Git & GitHub 101

Basic CLI Commands

1. To list all files or folder in a folder

   `ls`

2. Make a new folder

   `mkdir folder_name`

3. Go inside a folder

   `cd folder_name`

4. To delete a whole non-empty directory/folder

   `rm directory_name -rf`

5. Write a file in Git Bash Vim

   `vim file_name`

   1. use insert key to enable the writing mode in any file
   2. then after finishing edits, press the left-right arrow key to disable the writing mode and then write `:x` to exit out

6. Copy + Paste in CLI

   1. Use the insert key to paste in CLI or highlight the statement then right click and copy that statemen and then right-click on CLI shows the options.

Basic Git Commands

1. To make a new file

   `touch names.txt`

2. To check if git is installed in your PC

   `git`

3. To initialize an empty Git repository in your folder

   `git init`

4. To view the changes or the untracked files in the project that's not been saved yet

   `git status`

5. Staging the files

   `git add file_name` or `git add .` (to stage everything in the current folder)

6. Committing the files

Working with Existing Projects on GitHub

Use Git Bash for Windows.

You can't directly change the contents of a repo unless you have access to it. To solve this, you create a copy (fork) of this project in your own account. In our own copy, we can do anything we want with it. After forking, we:

1. Cloning the forked project to local machine

   `git clone forked_repo_url`

2. The public repo that we forked out local copy from is known as the upstream url. We can save it as

   `git remote add upstream insert_upstream_url`

3. Creating a new branch

   `git branch branch_name`

4. Then shift the head to the above branch using the checkout command
5. Then stage. Then commit.
6. Then push. We can't push to upstream (no access). Can push to our forked repo though (origin)

   `git push origin your_branch_name`

7. Always make different branches for different pull requests if you're working on different features. 1 branch = 1 pull request (never commit on main (2))
8. To remove a commit

   1. we can remove a commit with the reset command Now it's unstaged.
   2. then add. to stage the remaining files
   3. then we can use the stash command to stash it elsewhere
   4. then, we'll have to force push this branch since the online repo contains a commit which the local repo does not

   `git push origin your_branch_name -f`

9. To make forked project even (updated) with the main project