**Get Your Tools Installed (Mac)**

You will be installing the required tools and software for the course. There's a lot to get through, so buckle in and get ready!

**Before You Begin**

First, create accounts for the following services, which you'll need throughout the course. Don't just create logins; job recruiters often scour these sites in search of job candidates, so be sure to provide at least a headshot and contact information.

* [LinkedIn](https://www.linkedin.com/)

**Note:** You should create a full profile highlighting your skills and work experience, and include a headshot.

* [GitHub](https://github.com/)
* [Stack Overflow](http://stackoverflow.com/)

In addition, be sure to accept the invite for your section on [Slack](https://slack.com/). You will receive the link to your class-specific channel during orientation. If you don't receive this email, contact your Student Success Manager for assistance.

**Tool and Software Installations**

Follow the instructions below to complete the installation process for all of the required tools.

**Google Chrome**

1. If you don’t already have Chrome installed, visit the download page [here](https://www.google.com/chrome/browser/desktop/index.html).

2. Download, open, and run the installation file.

**Screencastify**

**Note:** You must be using Google Chrome in order to install and use Screencastify.

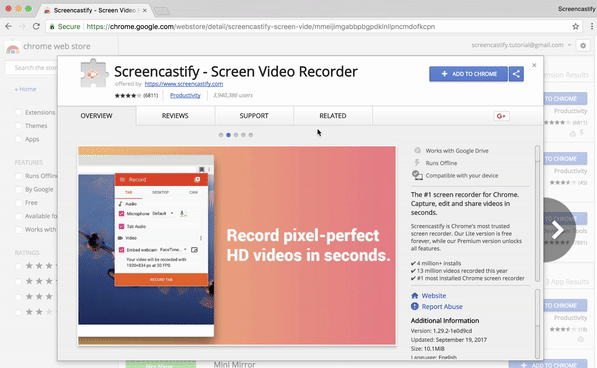
1. Open Google Chrome.

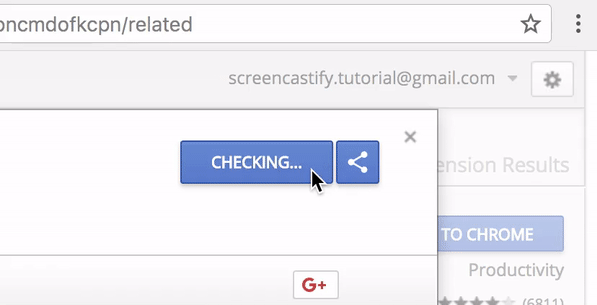
2. Go to the [Chrome Web Store](https://chrome.google.com/webstore/category/extensions) and type **Screencastify** in the **Search Extensions** bar in the top left of the page. **Note:** You can also start by going to the [Screencastify website](https://www.screencastify.com/) and clicking the **Add to Chrome** button, which will take you directly to its listing in the Chrome Web Store.

3. Click the **Add to Chrome** button.

4. Click **Add extension**.

5. When the download is complete, click the gray film strip icon in the top-right corner of your browser screen to launch Screencastify.



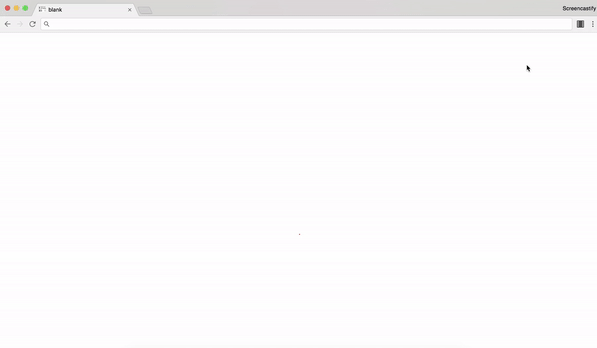


**Allow Webcam and Microphone Access**

The first time you click on the Screencastify icon, you'll be asked to give Screencastify permission to access your webcam and microphone, which enables you to narrate over your recordings as well as record your webcam. You are required to allow access in order for the Screencastify extension to function properly. You do not need to include audio or video in your recordings, and Screencastify never records audio or video unless you expressly ask it to.

1. Click the **Setup Camera Access** button.

2. Click **Allow** in the pop-up window.



**Note:** If Screencastify is unable to detect your microphone, click [here](https://help.screencastify.com/article/206-i-dont-hear-any-audio-from-my-microphone). If it's unable to detect your webcam, click [here](https://help.screencastify.com/article/205-my-webcam-is-not-detected).

**Connect Your Google Drive to Screencastify**

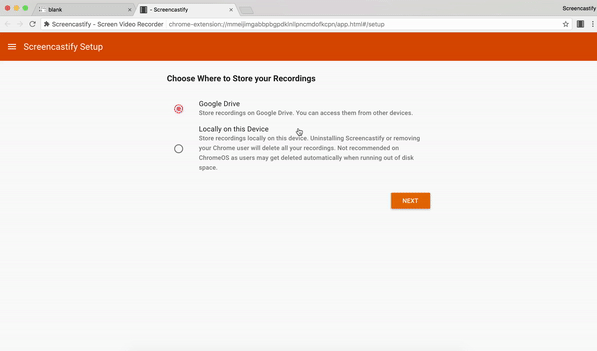
It is recommended that you save all of your Screencastify recordings to Google Drive, which is safer than storing them locally on your computer. By connecting your Google Drive to Screencastify, your recordings will be saved there automatically. The length of this process depends on how long a particular recording is. **Note:** Screencastify never views, modifies, stores, or in any way interacts with the other files in your Google Drive.

The first time you open the Screencastify extension, you'll be asked where you want to store your recordings (right after you allow webcam and microphone access). Follow these steps:

1. Confirm that Google Drive is selected, and then click **Next**.

2. Sign in to your Google account. **Note:** You can sign in with any Google-based account, not just an @gmail.com address.

3. Allow Screencastify permission to access your Google Drive.



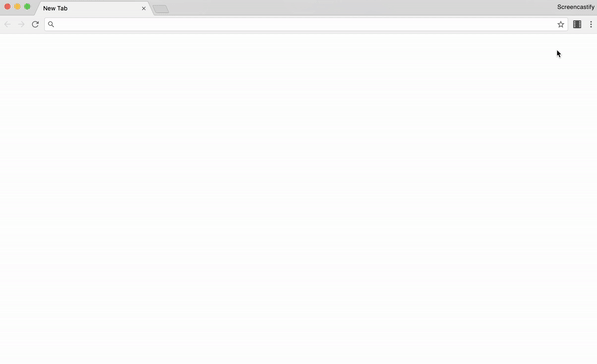
And that's all you need to do! As soon as you finish your first recording, a new folder named Screencastify will be created in your Drive. All of your recordings will be saved in this folder.

The videos you will see in Your Recordings library in Screencastify are synced with this Screencastify folder in your Drive. This means:

* If you delete a video in Screencastify, it will also be deleted from your Drive.
* If you move a video out of the Screencastify folder in your Drive, it will no longer be accessible from Your Recordings in Screencastify.
* If you rename a video in Screencastify, it will also be renamed in your Drive.

You can always change the default storage location for your recordings, even after initial setup. Simply open the extension menu and click **Options**. The first thing you'll see is an option to select where you want to store your recordings.

Want to know how to record, save to your Google Drive and share your recordings? Check it out [here](https://coding-bootcamp-fsf-prework.readthedocs-hosted.com/en/latest/modules/screencastify-record-save-to-your-google-drive-and-share-your-video/)!



**Slack**

1. Search for Slack in the App Store. Click **Get** under the app name and icon, and then click **Install**.

2. Open the Finder window, select the Applications folder from the sidebar, and then find and double-click the Slack icon to launch the program.

3. To keep Slack in your dock, right-click the Slack icon, click **Options**, and then click **Keep in Dock**.

4. Add our channel to the application and then click **Join Now**. You will receive the link to your class-specific channel during orientation.

5. Enter your full name and optional display name, create a password, and then click **Next**.

6. Click your workspace name in the top left to open the menu.

7. Select **Sign in to another team**.

8. Enter your team’s Slack domain that you received during orientation. Click **Continue**.

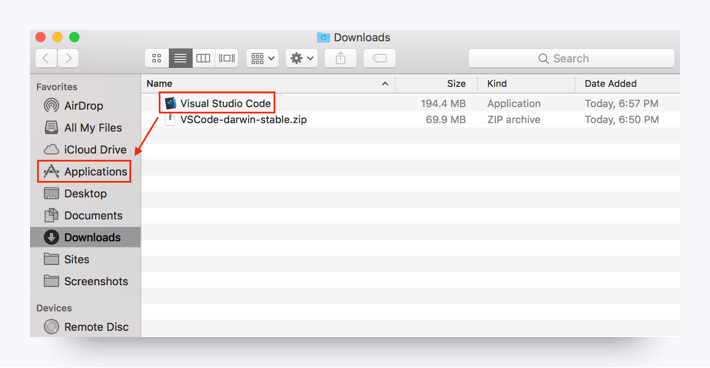
9. Enter your email address (the one we used to invite you) and password to sign in. When you see the chatroom, you're finished.

**VS Code**

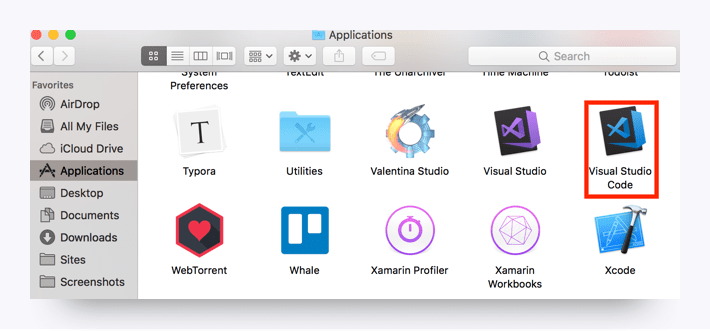
1. Go to the [setup page](https://code.visualstudio.com/docs/setup/setup-overview) on the VS Code website and select Mac as your platform.

2. Open your Downloads folder and click on the file to unzip.

3. Drag the file to your Applications folder.



You should now be able to access VS Code from your Applications folder.

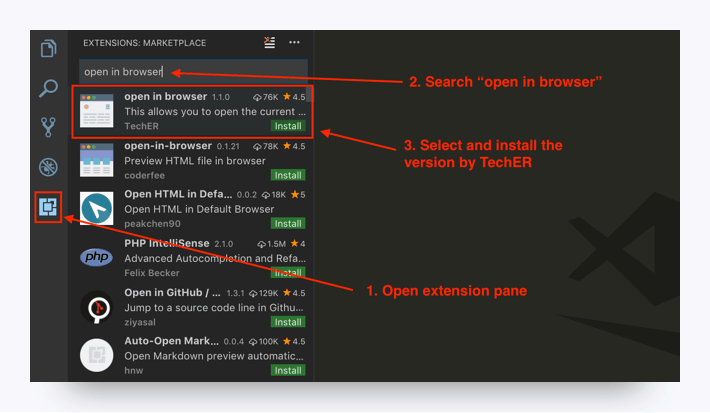


**Open in Browser (VS Code Extension)**

1. Open VS Code.

2. Open the extensions pane and search for **open in browser**.

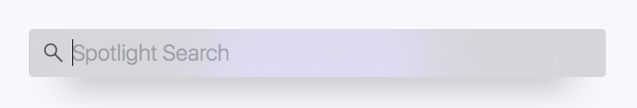
3. Select the version written by TechER and click **Install**.



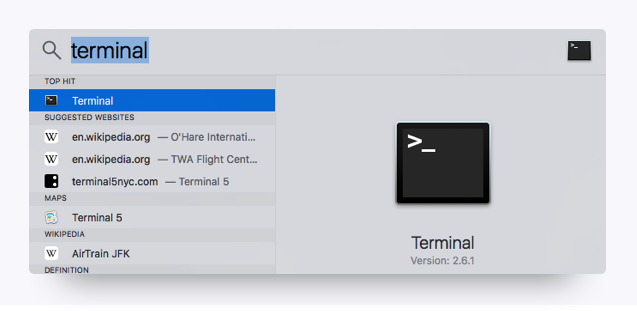
**Terminal**

You’ll be entering your command line code through this interface. Since you’re on a Mac right now, you already have it! Just follow these steps to open the program.

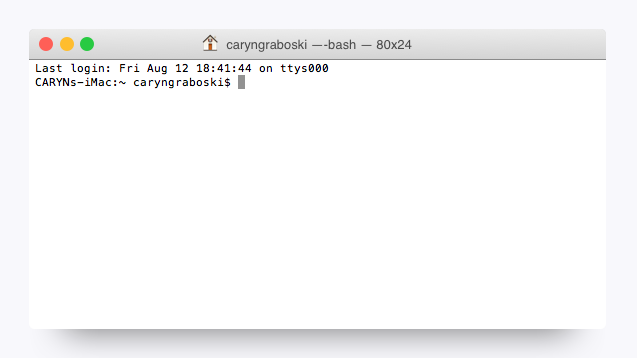
**Step 1.** Press command + spacebar to open Spotlight Search.



**Step 2.** Type **terminal** into the search and then press **Enter**.



**Step 3.** Keep this window open; you’ll need it for the next steps.



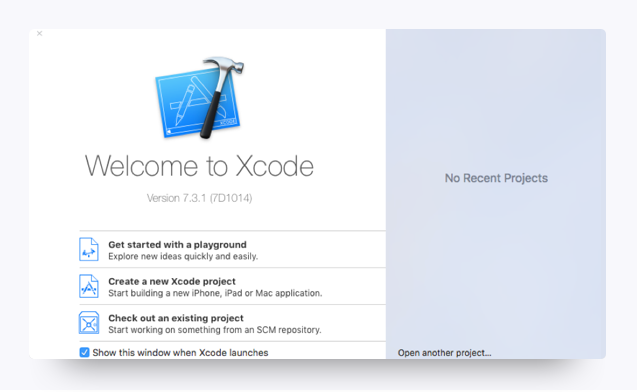
**Xcode**

**Step 1.** Open the App Store and search for Xcode.



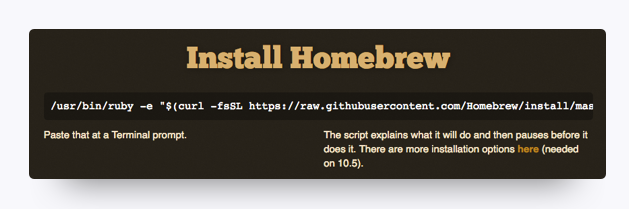
**Step 2.** Download and install Xcode. When the download is complete, open the program.

**Step 3.** Agree to the terms of service.

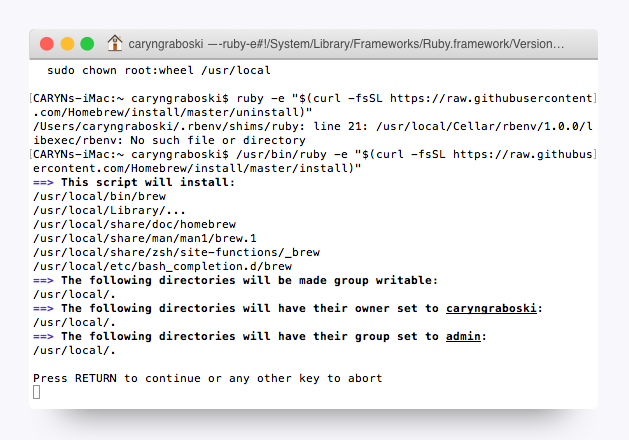
**Step 4.** When you see the window below, the installation is complete.

**Homebrew**

**Step 1.** Go to [Homebrew](http://brew.sh/). Copy the script listed under Install Homebrew.



**Step 2.** Paste the script into your Terminal window. Press **Enter**.



**Heroku Toolbelt**

**Step 1.** Sign up for a free [Heroku](https://signup.heroku.com/) account.

**Step 2.** Download the [installer](https://toolbelt.heroku.com/) and read the installation guide.

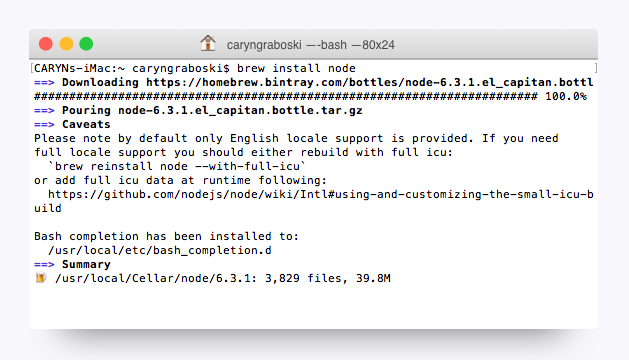
**Step 3.** Open Terminal. Type heroku login into the command line and press **Enter**.

![Mod5-31](assets/Mod5-31.png)

**Step 4.** When prompted, enter your Heroku credentials.

**Node.js**

**Step 1.** Run the following command in Terminal: brew install node. **Note:** Homebrew must be installed in order for you to run this command.



**Step 2.** Type node –v in Terminal, and press **Enter**. If Terminal returns a version number, then you’ve successfully installed Node.

**MySQL Workbench**

**Install MySQL Workbench**

1. Go to the [MySQL downloads](https://dev.mysql.com/downloads/workbench/) page.

2. Scroll to the bottom of the page and select Mac as your operating system.

3. Download the DMG Archive, not the TAR Archive.

4. Follow the installation instructions.

**SSH Keys**

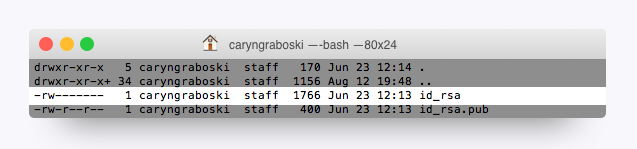
To complete these steps, you will need to sign up for a [GitHub](https://github.com/) account if you haven't already.

**Step 1.** Open Terminal.

**Step 2.** To make sure you don’t already have a set of keys on your computer, type the following in your Terminal window. (**Note:** Copying and pasting will not work!)

​ ls –al ~/.ssh

* If no keys pop up, move on to **Step 3**.
* If keys do pop up, check that none of them are listed under id\_rsa, like in this image:



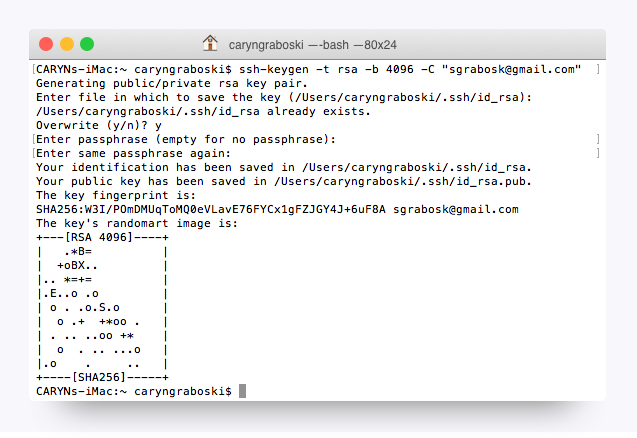
**Note:** If you find a key with a matching name, you can either overwrite it by following the next steps, or you can use the same key referenced in Step 8. If you decide not to overwrite it, you will need to remember the password tied to your key.

**Step 3.** Enter the following command along with your email to generate your keys.

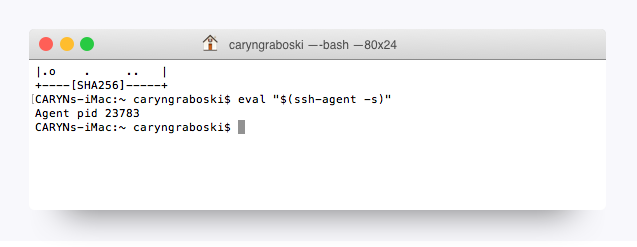
ssh-keygen –t rsa –b 4096 –C "YOURGITHUBEMAIL@PLACEHOLDER.NET"

**Step 3a. (optional)** If the command above causes an error, try the following command instead:

ssh-keygen -t rsa

**Step 4.** When prompted to enter a file to save the key, press **Enter**, and then enter a passphrase for your key. **Note:** You shouldn’t see any characters appear in the window while typing the password. When you’re finished, your window should look like this:

**Step 5.** Link your key to your machine using a tool called the ssh-agent. Run the following command to test whether the ssh-agent is running on your machine: eval "$(ssh-agent –s)". Your Terminal window should look like this:



**Step 5a. (optional)** If you're using macOS Sierra 10.12.2 or later, you need to modify your ~/.ssh/config file to the following:

Host \*

AddKeysToAgent yes

UseKeychain yes

IdentityFile ~/.ssh/id\_rsa

**Step 6.** Run the following command: ssh-add ~/.ssh/id\_rsa.

Depending on your version of ssh-add you may need the -K option. If the command above returns an error, run the following command instead: ssh-add -K ~/.ssh/id\_rsa

**Step 7.** When prompted, enter the passphrase associated with the key. **Note:** If you’ve forgotten this key, go back to Step 3.

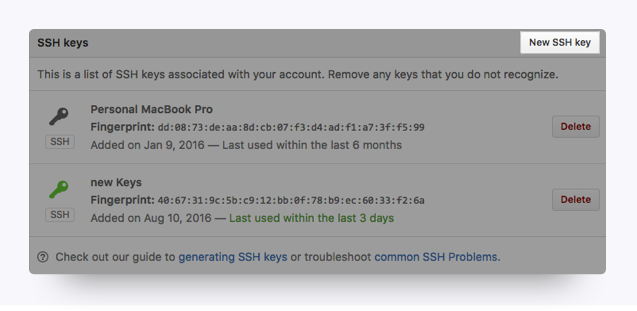
**Step 8.** To add the key to GitHub, first copy the key to your clipboard by entering the following command:

pbcopy < ~/.ssh/id\_rsa.pub.

You shouldn’t see any kind of message when you run this command. If you do, make sure you entered it correctly.

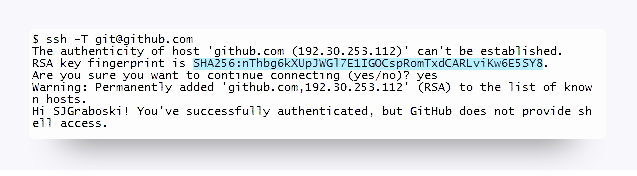
**Note:** Do not copy anything else to your clipboard until all steps are completed. Otherwise, you’ll need to enter the copy command again.

**Step 9.** Go to GitHub's [SSH key settings](https://github.com/settings/ssh). Click **New SSH key**.



**Step 10.** When the form pops up, enter a name for your computer in the Title input. In the Key input, paste the SSH key you copied in Step 8.

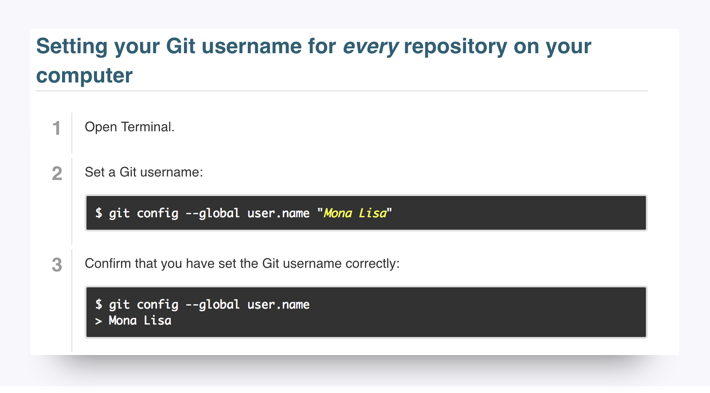
**Step 11.** To add GitHub to your computer’s list of acceptable SSH hosts, type the following command in your Terminal window: ssh –T git@github.com. - You should see an RSA fingerprint in your window. Enter **yes** only if it matches the one highlighted in the image below:



**Setting Your Git Username for *Every* Repository on Your Computer**

Git uses a username to associate commits with an identity. The Git username is not the same as your GitHub username.

You can change the name that is associated with your Git commits using the git config command. The new name you set will be visible in any future commits you push to GitHub from the command line. If you'd like to keep your real name private, you can use any text as your Git username. Changing the name associated with your Git commits using git config will only affect future commits and will not change the name used for past commits.

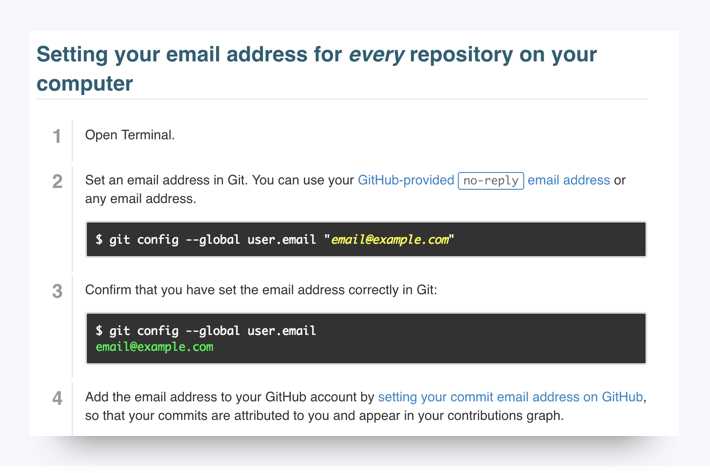


**Setting Your Email Address for *Every* Repository on Your Computer**

GitHub uses the email address set in your local Git configuration to associate commits pushed from the command line with your GitHub account.

You can use the git config command to change the email address you associate with your Git commits. The new email address you set will be visible in any future commits you push to GitHub from the command line. Any commits you made prior to changing your commit email address are still associated with your previous email address.

For more information on commit email addresses, including your GitHub-provided noreply email address, see "[About commit email addresses](https://help.github.com/articles/about-commit-email-addresses/)."



**You're Done!**

That's all for the installations, so give yourself a pat on the back! Installations are never fun, but just like taxes, you gotta do them.

Be sure to take a break before continuing with the rest of the prework.