



Sponsored by:



Git Install Guide - Mac

1. Open the Terminal application on your computer

Note: You will require sufficient permissions on your device to install git and its dependencies. This includes administrator permissions and the password for the user

To install git and its dependencies we will use the command line through the terminal on Mac.

2. Install Homebrew

In order to install git on a Mac we need to install Homebrew. Homebrew is a package manager that allows for the easy installation of packages commonly used in development.

The latest version of Homebrew can be installed using the command `/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"`

```

brown@Roberts-MacBook-Pro ~ % /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
Password:
==> This script will install:
/usr/local/bin/brew
/usr/local/share/doc/homebrew
/usr/local/share/man/man1/brew.1
/usr/local/share/zsh/site-functions/_brew
/usr/local/etc/bash_completion.d/brew
/usr/local/Homebrew
==> The following existing directories will be made writable by user only:
/usr/local/share/zsh
/usr/local/share/zsh/site-functions

Press RETURN to continue or any other key to abort
==> /usr/bin/sudo /bin/chmod u+rw /usr/local/share/zsh /usr/local/share/zsh/site-functions
==> /usr/bin/sudo /bin/chmod 755 /usr/local/share/zsh /usr/local/share/zsh/site-functions
==> /usr/bin/sudo /usr/sbin/chown brown:admin /usr/local/Homebrew
==> Downloading and installing Homebrew...
remote: Enumerating objects: 42, done.
remote: Counting objects: 100% (42/42), done.
remote: Total 110 (delta 42), reused 42 (delta 42), pack-reused 68
Receiving objects: 100% (110/110), 53.49 KiB | 750.00 KiB/s, done.
Resolving deltas: 100% (49/49), completed with 15 local objects.
From https://github.com/Homebrew/brew
 * [new branch]      dependabot/bundler/Library/Homebrew/nokogiri-1.11.0 -> origin/dependabot/bundler/Library/Homebrew/nokogiri-1.11.0
 * [new branch]      dependabot/bundler/docs/html-proofer-3.18.5 -> origin/dependabot/bundler/docs/html-proofer-3.18.5
 * [new branch]      dependabot/github_actions/dessant/lock-threads-486f7380c15596f92b724e4260e4981c68d6bde6 -> origin/dependabot/github_actions/dessant/lock-threads-486f7380c15596f92b724e4260e4981c68d6bde6
HEAD is now at 7eca0d77c Merge pull request #10206 from psobolewskiPhD/patch-1
Already up-to-date.
==> Installation successful!

==> Homebrew has enabled anonymous aggregate formulae and cask analytics.
Read the analytics documentation (and how to opt-out) here:
  https://docs.brew.sh/Analytics
No analytics data has been sent yet (or will be during this `install` run).

==> Homebrew is run entirely by unpaid volunteers. Please consider donating:
  https://github.com/Homebrew/brew#donations

==> Next steps:
- Run `brew help` to get started
- Further documentation:
  https://docs.brew.sh
brown@Roberts-MacBook-Pro ~ %

```

3. Install Git

Once Homebrew is installed you can use it to install git. The `brew install [name]` command tells Homebrew to install the package that you specify.

To install git use `brew install git`

```

brown@Roberts-MacBook-Pro ~ % brew install git
==> Downloading https://homebrew.bintray.com/bottles/git-2.30.0.big_sur.bottle.tar.gz
==> Downloading from https://d29vzk4ow07wi7.cloudfront.net/bd4dd57a4740df3c596377954d45fcf2b08db83f7d8853f5361db1d158a6bdb8?response-content-disposition=attachment%3Bfilename%3D%22git-2.30.0.big_sur.bottle.tar.gz%22&Policy=eyJTdGF0ZW11bnQiOiBbeyJSZXNvdXJjZSI6Imh0dHAQOi8##### 100.0%
==> Pouring git-2.30.0.big_sur.bottle.tar.gz
==> Caveats
The Tcl/Tk GUIs (e.g. gitk, git-gui) are now in the `git-gui` formula.

zsh completions and functions have been installed to:
  /usr/local/share/zsh/site-functions

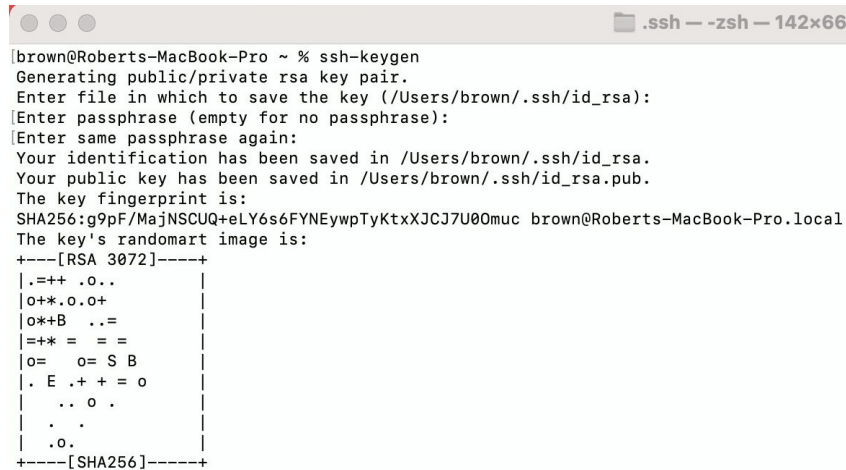
Emacs Lisp files have been installed to:
  /usr/local/share/emacs/site-lisp/git
==> Summary
📦 /usr/local/Cellar/git/2.30.0: 1,486 files, 40.5MB
Removing: /Users/brown/Library/Caches/Homebrew/git--2.29.2.big_sur.bottle.tar.gz... (14.6MB)
brown@Roberts-MacBook-Pro ~ % git --version
git version 2.30.0
brown@Roberts-MacBook-Pro ~ % git config --global user.name "your_github_username"
brown@Roberts-MacBook-Pro ~ % git config --global user.email "your_email@github.com"

```

Once git is installed you will need to set your username and email. This is done using the command `git config --global user.name "your_github_username"` and `git config --global user.email "your_email@github.com"`

4. Generate SSH Keys

To make our GitHub experience smoother we will generate ssh keys. This allows us to securely communicate with GitHub. If you're interested in how this works you can research public/private key encryption. To generate an ssh key pair type `ssh-keygen`. Accept the defaults.



```

brown@Roberts-MacBook-Pro ~ % ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/brown/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /Users/brown/.ssh/id_rsa.
Your public key has been saved in /Users/brown/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:g9pF/MajNSCUQ+eLY6s6FYNEywpTyKtxXJCJ7U00muc brown@Roberts-MacBook-Pro.local
The key's randomart image is:
+---[RSA 3072]-----+
| .==+ .O.. |
| o+*.O.O+ |
| o*B ..= |
| =+* = = |
| o= o= S B |
| . E .+ + = o |
| .. O . |
| . |
| .o. |
+---[SHA256]-----+

```

5. Add our ssh keys to GitHub

We now need to add the public key to our GitHub account.

- Type `cd ~/.ssh`. This will change directory to the folder we just made the keys in. If we type `ls` it will list the contents of this folder.
- Type `cat id_rsa.pub` to show the contents of your public key.
- Copy your public key

```

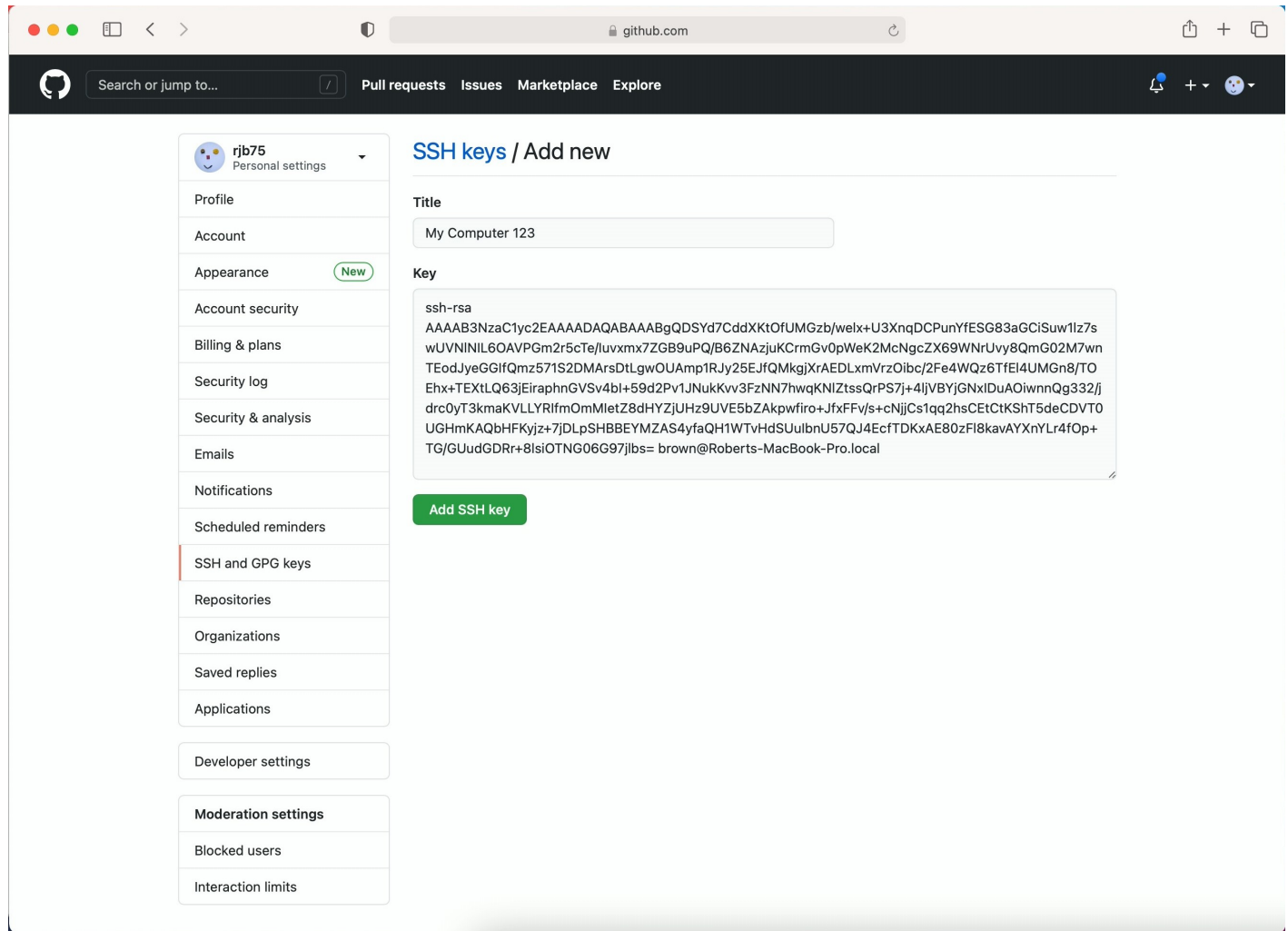
brown@Roberts-MacBook-Pro ~ % cd ~/.ssh
brown@Roberts-MacBook-Pro .ssh % ls
config      id_ed25519      id_ed25519.pub  id_rsa          id_rsa.pub      known_hosts
brown@Roberts-MacBook-Pro .ssh % cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQgQDSYd7CddXKtOfUMGzb/welx+U3XnqDCPunYfESG83aGciSuw1lz7swUVN1NIL60AVPGm2r5cTe/Iuvxmx7ZGB9uPQ/B6ZNAzjuKCz
0pWeK2McNgcZX69WNRUvy8QmG02M7wnTEodJyeGGIfQmz571S2DMArsDtLgwOUAmp1RJy25EJfQMkgjXrAEDLxmVrz0ibc/2Fe4WQz6TfE14UMGn8/TOEhx+TEXTLQ63jEiraphnGVs
I+59d2Pv1JNukKvv3FzNN7hwqKN1ZtssQrPS7j+41jVBYjGNxIDuA0iwnnQg332/jdrc0yT3kmaKVLlyRIfmOmMIetZ8dHYZjUH9UVE5bZakpwfiro+JfxFFv/s+cNjjCs1qq2hsCE
KShT5deCDVT0UGHmKAQbHFkyjz+7jDLpSHBBEYmZAS4yfaQH1WtVHdSUuIbnU57QJ4EcftDKxAE80zF18kavAYXnYLR4f0p+TG/GUudGDRr+81si0TNG06G97jlbs= brown@Robert
acBook-Pro.local

```

Next navigate to github.com and create an account if needed. To edit your ssh keys:

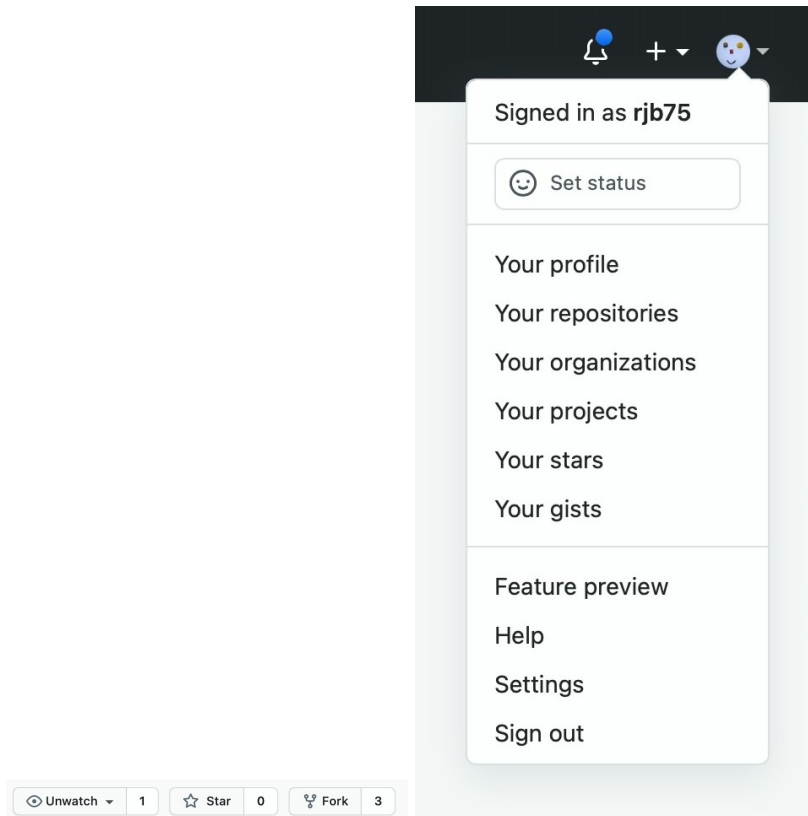
- Click your profile picture in the top right
- Go to settings
- Choose SSH and GPG keys from the sidebar
- Click "New SSH Key".
- Paste the contents of your public key in the key field. The title field doesn't really matter.
- Click "Add SSH key".

Your ssh keys are now set up.



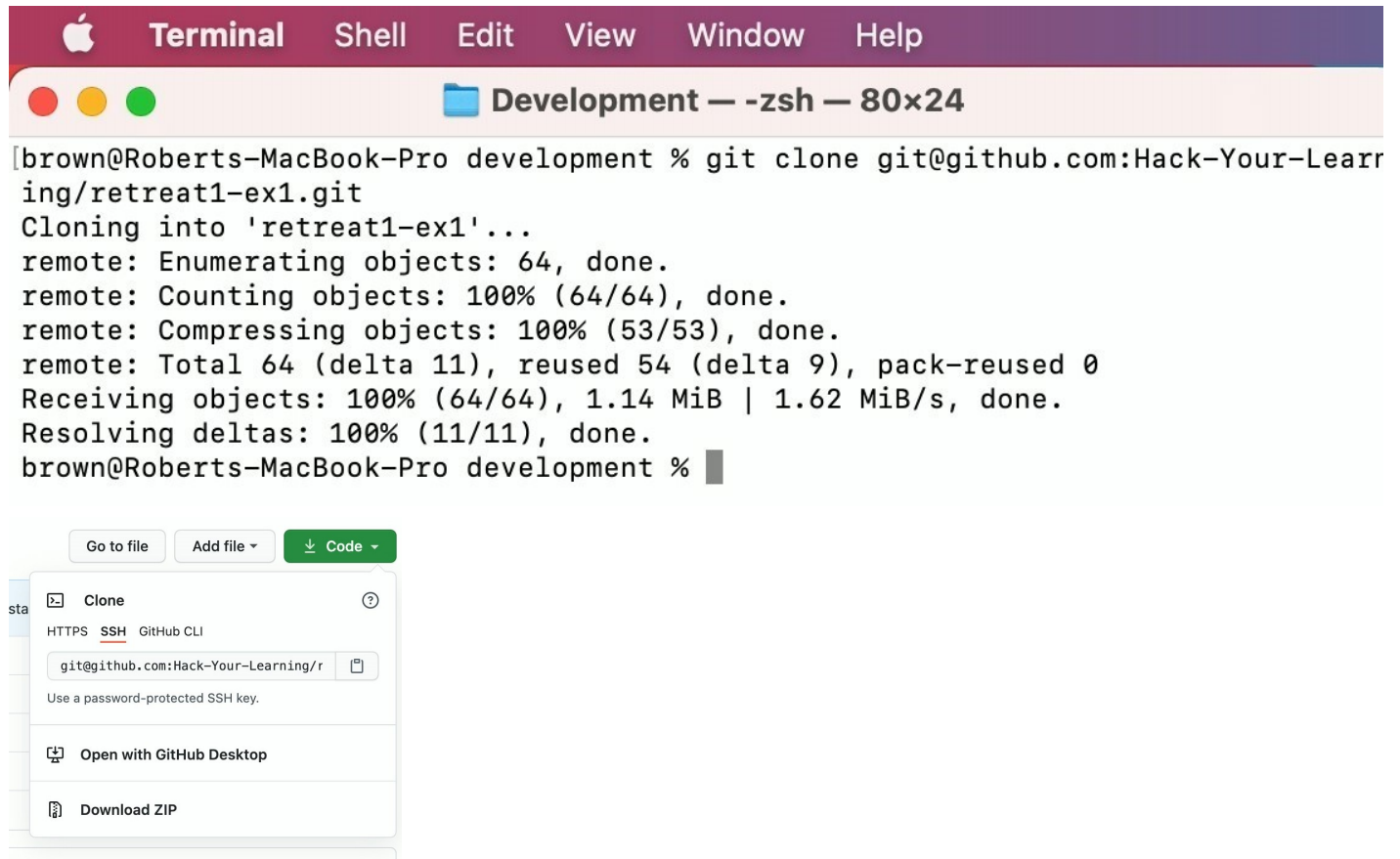
6. Fork the HYL Repository - only one team member needs to do this!

To work on the HYL project one member of your team will need to fork the repository. To do so, navigate to <https://github.com/Hack-Your-Learning/retreat1-ex1>. Click "Fork" in the top right corner. This will create a copy of this repository for your team to work on. GitHub should redirect you to your copy when you create it. If it did not you can find your copy by clicking on your profile picture and then "your repositories".



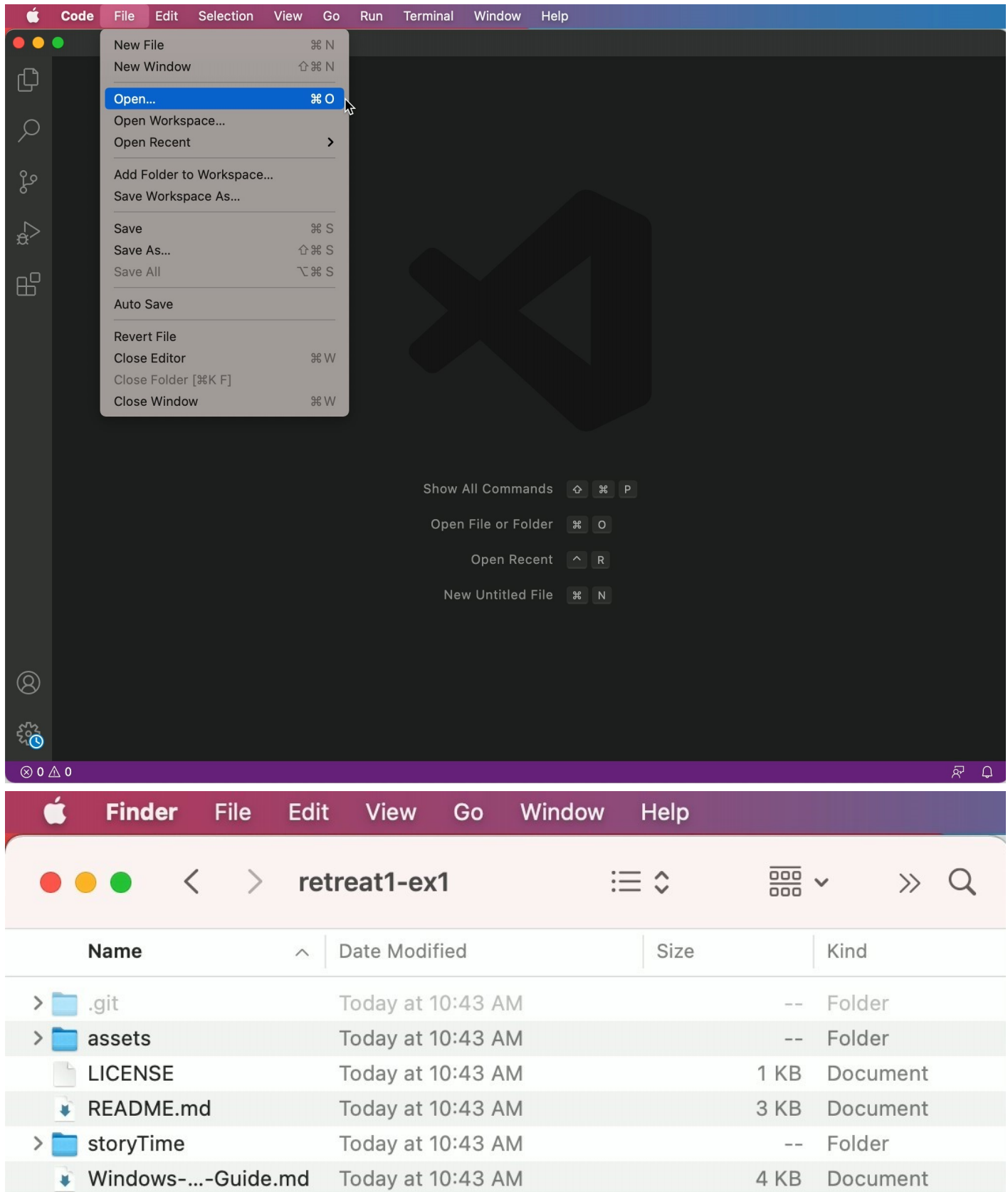
7. Clone the repository

Now that we have our fork on GitHub we need a local copy. We get this copy with the "clone" command. First copy the ssh link by clicking the green "Code" button, choose ssh and then copy the link. Go back to your git bash and `cd` to your project directory. We can clone the project with `git clone <your link>`. If it asks you about adding to known hosts say yes.



8. Open in your preferred text editor

You will now be able to open the files in your favorite text editor. Below you can see how to open it in VSCode.



9. Add collaborators

If you're working with a team you will want to add collaborators. From your repository's page on GitHub, click

- Settings

- Manage Access
- Invite Collaborator

You can add your collaborators by email or GitHub username. To contribute they should clone the repository as described above. Note that there is no need for your teammates to fork either the HYL repository or your forked repository! They should *clone* the repository that you have just *forked*.

