# Microsoft: Introduction to Bootstrap*

August 28, 2015

## Contents


**1 Getting Started with Bootstrap** **1**
- 1.1 Introduction to Bootstrap . . . 1
- 1.2 Page Layouts . . . 1
  - 1.2.1 Bootstrap Classes . . . 1
  - 1.2.2 Common Bootstrap Starting Classes . . . 2
  - 1.2.3 Bootstrap Grid System . . . 2
  - 1.2.4 Grid Notes . . . 2
- 1.3 Lab . . . 3
  - 1.3.1 Part 1 Notes, Steps and Scenario . . . 3
  - 1.3.2 Part 2 Wireframe . . . 3

**2 Page Design and Data Display** **5**

**3 Forms and User Interaction** **5**


# 1 Getting Started with Bootstrap

During our first module we'll cover the basics of Bootstrap and the grid system. You'll see how to obtain and add Bootstrap to your pages, and why Bootstrap makes developing responsive pages simpler than using CSS directly. We'll also explain how the grid system works to layout your pages.

## 1.1 Introduction to Bootstrap

## 1.2 Page Layouts

### 1.2.1 Bootstrap Classes

Most everything that you want to add onto a page by using Bootstrap is done by adding classes and HTML to the page. There are numerous Bootstrap classes. Fortunately, you don't have to know or memorize all of them.

Many of the classes are related to one another, so learning the basic concepts of what makes up certain classes is all that's needed. There are many that are pretty straight forward, such as left and right, which will do exactly what you expect. And there are still others which you may never find yourself using, which is just fine. You will notice that the goal of this course is not to teach you every Bootstrap class that's available, but rather how to use Bootstrap in your applications.

As has already been stated, Bootstrap uses CSS. This means you are free to customize all classes as needed by using standard CSS and cascading rules.

*Updated by jibi-edx

### 1.2.2 Common Bootstrap Starting Classes

When putting together the layout for a page, there are a few classes that you typically use to start. The first two help define how much space the content will take on the overall page, while the last will allow you to highlight content, such as the site's title and introductory information.

| Classes | Information |
|---|---|
| container | The container class places content inside of a horizontal container. The size of the container will vary based on screensize. The container will stay the same width for that particular screensize. This helps developers size content appropriately for different devices. |
| container-fluid | The container-fluid class places content inside of a container that will always be the width of the screen. This is helpful for scenarios where the content to be displayed must use the entire browser window. |
| jumbotron | The jumbotron class is a common class for displaying titles for different sections of a website, or the landing page. The jumbotron typically has a highlighted background and an increased fontsize. |

### 1.2.3 Bootstrap Grid System

Many web developers have used tables to layout pages. Tables made it easier for the web developer, because the developer didn't have to worry about either the complexities of CSS, or cross-browser issues.

Unfortunately, while tables may have made it easier for the developer, it makes it more of a challenge for both web browsers and accessibility. For traditional browsers, complex table structures can degrade the performance of page layout. For screen readers, tables are read assume tabular data, which may not be how the page is actually layed out and designed to be presented to the user.

With this in mind, it is never a best practice to do page layout using tables. Bootstrap offers a grid system so developers don't feel the urge to layout pages using tables.

The Bootstrap grid behaves like a table. There are rows and columns. However, unlike tables, Bootstrap uses CSS to manage the layout of the pages. In addition, Bootstrap offers support for mobile browsers and different sized screens by providing different grids for different screen sizes. There are four screensizes provided by Bootstrap grids. This allows developers to size content appropriately for various devices.

### 1.2.4 Grid Notes

1. Always 12 columns
2. Smaller sizes set the default for larger sizes

- col-sm- would set the size for small, medium and large screens unless overridden by a setting specific to one of the larger sizes

3. There is a 30px "gutter", or empty space, between each column (15px on each side of the column)
4. The gutter on the outside of the container varies based on the current screen size

- Extra small devices always use the entire width of the screen

| | Extra small devices e.g. phones (< 768 px) | Small devices e.g. tablets (≥ 768 px) | Medium devices e.g. laptops (≥ 992 px) | Large devices e.g. desktops (≥ 1200 px) |
|---|---|---|---|---|
| Container width | None (auto) | 750px | 970px | 1170px |
| Column class prefix | col-xs- | col-sm- | col-md- | col-lg- |
| Column width | Auto | 60px | 75px | 95px |

The screens sizes that Bootstrap uses are not random, but rather based on research on common device sizes. While you can change the screen sizes, because Bootstrap does use CSS, you want to use the defaults provided.

## 1.3 Lab

### 1.3.1 Part 1 Notes, Steps and Scenario

**Lab Notes**

Labs are designed to both be a challenge as well as real-world. As a result, full step-by-step instructions are not provided. However, all of the needed information, class names, and resources to complete the labs have been provided. This content is either in the videos provded, or in the resources section.

If you do get stuck you are free to discuss the lab with other students.

Keep in mind that future labs will continue to build on this one. Successful completion of the labs will require completing the labs from the module prior.

**Lab Steps**

Each lab will provide various resources. As you attempt the lab, try to solve the lab by using the earliest step possible. Here are the resources provided for this lab:

1. Wireframe.

In the real world, a wireframe is generated to represent the design the business owner desires. Being able to translate a wireframe into a final product is a key skill all web developers need to have. Remember, a wireframe does not need to be matched exactly; a wireframe is intended to be a rough sketch of the desired result.

2. Screenshot

If you're just getting started, or if the wireframe isn't clear about the target design, you can look at the screenshot. This is a sample of what the final result should look like.

3. Additional guidance

The additional guidance is provided to add clarity and help add context if needed.

**Lab Scenario**

You have been tasked with creating a page to display the details of a record album. You will build this page up over the course of the next several labs. The starting page only needs to target medium and large devices; support for smaller devices will be added later.

These are the necessary components:

1. A banner to display the name of the music store, Music Store
2. Album information

- An image on the left side to represent the cover art. The image is provided here.
- The basic information on the album on the right side.

Finally, you have been requested to implement a new theme that is different from the default.

You are free to choose whatever theme you like, and are free to determine the sizing that you believe will work best for the page.

### 1.3.2 Part 2 Wireframe

Use the wireframe to layout your page (fig 1).

**Part 3 Screenshot**

This is what the solution should look like (fig 2). The coloring will be different based on the theme you select.

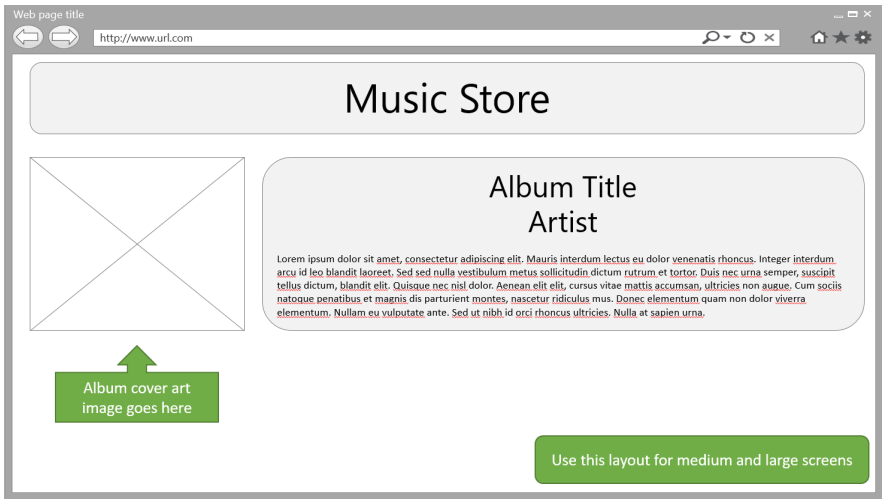


**Figure 1.** Use the wireframe to layout your page.

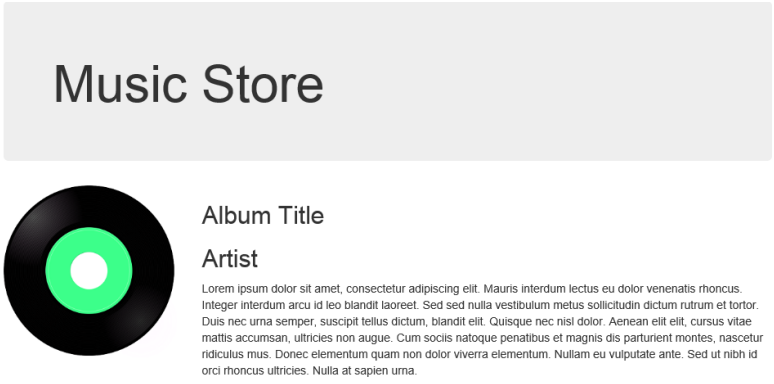


**Figure 2.** The coloring will be different based on the theme you select.

**Part 4 Additional Guidance**

1. Download Bootstrap components from www.getbootstrap.com.
2. Download the album art.
3. Create a new project in Visual Studio. Use the empty web template.
4. Add a folder named "Styles". Add the Bootstrap css files to this folder.
5. Add a folder named "Scripts". Add the JavaScript files to this folder.
6. Add a new HTML page named AlbumDetails.html.
7. Add references to the Bootstrap min CSS file to AlbumDetails.html.
8. Add the necessary HTML to implement the layout. Use the phrases Music Store, Album Title, and Artist as placeholders as shown in the wireframe.
9. Obtain a new theme from bootswatch.com. Implement the new theme.

# 2 Page Design and Data Display

Bootstrap offers several features to ensure your page is designed optimally for all screen sizes. You'll examine how to best use those capabilities to ensure users can easily consume the data they need regardless of their device type. You'll also see how to use some of the advanced components, or controls, that Bootstrap offers.

# 3 Forms and User Interaction

We'll close our exploration of Bootstrap by taking a look at how we can obtain data from the user and provide feedback. Bootstrap offers great support for customizing forms, and several options to let the user know when the form is correct or incorrect. You'll see how to add alerts and modal dialog boxes, as well as inline validation and data prompts.