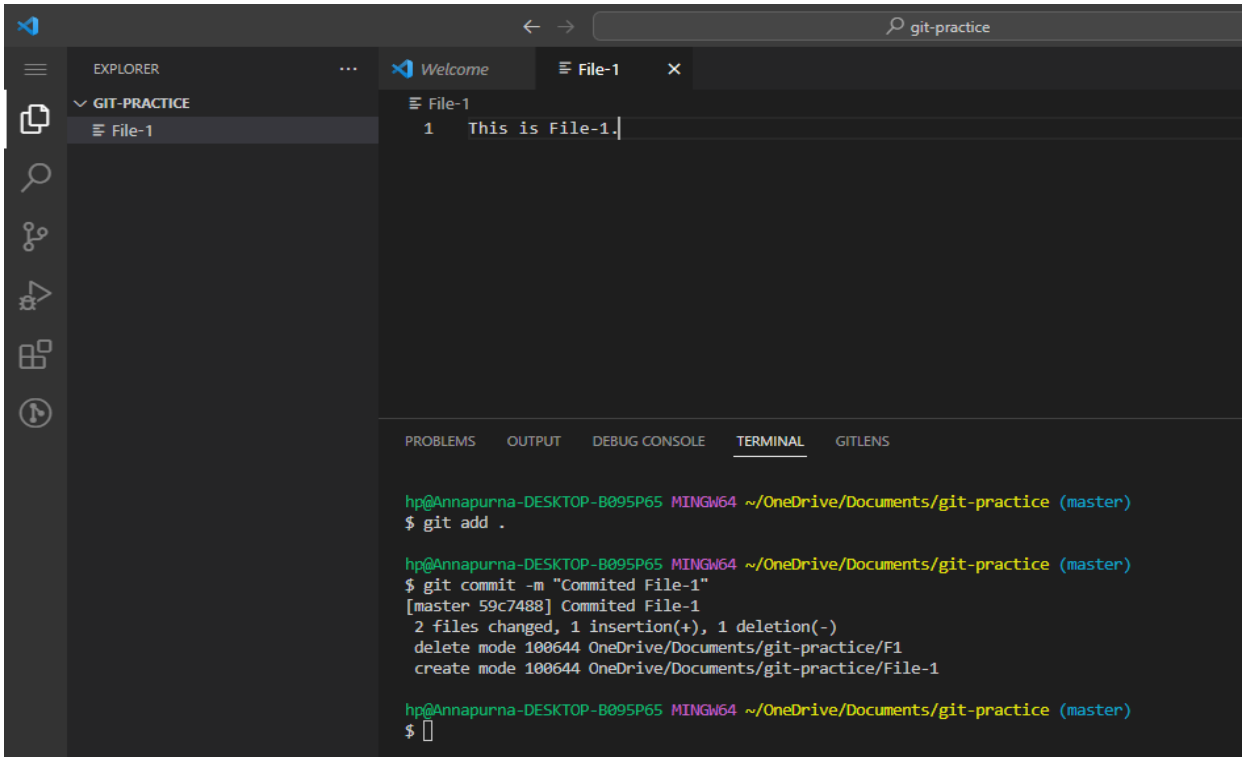
Q1. Describe the usage of the git stash command by using an example and also state the process by giving the screenshot of all the commands written in git bash.

git stash

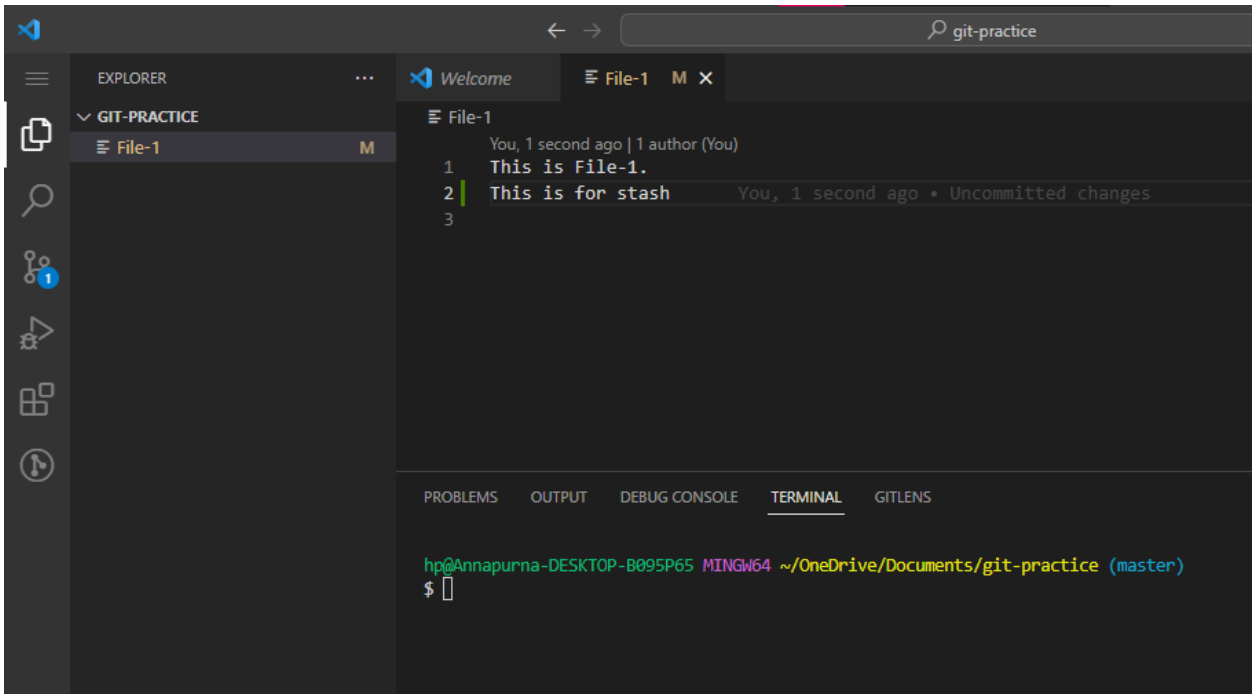
Git stash saves the uncommitted changes locally, allowing us to make changes, switch branches and can do all other git operations. We can reapply the stashed changes when we need them.

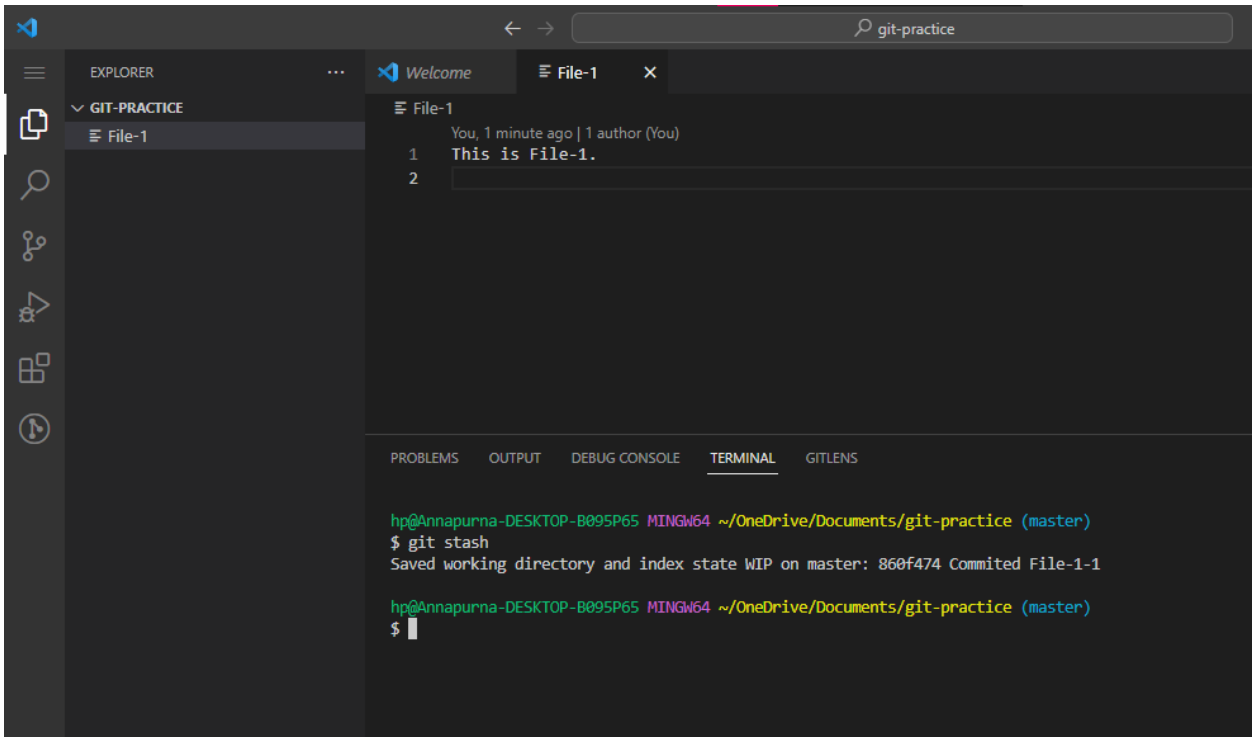
For example, if me and my friend are working with two branches A and B which are merged together. If suppose currently I am working in branch A with some file and I haven't committed the changes yet and my friend got a bug in the branch B and ask me to fix it so, when I switch to branch B then it will show me an error that "Your local changes to the following files would be overwritten by checkout … Please commit your changes or stash them before you switch branches.". So, to overcome from this problem we use stash it stores the changes and we can retrieve them whenever we want.

- First I have created a file named File-1 and added some content in it. I have added it and committed it.



Now I have added another line in the File-1and I have saved it.





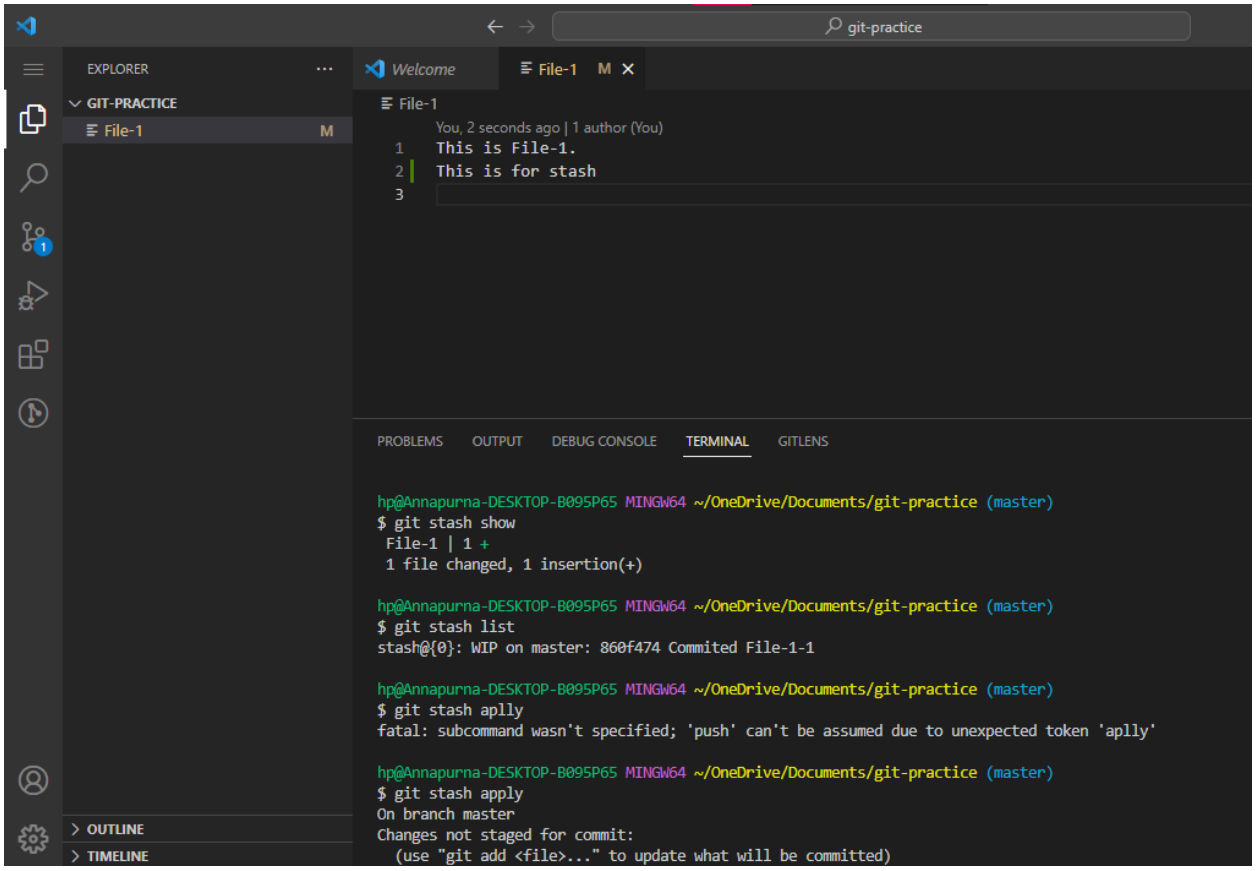
In the above screenshot what happened is I have run the **git stash** command so the line2 have been disappeared.

I have performed some commands related to git hub

git show – It shows the count of the changes that are made in the files.

git stash list – This command shows the details of the stashed changes.

git stash apply – It will show the changes which are stashed.



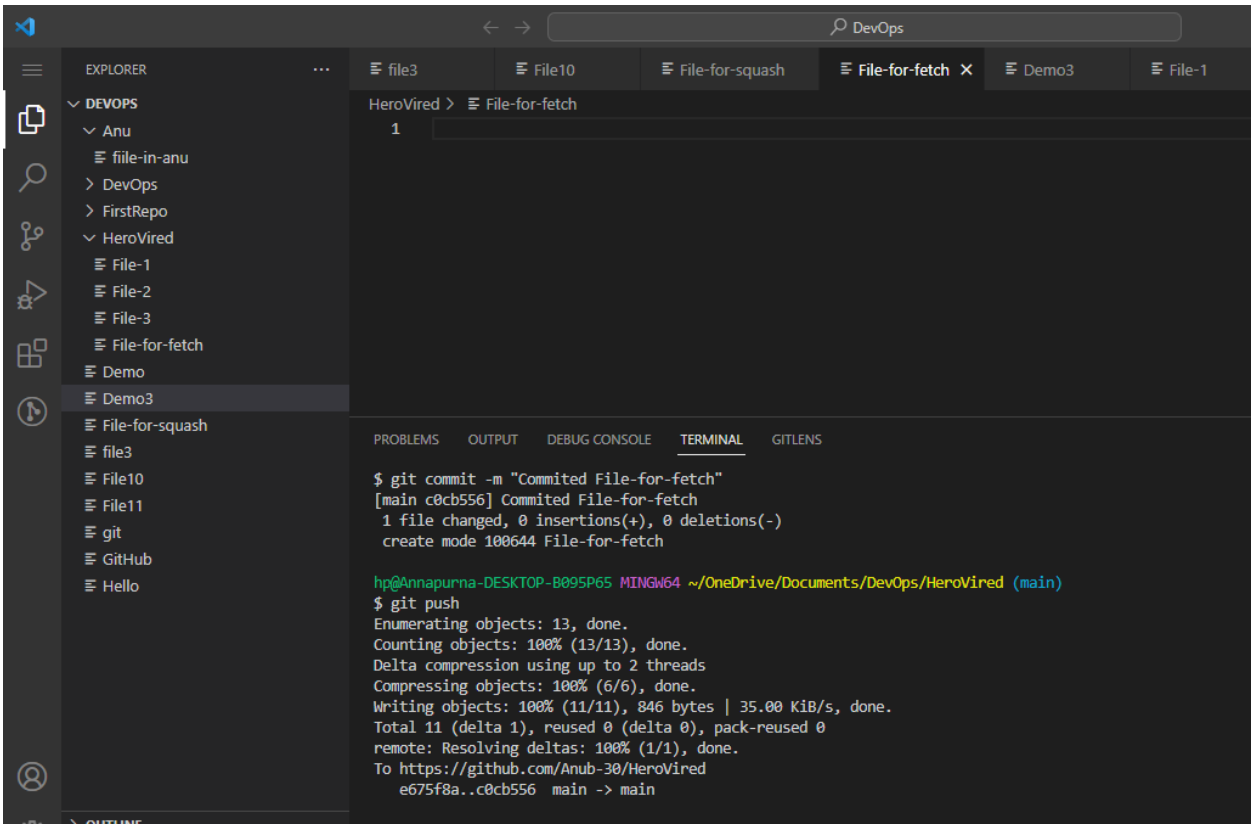
In the above screenshot when we run the command git stash apply it showed the change(line2) which was stashed.

Q2. By using a sample example of your choice, use the git fetch command and also use the git merge command and describe the whole process through a screenshot with all the commands and their output in git bash.

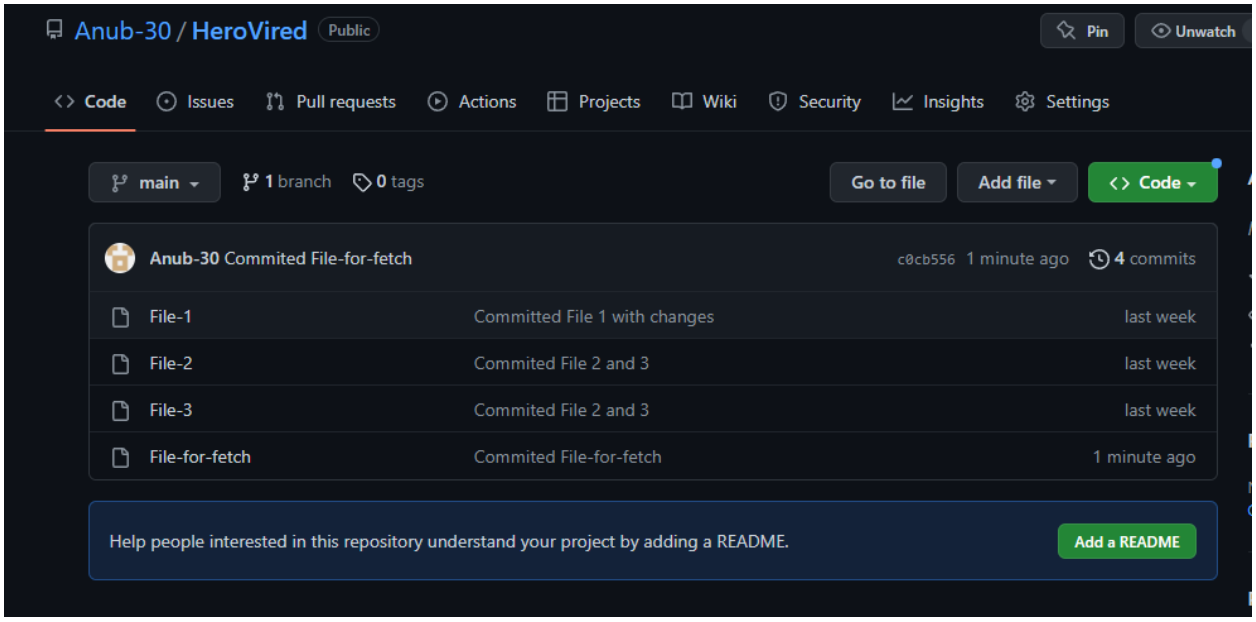
git fetch

git fetch is a command used to download contents from a remote repository.

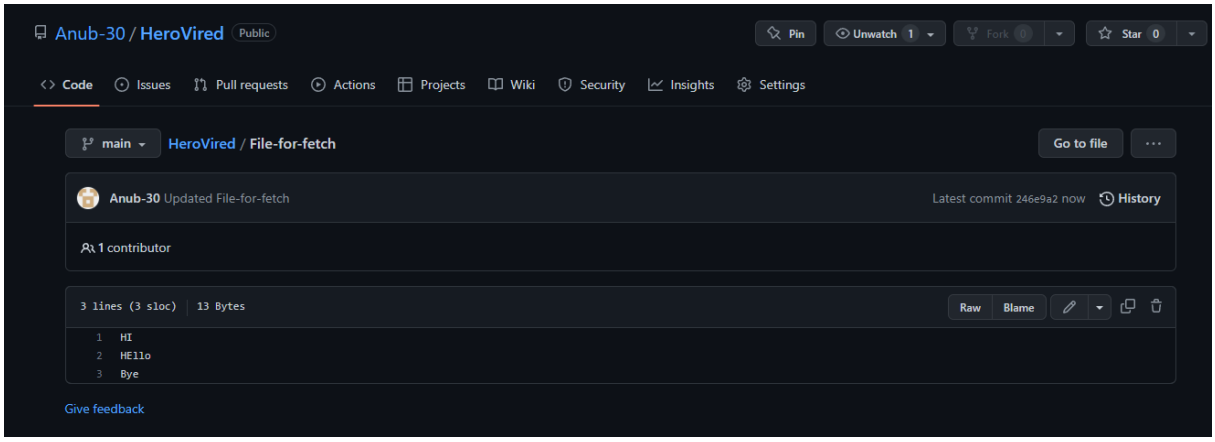
I have a repository in my GitHub called HeroVired. So I have accesed it through my local machine and I have added a file named File-for-fetch and I have pushed it to the repository with the help of **git push** command.



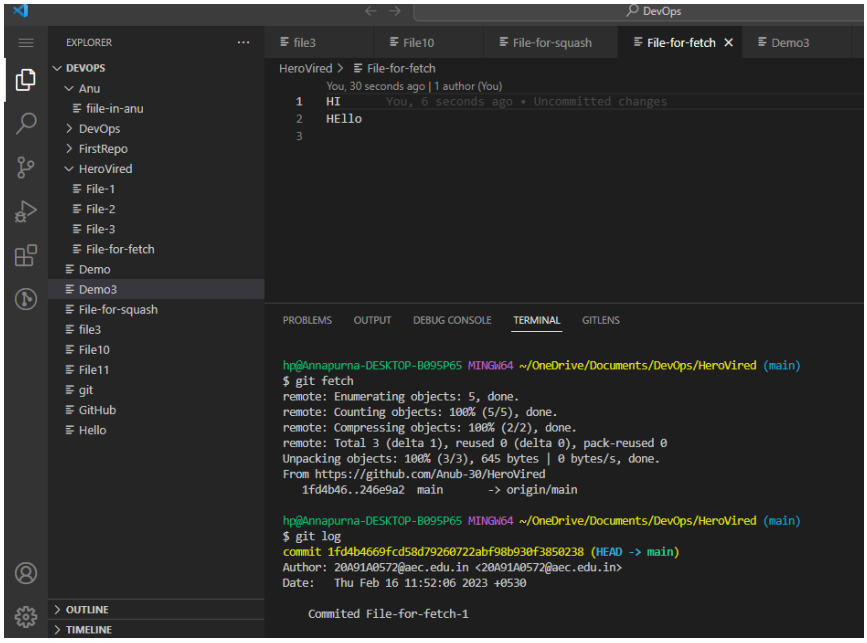
Now I have logged in to the my GitHub account and can able to see the file File-for-fetch as I have pushed it into the repository.



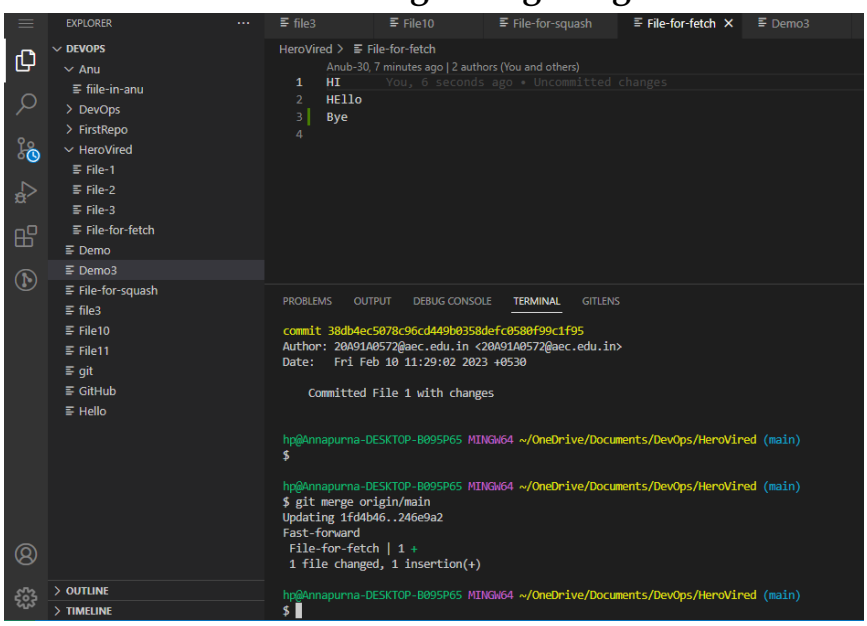
Now I have opened the file and added the line 3 through the edit option and I have committed it in the GitHub.



Now I have performed the **git fetch** command.



To merge the changes that are made in the remote repository to the local repository we have to use the command **git merge origin/<branch-name>.**



So, now we can see the line3 which we added into the repository in the github.

Q3. State the difference between git fetch and git pull by doing a practical example in your git bash and attach a screenshot of all the processes.

git fetch

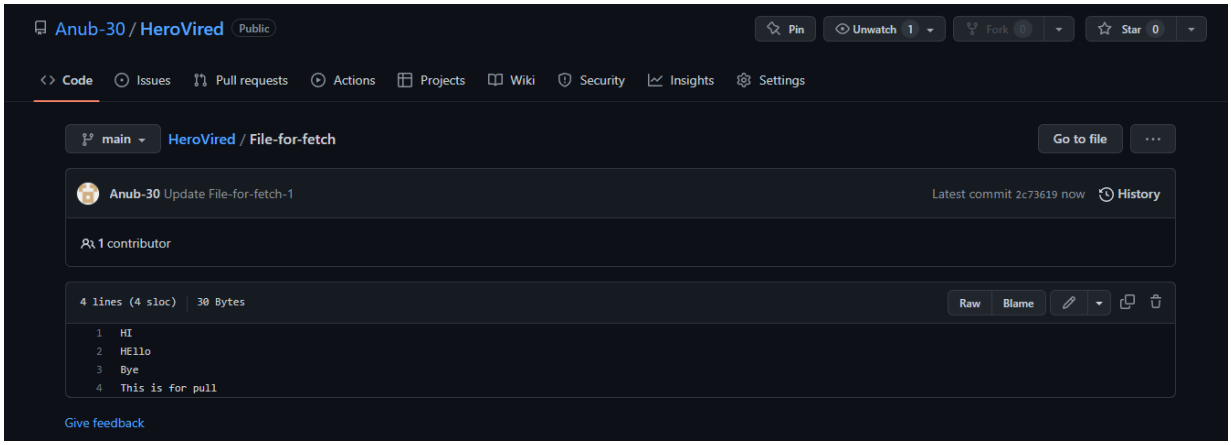
Git Fetch is the command that tells the local repository that there are changes available in the remote repository without bringing the changes into the local repository.

git pull

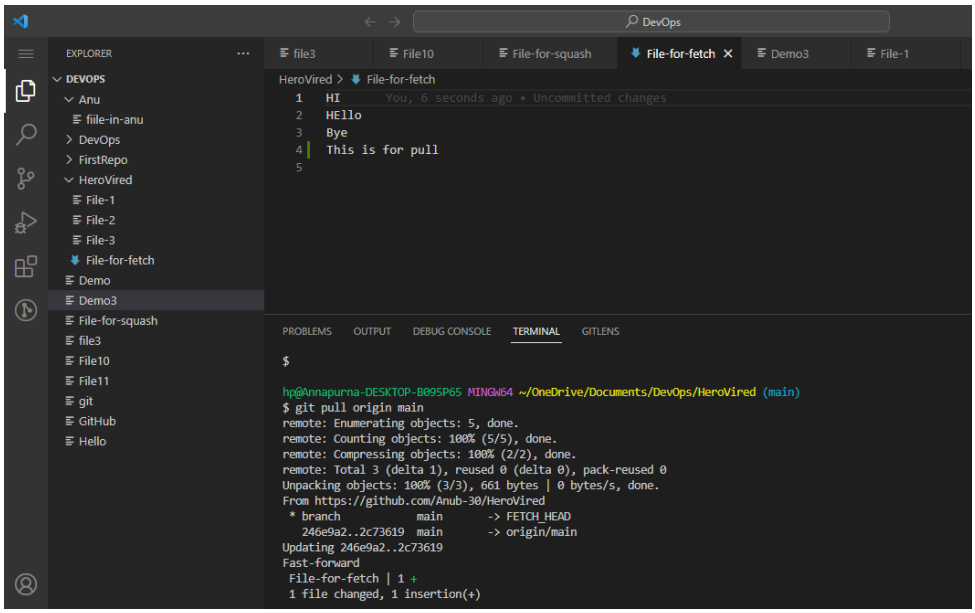
Git Pull on the other hand brings the copy of the remote directory changes into the local repository.

As I have seen the git fetch command in the above question it used to download the changes from the remote repository.

Now for the git pull first I have added line4 in the file File-for-fetch in the github website and committed the changes.



Now if we run the **git pull origin <branch-name>** command it do the operation of the **git fetch** and the **git merge.**



So, here after running **the git pull origin main** command it shows the changes that we made in the File-for-fetch file which we committed in the github website.

Q4. Try to find out about the awk command and use it while reading a file created by yourself. Also, make a bash script file and try to find out the prime number from the range 1 to 20.
The whole process should be carried out and by using the history command, give the screenshot of all the processes being carried out.

<u>AWK COMMAND:</u>

AWK is used for Pattern Scanning and Processing. It is used for Reading the Files. We can specify the patterns and fetch the data from the file. We can also count the number of input records and fields in the File.

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    GITLENS

hp@AnnapurnaDESKTOP-B095P65 MINGW64 ~/OneDrive/Documents/git-practice (master)
$ cat demoawk
EmployeeId      Place           Role
123             Kakinada        Developer
213             Rajahmundry     Tester
513             Kakinada        Tester
672             Hyderabad       Developer

hp@AnnapurnaDESKTOP-B095P65 MINGW64 ~/OneDrive/Documents/git-practice (master)
$
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    GITLENS

513            Kakinada        Tester
672            Hyderabad       Developer

hp@AnnapurnaDESKTOP-B095P65 MINGW64 ~/OneDrive/Documents/git-practice (master)
$ awk '{print}' demoawk
EmployeeId      Place           Role
123            Kakinada        Developer
213            Rajahmundry     Tester
513            Kakinada        Tester
672            Hyderabad       Developer

hp@AnnapurnaDESKTOP-B095P65 MINGW64 ~/OneDrive/Documents/git-practice (master)
$
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    GITLENS

hp@AnnapurnaDESKTOP-B095P65 MINGW64 ~/OneDrive/Documents/git-practice (master)
$ awk '/Developer/ {print}' demoawk
123            Kakinada        Developer
672            Hyderabad       Developer

hp@AnnapurnaDESKTOP-B095P65 MINGW64 ~/OneDrive/Documents/git-practice (master)
$
```

Program of finding prime numbers between 1 to 20 numbers using bash script.

```
EXPLORER
GIT-PRACTICE
File-1
$ primenumber.sh U

Welcome    $ primenumber.sh U

$ primenumber.sh
#!/bin/bash
for((i=2;i<20;))
do
    for((j=i-1;j>=2;))
    do
        if [ `expr $i % $j` -ne 0 ] ; then
            prime=1
        else
            prime=0
            break
        fi
        j=`expr $j - 1`
    done
    if [ $prime == 1 ] ; then
        echo $i
    fi
    i=`expr $i + 1`
done

PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    GITLENS

hp@AnnapurnaDESKTOP-B095P65 MINGW64 ~/OneDrive/Documents/git-practice (master)
$ bash primenumber.sh
primenumber.sh: line 14: [: ==: unary operator expected
3
5
7
11
13
17
19

hp@AnnapurnaDESKTOP-B095P65 MINGW64 ~/OneDrive/Documents/git-practice (master)
$

OUTLINE
TIMELINE
```

Q5. Set up a container and run a Ubuntu operating system. For this purpose, you can make use of the docker hub and run the container in interactive mode. All the processes pertaining to this should be provided in a screenshot for grading.

```
C:\Users\hp>docker run -it ubuntu
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
677076032cca: Pull complete
Digest: sha256:9a0bdde4188b896a372804be2384015e90e3f84906b750c1a53539b585fbbe7f
Status: Downloaded newer image for ubuntu:latest
root@f2225e131dab:/# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@f2225e131dab:/#
```