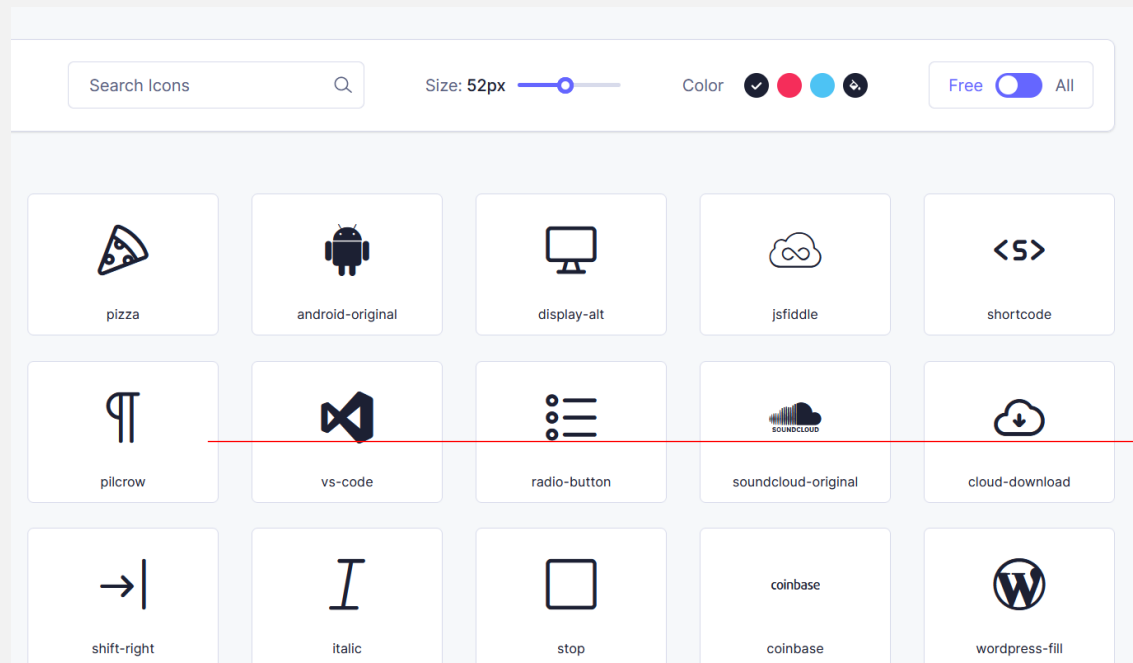


How this works: Line Icons is a font that includes a bunch of icons as characters.

Since fonts are vectors, they can be resized without losing quality  
(This also means they have to be resized with **font-size**)

# Line Icons

Go to <https://lineicons.com/icons/> to browse the list of icons  
(Use the button at the top-right to only select freely available icons)



```
<i class="lni lni-pilcrow"></i>
```

# JavaScript Functionality

The JavaScript in bootstrap.js (or bootstrap.min.js) includes functionality for some common website functionality, such as:

- Modals
- Tabs
- Tooltips
- Popovers
- Etc.

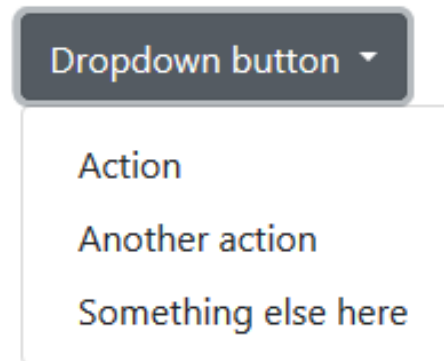
# JavaScript Functionality

You generally don't have to write any JavaScript yourself to use these included functionalities, except for options/customization/manually overriding.

# JavaScript Functionality

For example:

```
<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" type="button"
id="dropdownMenuButton" data-bs-toggle="dropdown" aria-expanded="false">
    Dropdown button
  </button>
  <div class="dropdown-menu">
    <a class="dropdown-item" href="#">Action</a>
    <a class="dropdown-item" href="#">Another action</a>
    <a class="dropdown-item" href="#">Something else here</a>
  </div>
</div>
```



# JavaScript Functionality

Many of the components include JavaScript functionality:

<https://getbootstrap.com/docs/5.3/components>

# Conclusion

Bootstrap doesn't **decorate** a webpage, but rather styles and formats it according to some accepted standard.

To use Bootstrap (generally speaking) check if the style/functionality that you want is on <http://getbootstrap.com/> and get the code there.

The official documentation can be found here:  
<https://getbootstrap.com/docs/5.3>



# Important

You still need to add your own style sheets to decorate your website.

Project: have to redesign all form elements (inputs, buttons, etc.)

Look for inspiration on web design, including specific aspects, such as forms, e.g.,

- <https://www.awwwards.com/>
- <https://www.behance.net/search/projects?search=web+forms+design>

# Important

There's only one way to really know if your site is usable...

...you have to do user testing

- Ask people (especially non-IT people) to use your site.
- Observe them as they do so (don't help them or say anything)
- If they struggle, you need to improve it

<https://www.nngroup.com/videos/user-testing-jakob-nielsen/>

# Sources

- <http://getbootstrap.com/>
- <https://blog.twitter.com/2011/bootstrap-twitter>
- <https://lineicons.com/>