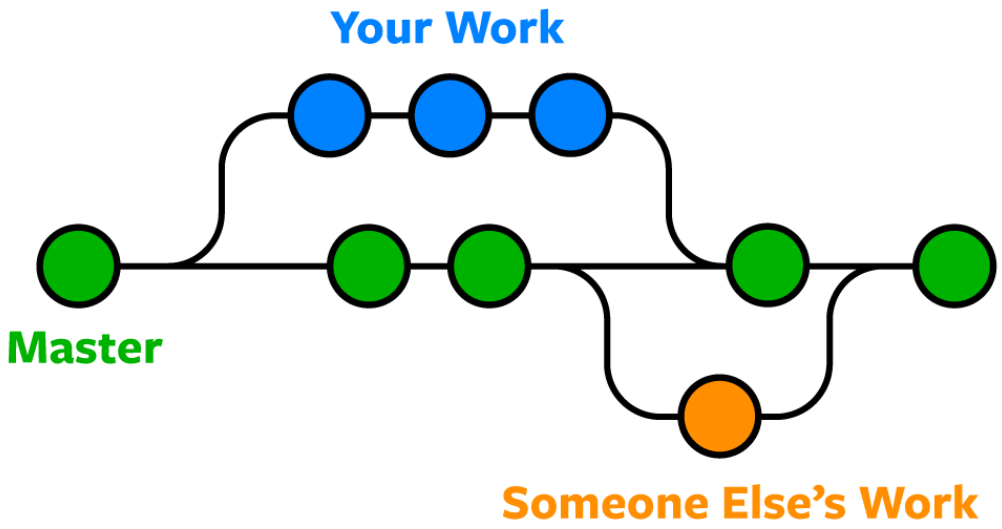
SET-UP FOR TEAM CODING

Working with Git and Terminal

How is team-based coding different?

- Usually working with a larger body of code, and so **maintaining a robust version history becomes important** (in case a new change has broken the code)

- Team members work on different sections of the code, and so **individuals' codes must be combined without anyone's work being overwritten**



How does **Git** help?

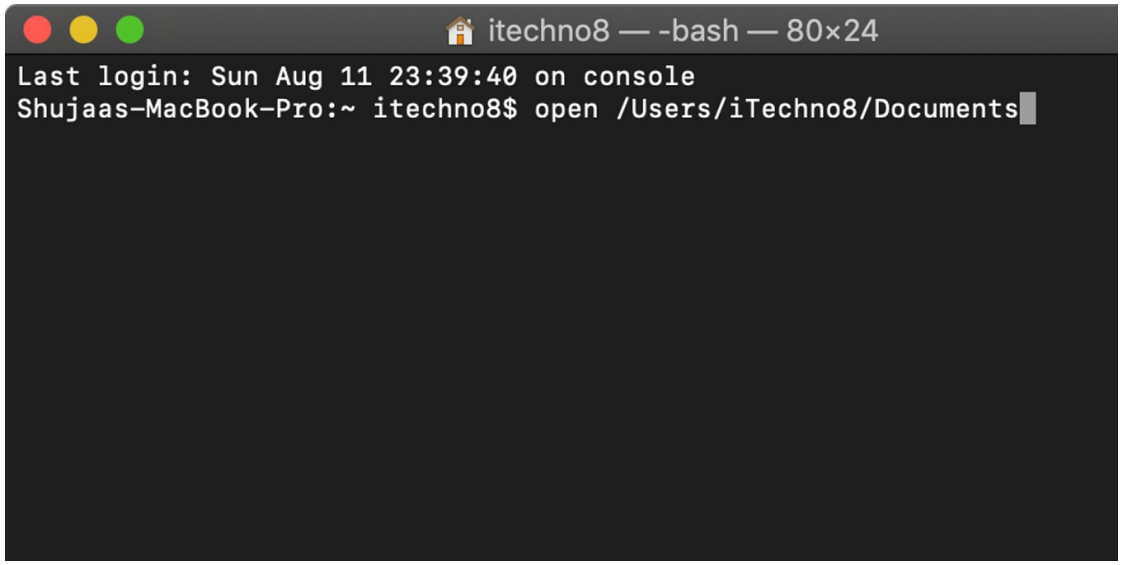
- Git is the most popular **version-control system** used among software developers

- Git **tracks changes to files in a repository** in order to create a "series of snapshots" or a history of this repository's development through time

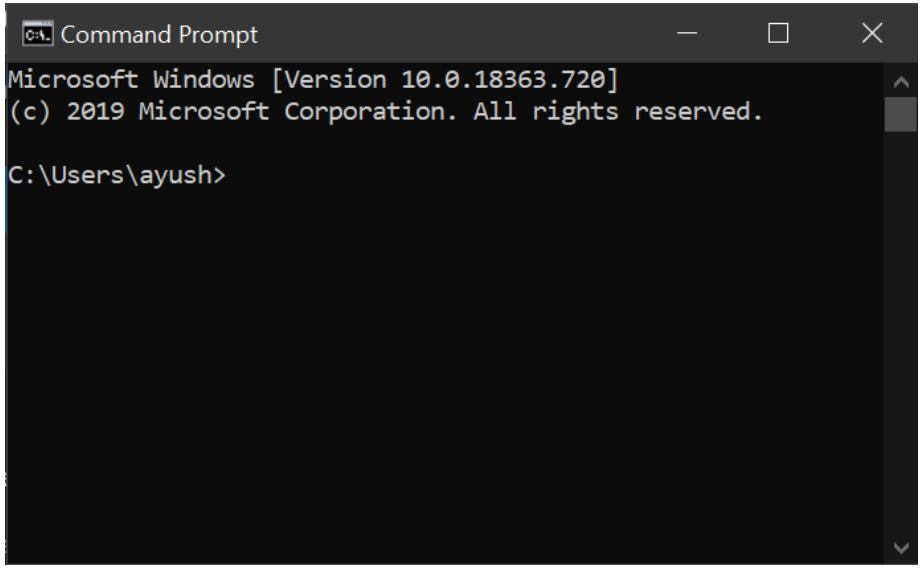
How do we use **Git**?

- Using your computer's **Terminal** or **Command Line Interface (CLI)**! The terminal takes in text-based commands to be completed by your operating system

MacOS



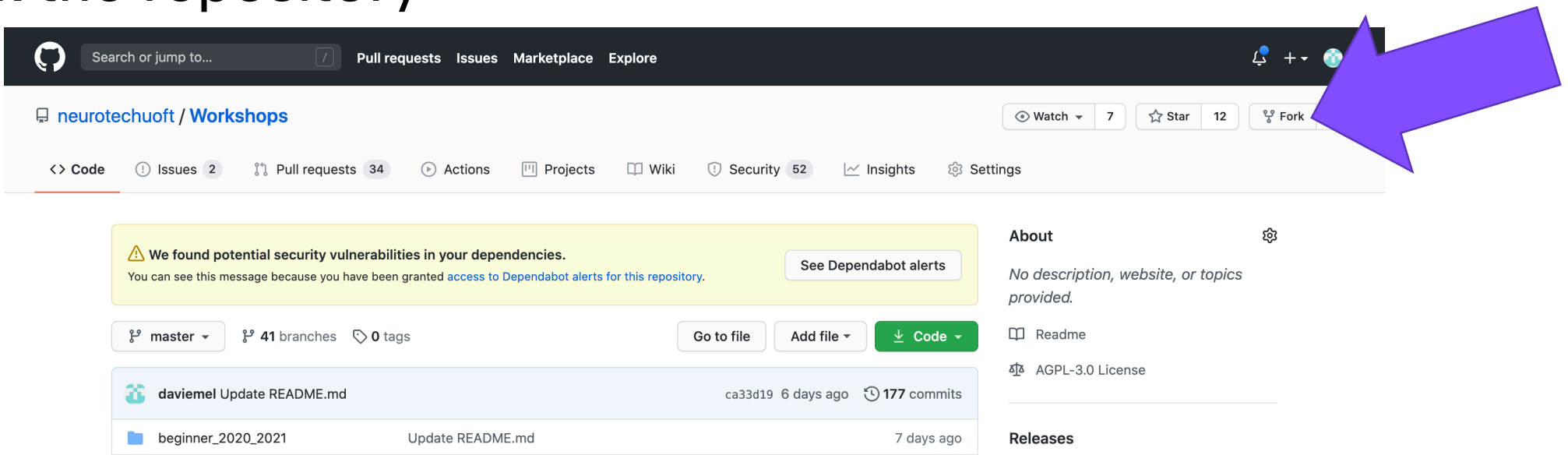
Windows



Let's try some basic **Git** commands:

- **Note:** make sure that you have downloaded and installed Git before trying these commands

1. Create a free **GitHub** account and sign in
2. Navigate to https://github.com/neurotechuoft/workshops
3. Fork the repository



Let's try some basic **Git** commands:

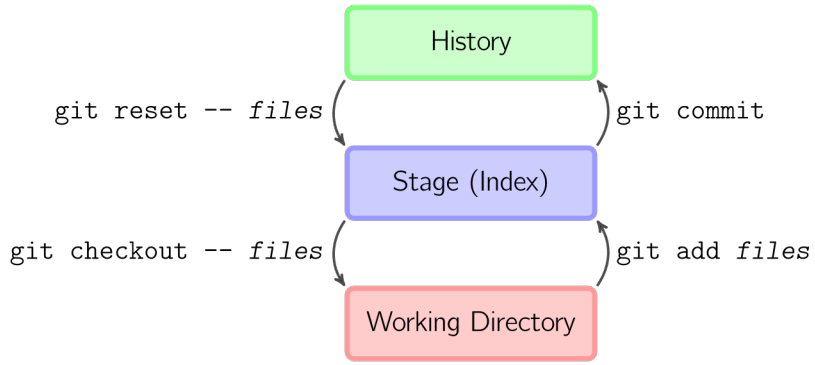
4. Navigate to "Your repositories" (click on the arrow next to your profile icon in the top right)
5. Access your newly forked repository and click the 

button—copy the link that is displayed
6. Now, go back to your terminal and navigate to the folder you'd like your working directory to appear in using **cd** (e.g. cd Desktop)
7. Type in **git clone** followed by the URL you just copied, and run the command
   - This will create a copy of the full repository on your computer that is tied to the GitHub repository it was created from (now you have your own working directory)

Let's try some basic **Git** commands:

8. Use your computer's text editing app to create a new text file named "Testing.txt" (it can be empty) and place it in the **Workshops** folder that was created (you can just drag and drop)
9. Navigate to the Workshops folder in your terminal using **cd**, and now run **git add "Testing.txt"**

- This adds your new file to your **Stage** (this is where we store files we want to "commit" or save to our working directory so that we can commit them all at once)



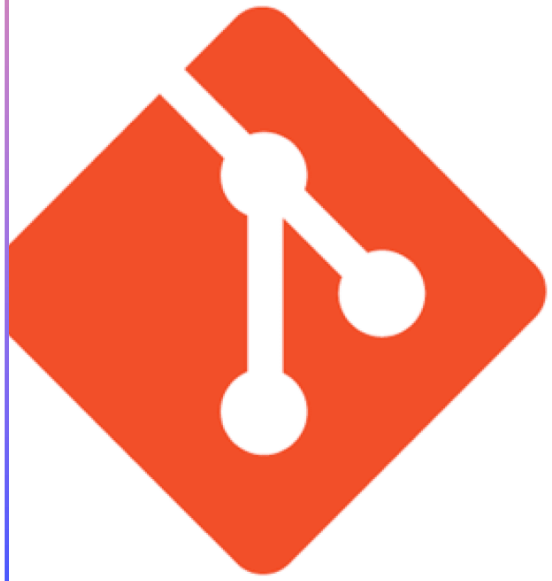
Let's try some basic **Git** commands:

Now we are ready to save the changes we've made to our local repository (i.e. the file we've added)!

10. In your terminal, run '**git commit –m "Testing git commands"**'
    - Now our new file is saved to the history of our local repository with a message that tells us why we made this change
11. Let's say we're ready to save this change to our remote directory (i.e. our forked directory on GitHub). Now, we will run '**git push**'
    - Check your GitHub repository—you should see Testing.txt there!

What else can I do with Git?

- A lot! Check out this online tutorial that lists a lot of the other Git commands that you have available to you, or check out the info for Week 3 on our GitHub



- $git config
- $git init
- $git clone <path>
- $git add <file_name>
- $git commit
- $git status
- $git remote
- $git checkout <branch_name>
- $git branch
- $git push
- $git pull
- $git merge <branch_name>
- $git diff
- $git reset
- $git revert
- $git tag
- $git log

We'll get back to Git later… let's play with some EEG data now!

- Access your web browser and go to **Google Colab**
  - We can use this tool to write and run Python code within a notebook, which is great for organization
- Select the GitHub tab and paste the URL for your forked repository, then navigate to and open **week_4_loading_graphing_data/Exercises/graphing_loading_data.ipynb**
- Once your notebook has opened, click the folder icon on the left and upload the file from **Workshops/beginner_2020_2021/week_4_loading_graphing_data/data** called "**stare_blink.csv**"
- Click the 3 dots on the right of the file once it's uploaded, Copy Path, and paste it where you see "./../data/stare_blink.csv" in the notebook