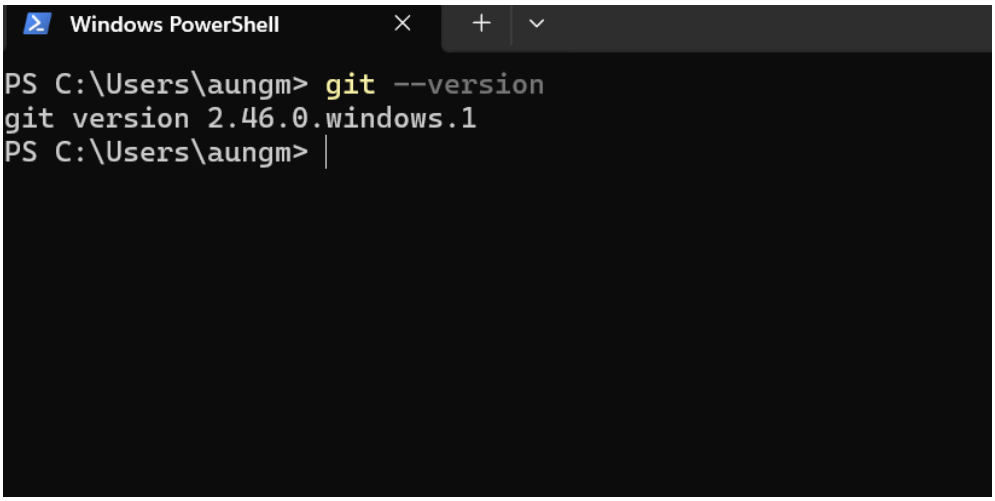
# BIGDOSE GitHub Guides by using Command Line

## Checking your device installed git?

Git and GitHub are different topics and if you want to use github in your terminal window (cmd), simply open a terminal window and type “git –version”. After that you will see that




## What is a pull request?

A pull request (often referred to as “PR”) is a proposal to merge a set of changes from one branch into another. By creating a pull request, you can review a set of changes with others before they are incorporated into the main code base. Before getting to that, it’s helpful to define a couple of terms.

- **Source branch:** the branch containing your changes.
- **Target branch:** the branch you are trying to merge your changes into.

Pull requests provide a visual representation of the differences in the content between the source branch and the target branch. This is what enables you to review the changes before accepting them and pulling them into the target branch.

## Creating your pull request

Step1: you need to clone and fork the project on your local device (personal computer) on your window terminal or IDE terminal. Then forked the project on the github.

git clone https://github.com/..... (…… means the project repo url)

Step2: On your terminal (wherever it is IDE terminal or window terminal you must be in the right path to the file, see the image below.

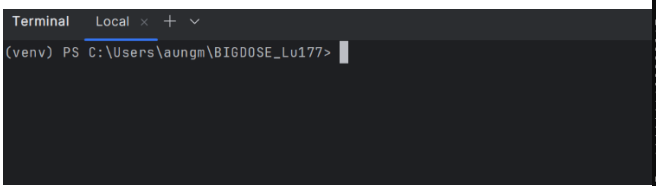



Fig: Pycharm IDE terminal

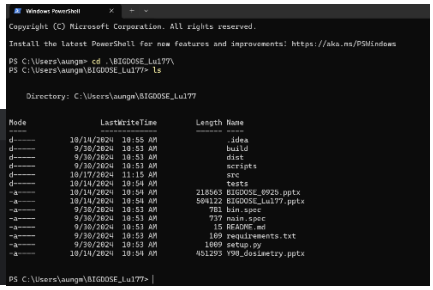



Fig:Window terminal

You can see C:\Users\aungm\BIGDOSE_Lu177 which is my file path.

If you are setting up the git for the first time, we can configure the git with name & email.

git config --global user.name “Your_Name” (eg- git config user.name “bob88”

git config –global user.email “Your github email”

Please don’t forget you need to forked the project you wanna make contribution. The forked repo will be appeared on you github repo.

Step3: Initialize git repo

When you are in the right path of your code file you should start with

git init

Step4: Committing files into the git repo

git add . (. means adding all the files you’ve made changes)

you can also use

git add <file name> (for single file)

To check if files are added or not use this

git status

you may see the added file and your current branch.

```
venv) PS C:\Users\aungm\BIGDOSE_Lu177> git init
einitialized existing Git repository in C:/Users/aungm/BIGDOSE_Lu177/.git/
venv) PS C:\Users\aungm\BIGDOSE_Lu177> git add .
venv) PS C:\Users\aungm\BIGDOSE_Lu177> git status
n branch main
our branch is up to date with 'origin/main'.

hanges to be committed:
 (use "git restore --staged <file>..." to unstage)
       modified:   src/main.py

venv) PS C:\Users\aungm\BIGDOSE_Lu177>
```

Step5: Push the file into a remote repository.

git commit -m “first commit” (instead of first commit, you can change the commit message whatever you like for instance- add a new change)

now we’re going to push our changes to our forked repo.

git branch -M main (change your branch )

```
(venv) PS C:\Users\aungm\BIGDOSE_Lu177> git commit -m "latest version"
[main 9538949] latest version
 1 file changed, 1 insertion(+)
(venv) PS C:\Users\aungm\BIGDOSE_Lu177> git branch
* main
(venv) PS C:\Users\aungm\BIGDOSE_Lu177>
```

In the above figure: you can see that I am already on the main branch of my forked repo.

git push -u origin main

```
(venv) PS C:\Users\aungm\BIGDOSE_Lu177> git push origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 16 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 522 bytes | 522.00 KiB/s, done.
Total 5 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 3 local objects.
To https://github.com/aungminnkhant9400/BIGDOSE_Lu177
   07f9fd5..40d4aa0  main -> main
(venv) PS C:\Users\aungm\BIGDOSE_Lu177>
```

If you see the response like this, you've successfully push your changes to your forked repo.

Step5: After pushing everything go to your forked repo and check sync fork first if there's any branch to update. Then, click contribute and open pull request. Now you finally make contributions to the project. For the merging, please don't merge if it is showing any conflict. It is better to ask other contributors about that. Every project has only one or two person who did the merging process.


Fig: My forked repo

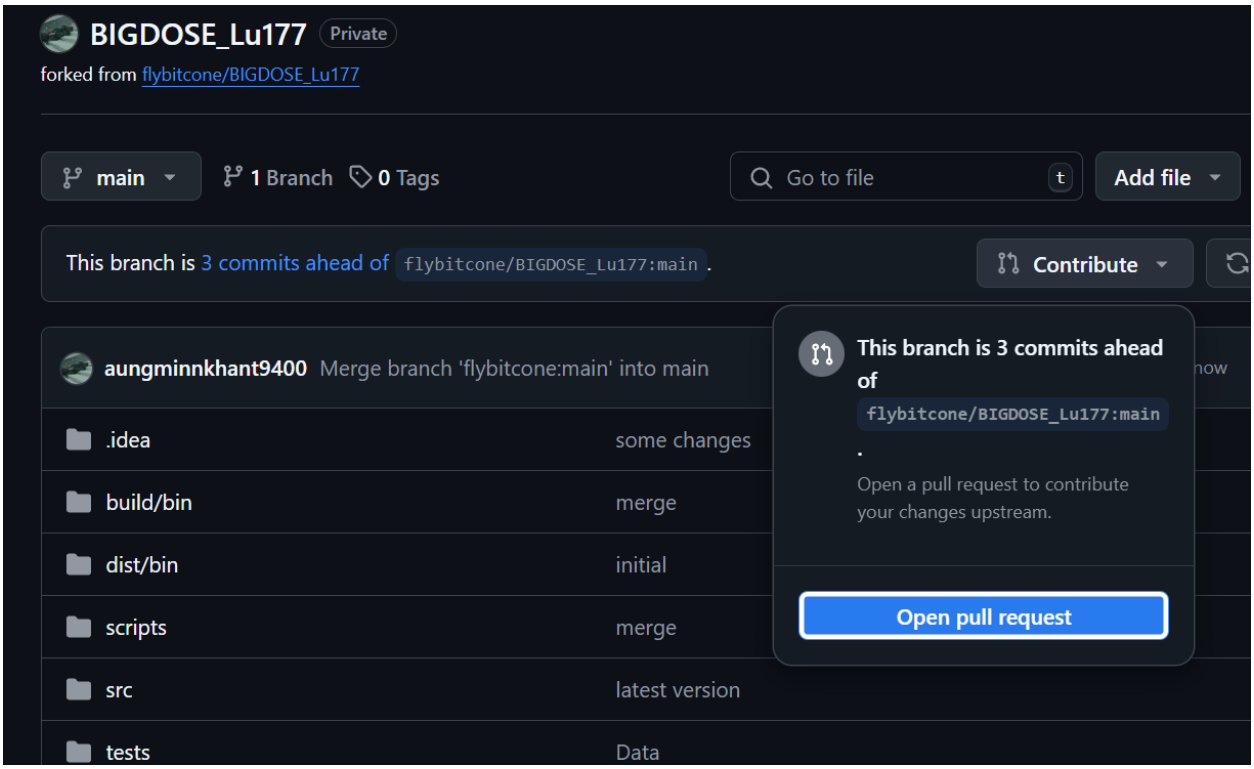



Fig: Open pull request

Step6: Everyday before you make changes, please use this command line

```
git pull
```

to pull every changes that've been made by other contributors and you can see the changes. Also don't forget to sync fork.