

Git and GitHub

Assignment:2(a)

Create version control account on GitHub and using Git Commands to create a repository and push your code on github.

What is Git?

- Git is a **distributed version control system (DVCS)** used for tracking changes in source code during software development.
- It helps developers collaborate efficiently, maintain code history, and manage different versions of a project.
- Key Features:
 - Distributed System
 - Version Control
 - Branching and Merging

What is GitHub?

- GitHub is a cloud-based platform for **version control and collaboration**, built on **Git**.
- It allows developers to **store, manage, and share** their code while working on projects with others.
- Key Features
 - Repository Hosting
 - Security & Access Control
 - Pull Requests & Code Review

Difference between Git And Github:



Git

Software

Version control

Maintained by Linux

Open-Source

No user management

Locally installed

Minimal external tool
configuration

Little to no competition



GitHub

Service

Git repository hosting

Maintained by Microsoft

Free or paid membership

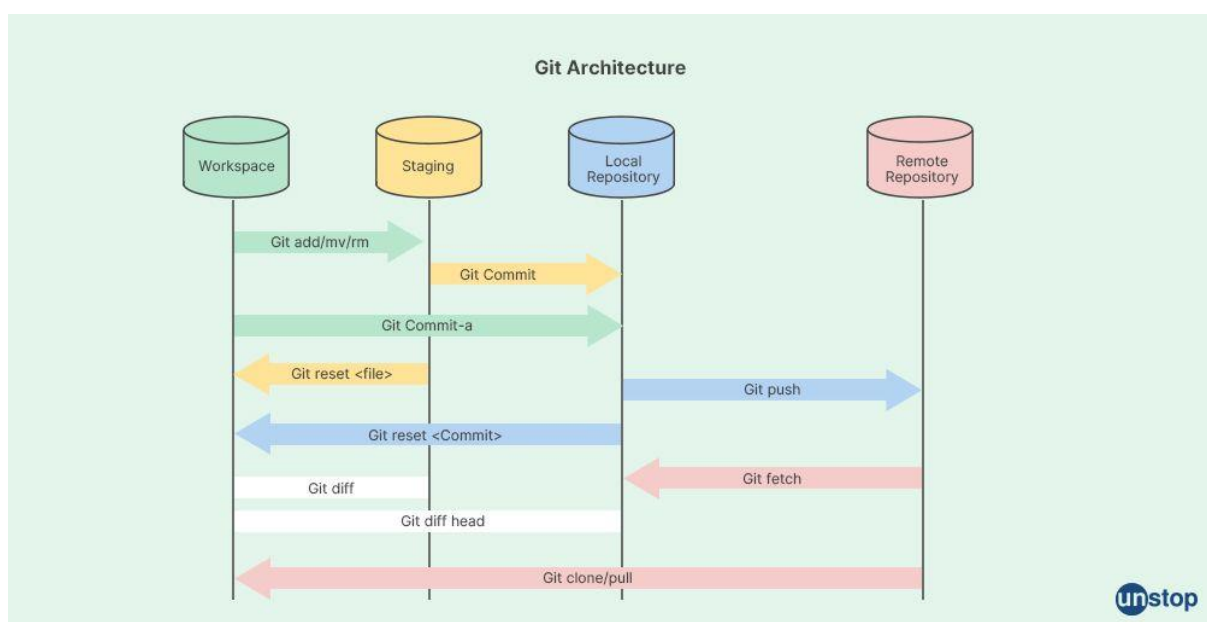
Built-in user management

Hosted on the web

Active marketplace for
tool integration

High competition

Git Architecture



Components of Git architecture:

1. Working Directory

- Stores the current version of your project files.
- Any modifications made here are untracked until staged.

2. Staging Area (Index)

- Holds changes before committing to the repository.
- Allows selective commits by staging specific files.

3. Local Repository

- A hidden .git folder storing all commit history.
- Enables version control even without internet access.

4. Remote Repository

- A shared Git repository hosted on platforms like GitHub.
- Used for collaboration and code sharing among developers.

GIT COMMANDS

```
MINGW64:/d/WAD
vedan@VEDANT MINGW64 /d/WAD
$ git init
Initialized empty Git repository in D:/WAD/.git/

vedan@VEDANT MINGW64 /d/WAD (master)
$ git status
On branch master

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)

vedan@VEDANT MINGW64 /d/WAD (master)
$ git add *

vedan@VEDANT MINGW64 /d/WAD (master)
$ git commit -m "New text added"
[master (root-commit) 2e5cb86] New text added
1 file changed, 0 insertions(+), 0 deletions(-)
```

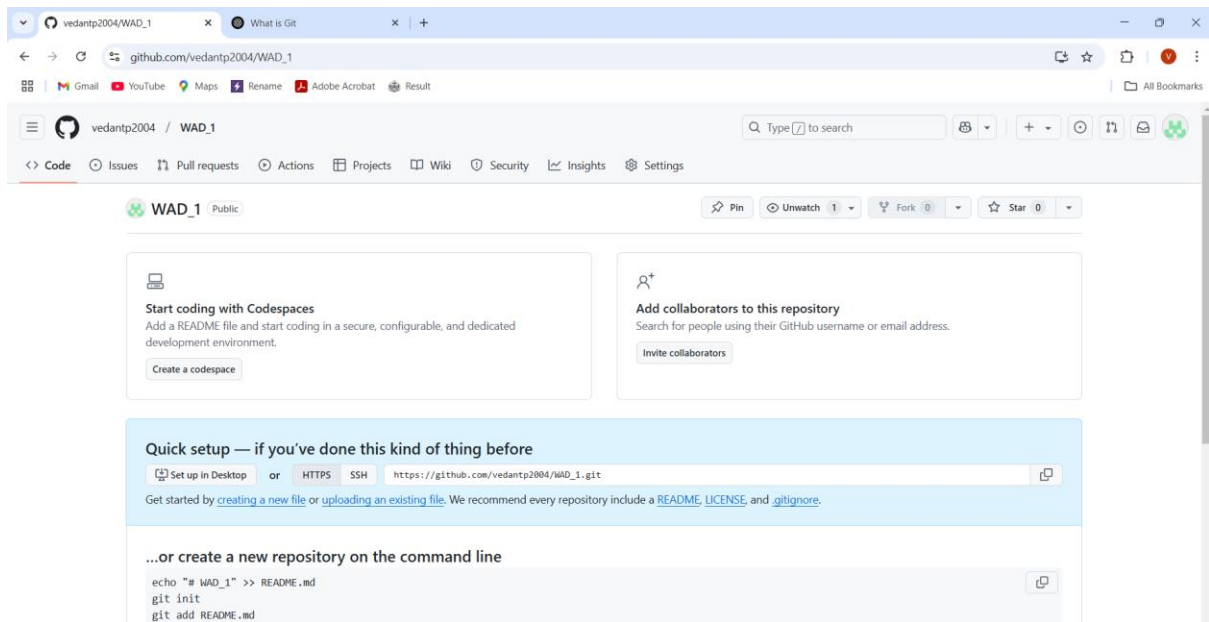
```
MINGW64:/d/WAD
create mode 100644 New Doc.txt

vedan@VEDANT MINGW64 /d/WAD (master)
$ git remote add origin https://github.com/vedantp2004/WAD_1.git

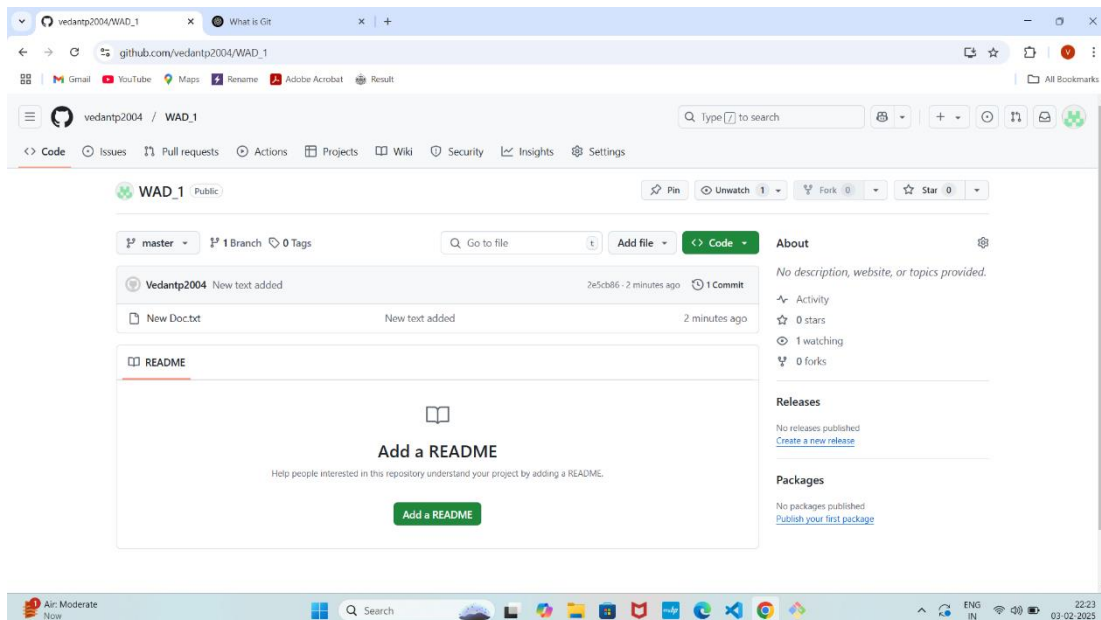
vedan@VEDANT MINGW64 /d/WAD (master)
$ git remote add origin https://github.com/vedantp2004/WAD_1.git
error: remote origin already exists.

vedan@VEDANT MINGW64 /d/WAD (master)
$ git push -u origin master
info: please complete authentication in your browser...
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 219 bytes | 219.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/vedantp2004/WAD_1.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

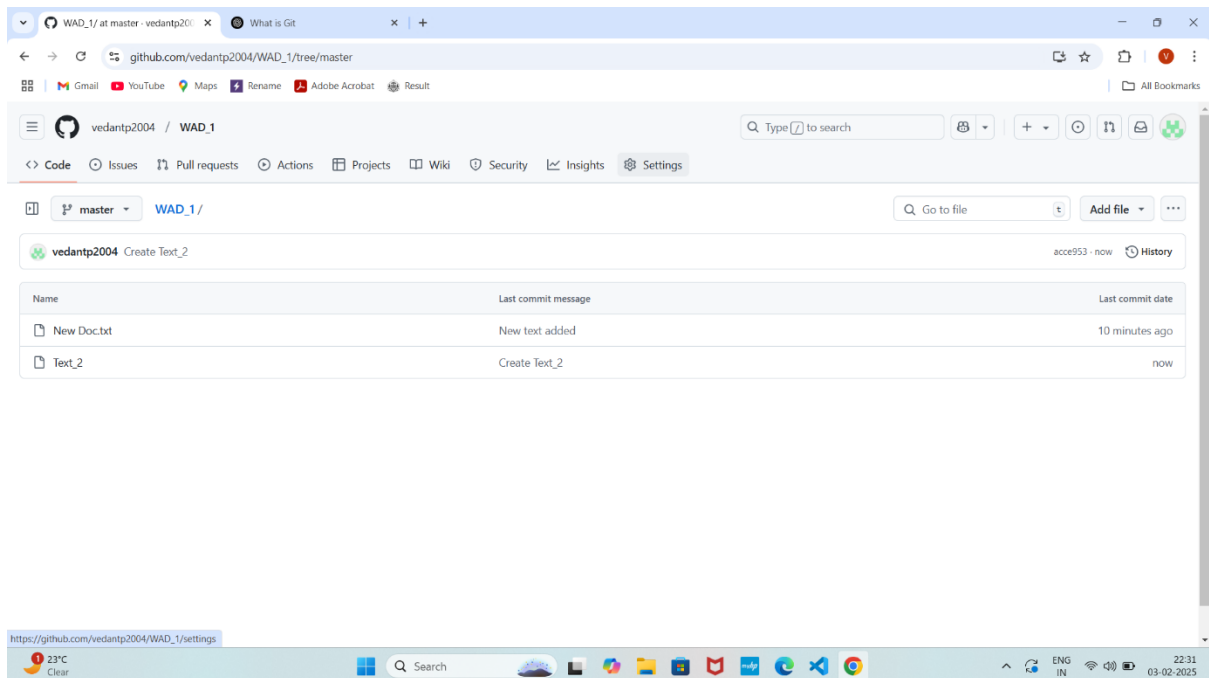
PUSH(Before)



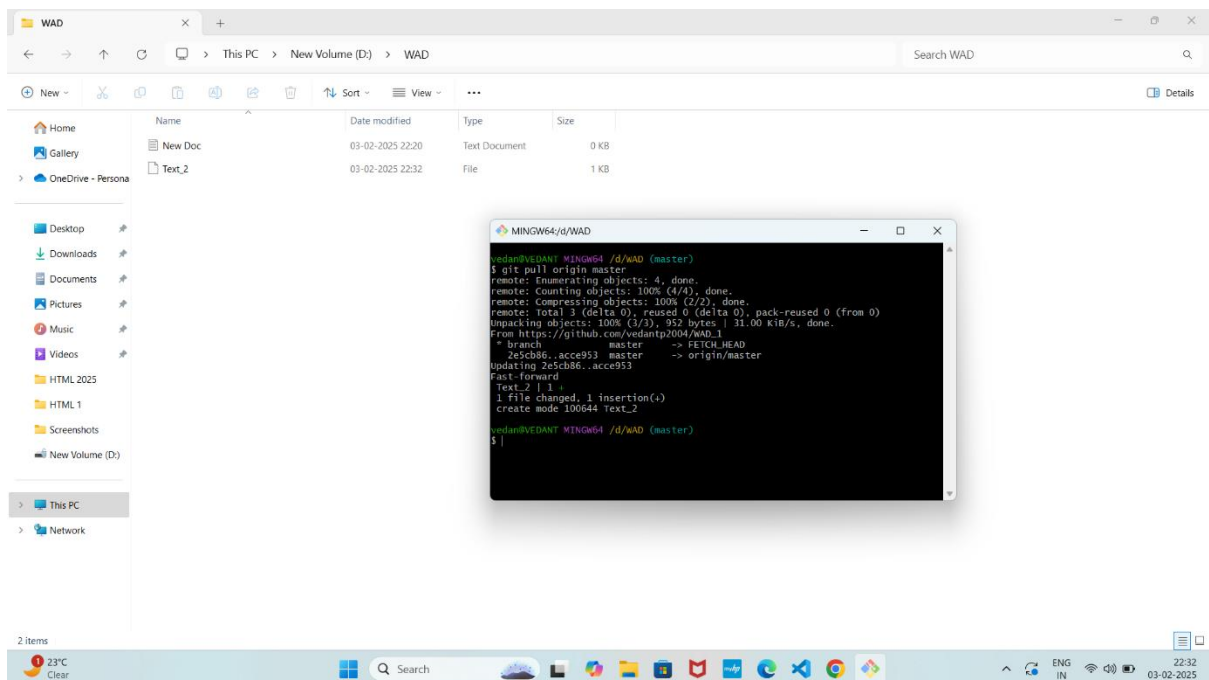
PUSH(After)



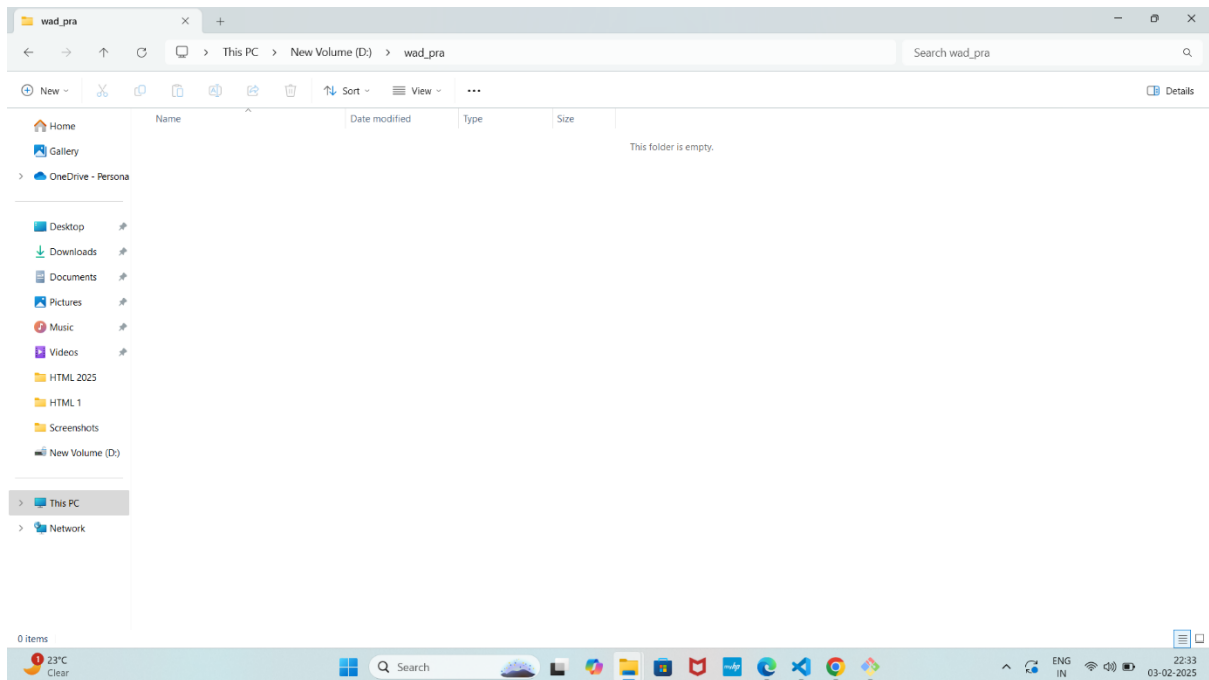
PULL(Before)



PULL(After)



CLONE(Before)



CLONE(After)

