



Sponsored by:



Git Install Guide - Windows

1. Download the installer from <https://git-scm.com/downloads>

Click the download button and your download should start immediately. If it doesn't you can manually click Windows 64 bit as shown below. Keep track of where you saved the installer.

Downloading Git



Your download is starting...

You are downloading the latest (**2.30.0**) **64-bit** version of **Git for Windows**. This is the most recent **maintained build**. It was released **6 days ago**, on 2020-12-28.

[Click here to download manually](#), if your download hasn't started.

Other Git for Windows downloads

Git for Windows Setup

32-bit Git for Windows Setup.

64-bit Git for Windows Setup.

Git for Windows Portable ("thumbdrive edition")

32-bit Git for Windows Portable.

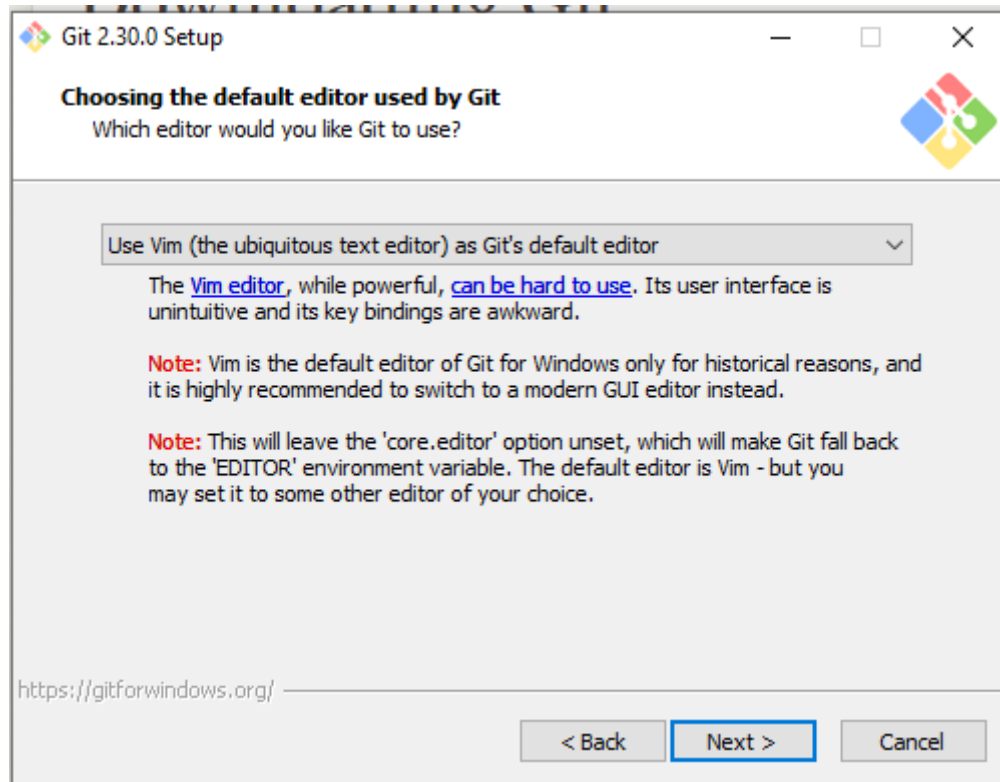
64-bit Git for Windows Portable.

The current source code release is version **2.30.0**. If you want the newer version, you can build it from [the source code](#).

2. Run the installer

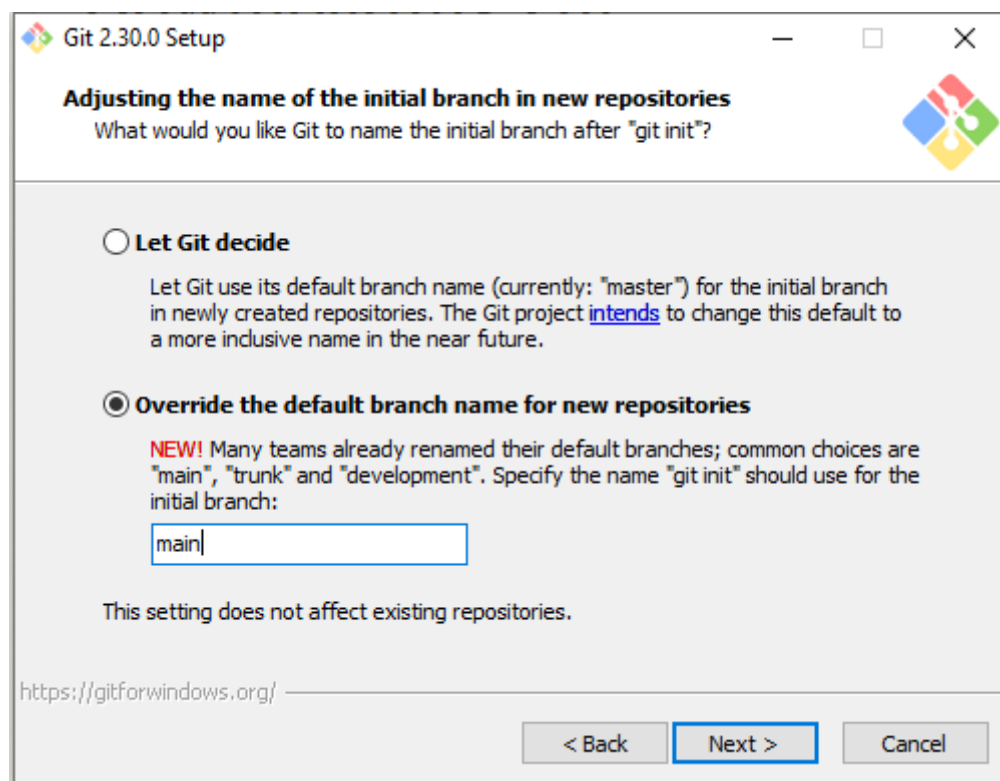
You can accept the default for most of the install. However, it is recommended to pause at the following critical steps to double check the information.

Text editor choice



This step is where you choose your text editor for git bash. It's ok to stick with the default, but if you have a preferred text editor you should specify that here.

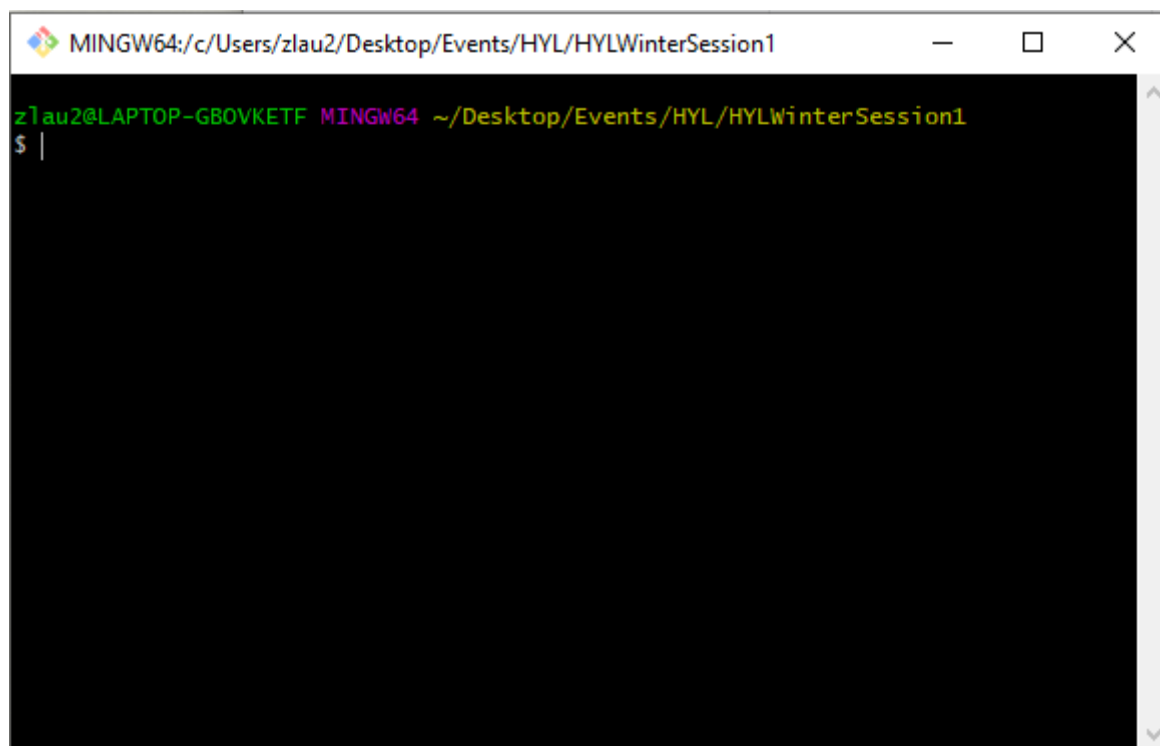
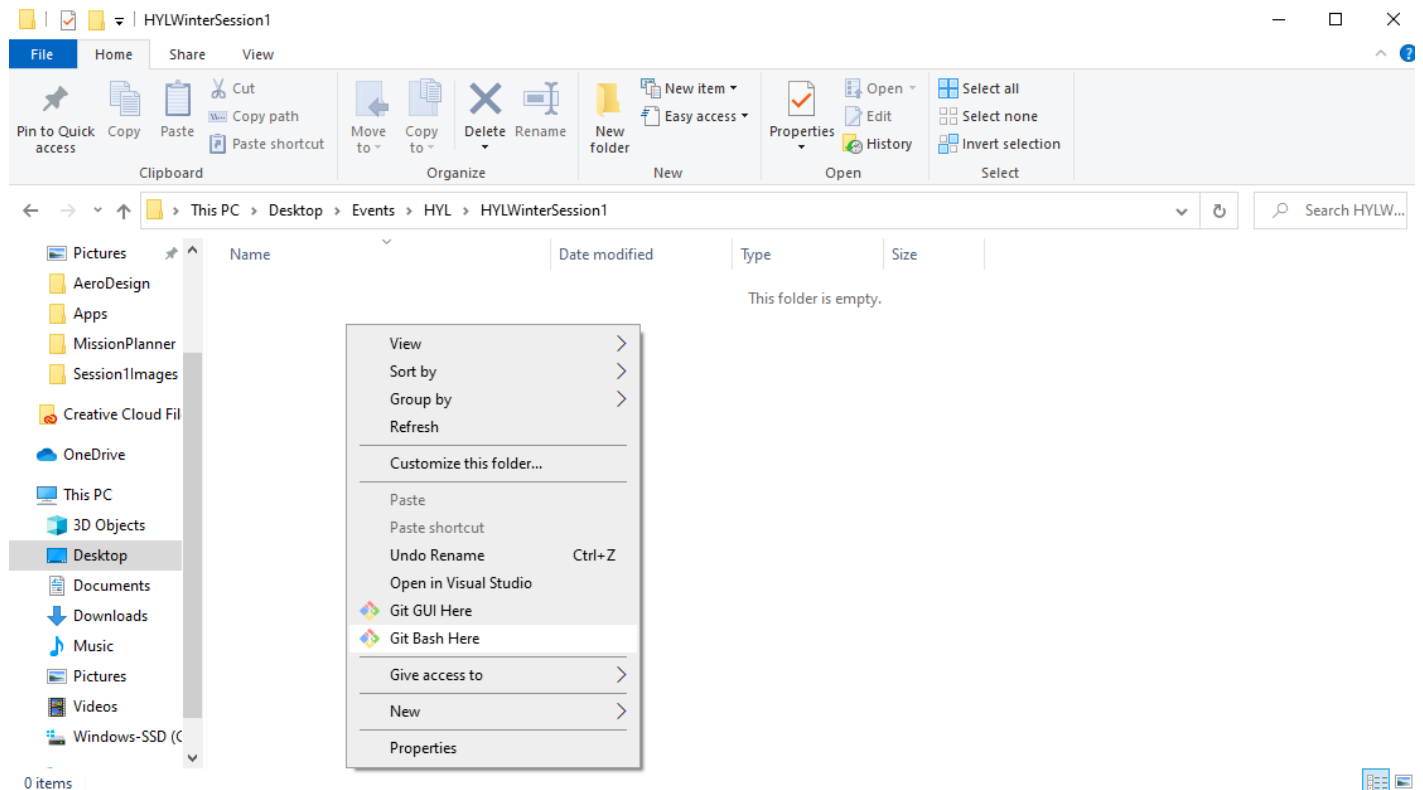
Default branch



It is recommended to choose the default branch name to be "main" since this is what Github uses.

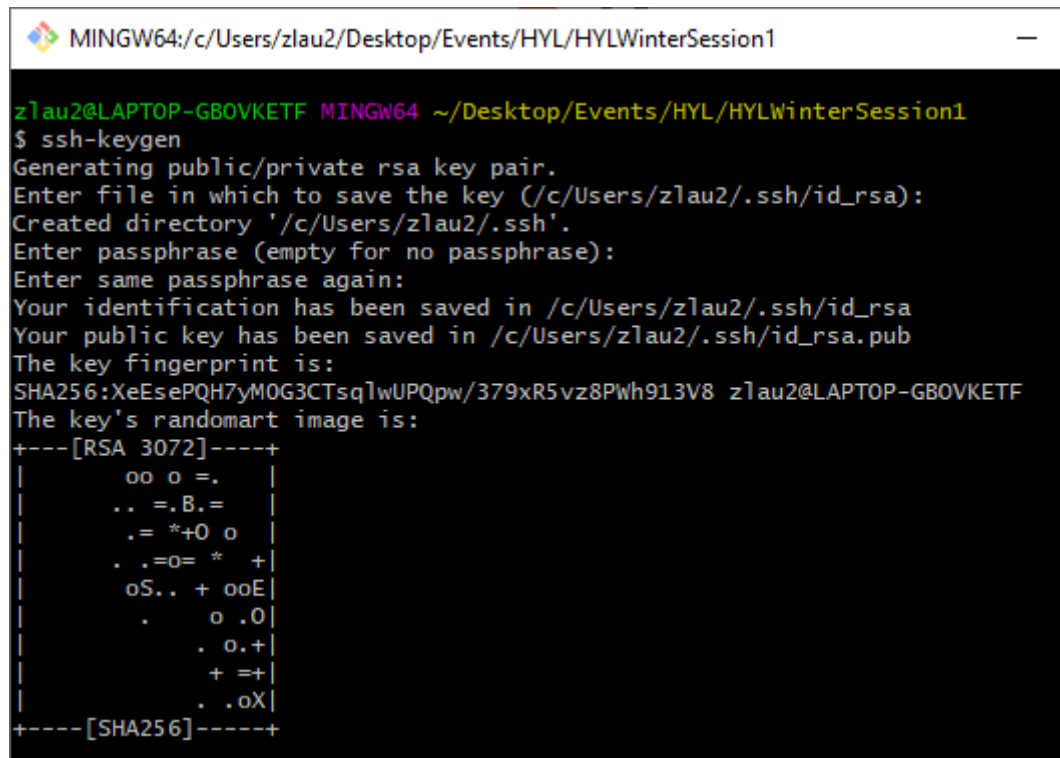
3. Start your git bash

- Create a folder to house your project. Change into this folder
- Start a git bash by right clicking and selecting "Git Bash Here". This will open a terminal like that shown below. All your commands in this workshop will be executed in the git bash terminal.



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.



```
MINGW64:/c/Users/zlau2/Desktop/Events/HYL/HYLWinterSession1
zlau2@LAPTOP-GBOVKETF MINGW64 ~/Desktop/Events/HYL/HYLWinterSession1
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/zlau2/.ssh/id_rsa):
Created directory '/c/Users/zlau2/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/zlau2/.ssh/id_rsa
Your public key has been saved in /c/Users/zlau2/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:XeEsePQH7yM0G3CTsq1wUPQpw/379xR5vz8PWh913V8 zlau2@LAPTOP-GBOVKETF
The key's randomart image is:
+---[RSA 3072]-----+
|      oo o =.      |
|    .. =.B.=      |
|    . = *+0 o      |
|    . .0= * +      |
|    oS.. + ooE      |
|    .   o .0       |
|    . o.+          |
|    + =+          |
|    . .oX          |
+-----[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

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".

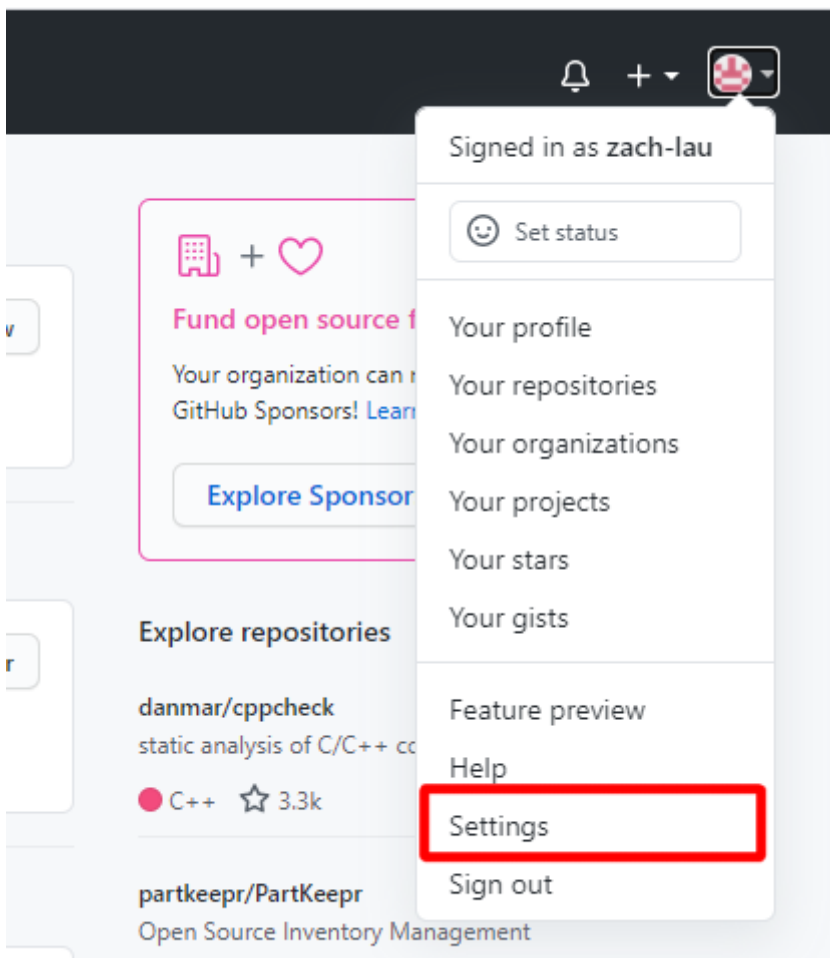
Your ssh keys are now set up.

```
MINGW64:/c/Users/zlau2/.ssh
|
|  oo o =. |
| .. =.B.= |
| . = *+0 o |
| . =0= * + |
| oS.. + ooE |
| .   o .0 |
| . o.+ |
| + =+ |
| . .oX |
| +----[SHA256]-----+
|
z1au2@LAPTOP-GBOVKETF MINGW64 ~/Desktop/Events/HYL/HYLWinterSession1
$ cd ~/.ssh

z1au2@LAPTOP-GBOVKETF MINGW64 ~/.ssh
$ ls
id_rsa  id_rsa.pub

z1au2@LAPTOP-GBOVKETF MINGW64 ~/.ssh
$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAgQCI11QVUx2BPqQJ2kn7WRp4yAKq/sd7oLOGMA1ERicZ1aQR/INe1E9vTbBXGu+Pge74MouJ0uwTIivTHYLVqgJc54y8iw61y/vAZ9TfSAxyvCgIzFaUHzPLimhxBendYhSJdFn
n00uWc00uykEtFvi/wDDn42Pn7NGAFdnERdemY5zT6wRgLUYHpBRR3UTaH8wV/hsuLg0kJPppcHPsx8D+Hn188
3cmN9YYnWjZiduUmMY2kqVQW08oY61SK1fRZ0aCwokH1pp31JHnxxRK35dPYh1rVv1Io1/mJI6d5AvMg6wzYh
D1JSuwtOEGgIOYGWUkzzszCHrVQ8I4kvWi7JcHyjGq//AhP0GjNiQnStbc7SV7bNoxwC5a1gUsvvqTP4v/4fqC
Ckiiw9dmFHJE02k11IxFSRjbbboYU0s5uVQ7v2eUt/7sIEjpMH+3K9tY219W96ChVckFd/nQY1HDVPrWJIj2Z+R
WwDQNhSi0tEYjnAR5V8IQ02MYDTg17vfODk= z1au2@LAPTOP-GBOVKETF

z1au2@LAPTOP-GBOVKETF MINGW64 ~/.ssh
$ |
```



JavaScript ☆ 795

Okay, you have successfully deleted that key. ×

zach-lau
Personal settings

Profile

Account

Appearance New

Account security

Billing & plans

Security log

Security & analysis

Emails

Notifications

Scheduled reminders

SSH and GPG keys





Repositories

Organizations

Saved replies

SSH keys

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

 SSH	Lenovo SHA256:HISTuf8RQX35hw6xDi1eH0RhcZF64RMQc1zHe+tMFqc Added on Jul 29, 2019 Last used within the last week — Read/write	Delete
 SSH	Pi SHA256:79uz1v709dNmzzE53aZ62gX476M10cObWDzhI3R65mY Added on Feb 29, 2020 Last used within the last 10 months — Read/write	Delete
 SSH	Work Computer SHA256:M1xLxgmxDyrJ9+d8eQ754yLPJMB7qCZEMS1BmrKwDLg Added on Jul 14, 2020 Last used within the last 6 months — Read/write	Delete
 SSH	Hackathon Box SHA256:Y7HRYdvdRXoic1SLI376sRvgPL1w3jw0xs7k589x0E Added on Jul 15, 2020 Never used — Read/write	Delete

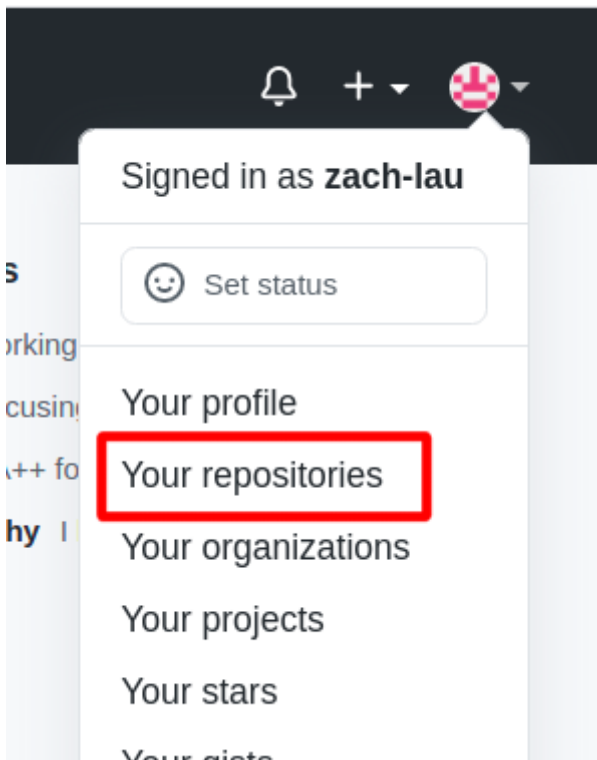
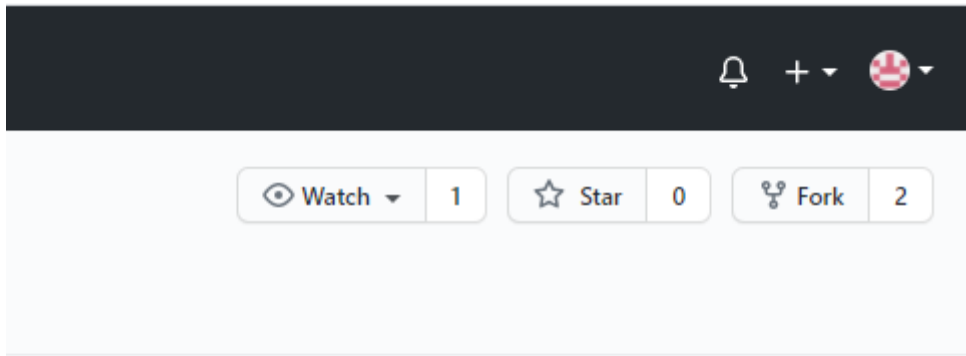
1

2

New SSH key

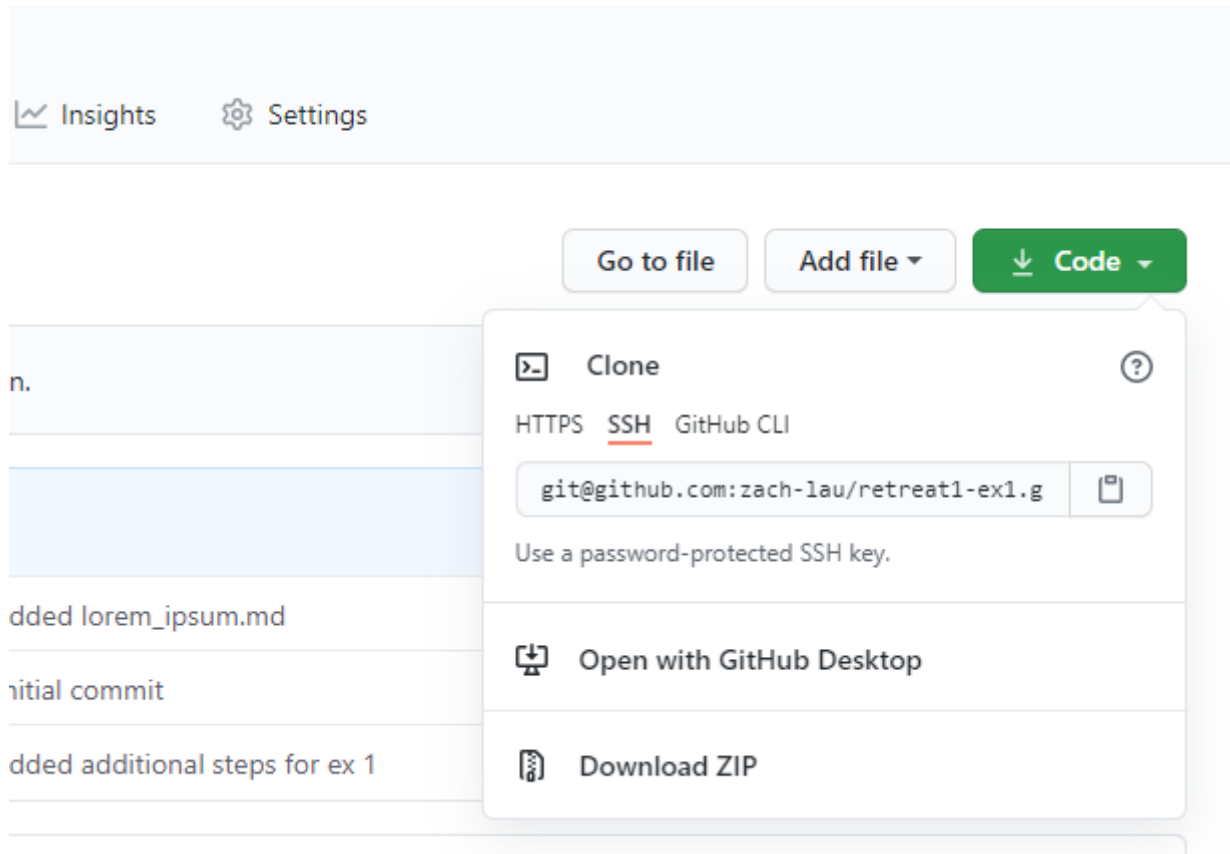
6. Fork the HYL Repository - only one team member needs to do this!

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.

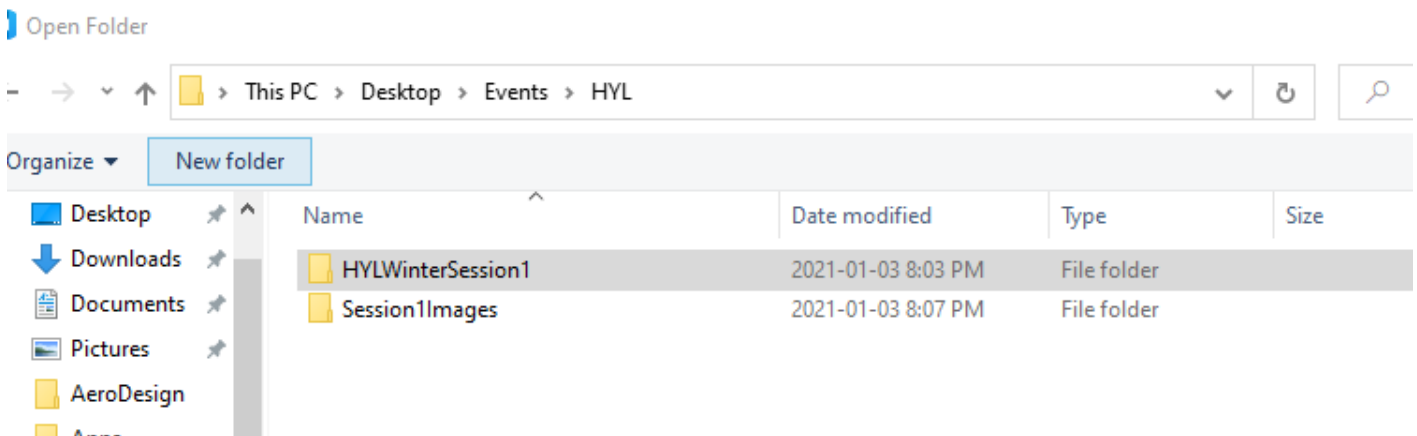
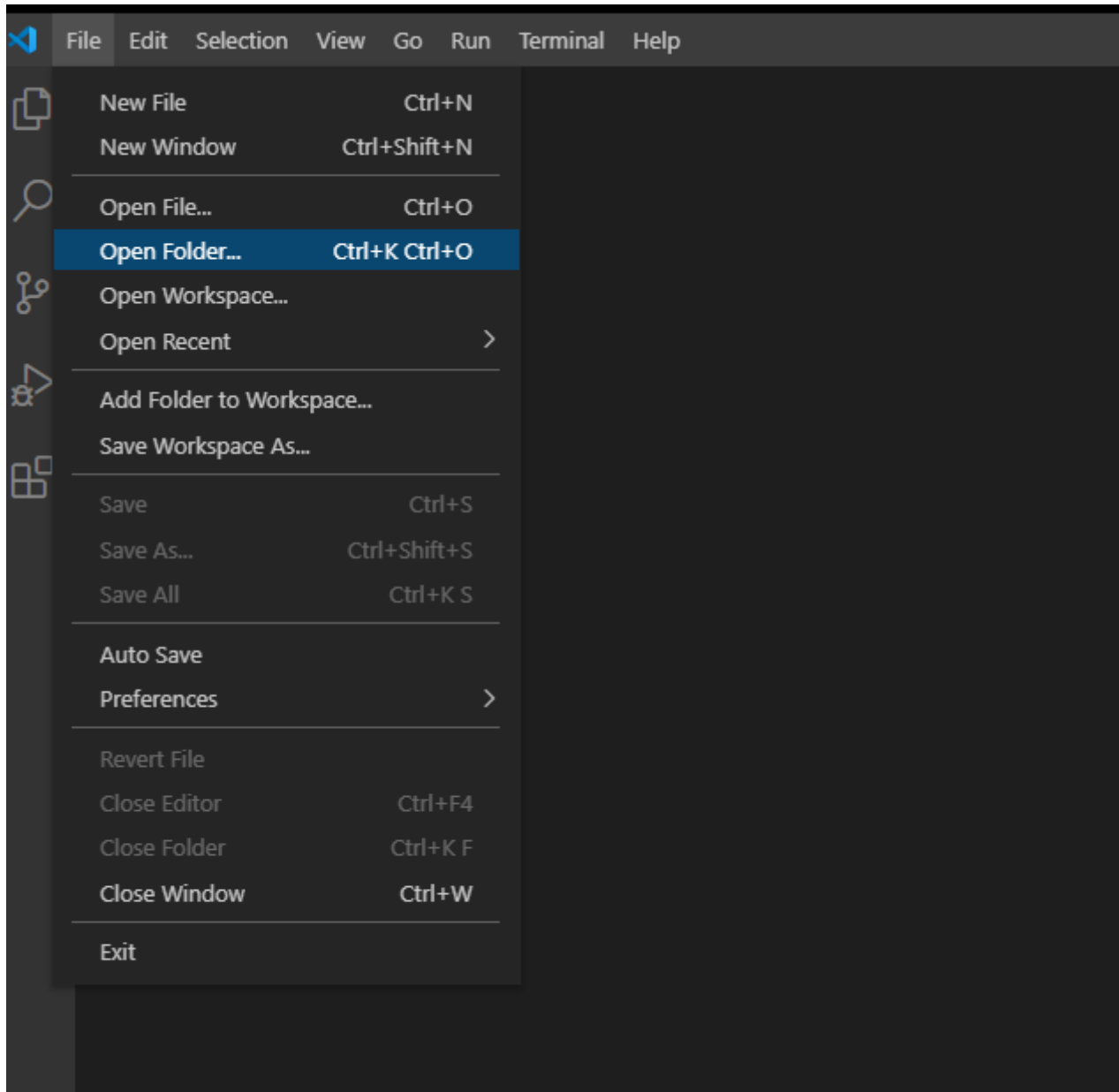


```
z1au2@LAPTOP-GBOVKETF MINGW64 ~/.ssh
$ cd ~/Desktop/Events/HYL/HYLWinterSession1/

z1au2@LAPTOP-GBOVKETF MINGW64 ~/Desktop/Events/HYL/HYLWinterSession1
$ git clone git@github.com:zach-lau/retreat1-ex1.git
Cloning into 'retreat1-ex1'...
Warning: Permanently added the RSA host key for IP address '140.82.112.4' to the list of
known hosts.
remote: Enumerating objects: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (20/20), done.
remote: Total 27 (delta 6), reused 19 (delta 5), pack-reused 0
Receiving objects: 100% (27/27), 6.54 KiB | 3.27 MiB/s, done.
Resolving deltas: 100% (6/6), done.
```

8. Open in your preferred text editor

You will now be able to open the files in your favourite 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 repository! They should *clone* the repository that you have just *forked*.

