Lesson **Thursday**

Introduction to Programming (/introduction-to-programming) / Git, HTML and CSS (/introduction-to-programming/git-html-and-css) / Introduction to Bootstrap and Front-End Frameworks

Text Cheat sheet

Even if you don't pursue a career in web design, you'll need to know enough about CSS to apply the correct classes to your code to integrate a designer's CSS. You'll also need to know how to make small changes to CSS as required by your project. This is why it's so important for both web developers *and* designers to understand CSS basics.

However, it's possible to create a website that looks nice and is responsive to a user's screen size without spending a lot of time designing the user interface. To do this, developers make use of pre-built tools from front-end frameworks.

In this lesson we'll learn what a front-end framework is, how they work, and we'll begin familiarizing ourselves with **Bootstrap**, one of the most popular front-end frameworks for design (https://getbootstrap.com/). We'll also take a moment to learn other frequently used terms like "library" and "toolkit" that offer similar functionality to what frameworks offer.

Then, in the following lessons we'll learn how to use Bootstrap in a project.

While we do want to learn how to make our projects look nice without much effort, an equally important goal is to learn how to use external tools in our projects. That's because when we learn how to use Bootstrap, we'll also learn the general process of learning to work with any library, toolkit, or framework. This includes understanding three important things:

1. How to download a tool and load it in our project.
2. How the tool works and how to use in our projects.
3. How to navigate the tool's online documentation.

Let's get started!

Front-End Frameworks

When we use the term **front-end** in web development, we're referring to the portion of a website or application that's run in the browser. This includes menus, navbars, links, and CSS designs. This is in contrast to back-end code, which doesn't run in the browser. A browser application can make a request to a server where back-end code is stored. **Back-end** code is all of the code that serves data or manages a database that cannot be accessed by a user. During the Intro to Programming and Intermediate JavaScript courses, we'll create applications with a front-end and no back-end.

The term **framework** refers to a foundation of pre-built tools, software, code, and other resources we can use to create a project more easily than if we created it from scratch.

Front-end frameworks are pre-built code that assist us in creating the front-end of our website. There are frameworks more focused on design like Bootstrap, which we'll discuss in a moment. There are also frameworks and libraries that are focused on JavaScript code, like Angular and React.

Bootstrap

Bootstrap is one of the most popular, widely used front-end design frameworks. It allows developers to quickly set up professional looking websites with minimal effort. It was originally built at Twitter to address inconsistencies in front-end design. It was later released to the public in 2011.

Bootstrap is essentially a big package containing all the code (CSS and JavaScript) necessary to make a wide variety of commonly-used components like navigation bars, forms, headings, tables, buttons, and more. It also has built-in media queries so any Bootstrap elements we use in our own site will automatically re-style and resize themselves to best fit the user's viewport.

You can read more about Bootstrap on their website, getbootstrap.com (https://getbootstrap.com/).

Websites Built with Bootstrap

Check out the following link to see some example sites built with Bootstrap:

- Sites Built with Bootstrap from Expo.GetBootstrap.com (https://expo.getbootstrap.com/)
- Sites Built with Bootstrap from Awwwards (https://www.awwwards.com/websites/bootstrap/)

Frameworks, Libraries, and Toolkits

There are other terms that are commonly used to describe pre-built code that we can download and use in our apps:

Library. A library is a smaller set of pre-defined code that we can download and use in our apps. It is quite similar to a framework, though frameworks are always larger and often comprise multiple libraries and other tooling.

Toolkit. A toolkit is a set of pre-built tools used for a particular purpose. Often, a toolkit is defined as multiple and related libraries, packaged into one. Usually, frameworks are much larger than toolkits, and a toolkit is larger than a library.

Framework. A set of pre-built tools that somprises software, code, and other resources like other frameworks, libraries, and toolkits. Frameworks often contain constraints about how you can build your code within the framework. We've already mentioned Bootstrap — we'll also learn how to use back-end frameworks when we begin learning C# or Ruby.

The nuances within the definitions for each term is a topic of discussion and study within the developer community. To see an example of the discussion around these terms, visit this Stack Overflow article (https://stackoverflow.com/questions/3057526/framework-vs-toolkit-vs-library).

The bottom line with all these terms is that they represent pre-built code that we can download and use in our applications. And when we learn how to use Bootstrap in our applications, we'll also learn the general process of learning to work with any library, toolkit, or framework.

Up next, we're going to start integrating Bootstrap into our sites by walking through how to download and install Bootstrap into a project.

Previous (/introduction-to-programming/git-html-and-css/practice-css-media-queries-and-responsive-design)
Next (/introduction-to-programming/git-html-and-css/downloading-and-installing-bootstrap)

Lesson 58 of 64
Last updated March 24, 2023

disable dark mode

Epicodus (http://www.epicodus.com)