

Setup & Installations on macOS Catalina or Mojave

Being a developer isn't just about hacking away into the wee hours of the morning or debugging a new feature. All craftspeople must have mastery of their tools to be successful in their trade, and programmers are no different. For a developer the most important tools are our CLI, text editor, web browser, compiler, package manager, and Node environment for running JavaScript. Mastery of these tools will be invaluable for the entire duration of our careers.

This reading will cover the installation of the basic tools you'll need to run code on your computer.

When you finish this reading, you should have:

- Installed Visual Studio Code (VS Code)
- Installed Node & NPM (Node Package Manager)
- Installed Google Chrome
- Installed Xcode & Homebrew (Mac)
- Installed Python 3

Preparing your machine

The commands you need to enter are listed below. Here we will install basic developer tools, such as [homebrew](#) (a 3rd party package manager for MacOS), [Xcode](#) (a library of developer tools provided by Apple), VS Code (a full-featured text-editor), and Node (a JavaScript runtime environment).

Chrome

Here at App Academy, our browser of choice is Google Chrome. This isn't super important at the beginning of the course, but once we get into frontend development (writing code that runs in a web browser) the Chrome Devtools will be crucial for debugging every manner of issue.

To install Google Chrome, download the necessary files and follow the instructions on the [Google Chrome website](#).

Xcode

Let's start with Xcode. The Xcode command line tools are a requirement for installing the homebrew package manager in the next step.

NOTE: If you are using a Linux machine you will not be able to install Xcode or homebrew.

Install the Xcode command line tools by running the following from the console.

```
$ xcode-select --install
```

To conclude the installation you will need to agree to the Xcode license. Start the Xcode app, click "Agree", and allow the installation to finish. Then you can go ahead and quit the Xcode app.

Homebrew

Homebrew is kind of like a low-tech App Store. It allows us access to and the ability to install a wide variety of software and command line tools from the

console. These are distinct from those hosted on the App Store and will need to be managed by Homebrew.

Enter the following in your terminal to download and install Homebrew:

```
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install)
```

You will be given a list of dependencies that will be installed and prompted to continue or abort. Press `RETURN` to continue.

Let's break this command down a bit. `curl`, a command-line tool commonly used for downloading files from the internet, is used to download the Homebrew installation file. The `"$(...)"` transforms the file content into a string. Finally, the string is passed to a Ruby language executable (`/usr/bin/ruby` is where the system Ruby executable file is stored on our machine) with the `-e` flag to tell Ruby to run the argument as code.

Check out the [Homebrew website](#) to learn the basic commands.

Node.js & NPM

[Node.js](#) is a very powerful runtime environment built on Google Chrome's JavaScript V8 Engine. It is used to develop I/O intensive applications like video streaming sites, robots, and other general purpose applications. For our purposes Node provides a way for us to run JavaScript outside of the browser.

We want to use a version manager with Node to help us manage potential conflicts between versions and dependencies. In this case we will be using [NVM](#) (Node Version Manager) to install/manage Node.js.

Open up your console (the Terminal application on Mac) and run the following:

```
# download and run the official install script
curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.33.6/install.sh | ba

# update your terminal config (you will now have access to the nvm command)
source ~/.bashrc

# install a stable version of node
nvm install 10.16

# set version 10.16.0 as default version
nvm use 10.16

# verify install/config
which node # => /Users/username/.nvm/versions/node/v10.16.0/bin/node
```

Node comes with a package manager called [NPM](#), which provides access to a whole ecosystem of libraries and tools we can use. NPM comes pre-bundled with Node, so there is no additional work for us to do. By default we don't need any additional libraries, and any additional packages we do need to use will be installed on a project-by-project basis.

VS Code

This one is pretty easy. Go to website for [Visual Studio Code](#), then download and install VS Code.

To verify that the shell commands were installed correctly, run `which code` in your terminal. If `code` is not a recognized command, open the VS Code editor, open the Command Palette (`Cmd+Shift+P` on macOS, `Ctrl+Shift+P` on Linux) and type `shell command` to find the `Shell Command: Install 'code' command in PATH` command. Then restart the terminal. This will now allow

you to easily open files in VS Code from the terminal using the `code` command followed by a file or directory.

Next, we'll want to install a few useful VS Code extensions and configure VS Code to play nice with these extensions. Download [this zip file](#), which contains a script that will do the work for you. Unzip the file and open the `setup_vscode` directory. Navigate into that directory in the terminal (drag and drop the folder over to the terminal icon on macOS or right click in the directory and select `Open in Terminal` on most Linux distributions).

To run the script, run the command:

```
~ ./setup-vs-code.sh
```

The script will do the rest. Now restart VS Code and you'll be good to go.

Mocha testing framework

The last thing we'll be installing will be Mocha. Mocha is a JavaScript testing framework that we will be using to test our work in the future. Here are the instructions of how to install `mocha`!

Installing Mocha

1. Open Terminal
2. Enter this command: `npm install -g mocha`
3. To test your installation, run the command: `mocha --version`. If it returns a version number, you've successfully installed mocha! Otherwise, let your instructor know and they'll help you fix things.

Installing Python3

You can just use *homebrew* for this install, too.

```
brew install python
```

What you learned

How to install the various tools you'll need to create a development environment.

After you finished this reading you should have the following installed:

- Virtual Studio Code (VS Code)
- Node & NPM (Node Package Manager)
- Mocha the JavaScript testing framework
- Google Chrome
- Xcode & Homebrew (Mac)
- Python 3