**Setting Up Your Computer For App Academy**

When preparing for App Academy, it is vitally important that your computer be set up in a very specific way. Not taking the time to do this correctly the first time will likely result in repeated issues and distractions, preventing you from concentrating on the curriculum.

To find the instructions for installing the components that you will need, navigate to the guide for your particular operating system in the left-hand sidebar menu:

* macOS with an M1 chip
* macOs without an M1 chip
* Ubuntu
* Windows

(The macOs guides will tell you how to determine whether or not you have an M1 chip; if your Mac is newer than late 2020, you likely have an M1.)

# macOS M1 Setup

**IMPORTANT**  
These instructions are for **Macs with an Apple Silicon (M1) chip**. You can find out if you have this chip by going to the Apple menu and choosing About This Mac. If you see Chip: Apple then you are in the right place: you have an Apple Silicon Mac.

## Installing the software

Install all of the following software in order.

### Google Chrome

1. Using your default internet browser, navigate to the [Google Chrome Download](https://www.google.com/chrome/) page.
2. Download and run the installer.
3. When prompted, set Google Chrome as your default internet browser.

### Visual Studio Code

1. Using Google Chrome, navigate to the [VS Code Download](https://code.visualstudio.com/Download) page.
2. Download and run the Mac installer.
3. After installation completes, drag the VS Code app icon into your Applications folder.
4. Run VS Code and open the Command Palette menu option.
5. Type shell command into the search bar that pops up and select the option Shell Command: Install 'code' command in PATH.

### Xcode Command Line Tools

Run the following command in your terminal to install the Xcode CLI tools:

xcode-select --install

### Homebrew

Run the following commands in your terminal to install and configure Homebrew:

/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

echo 'eval "$(/opt/homebrew/bin/brew shellenv)"' >> ~/.zprofile

### **git**

Run the following commands in your terminal to install and initialize your git configuration. (Replace "Your Name" and "Your Email" with your actual name and email--respectively--**in quotation marks**.)

brew install git

git config --global user.name "Your Name"

git config --global user.email "Your Email"

git config --global init.defaultBranch main

### **rbenv**

Run the following commands in your terminal to install and configure rbenv:

brew install rbenv

echo 'eval "$(rbenv init -)"' >> ~/.zshrc

source ~/.zshrc

### Ruby

Run the following commands in your terminal to install the correct version of Ruby and the gems you'll be using in the course:

rbenv install 3.1.1

rbenv global 3.1.1

rbenv rehash

gem install bundler pry byebug

gem update --system

brew install shared-mime-info

gem install rails -v 7.0.3

rbenv rehash

### PostgreSQL

1. Using Google Chrome, navigate to the [Postgres.app Download](https://postgresapp.com/downloads.html" \t "_blank) page.
2. Download and run the Latest Release installer.
3. Run the following commands in your terminal to access the corresponding CLI tools

sudo mkdir -p /etc/paths.d && echo /Applications/Postgres.app/Contents/Versions/latest/bin | sudo tee /etc/paths.d/postgresapp

1. Restart your terminal to complete the installation.

### **nvm** & Node.js

Run the following commands in your terminal to install nvm and the correct version of Node.js:

curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh | bash

source ~/.zshrc

nvm install 16

nvm use 16

# macOS Setup

**IMPORTANT**  
These instructions are for Macs **without** an Apple Silicon (M1) chip. You can find out if you have this chip by going to the Apple menu and choosing About This Mac. If you see Chip: Apple, then you have an Apple Silicon Mac and should follow the Apple Silicon (M1) Setup instead. (See the left-hand sidebar menu.)

## Requirements

You need to be running macOS Mojave or newer to follow these instructions.

## Installing the software

Install all of the following software in order.

### Google Chrome

1. Using your default internet browser, navigate to the [Google Chrome Download](https://www.google.com/chrome/) page.
2. Download and run the installer.
3. When prompted, set Google Chrome as your default internet browser.

### Visual Studio Code

1. Using Google Chrome, navigate to the [VS Code Download](https://code.visualstudio.com/Download) page.
2. Download and run the Mac installer.
3. After installation completes, drag the VS Code app icon into your Applications folder.
4. Run VS Code and open the Command Palette menu option.
5. Type shell command into the search bar that pops up and select the option Shell Command: Install 'code' command in PATH.

### Xcode CLI & ZSH

1. Run the following commands in your terminal to install the Xcode CLI tools and set ZSH as your default shell
2. xcode-select --install

chsh -s /bin/zsh

1. Restart your terminal to complete the transition to ZSH

### Homebrew & git

1. Run the following commands in your terminal to install Homebrew and git:
2. /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"

brew install git

1. Run the following commands in your terminal to initialize your git configuration. (Replace "Your Name" and "Your Email" with your actual name and email--respectively--in quotation marks.)
2. git config --global user.name "Your Name"
3. git config --global user.email "Your Email"

git config --global init.defaultBranch main

### rbenv & Ruby

1. Run the following commands in your terminal to install rbenv:
2. brew install rbenv
3. echo 'export PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.zshrc
4. echo 'eval "$(rbenv init -)"' >> ~/.zshrc

source ~/.zshrc

1. Run the following commands in your terminal to install the correct version of Ruby and the gems you'll be using in the course:
2. rbenv install 3.1.1
3. rbenv global 3.1.1
4. rbenv rehash
5. gem install bundler pry byebug
6. gem install rails -v 7.0.3

rbenv rehash

*If installing Rails is throwing an error, try running brew install shared-mime-info and trying again.*

### PostgreSQL & SQLite

1. Using Google Chrome, navigate to the [Postgres.app Download](https://postgresapp.com/downloads.html" \t "_blank) page.
2. Download and run the Latest Release installer.
3. Run the following commands in your terminal in order to access the corresponding CLI tools and install SQLite:
4. sudo mkdir -p /etc/paths.d && echo /Applications/Postgres.app/Contents/Versions/latest/bin | sudo tee /etc/paths.d/postgresapp

brew install sqlite

1. Restart your terminal to complete the installation.

### nvm & Node.js

1. Run the following commands in your terminal to install nvm and the correct version of Node.js:
2. curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.0/install.sh | bash
3. source ~/.zshrc
4. nvm install 16

nvm use 16

# Windows Setup

Welcome Windows users! Your operating system comes from a different lineage than most operating systems: Windows has its roots in MS-DOS and Windows NT, while macOS and Linux are both "Unix"-style operating systems. Everyone in this course will be using the Unix command line with its associated tools. Windows does not include a Unix-style system by default, so you are going to use an add-on piece of Microsoft Technology called "Windows Subsystem for Linux", or WSL.

## WSL

Windows Subsystem for Linux is a Linux Virtual Computer that you run on your PC. It gives you all the same tools and command line utilities that your Mac- and Linux-using classmates use.

To install WSL you must have a specific version of Windows 10.

*For x64 systems: Version 1903 or higher, with Build 18362 or higher.*

If you do not have this version or cannot update to this version, please contact an instructor.

## Installing WSL 2

Future versions of Windows will include an automatic installer for WSL, but for now you will have to use the manual install instructions:

[Windows Subsystem for Linux Installation Guide for Windows 10](https://docs.microsoft.com/en-us/windows/wsl/install-win10#manual-installation-steps)

Follow this guide to install WSL 2 and Ubuntu Linux. We recommend installing Ubuntu 18.04 LTS, found [here](https://www.microsoft.com/store/apps/9N9TNGVNDL3Q).

## Tips for using WSL

You will launch the Ubuntu terminal when the instructions in the curriculum tell you to open a Terminal.

You should always store your code files in your Ubuntu home directory and not in your Windows users home directory on your C: drive. Many of the tools you will be using will perform better and be more stable if the files exist on the Ubuntu filesystem.

If you do need to access the files in your Ubuntu home directory from Windows Explorer, you can type Windows + R to bring up the run command dialog and type \\wsl$\home\<your-username> replacing <your-username> with your Ubuntu username. (You can find your Ubuntu username by typing whoami at your Ubuntu terminal prompt.)

If you want to access your Windows hard drive from Ubuntu, you can use the path /mnt/c inside of the Ubuntu virtual machine.

If you ever need to restart the Ubuntu virtual machine, you need to open a Powershell window and run the following command:

wsl --shutdown

The next time you open the Ubuntu terminal, it will start the virtual machine back up.

## Install everything else

Now that you have WSL installed you can install all the rest of these tools in this order:

### Google Chrome

1. Using your default internet browser, navigate to the [Google Chrome Download](https://www.google.com/chrome/) page.
2. Download and run the installer.
3. When prompted, set Google Chrome as your default internet browser.

### Visual Studio Code

1. Using Google Chrome, navigate to the [VS Code Download](https://code.visualstudio.com/Download) page.
2. Download and run the Windows installer.

### rbenv & Ruby

1. Run the following commands in your Ubuntu terminal to install rbenv:

sudo apt install git curl libssl-dev libreadline-dev zlib1g-dev autoconf bison build-essential libyaml-dev libreadline-dev libncurses5-dev libffi-dev libgdbm-dev

curl -fsSL https://github.com/rbenv/rbenv-installer/raw/HEAD/bin/rbenv-installer | bash

echo 'export PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.bashrc

echo 'eval "$(rbenv init -)"' >> ~/.bashrc

echo 'export EDITOR="code --wait"' >> ~/.bashrc

source ~/.bashrc

1. Run the following commands in your Ubuntu terminal to install the correct version of Ruby and the gems you'll be using in the course:

rbenv install 3.1.1

rbenv global 3.1.1

rbenv rehash

gem install bundler pry byebug

gem install rails -v 7.0.3

rbenv rehash

### PostgreSQL & SQLite

1. Run the following commands in your Ubuntu terminal to install PostgreSQL:

sudo apt-get install postgresql libpq-dev

sudo service postgresql start

source ~/.bashrc

*Note that you will need to run the command sudo service postgresql start****every time you restart the virtual machine****. We recommend creating an alias for this and any other commands you find yourself running on a regular basis.*

1. Run the following commands in your Ubuntu terminal to create a new PostgreSQL user

*Note that you need to replace your\_username with the username that you created for your Ubuntu virtual machine*

sudo -u postgres psql

CREATE USER your\_username WITH SUPERUSER CREATEROLE CREATEDB REPLICATION;

ALTER ROLE your\_username WITH BYPASSRLS;

CREATE DATABASE your\_username;

\q

1. Run the following command in your Ubuntu terminal to install SQLite:

sudo apt-get install sqlite3 libsqlite3-dev

### nvm & Node.js

1. Run the following commands in your Ubuntu terminal to install nvm and the correct version of Node.js:

curl -sL https://raw.githubusercontent.com/creationix/nvm/v0.33.11/install.sh -o install\_nvm.sh

bash install\_nvm.sh

source ~/.bashrc

nvm install 16

nvm use 16

source ~/.bashrc

# Ubuntu Setup

## Requirements

You must be running either of the two most recent LTS versions - 18.04 or 20.04.

## What to install

### Google Chrome

1. Using your default internet browser, navigate to the [Google Chrome Download](https://www.google.com/chrome/) page.
2. Download and run the installer.
3. When prompted, set Google Chrome as your default internet browser.

### Visual Studio Code

1. Using Google Chrome, navigate to the [VS Code Download](https://code.visualstudio.com/Download) page.
2. Download and run the Linux .deb installer.

### rbenv & Ruby

1. Run the following commands in your Ubuntu terminal to install rbenv:

sudo apt install git curl libssl-dev libreadline-dev zlib1g-dev autoconf bison build-essential libyaml-dev libreadline-dev libncurses5-dev libffi-dev libgdbm-dev

curl -fsSL https://github.com/rbenv/rbenv-installer/raw/HEAD/bin/rbenv-installer | bash

echo 'export PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.bashrc

echo 'eval "$(rbenv init -)"' >> ~/.bashrc

echo 'export EDITOR="code --wait"' >> ~/.bashrc

source ~/.bashrc

1. Run the following commands in your Ubuntu terminal to install the correct version of Ruby and the gems you'll be using in the course:

rbenv install 3.1.1

rbenv global 3.1.1

rbenv rehash

gem install bundler pry byebug

gem install rails -v 7.0.3

rbenv rehash

### PostgreSQL & SQLite

1. Run the following commands in your Ubuntu terminal to install PostgreSQL:

sudo apt-get install postgresql libpq-dev

sudo service postgresql start

source ~/.bashrc

*Note that you will need to run the command sudo service postgresql start every time you restart the virtual machine. We recommend creating an alias for this and any other commands you find yourself running on a regular basis.*

1. Run the following commands in your Ubuntu terminal to create a new PostgreSQL user:

*Note that you need to replace your\_username with the username that you created for your Ubuntu virtual machine*

sudo -u postgres psql

CREATE USER your\_username WITH SUPERUSER CREATEROLE CREATEDB REPLICATION;

ALTER ROLE your\_username WITH BYPASSRLS;

CREATE DATABASE your\_username;

\q

1. Run the following command in your Ubuntu terminal to install SQLite:

sudo apt-get install sqlite3 libsqlite3-dev

### nvm & Node.js

1. Run the following commands in your Ubuntu terminal to install nvm and the correct version of Node.js:

curl -sL https://raw.githubusercontent.com/creationix/nvm/v0.33.11/install.sh -o install\_nvm.sh

bash install\_nvm.sh

nvm install 16

nvm use 16

source ~/.bashrc