console.log("Hello World!")

Welcome to the first module of ProductLab! We're giving you two weeks to complete these assignments and it will likely feel like one of the more challenging parts of the whole course. This is intentional. There is a lot for you to learn, but before you get there you have to know the basics. This class will set you up for success by providing you with:

1. A fully set up development environment
2. Foundational knowledge on technology, development and developer tools

Getting started

To be successful in this course you MUST have a fully set up development environment, if you do nothing else make sure you accomplish this piece. Additionally, ProductLab builds on the topics from each previous module. So, it is highly encouraged that you dedicate the time to closely follow the Javascript and HTML sections of this module. Ideally you should follow as many of the hands-on assignments as you can.

But before you dive in, read this article:

- Don't Believe Anyone Who Tells You Learning To Code Is Easy | TechCrunch

A lot of what you'll do today may not work on the first try, don't panic, that's part of what you're going to learn in this course. That coding takes time!

- (This is optional but a good article on how to approach learning to code in general)

This week's class will lean heavily on the Odin Project, which is a great resource for people interested in learning the basics of full stack development. In places it can go a little too in-depth for our needs and in others it does not cover all the topics you'll need to learn. But it's a great place to start!

On the following pages we detail the recommended parts of the Odin Project to review. However, you can do additional sections if you're interested!

These are the first two components to review:

- Computer Basics | The Odin Project
- How Does the Web Work? | The Odin Project
- Stop after you finish the "How Does the Web Work?" page

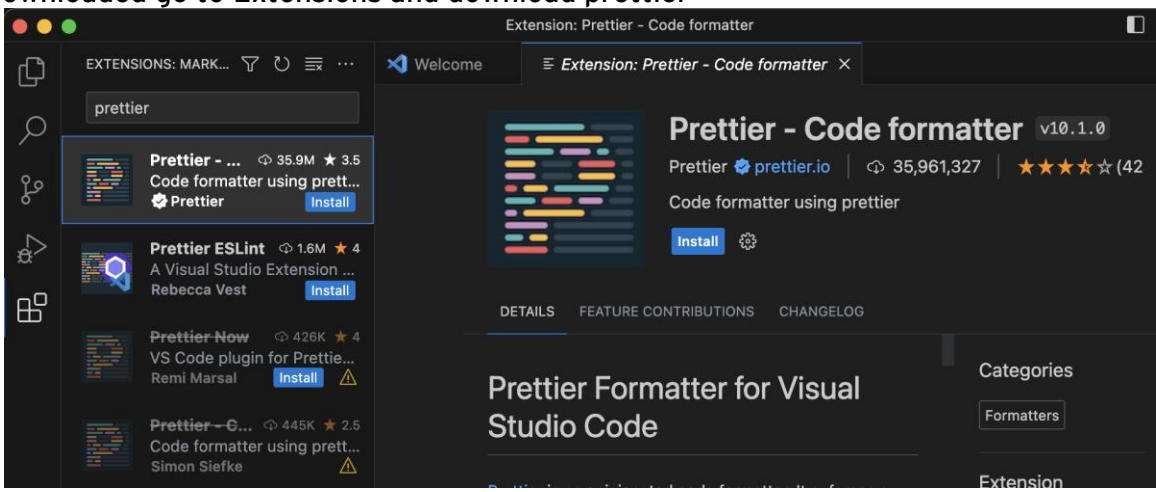
Setting up your Environment (for Mac Users ONLY)

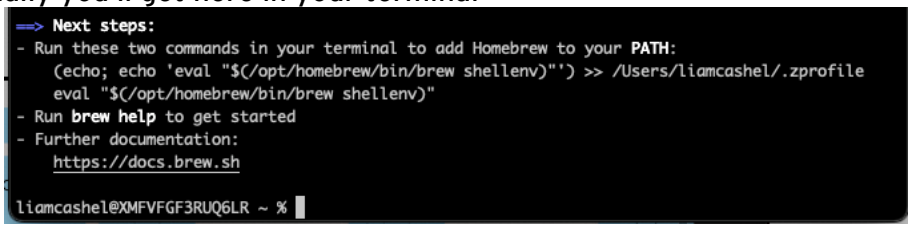
If you're using a windows machine keep scrolling, there are Windows specific instructions on subsequent pages.

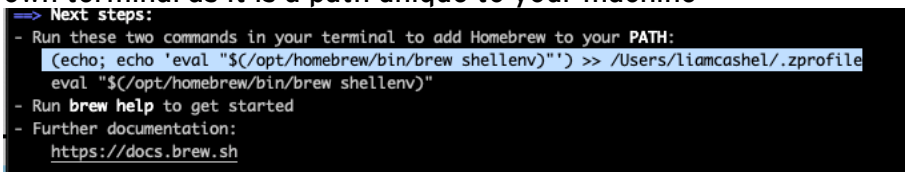
During these steps, please pay careful attention and follow the directions exactly. Don't worry you won't break your computer, but if you don't follow the instructions you may end up very frustrated!

- Follow along with the guide here: Installation Overview | The Odin Project

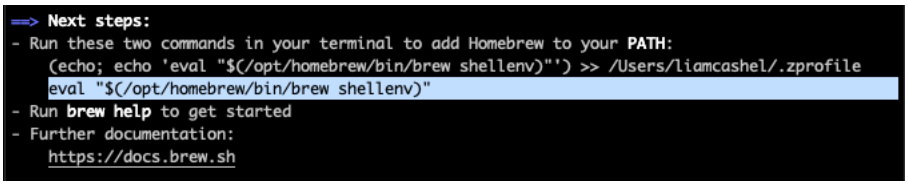
Basics Module

- A note on Downloading VScode
  - You'll need to elevate your permissions in the Mac@EY store
  - Once downloaded go to Extensions and download prettier
    - 

- "Setting up Git" module of the Odin Project
  - Github setup
    - Note: when creating your github profile create a personal account not an enterprise account
  - Homebrew installation
    - During the homebrew step, you'll have to enter your password. When you type it won't show up, this is a feature of the terminal's security. It's logging your password. Just type it in as usual then hit enter
    - ALSO: you'll have to enter your password multiple times, just be patient
    - ALSO: the download will take a while, shake your computer a few times to speed it up *(don't actually do this)*
    - Eventually you'll get here in your terminal
      - 

      - First copy&paste the highlighted line below into your CLI and hit enter
        - Don't copy the line in the screenshot, copy from your own terminal as it is a path unique to your machine
        - 

      - Then copy&paste the next highlighted line below into your CLI and hit enter

    - 

- Finally, type *brew help* into your CLI and hit enter
    - You should see this
    - 

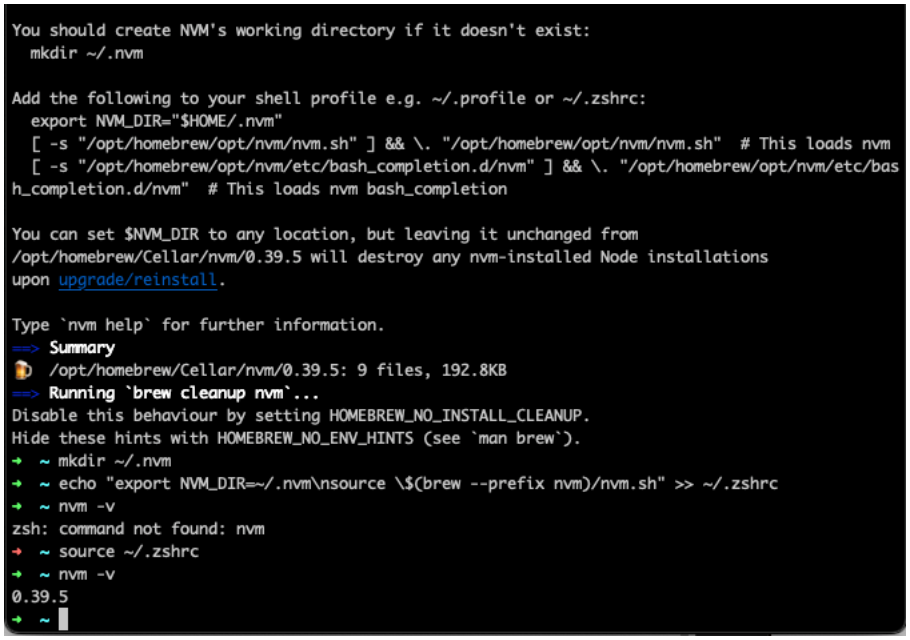
- Now go back to the Odin Project to install git
    - Note: when you are verifying that you correctly set up your SSH key you will be on the github tutorial page. In your CLI you have to type *yes* then hit enter when prompted, if you just hit enter it'll error out

- OPTIONAL: After you finish "Setting up Git" module
    - leave the odin project and open up another tab to this site: [The definitive iTerm2 & Oh-my-ZSH Setup on macOS (larsbehrenberg.com)]
    - ignore the step on homebrew as you'll already have set that up
    - follow the rest of the instructions which will guide you through setting up a third party terminal called iTerm2 which can be a very powerful tool, however it is not necessary
    - you can also then download "Oh my ZSH" which will allow you to customize the way your terminal looks

Ok you're almost ready with a fully functional coding environment on your computer! Just three more things:

- Github desktop:
    - developers should be comfortable using git commands inside their CLI, but you don't need to stress about it too much. Github desktop has a great UI that will make pulling and merging repositories extremely easy!
    - You can download it here: [GitHub Desktop | Simple collaboration from your desktop]
- Node.js

- Node.js lets you run JavaScript outside of a web browser (this will make more sense later) You won't use it quite yet but its important you set it up right.
- Navigate to this site: [Download | Node.js (nodejs.org)]
  - Select the appropriate operating system
  - Select the "LTS" version
  - Follow the download instructions
- Navigate to your terminal and type *npm -v*
  - If your terminal returns a series of numbers, you have successfully downloaded node.js!

- NVM
  - NVM stands for node version manager. Some projects you may utilize a different version of node.js, in this case you can use NVM to switch to an earlier version that the source code of your project will support
  - **NOTE: FOLLOW THE BELOW INSTRUCTIONS EXACTLY**
  - In your CLI enter the following commands (the italics might mess up the copy paste, if its not working try typing it in by hand)
    - *brew install nvm*
    - *mkdir ~/.nvm*
    - *echo "export NVM_DIR=~/.nvm\nsource \$(brew --prefix nvm)/nvm.sh" >> ~/.zshrc*
    - *source ~-/.zshrc*
  - next enter the following into your CLI
    - *nvm -v*
    - you should see a response something like "0.39.5"
  - here is a view of a successful download of NVM
    - 

  - if you are confused this guide may be helpful
    - [How To Install NVM (Node Version Manager) on macOS with Homebrew | by Paris Nakita Kejser | DevOps Engineer, Software Architect and Software Developering | Medium]

Setting up your Environment (for Windows Users ONLY)

During this process, please pay careful attention and follow the directions exactly. Don't worry you won't break your computer, but if you don't follow the instructions you may end up very frustrated!

First, follow the steps outlined on the Microsoft's Learn site.

Follow the entire process on this page: Install NodeJS on Windows

- DO NOT follow links to "install Node.js on Windows Subsystem for Linux or WSL2. After a lot of troubleshooting, we determined that setting up directly on Windows is the better alternative of the two.
- Make sure any existing installations of Node.js are removed from your operating system before downloading
    - When prompted to run a command – these will be run in PowerShell or Windows Command Prompt – you should already have both applications. Type either application name into the Windows application search bar.
- Follow the link to the github for nvm-windows releases, and click on this link:
    - 

    - Go back to the previous instructions webpage and follow the setup steps
    - You will be prompted to input your windows password and reason for approval. Select the reason "Required for development/testing/troubleshooting" and proceed.
    - You can proceed with your "Current" version rather than LTS for the time being.
- Download VS Code: This part should be straightforward
    - Enable Setting Sync via windows login (not via the github option)
    - You can always come back to this page if you have overview questions
        - We recommend watching the helpful introductory videos, **but hold off until you have downloaded your node.js extension pack** (as a part of the Windows Install Instructions) and github (instructions below). The videos are helpful tools, but you want to make sure everything is installed first

    - You can always come back to this page if you have questions about setup on Windows

Once you have completed this, it's time to setup Git, your new go to version control system.

- Download for Windows, DO NOT download anything related to Windows Subsystem for Linux
- Use the recommended settings during your install. This should mostly just be you clicking "Next" until it's time to Install.

Finally, download Github desktop:

- developers should be comfortable using git commands inside their CLI, but you don't need to stress about it too much. Github desktop has a great UI that will make pulling and merging repositories extremely easy!
- You can download it here: GitHub Desktop | Simple collaboration from your desktop
- you're going to have to set up a github account. Create a personal account, not an enterprise account.
    - We recommend you use your personal email address for this

Congratulations! You have a fully set up developer environment on your computer. While it may have been a headache, the work you just did will pay huge dividends later down the road! If you completed this all in one sitting so far, take a break. The next stage of this class will teach you the fundamentals of coding languages.

Coding Fundamentals

Now we can learn a little about coding. Starting out with HTML

- Introduction to HTML and CSS | The Odin Project
    - Follow along until you get to the "Lists" section.
        - Here you can stop being so hands on and skim read. Knowing HTML is a helpful skill and you should bookmark these sections for reference.
- Stop when you get to the "Commit Messages" section

Intro to CSS. You don't need to do the whole CSS module in the Odin Project. Just through the first page

- Intro to CSS | The Odin Project
- When you finish the rest of the material for the Basics Class you can come back and learn more about CSS if you'd like

Now you're ready to learn some JavaScript. JavaScript is likely not the language the backend devs you'll be working with will use. BUT, it's a great tool for learning because if you know JavaScript you can code fullstack, as opposed to other languages that are exclusive to the front or back end.

Now, there is a lot of content here, and we do not expect you to finish all of it. But even so this will take time. Like we've said before, this Basics Class is a lot of content and a larger time investment than any of the other classes. But establishing a strong foundation is important. Invest the time to learn this material!

Start the JavaScript modules here

- There is a lot of hands on practice material in these modules. Use your judgement on how many you want to practice on. If you feel you have the hang of a subject, feel free to move on.
- Note: In the "Fundamentals part 2" module the assignment is optional. Come back to it if you have time after completing the rest of the material
- Note: In the "Fundamentals Part 3" module, the assignment is required. Create a repository in your github account and upload your file with the functions you wrote

Required Projects:

- Fundamentals Part 3: upload repository of mathematical functions
- Project: Rock Paper Scissors | The Odin Project

STOP: after you complete the RPS project you do not need to go any farther in the Odin Project.

Conclusion

Everything after the Rock Paper Scissors project is optional. If you have time feel free to go through the rest of the material. If not, no problem! However, if you find you have time later on during ProductLab we encourage you to continue to hone your JavaScript abilities and the additional modules on the Odin Project are a great place to start.

Optional Project

- Recipes HTML Project
    - You can use the git commit commands or you can use the github desktop app, whichever you'd prefer
    - (you can also set up repositories on the desktop app)
    - If you login to the Odin Project you will be able to see user submissions for examples of solution code