

COMM 2555: Interactive Digital Communication / Spring 2018

Lab 10: Simplifying CSS layout with Bootstrap

Goals

- Learn a layout technique that may be simpler than manually specifying all the margins, padding, floats and other CSS properties (likely you did in the previous lab).
- Learn about responsive design (it's really layout, not design) using Bootstrap, a CSS framework that does the heavy lifting for you.

Step 1. Create your infrastructure

1.1. Create new folder, call it **lab10**.

1.2. Download the file lab10.zip from the course website and extract its contents into your **lab10** folder. You'll see these files:

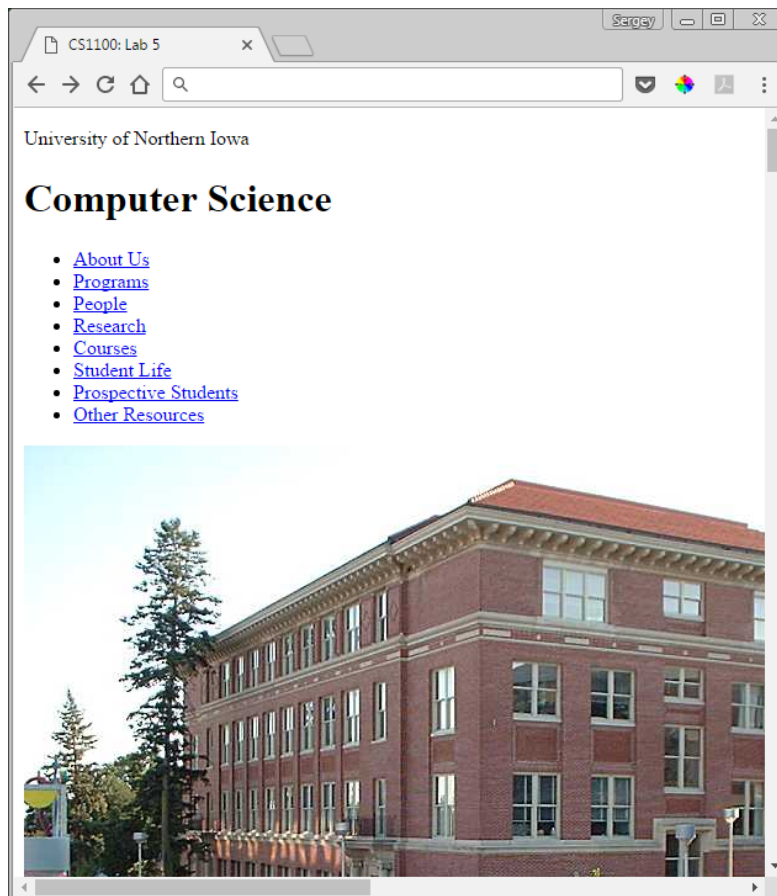
- index.html
- several txt files

The html file contains the bare-bones HTML - that's what you'll start with.

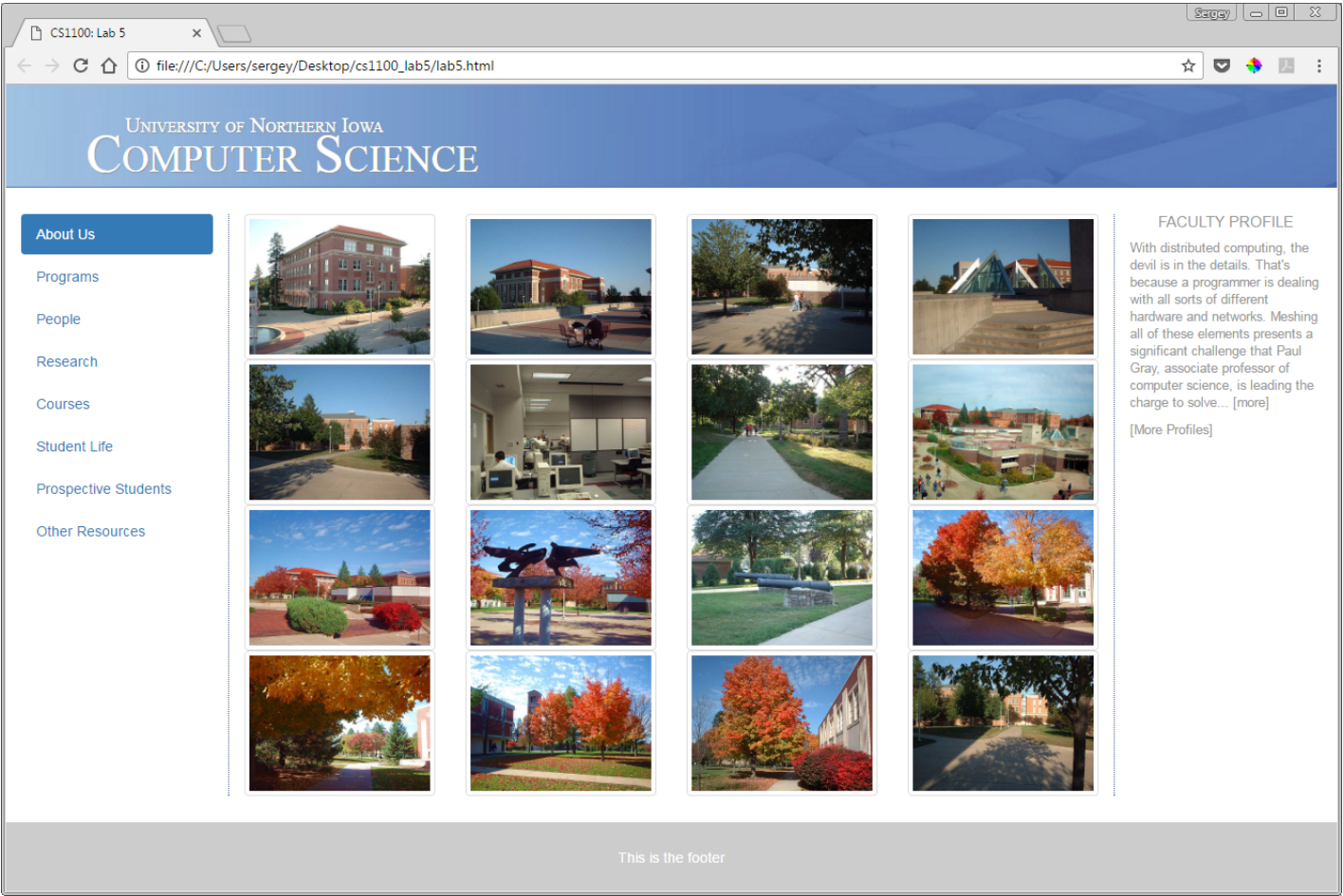
The other files are part of the code you will be adding.

1.3. In the same folder, create a **style.css** file and link it to your web page.

Open your index.html page in the browser. Yes, of course, you start with nothing:



This is your goal:



...which will look like this when viewed on a smaller screen (when you make your screen narrow, or when you view it on a smaller device like a phone)



Step 2. Introduction to Bootstrap

We will achieve our today's goal using the Bootstrap framework.

The best resource on Bootstrap is the Bootstrap web site: <http://getbootstrap.com/>

However, before that, try a brief introduction to Bootstrap in this tutorial: <https://www.w3schools.com/>

Click the "Start learning Bootstrap" green button to start - you will be walked through the main concepts, page by page (you can view your progress on the left side bar).

The Get Started and Grid Basic (the grid system) are the two most important pages. The rest are optional - they all offer you more layout (and more!) that you can use to easily layout and style your webpages.

Once you have a good idea about what Bootstrap is, look at the following pages (you know enough to understand anything here):

<http://getbootstrap.com/getting-started/>

<http://getbootstrap.com/css/>

<http://getbootstrap.com/components/>

You **DON'T NEED TO READ ALL OF THEM** carefully. They offer the same type of information as does the w3schools links, but that's the official guide so it will offer you more depth.

As a recap, the following **essential** information is borrowed from w3schools

What is Bootstrap?

- Bootstrap is a free front-end framework for faster and easier web development

- Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins
- Bootstrap also gives you the ability to easily create responsive designs

Advantages of Bootstrap:

- Easy to use: Anybody with just basic knowledge of HTML and CSS can start using Bootstrap
- Responsive features: Bootstrap's responsive CSS adjusts to phones, tablets, and desktops
- Mobile-first approach: In Bootstrap 3, mobile-first styles are part of the core framework
- Browser compatibility: Bootstrap is compatible with all modern browsers

Where to Get Bootstrap?

There are two ways to start using Bootstrap on your own web site.

You can:

- a. Download Bootstrap from getbootstrap.com
- b. Include Bootstrap from a CDN (Content Delivery Network)

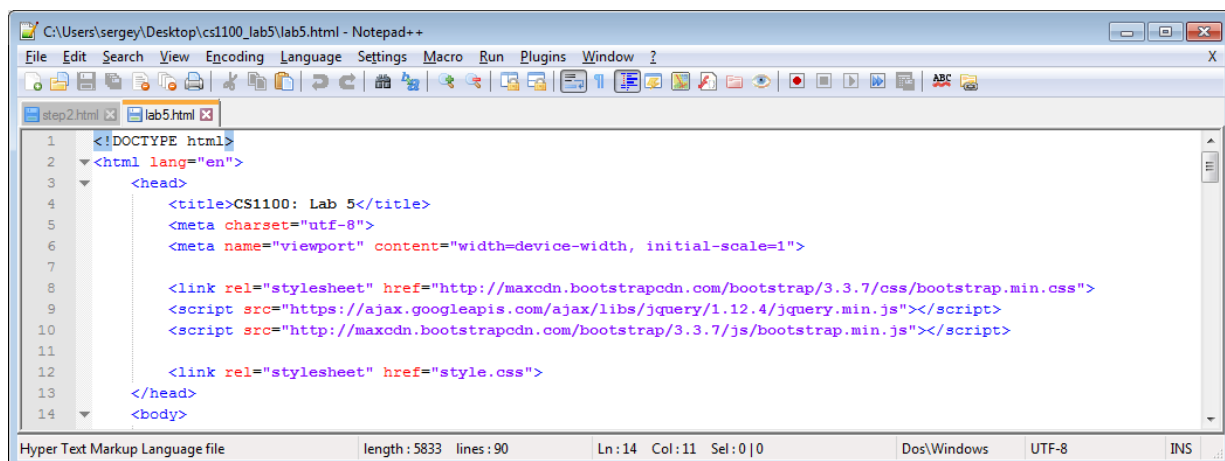
Advantages of using a CDN:

Many users already have downloaded Bootstrap from MaxCDN when visiting another site. As a result, it will be loaded from cache when they visit your site, which leads to faster loading time. Also, most CDN's will make sure that once a user requests a file from it, it will be served from the server closest to them, which also leads to faster loading time.

Step 3. Add essential code to your HTML

Place the code from code_header.txt inside your HEAD element (between the <head></head> tags, but ABOVE the link to your own style.css). Refresh your browser. As you see, Bootstrap comes with its own styles (and we can override any of them!)

Note: this lab uses Bootstrap version 3. Bootstrap has recently released version 4. For your website projects, if you decide to use Bootstrap, you may use either one.



```

1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <title>CS1100: Lab 5</title>
5     <meta charset="utf-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7
8     <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
9     <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
10    <script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
11
12    <link rel="stylesheet" href="style.css">
13  </head>
14  <body>

```

Step 4. Build your grid with Bootstrap

4.1. Bootstrap uses a grid that contains up to 12 columns. Here's a brief overview:

http://www.w3schools.com/bootstrap/bootstrap_grid_system.asp

So, you build your grid by placing your content inside DIV elements, and then assign special classes to these elements. Here's an example of a simple 1-column grid:

```

<div class="container-fluid">
  <div class="row">
    <div class="col-sm-12">
      Your content goes here!
    </div>
  </div>
</div>

```

Here are 3 columns:

```

<div class="container-fluid">
  <div class="row">
    <div class="col-sm-3">
      Your content goes here!
    </div>
    <div class="col-sm-6">
      Your content goes here!
    </div>
    <div class="col-sm-3">
      Your content goes here!
    </div>
  </div>
</div>

```

Here's the idea:

1. Everything goes inside a "container" div
2. The "container" div may contain ONLY "row" divs
3. The "row" divs may contain ONLY "column" divs
4. The roles of the divs are designated by assigning class names:
 - a. container OR container-fluid
 - b. row
 - c. col-[*type*]-[*number*] - this is a template for class names

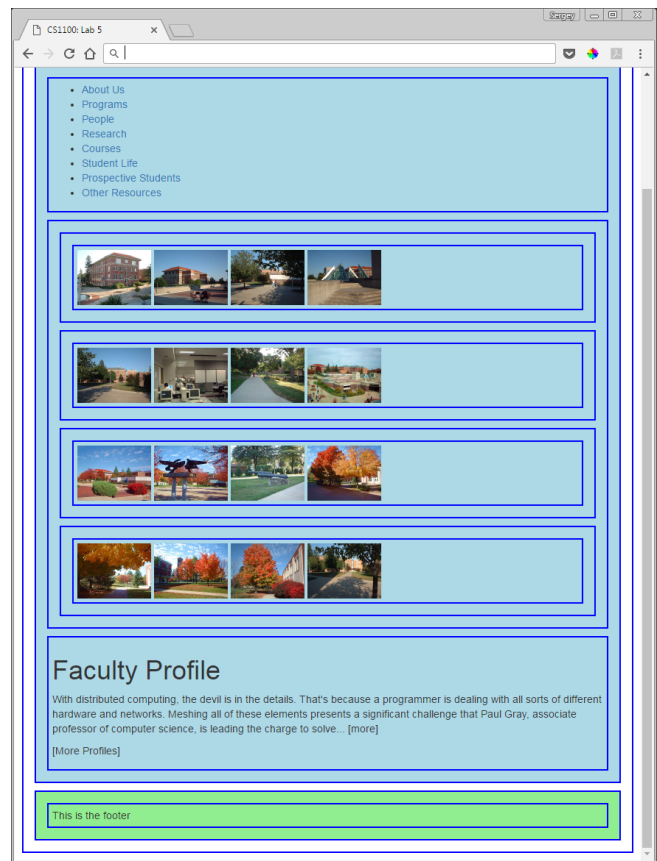
The Bootstrap grid system has four *types*:

- xs (for phones)
- sm (for tablets)
- md (for desktops)
- lg (for larger desktops)

The *number* in the class name can be anything from 1 to 12, as long as **all the columns in a row add up to 12**. (If they don't add up, the remainder will carry over to a new line, which may or may not be what you intended.)

4.2. To build our grid, we need:

1. Place everything inside one div
2. We start with 3 main rows (*I added this to my style.css for now: `img { width: 100px; }`*)



3. Each row has columns (1, 3, 1)

4. Add the following class names to your divs:

- container-fluid
- row
- col-sm-[number] (where number is between 1 and 12).

Here is your grid:


```
15 <body>
16
17 <div class="container-fluid">
18
19 <div class="row">
20 <div class="col-sm-12">
21 header goes here
22 </div>
23 </div>
24
25 <div class="row">
26 <div class="col-sm-2">
27 menu goes here
28 </div>
29
30 <div class="col-sm-8">
31 images go here (we'll do a nested row)
32 </div>
33
34 <div class="col-sm-2">
35 faculty profile goes here
36 </div>
37 </div>
38
39 <div class="row">
40 <div class="col-sm-12">
41 footer goes here
42 </div>
43 </div>
44
45 </div>
46
47 </body>
```

4.3. We'll nest 4 rows inside the center column (yes, we can do that!). Open the code_images.txt file and paste its contents inside the "col-8'sm" column.

*Make sure to remove the temporary **img { width: 100px; }** code you made added to your style.css.*

Look at the code we've added:

- Each image is inside a DIV - that's a Bootstrap column!
- There are 4 such columns in each nested row, and their width is 3 units. $3 \times 4 = 12$
- Each image has a class: **class="img-thumbnail"** - that is a Bootstrap style (which you may override)
- The top row images have an additional class added: "col-xs-6" - they will be displayed as 2 columns on a small device, unlike the rest.

4.4. The thumbnails look very nice! Why not use Bootstrap to style the menu? Again, we simply add several class names: change your **** to **<ul class="nav nav-pills nav-stacked">**

(Remember not to copy and paste from this pdf - type it yourself.)

You can also add this to any of the menu items: **<li class="active">**

Step 5. Add your own styling

Of course Bootstrap would be of little utility if we couldn't override its styles. But we can!

5.1. Add the code from code_your_styles.txt to your style.css file.

Of course, nothing happens. That's because we need to add our own class names to our HTML! (yes, we could override Bootstrap classes, but we would then risk breaking our layout - and who needs that! Here's what we do:

1. Add "header" class name to the top row
2. Add "center" to the column that contains the nested image rows
3. Add "right" to the right column
4. Add "footer" to the bottom row

Here's your new code:

```
<div class="row header">
  <div class="col-sm-8 center">
    <div class="col-sm-2 right">
      <div class="col-sm-12 footer">
```

And that's it!

Look at your page in a browser. If you are happy with what you see, you may upload your code to weblab. I suggest placing it under weblab.uni.edu/YOU/labs/10/ . You can now see what this looks like on your cell phone.

Submit your work

Save your lab10 folder as a zip file and submit the zip file to eLearning:

<https://bb9.uni.edu> > log in with CatID > our course > Course Content > Labs > Lab 10

And you're done!