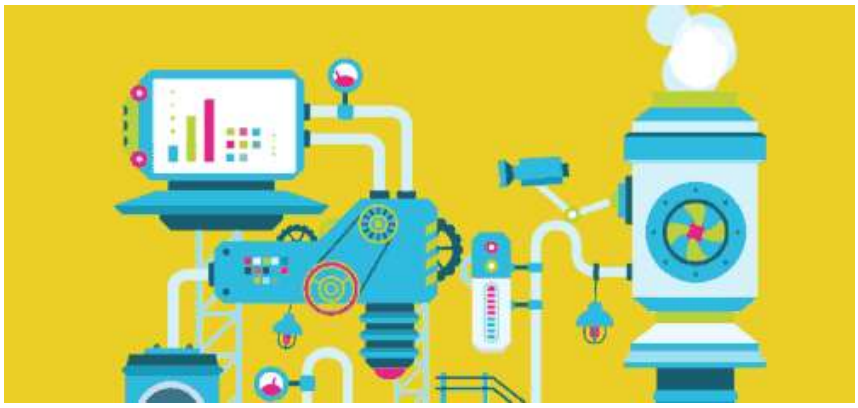
Module #5 - Machine Ready (Required)

More Than Programming...

As every developer knows, becoming proficient at web and software development requires more than just a grasp of programming languages alone. Instead, successful developers must also possess a comfortable familiarity with a wide assortment of tools and technologies. These tools enable software to be created, to be collaborated over, and to be shared with users around the world.

In this class, you can expect to be exposed to a wide range of tools with names you've never heard—tools like *Heroku*, *Git*, *Emmet*, and *Robomongo*. At first, the sheer number of tools you'll be expected to use may seem overwhelming but trust us! With a little time and with a little effort, they will be as familiar to you as a scalpel is to a surgeon or a sewing needle is to a seamstress.

They're all just tricks of the trade.



Ready for Action!

Coming into your first day of class, you will be expected to have a number of tools already installed. This will ensure that you are ready to start coding right from the start. The purpose of this module is to walk you through the process of installing all your tools and to give you a brief primer on the roles they play.

After completing this pre-work module, you will have each of the following installed:

- Google Chrome
- Screencastify
- Slack
- Visual Studio Code
- "Open in Browser" Visual Studio Code extension
- Git
- Git Bash (Windows)
- Terminal (Mac, Pre-Installed)
- Homebrew (Mac)
- Heroku Toolbelt
- Node.js
- MAMP

In addition, you will also have accounts on each of the following websites:

- LinkedIn
- GitHub
- Stack Overflow
- Slack

Having trouble with set up? Not to worry. Your instructional staff will help you troubleshoot any errors and answer any questions on the first day of class. Just sit tight until then!

Tools for Fools

Before we start installing everything willy-nilly, let's take a moment to examine each of these tools to better understand the role they play.

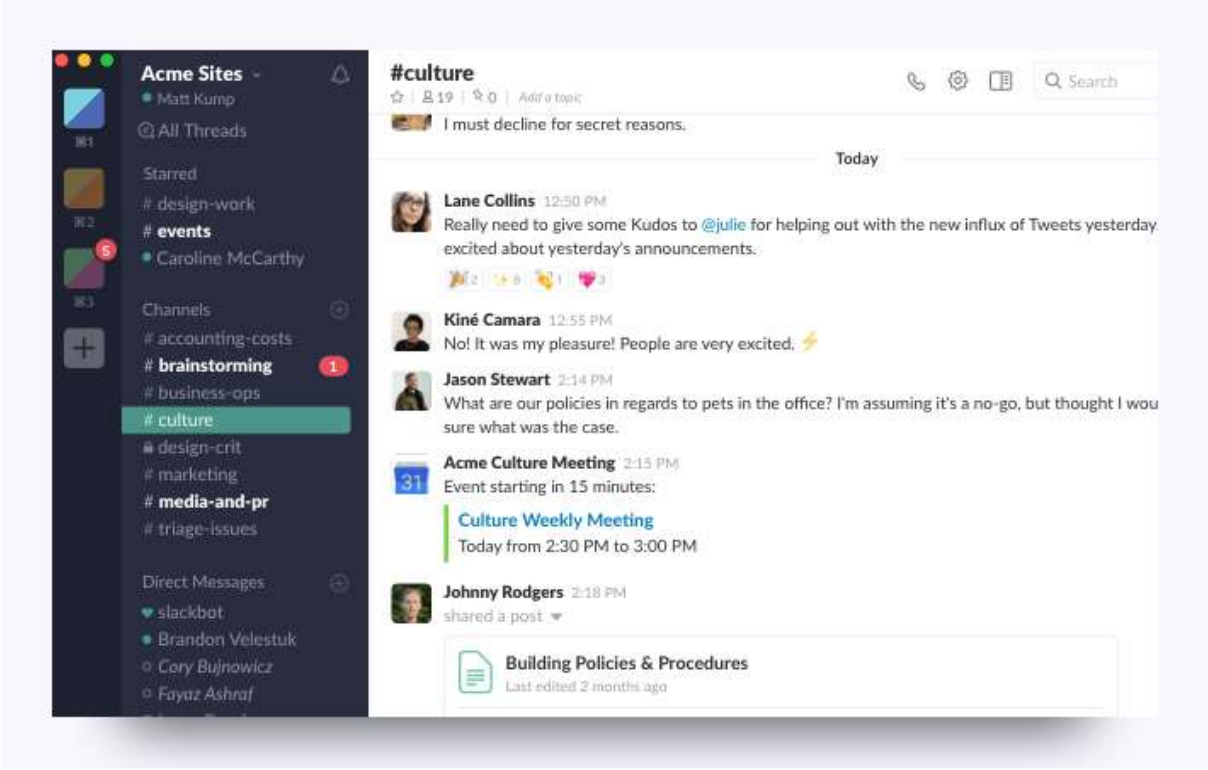
Google Chrome

This one is pretty straightforward. It's a web browser. In our case, we'll be using it to quickly *see* if our code is working. While in truth, you can use any web browser, Google Chrome has a number of tools that make it an ideal platform for developing so we strongly encourage you to make the switch.

Slack

This is one you will be using literally every single day for the next six months. Slack is an online communication tool that is a mix of forum, of instant messenger, and of email—all rolled in one. It's a tool that is used by countless organizations worldwide.

In our Bootcamp, we'll be using Slack extensively to send code snippets during class, to relay important announcements, and to facilitate group exercises. You will receive the link to your class-specific channel during orientation. You will definitely want to have this *installed* on Day 1. (Note how we said *installed* and not simply logged into. While the web client is good, for our class, you will want to install the actual program on your machine.)



Visual Studio Code

Oh, the power of Visual Studio Code! A little program that does so much!

Visual Studio Code is a free, open-source "text-editor". Now, for the uninitiated, the first thought that comes to mind when we say "text editor" may be something basic like Notepad or TextEdit. But for developers, text-editors are like the cozy pillow on which they rest their head. This is because fundamentally programming is all about creating "text"

in files with various extensions. When we create a block of HTML, like the one below:

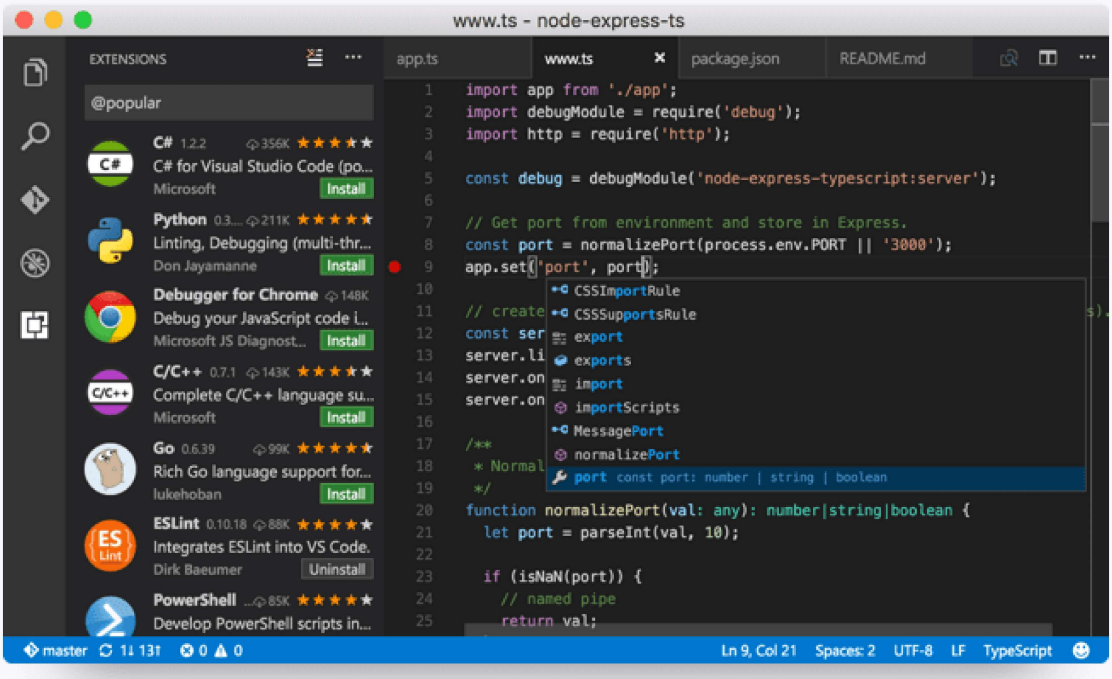
```
<!DOCTYPE html>
        <html lang="en">
        <head>
                <meta charset="UTF-8">
                <title>Document</title>
        </head>
        <body>

                <h1>Yo Yo Yo</h1>
                <h2>I'm the big man money</h2>

        </body>
</html>
```

What we've really done is just created a block of text. There are funny symbols in there like those angle brackets and what not, but at its most base level -- it's just text.

Now to a simple text editor this is where the comprehension stops. Our block of HTML remains a block of text. But to more powerful text editors like VSCode, on the other hand, these blocks of "text" are immediately recognized as code (so long as you include the right file extension). This means that VSCode can give us a more visually intuitive understanding of the code through indicative coloring, "smart" tabs, and autocomplete functionality. The end result is that creating the above block of HTML becomes a more natural process and one that can be debugged more quickly.



Additionally, as you will find in this class, VSCode is also powerful in its ability to be extended through the use of plugins. This means that we can easily incorporate free add-ons that enable VSCode to make the process of coding even easier than before.

Open in Browser

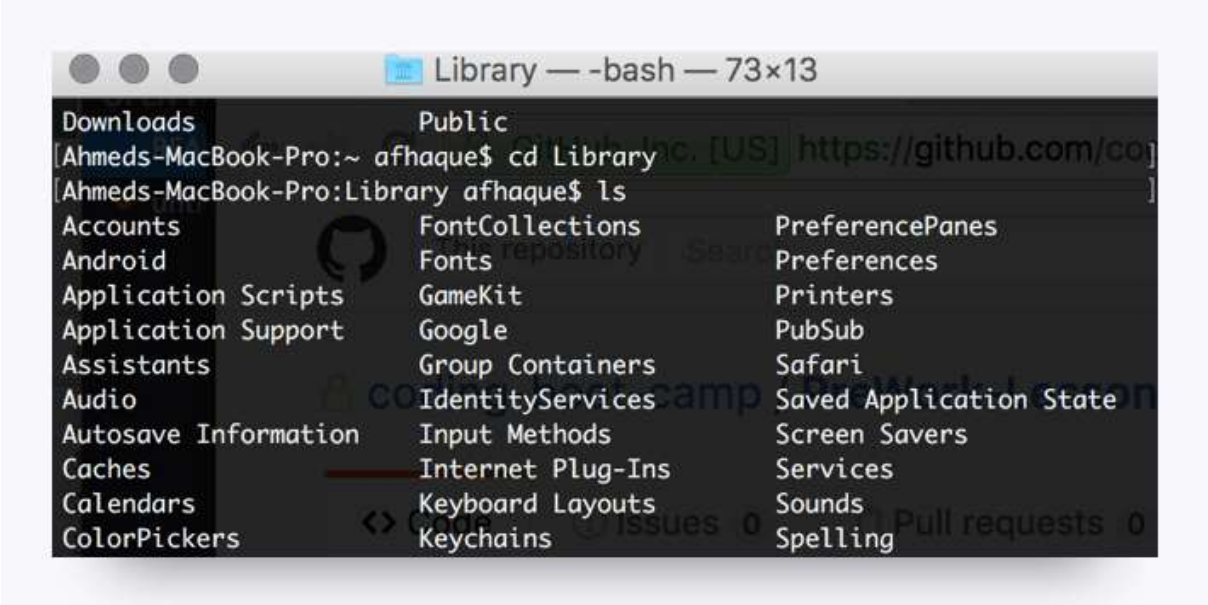
For now, the only extension we recommend installing with Visual Studio Code is the "Open in Browser" extension. This will allow us to open HTML files we are editing in VSCode in our web browsers without having to go through the file explorer/finder.

Git Bash / Terminal

Git Bash (PC) and Terminal offer a command line interface for working with the files and folders on your computer.

So, is it like Finder or Windows Explorer?

Kind of... except there are no pictures or visuals. It's just a box with text.

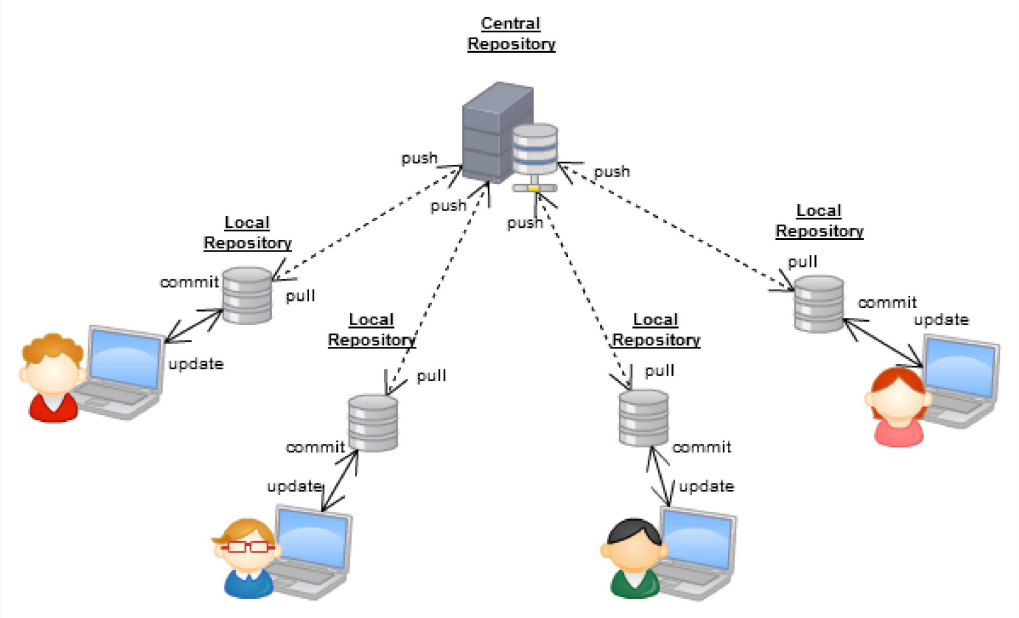


Uh, why would I want that?

You'll come to understand over time, but in many situations, utilizing a command line interface can be faster and can be more powerful than relying on the operating system's GUI. You'll get plenty of exposure to the command line on day one.

Git / GitHub

Because code files are ones in which multiple developers need to carefully build upon each other's work, Git offers a specialized set of strategies for orchestrating the collaboration. GitHub then takes all these collaborative actions and stores them online. In a way, GitHub serves as a sort of Dropbox for coders. It offers a central place for individuals or teams to upload their code, to view revision history, and to make changes to a master set of files. You'll come to learn a lot about Git and GitHub in your first week of class. You will receive the link to your class-specific repository during orientation.



Homebrew

Homebrew is a Mac-specific toolkit that makes it easy to install, using the command line, a variety of applications. It can greatly simplify the installation process for various tools you'll be using in your development career.

Heroku / Heroku Toolbelt

While you may not feel ready yet, very quickly, you'll be creating complete websites on your computers. But once you have these sites made, how do you get them online for the world to see?

Hosting platforms like Heroku effectively serve as a dumping ground for web applications. These platforms are set up to take your web applications' code, to activate them, and to then assign them to a URL for other visitors to see. In a sense, they *host* your applications so that every internet user has access to them. You'll learn a lot about how this works towards the tail end of the course.

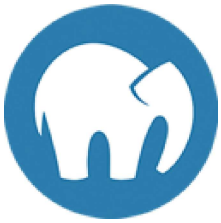
As it relates to the pre-work, you'll be installing the Heroku Toolbelt, which offers a set of easy-to-use tools for interacting with the Heroku platform online.

Node.js

Oh, Node! (Get the pun?)

Node.js is a JavaScript library that we'll be using extensively for the back-end of our applications. Don't worry if that doesn't make much sense yet. We'll be spending effectively half the program working on Node. You'll become very proficient in its use and in its utility by the end of the course.

MAMP



MAMP is a popular *stack* of tools that is used by many web developers. In our case, we'll install it as a back-up, in case your machine runs into any issues with MySQL. (It happens. Tools are finicky sometimes.)

Collect Your Tools!

And that's it!

It's time to collect your tools and begin. As you'll see in the links below, we have guides for both Mac and Windows users on the process for getting setup. Follow the instructions closely and do your best with the information you have. (Yes, we know there is a lot to install.)

Just one bit of advice! As future coders, you will frequently be looking at documentation to install and to utilize unfamiliar tools. Resist the urge to stop and ask, "Am I doing this right?" Instead, just take your best stab at it. This is an important attitude to start developing *now*. A lot of what you'll be exposed to over the next six months will be unfamiliar. Don't be phased by it, and don't get bogged down by it. Trust your instincts! We'll make sure that anyone who is lost gets the help they need on Day One.

Good luck! Make sure your tools are extra pointy.

Assignment (Required):

- Get Yo' Tools Installed on Windows
- Get Yo' Tools Installed on Mac

Copyright