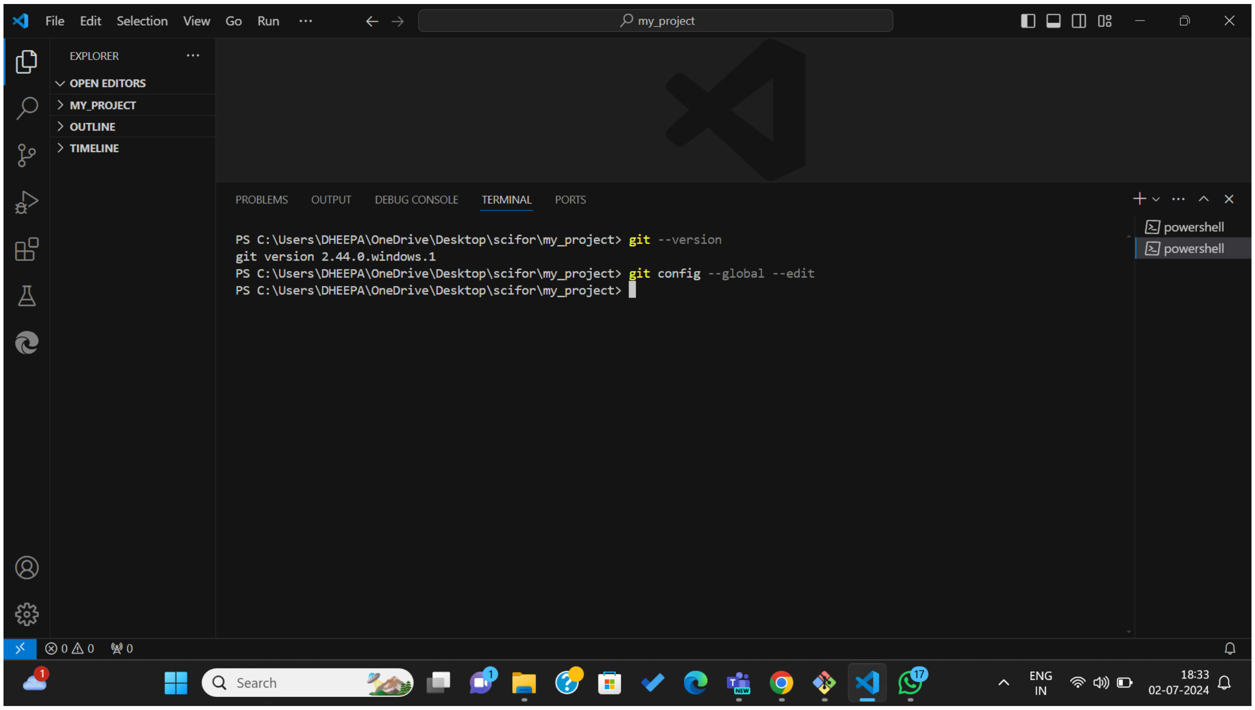
Git vs Github

Git	Github
It is a software	It is a service
It is installed locally on the system	It is hosted on web
It is a command line tool	It is a graphical interface
It is a version control tool	Git repository hosting tool
No user management	Built in user management

Git commands

1. git config -The git config command is used initially to configure the user.name and user.email. This specifies what email id and username will be used from a local repository.

   When git config is used with –global flag, it writes the settings to all repositories on the computer.



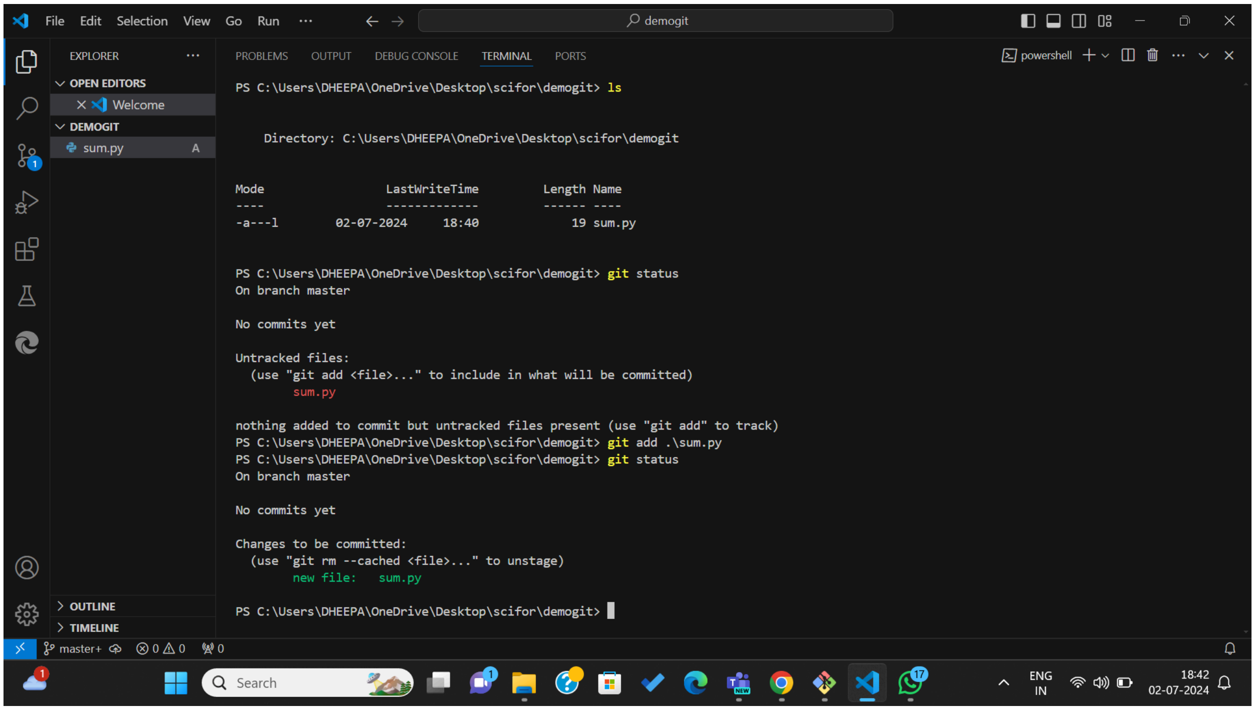
```
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\my_project> git --version
git version 2.44.0.windows.1
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\my_project> git config --global --edit
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\my_project>
```

2. git init- Initialize a local Git repository

   After the git init command is used, a .git folder is created in the directory with some subdirectories. Once the repository is initialized, the process of creating other files begins.

3. git status- The git status command tells the current state of the repository.

   The command provides the current working branch. If the files are in the staging area, but not committed, it will be shown by the git status. Also, if there are no changes, it will show the message no changes to commit, working directory clean.



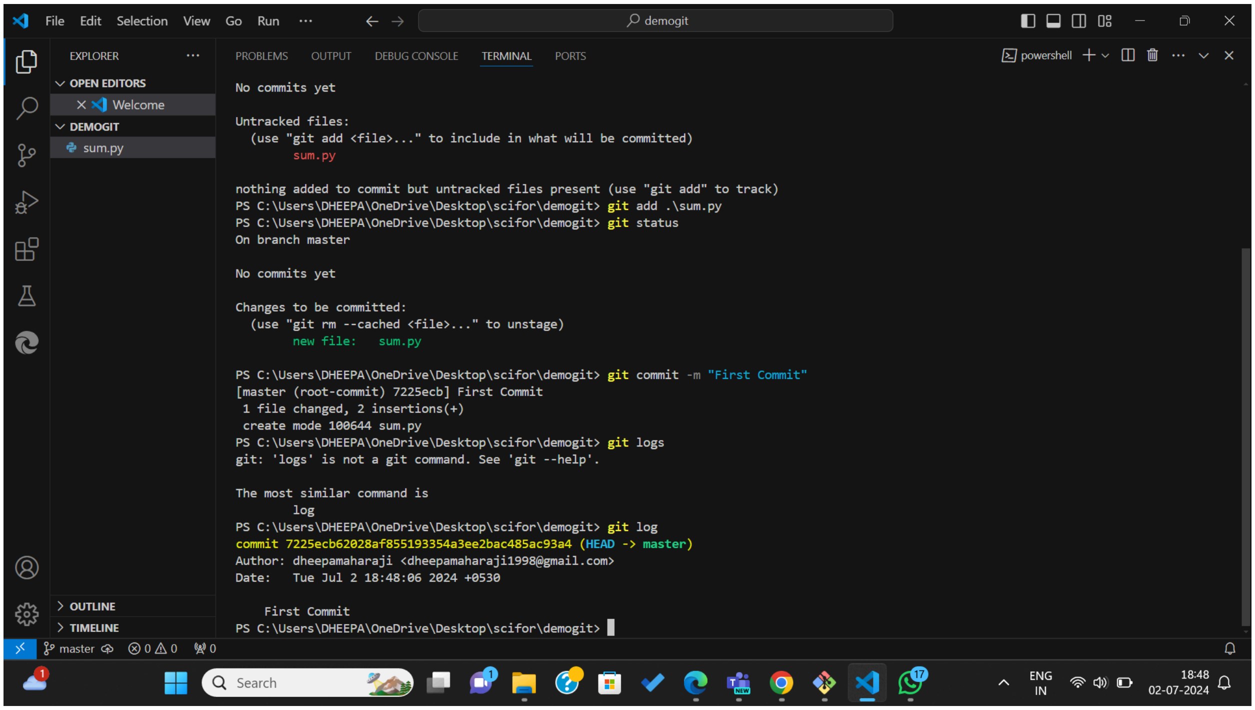
```
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> ls


    Directory: C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a---l        02-07-2024     18:40             19 sum.py


PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        sum.py

nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git add .\sum.py
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   sum.py

PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> 
```

4. git add - Add command is used after checking the status of the files, to add those files to the staging area.

   Before running the commit command, "git add" is used to add any new or modified files.

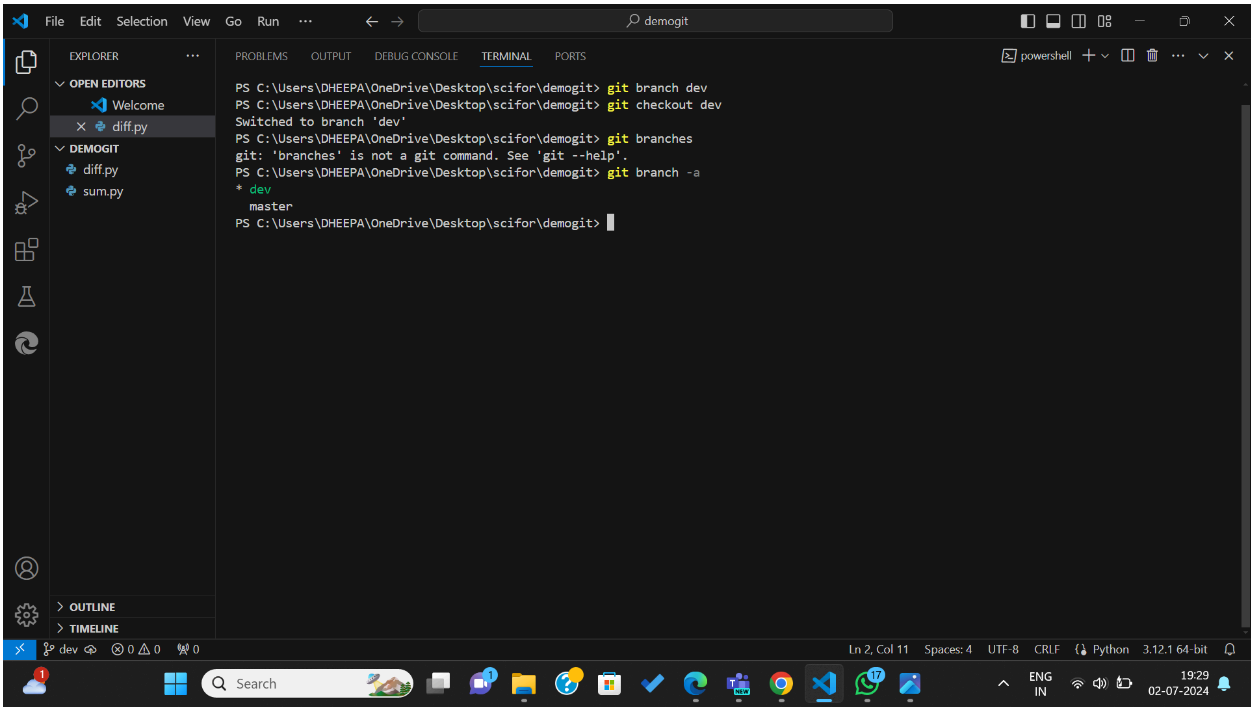
5. git commit - The commit command makes sure that the changes are saved to the local repository.

   The command "git commit –m <message>" allows you to describe everyone and help them understand what has happened.



```
No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        sum.py

nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git add .\sum.py
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   sum.py

PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git commit -m "First Commit"
[master (root-commit) 7225ecb] First Commit
 1 file changed, 2 insertions(+)
 create mode 100644 sum.py
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git logs
git: 'logs' is not a git command. See 'git --help'.

The most similar command is
        log
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git log
commit 7225ecb62028af855193354a3ee2bac485ac93a4 (HEAD -> master)
Author: dheepamaharaji <dheepamaharaji1998@gmail.com>
Date:   Tue Jul 2 18:48:06 2024 +0530

    First Commit
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit>
```

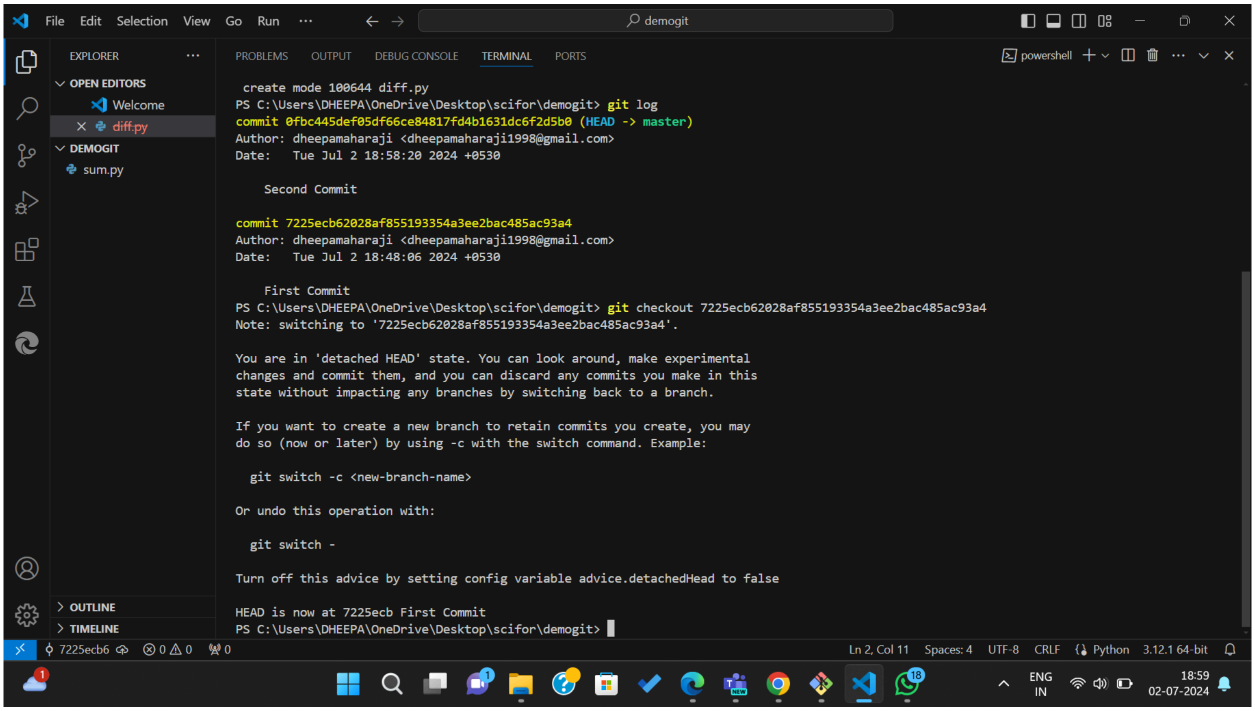
6. git branch - The git branch command is used to determine what branch the local repository is on.
   The command enables adding and deleting a branch.



```
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git branch dev
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git checkout dev
Switched to branch 'dev'
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git branches
git: 'branches' is not a git command. See 'git --help'.
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git branch -a
* dev
  master
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit>
```

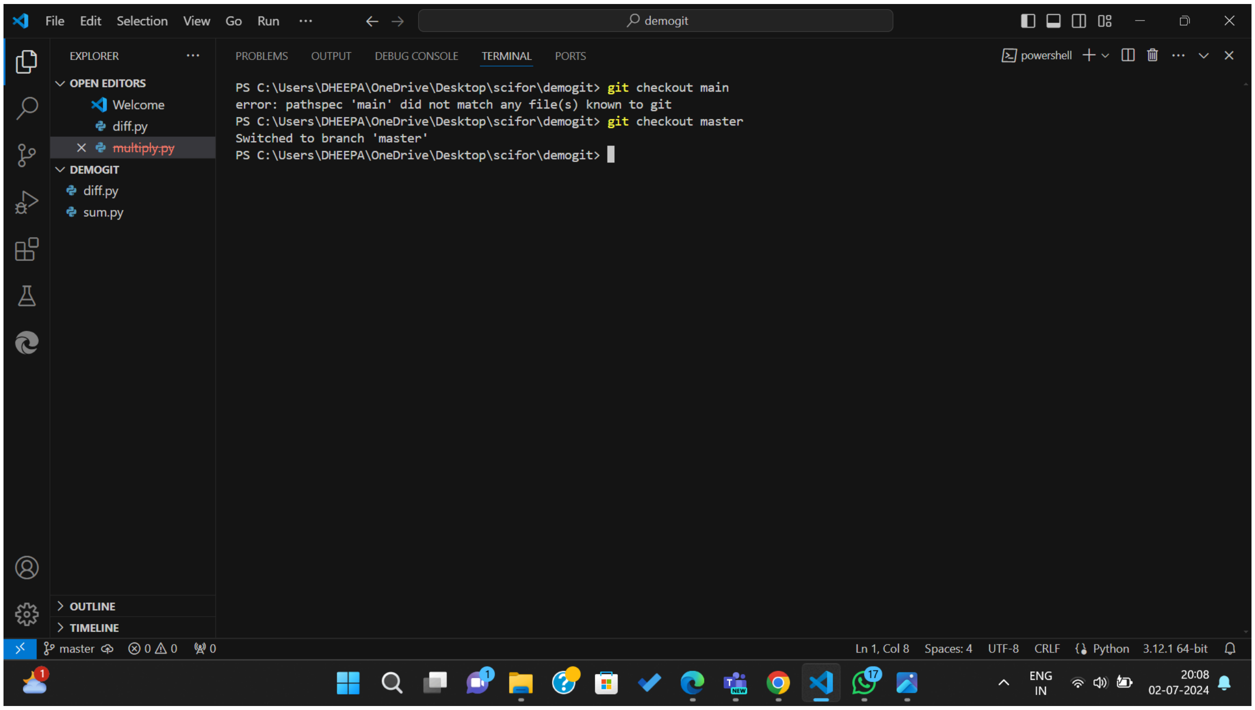
7. git checkout - The git checkout command is used to switch branches, whenever the work is to be started on a different branch.

   The command works on three separate entities: files, commits, and branches.

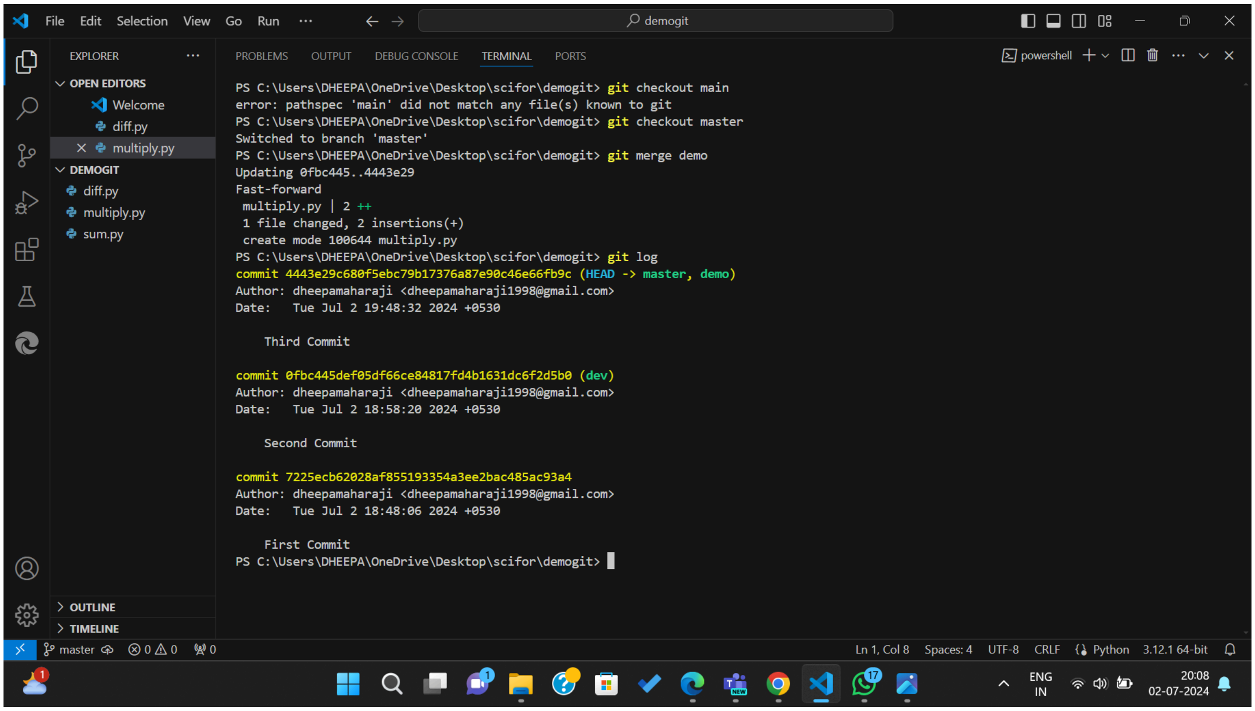


```
 create mode 100644 diff.py
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git log
commit 0fbc445def05df66ce84817fd4b1631dc6f2d5b0 (HEAD -> master)
Author: dheepamaharaji <dheepamaharaji1998@gmail.com>
Date:   Tue Jul 2 18:58:20 2024 +0530

    Second Commit

commit 7225ecb62028af855193354a3ee2bac485ac93a4
Author: dheepamaharaji <dheepamaharaji1998@gmail.com>
Date:   Tue Jul 2 18:48:06 2024 +0530

    First Commit
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git checkout 7225ecb62028af855193354a3ee2bac485ac93a4
Note: switching to '7225ecb62028af855193354a3ee2bac485ac93a4'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

  git switch -c <new-branch-name>

Or undo this operation with:

  git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 7225ecb First Commit
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit>
```

8. git merge - The git merge command is used to integrate the branches together. The command combines the changes from one branch to another branch. It is used to merge the changes in the staging branch to the stable branch.



```
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git checkout main
error: pathspec 'main' did not match any file(s) known to git
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git checkout master
Switched to branch 'master'
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> 
```



```
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git checkout main
error: pathspec 'main' did not match any file(s) known to git
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git checkout master
Switched to branch 'master'
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git merge demo
Updating 0fbc445..4443e29
Fast-forward
 multiply.py | 2 ++
 1 file changed, 2 insertions(+)
 create mode 100644 multiply.py
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git log
commit 4443e29c680f5ebc79b17376a87e90c46e66fb9c (HEAD -> master, demo)
Author: dheepamaharaji <dheepamaharaji1998@gmail.com>
Date:   Tue Jul 2 19:48:32 2024 +0530

    Third Commit

commit 0fbc445def05df66ce84817fd4b1631dc6f2d5b0 (dev)
Author: dheepamaharaji <dheepamaharaji1998@gmail.com>
Date:   Tue Jul 2 18:58:20 2024 +0530

    Second Commit

commit 7225ecb62028af855193354a3ee2bac485ac93a4
Author: dheepamaharaji <dheepamaharaji1998@gmail.com>
Date:   Tue Jul 2 18:48:06 2024 +0530

    First Commit
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> 
```

9. git remote - The git remote command is used to create, view, and delete connections to other repositories.

The connections here are not like direct links into other repositories, but as bookmarks that serve as convenient names to be used as a reference.

10. git remote- The git remote command is used to create, view, and delete connections to other repositories.

    The connections here are not like direct links into other repositories, but as bookmarks that serve as convenient names to be used as a reference.

11. git clone - The git clone command is used to create a local working copy of an existing remote repository.

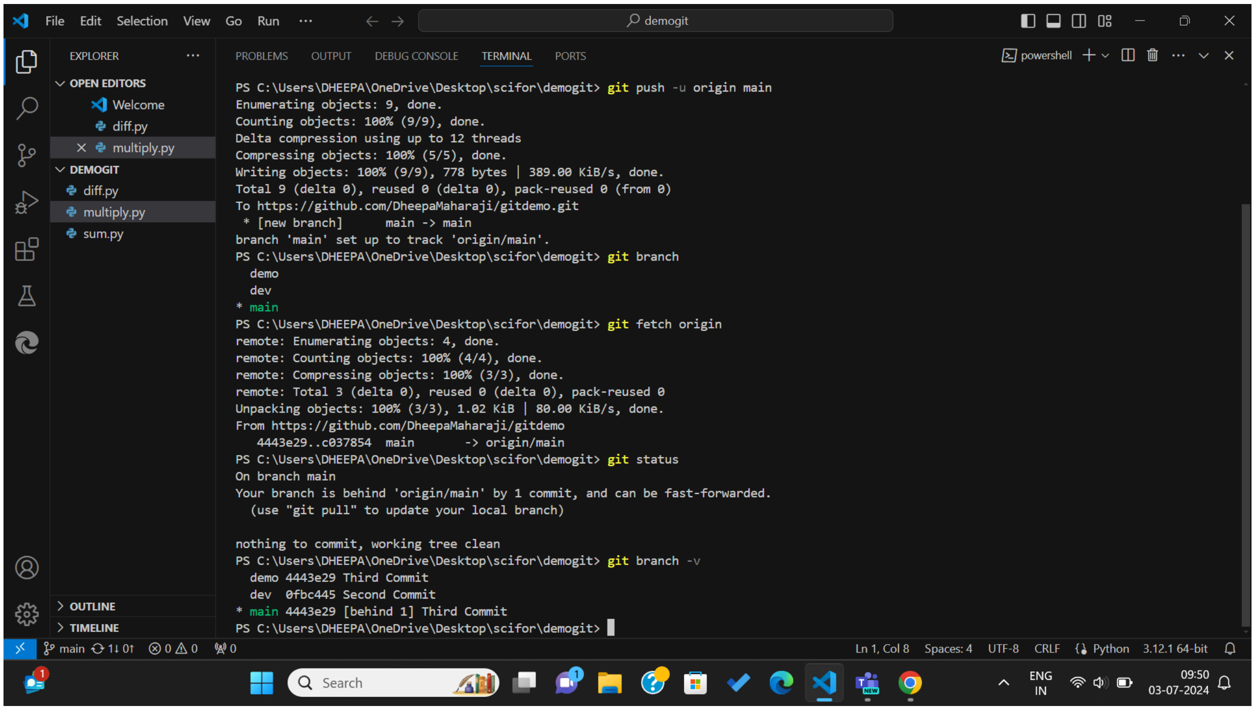
    The command downloads the remote repository to the computer. It is equivalent to the Git init command when working with a remote repository.

12. git pull - The git pull command is used to fetch and merge changes from the remote repository to the local repository.

    The command "git pull origin master" copies all the files from the master branch of the remote repository to the local repository.

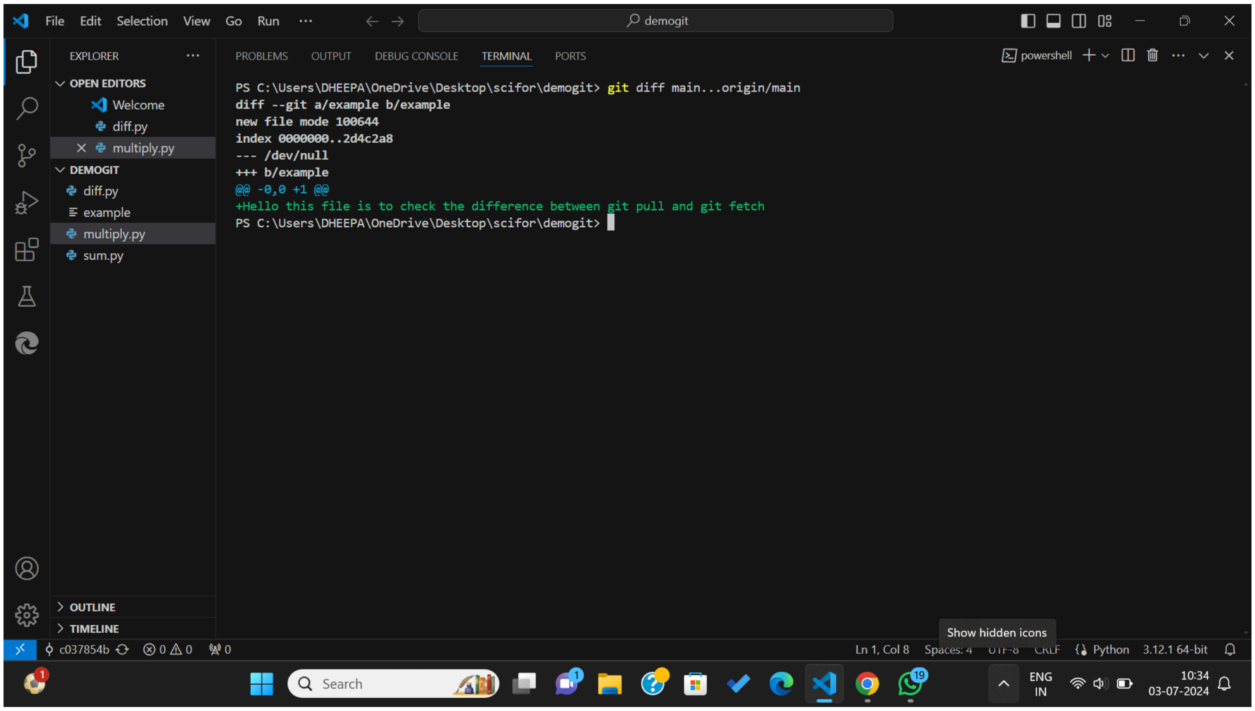
13. git push -The command git push is used to transfer the commits or pushing the content from the local repository to the remote repository.

    The command is used after a local repository has been modified, and the modifications are to be shared with the remote team members.



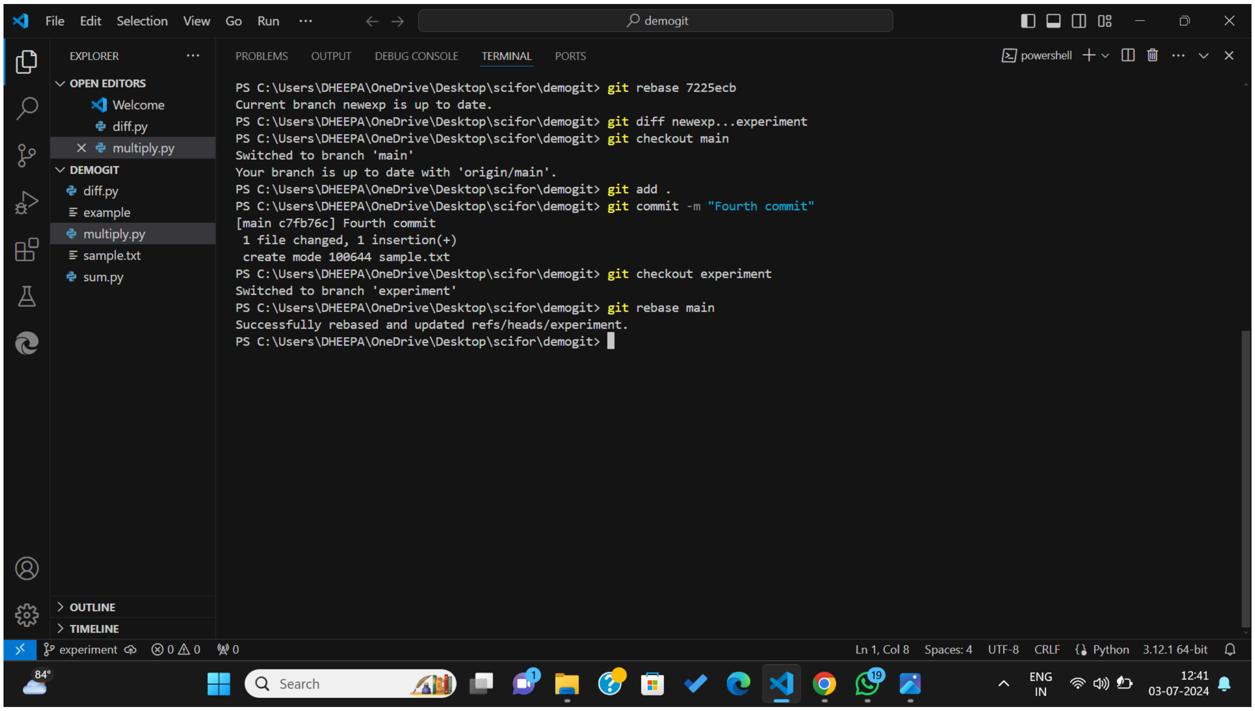
```
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git push -u origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (9/9), 778 bytes | 389.00 KiB/s, done.
Total 9 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/DheepaMaharaji/gitdemo.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git branch
  demo
  dev
* main
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git fetch origin
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 1.02 KiB | 80.00 KiB/s, done.
From https://github.com/DheepaMaharaji/gitdemo
   4443e29..c037854  main       -> origin/main
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git status
On branch main
Your branch is behind 'origin/main' by 1 commit, and can be fast-forwarded.
  (use "git pull" to update your local branch)

nothing to commit, working tree clean
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git branch -v
  demo 4443e29 Third Commit
  dev  0fbc445 Second Commit
* main 4443e29 [behind 1] Third Commit
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit>
```

14. git fetch - The git fetch command downloads all new commits from the remote repository but does not merge them with your local copy



```
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git diff main...origin/main
diff --git a/example b/example
new file mode 100644
index 0000000..2d4c2a8
--- /dev/null
+++ b/example
@@ -0,0 +1 @@
+Hello this file is to check the difference between git pull and git fetch
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit>
```

15. git rebase - Git rebase in standard mode will automatically take the commits in your current working branch and apply them to the head of the passed branch.



```
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git rebase 7225ecb
Current branch newexp is up to date.
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git diff newexp...experiment
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git add .
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git commit -m "Fourth commit"
[main c7fb76c] Fourth commit
 1 file changed, 1 insertion(+)
 create mode 100644 sample.txt
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git checkout experiment
Switched to branch 'experiment'
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit> git rebase main
Successfully rebased and updated refs/heads/experiment.
PS C:\Users\DHEEPA\OneDrive\Desktop\scifor\demogit>
```