**Version Control with Git**

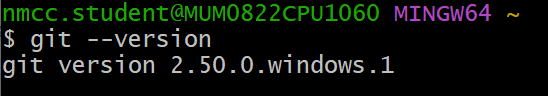
19th June

* Practical 1: To understand the concept of version control, its importance in managing project changes, and explore different version control system.

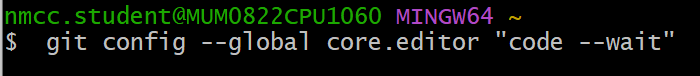
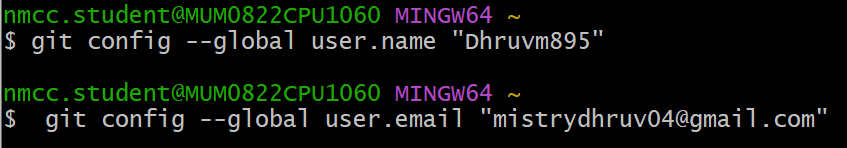
Installation

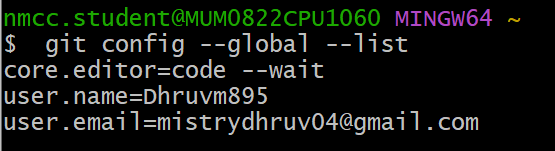
git --version

* Practical 2: To set up git on a windows system configure git for the first time and understand the basic configuration options.

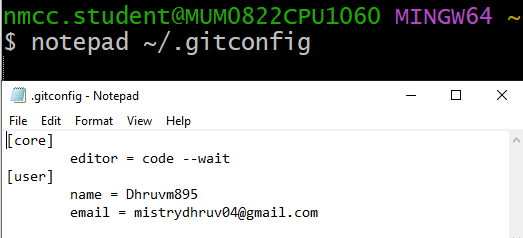
A. Install git on windows and verify the installation.

B. Configure git with your username, email and default editor.



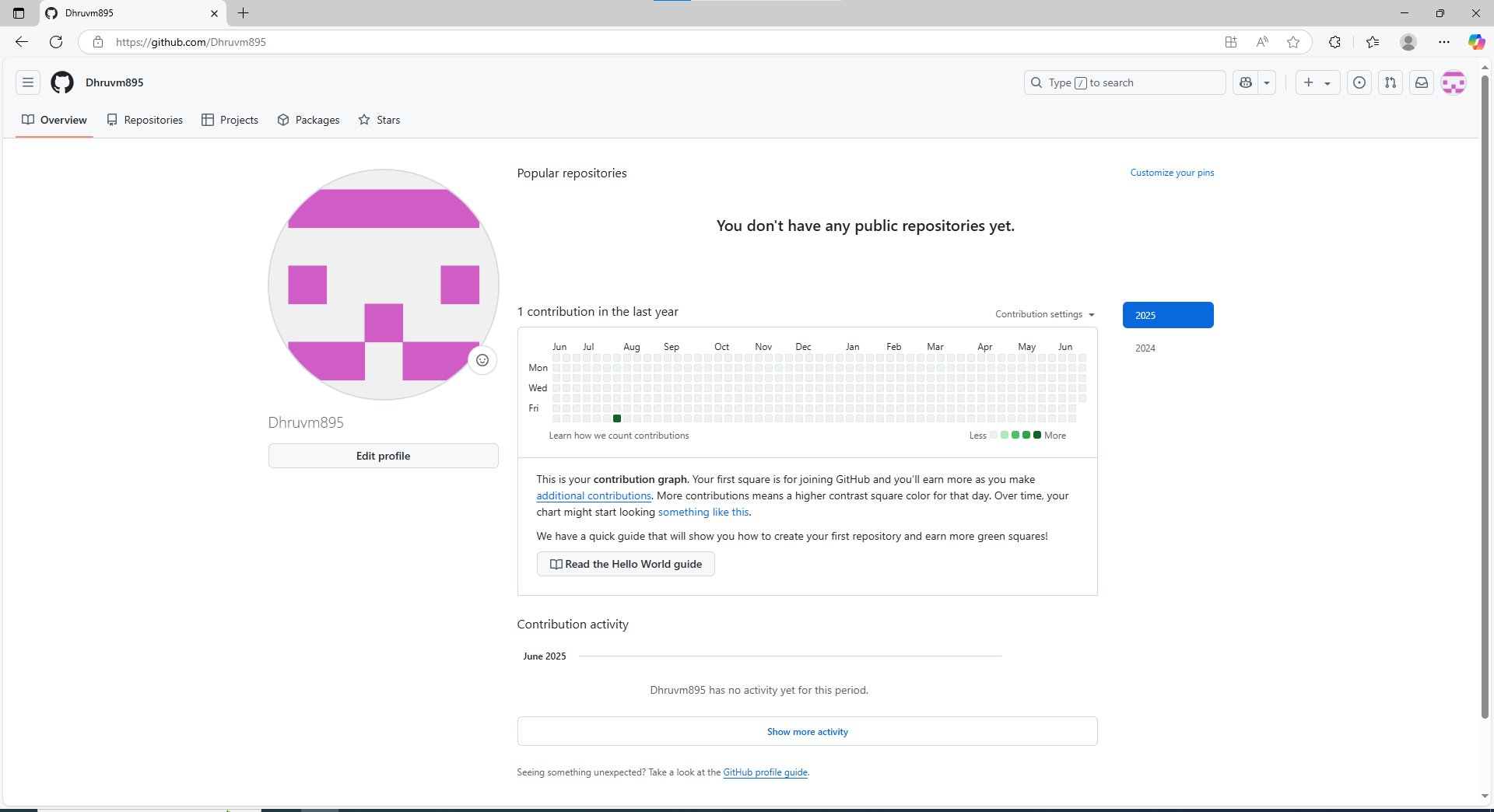


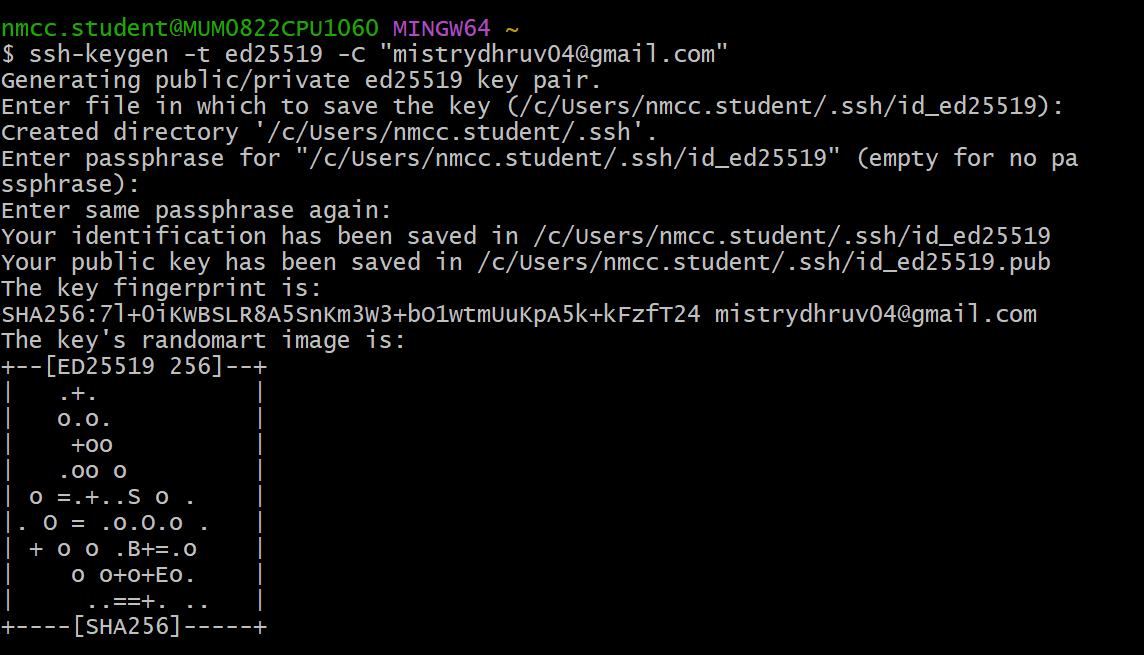
C. Explore and understand the .gitconfig file.

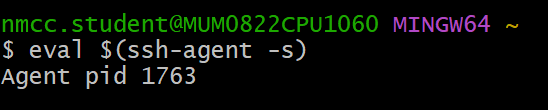


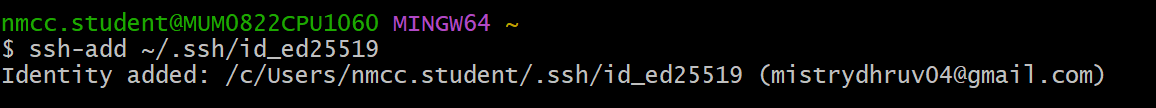
* Practical 3: To explore Github as a cloud-based hosting platform for Git repositories and understand it’s features and benefits for collaboration and project management.

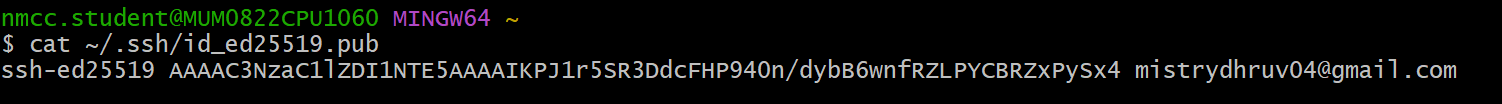
A. Creating a Github account.

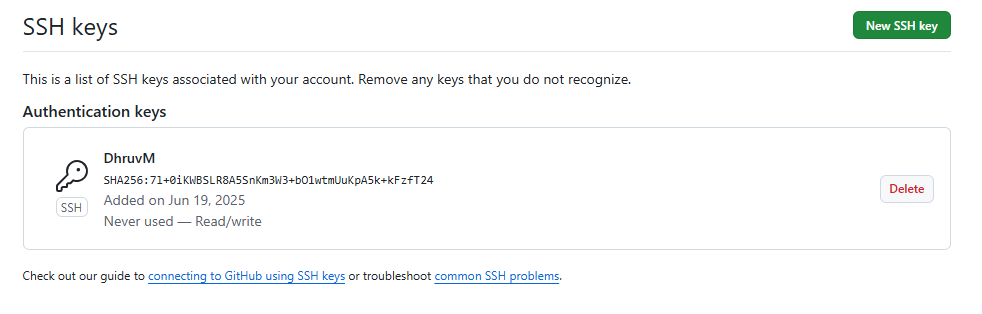


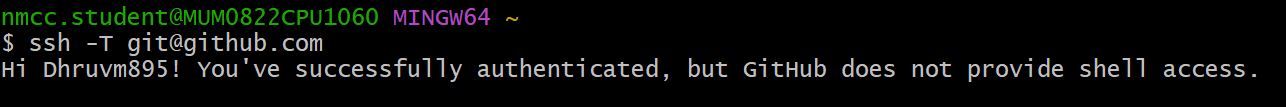
B. Generate and SSH keys to Github. 





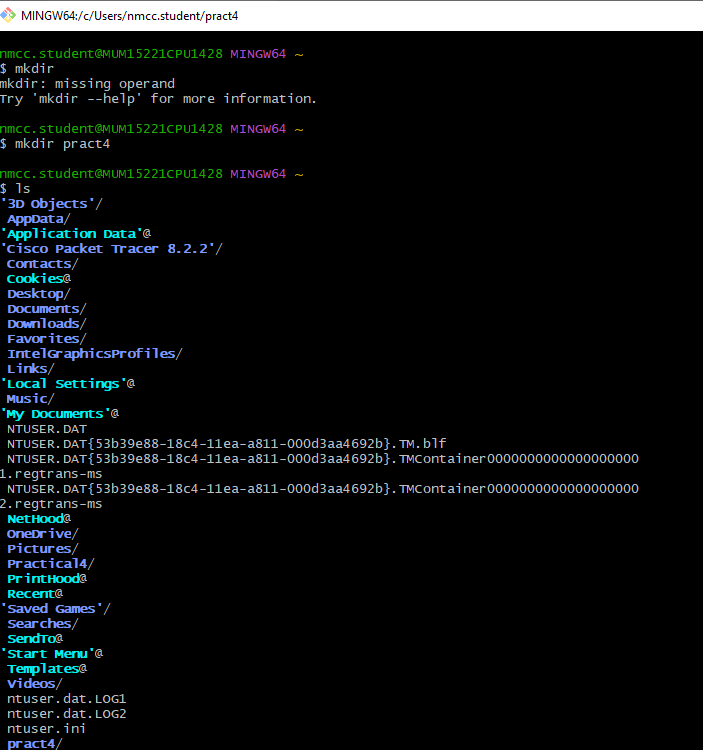






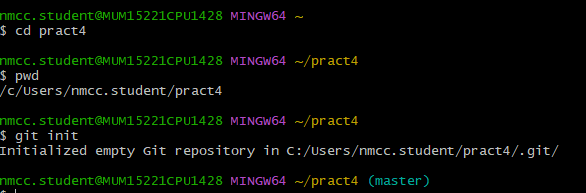
26th June

* Practical 4: To understand the concept of a Git repository and learn to create and initialize a new local repository for tracking project files.

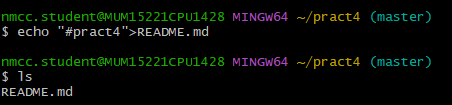
1. Create a new local repository using git init.  
   

1. Initialize the Git Repository

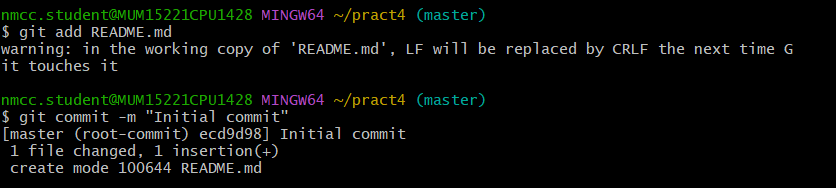
-This creates a hidden .git folder, which contains the repository history and configuration.



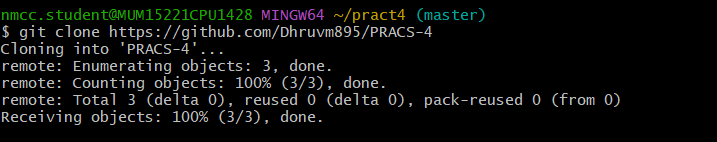
2. Add and commit files



3. Add the file to staging

****

1. Clone an existing remote repository using git clone.



1. Compare the structure of a local and remote repository.

1. Local Repository (git init)

-Exists only on your computer

-Contains a .git folder for version control

- Does not have a connection to a remote repository unless

added later (git remote add origin <URL>).

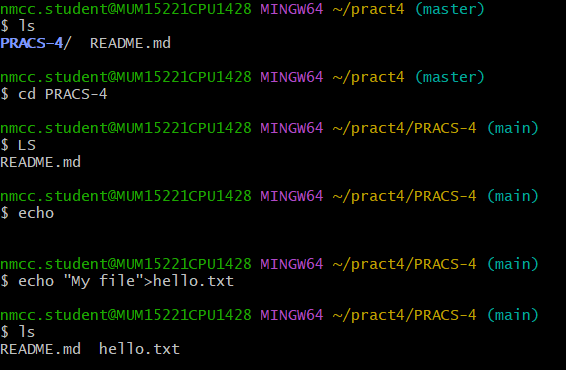
2. Remote Repository (git clone)

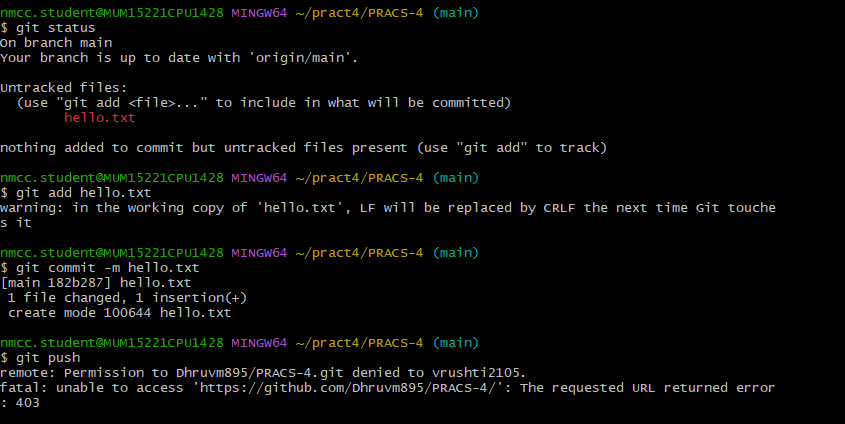
-Hosted on platform like GitHub

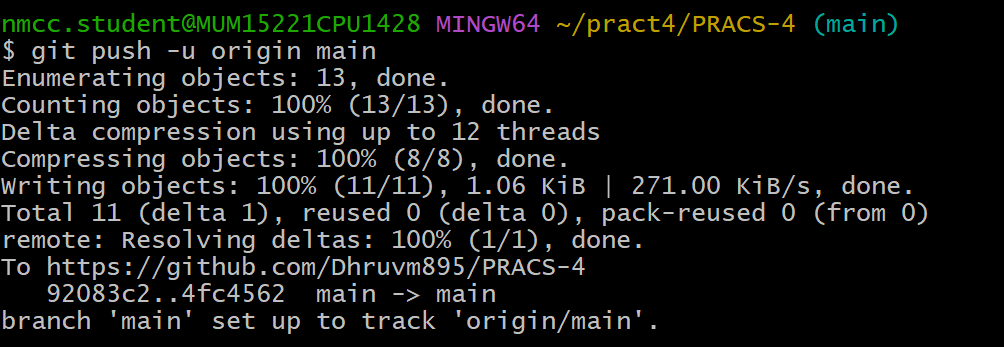
- Can be accessed by multiple uses.

- Already has a .git structure and commit history.

- The cloned local copy automatically tracks the remote repository.





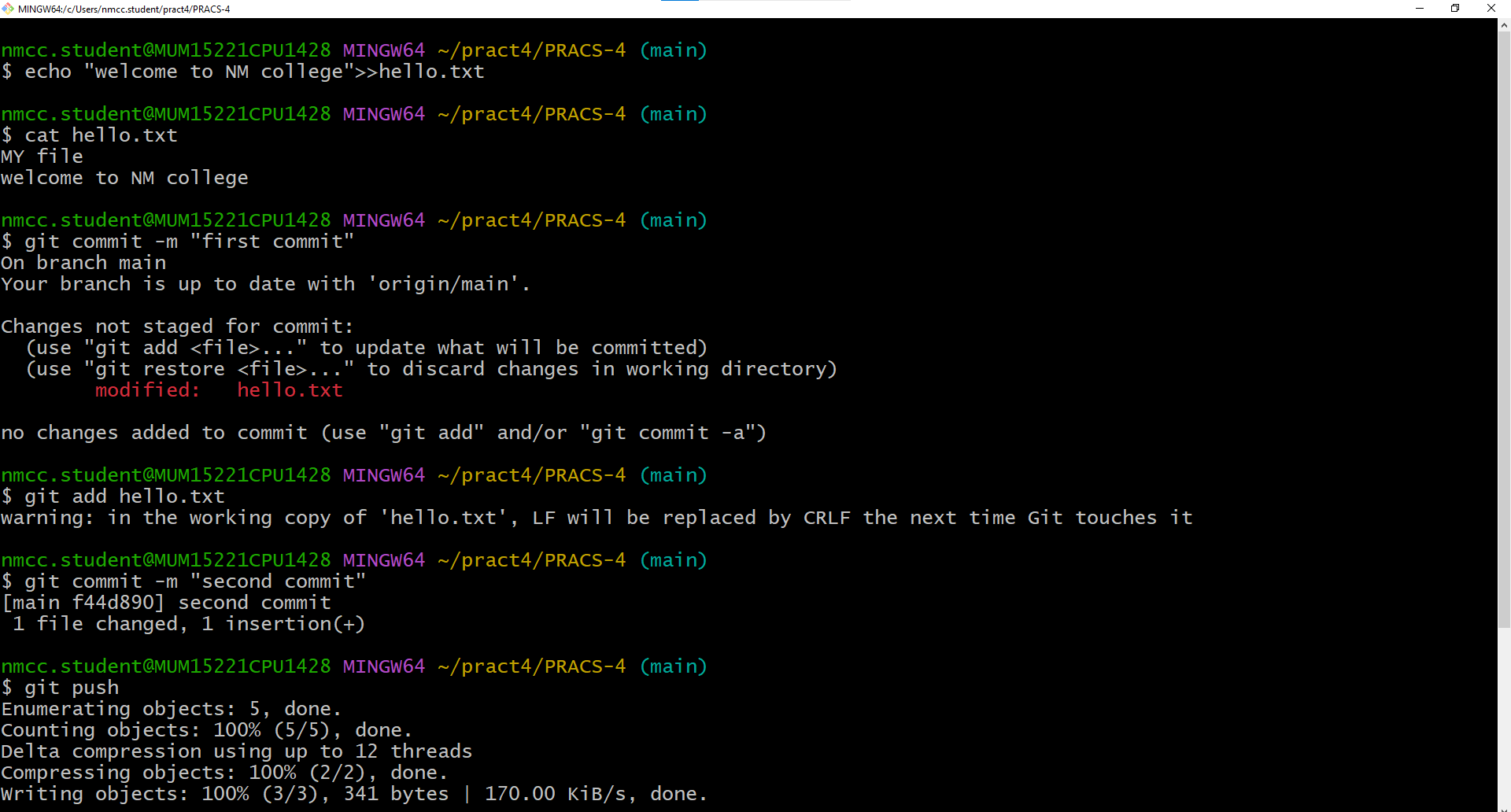


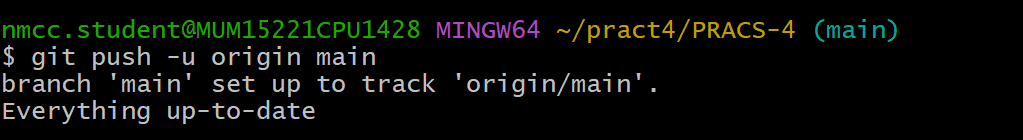
* Practical 5: To practice recording changes to files in a Git repository, checking file statuses, and managing tracked and untracked files.

1. Track new files in a repository using git add.

1. Create a New file

-Create a sample file





2. Check the Status of files

-Run

* Untracked files(red) mean they are not being tracked by Git yet

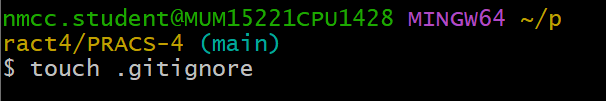
3. Add files to Staging

- Track the new file

* The file should now be in staged(green) state.

1. Create and apply a .gitignore file to exclude unnecessary files.
2. Create a .gitignore File

-Open Git Bash or a text editor and create .gitignore:



1. Add unwanted files to .gitignore

-Edit .gitignore and add file patterns to exclude:

3. Verify .gitignore is working

-Add all files

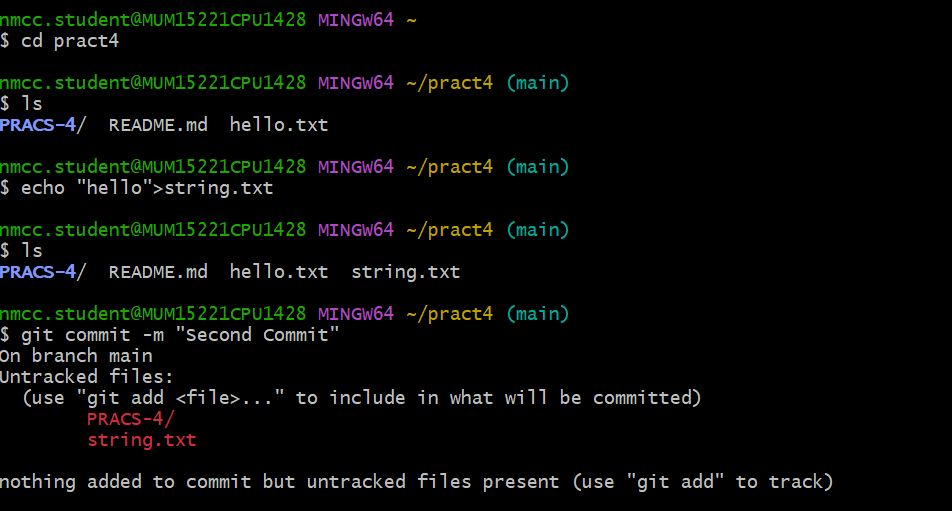
-Check status:

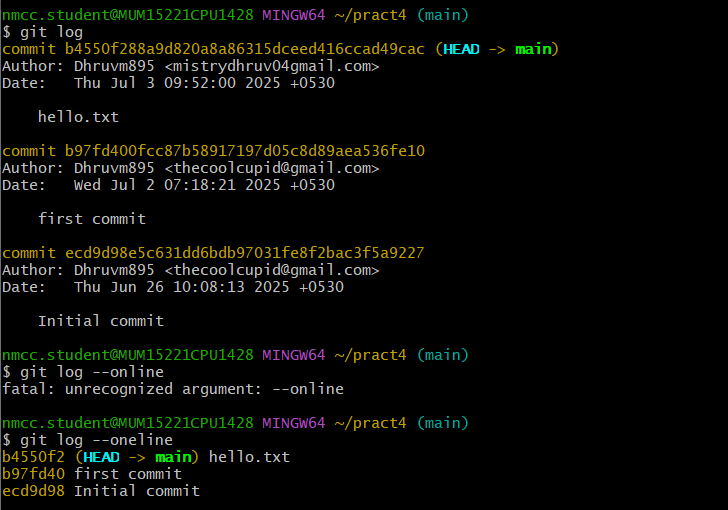
-File listed in .gitignore should now be ignored.

1. Perform first commit and commit changes with detailed messages using git commit.

1.Commit Staged Changes

- First Commit

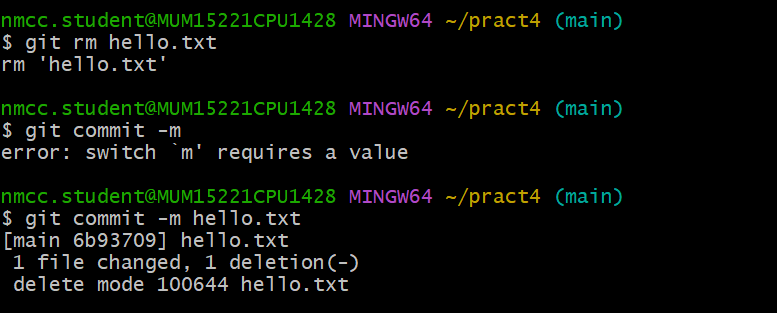




1. Demonstrate removing files (git rm) and moving files (git mv).

1.Remove a tracked file (git rm)

-Remove a file and commit:



-To remove a file but keep it in the working directory:

1. Move or Rename a file (git mv)

-Rename hello.txt to greeting.txt:

git mv hello.txt string.txt

-Commit the rename:

git commit -m ”Renamed hello.txt to greeting.txt”

* Practical 6: To explore commit history in a Git repository and learn techniques to undo changes or revert to previous states when needed

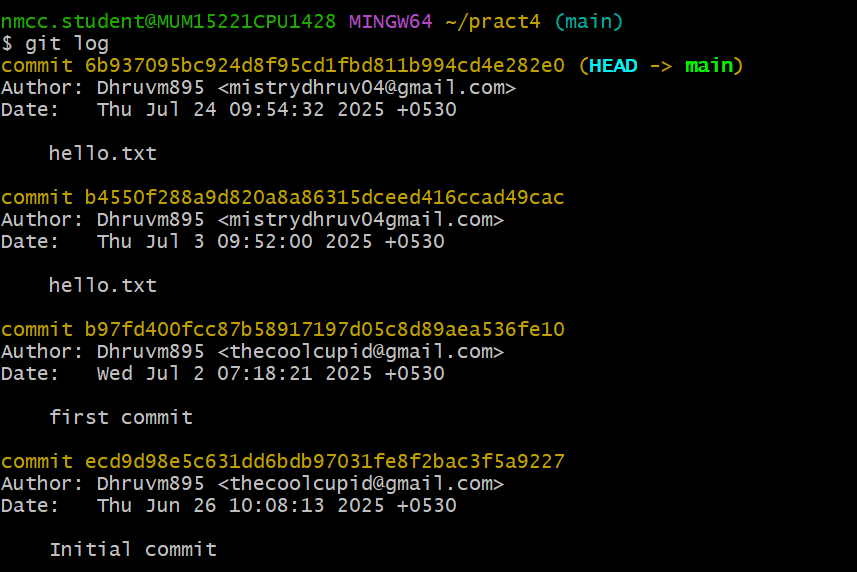
1. View commit history using git log, git log --oneline, and git log –graph
2. View Commit history(git log)

-Check full commit history

Run the following command to see a detailed history of commits:

git log

It displays:

-Commit hash (ID) 

--Author name and email

--Date

--Commit message

View a simplified History (git log –oneline)

To see a compact

1. Undo a staged file using git reset
2. Stage the file:

git add test.txt

1. Check status:

Git status

The file appears in staged(ready for commit).

1. Create a scenario to demonstrate reverting to an older commit using git revert