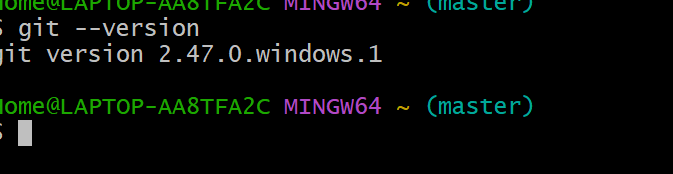
# GIT COMMANDS

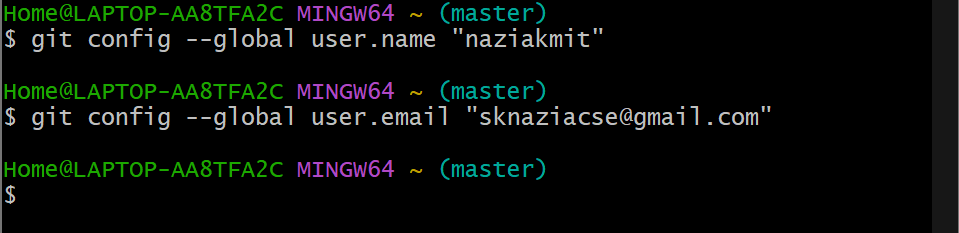
# 1.Git Version:

Check if Git is installed using **git --version**.



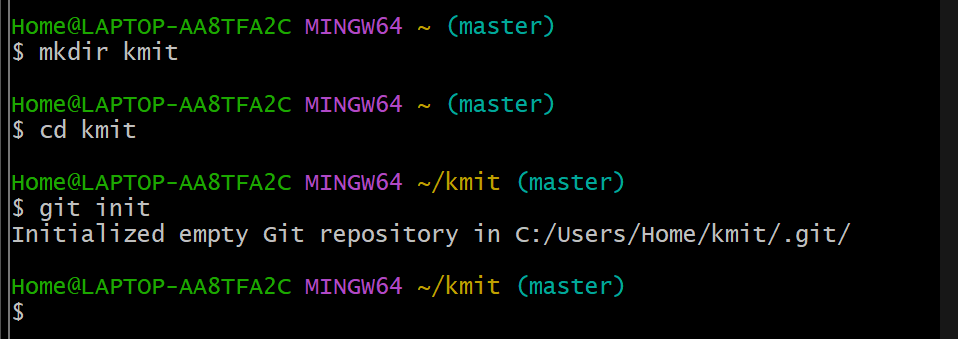
# 2.Git config:

* Set up your name and email using **git config.**
* Essential for setting up your user name and email, ensuring that every commit you make is associated with the correct identity in the project history.



# 3.Git init:

* Start a new project by creating a Git repository.

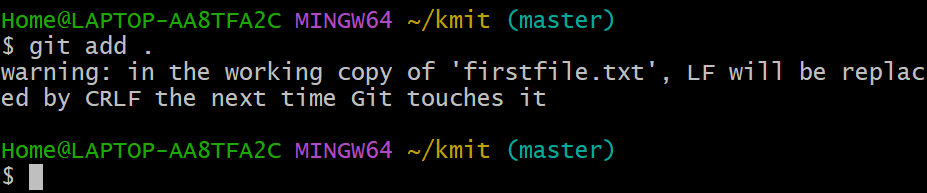


Creating a file in the name of f1.txt



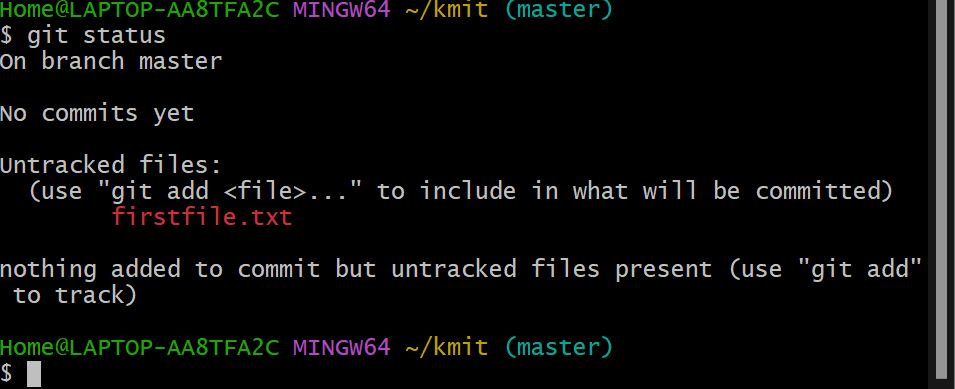
# 4.Git add:

Stage changes by adding files to the staging area before saving.



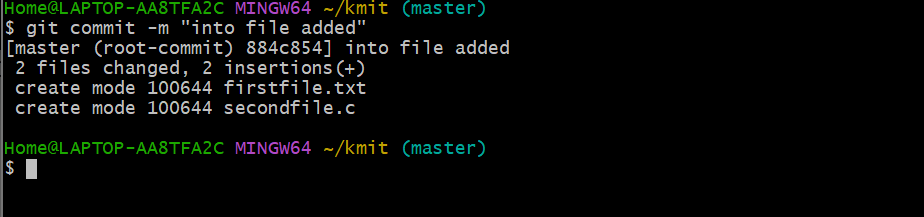
# 5.git status:

* Check the current state of your files and see if any changes need to be saved.



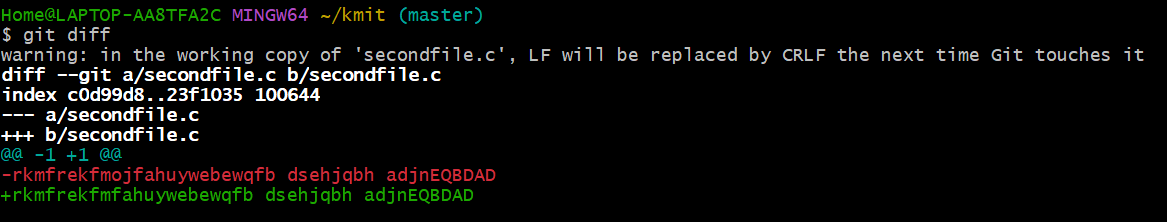
6.Git commit:

* Save your changes with a message explaining what was changed.



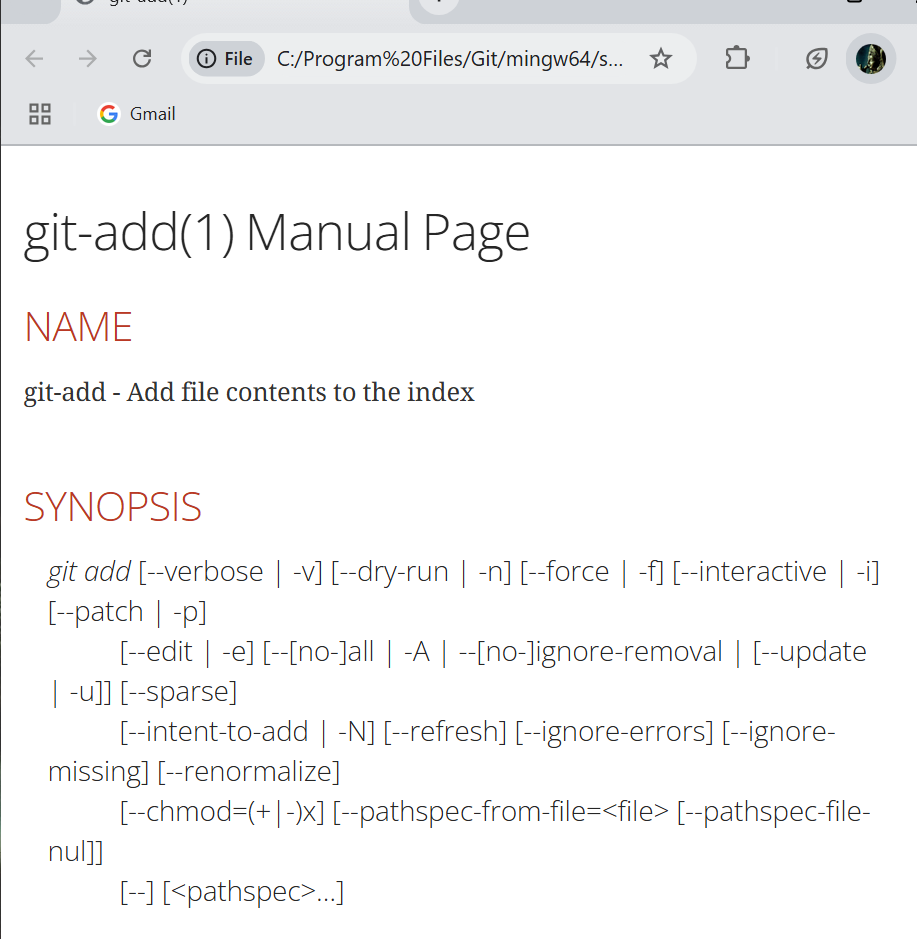
# 7.git diff:

* Compare changes between your current files and the last commit.



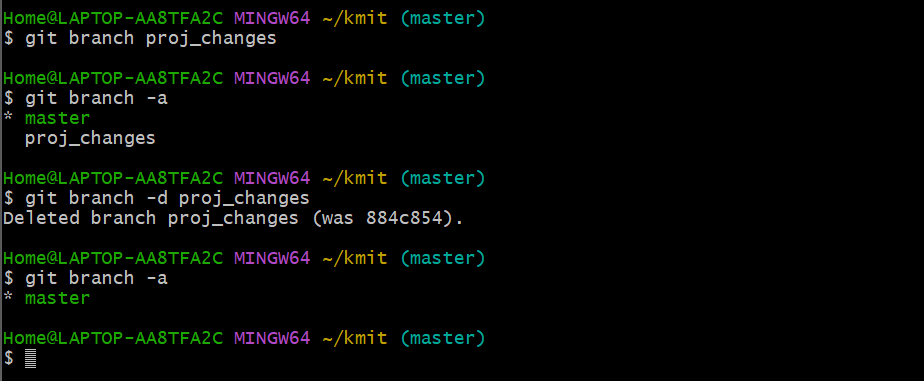
# 8.Git help add:

Screenshot 2024-11-14 101905



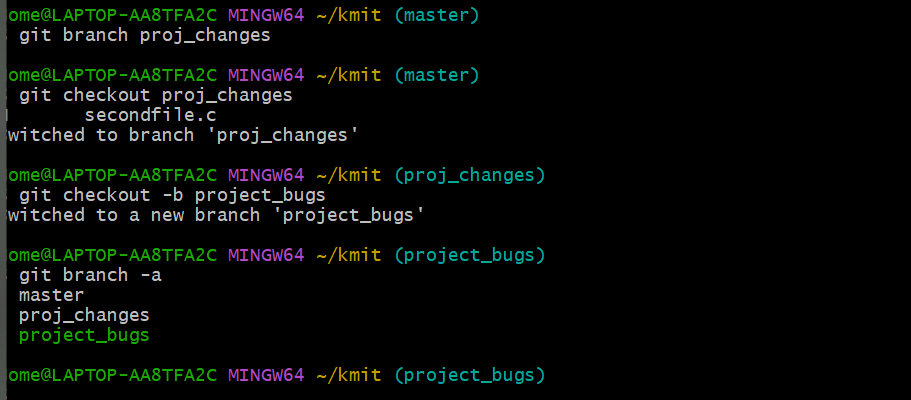
# 9.git branch:

A branch is like a separate workspace where you can work on new features or fixes without affecting the main project.



# 10.Git checkout:

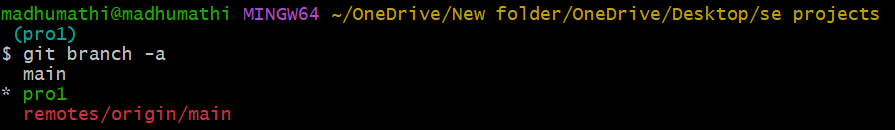
Switching between branches.



s

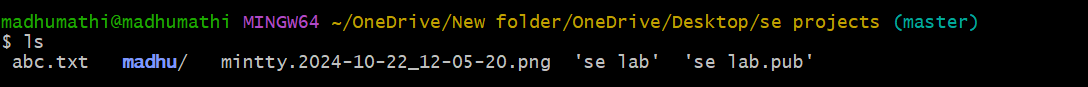
# 11.Git branch -a:

List the branches

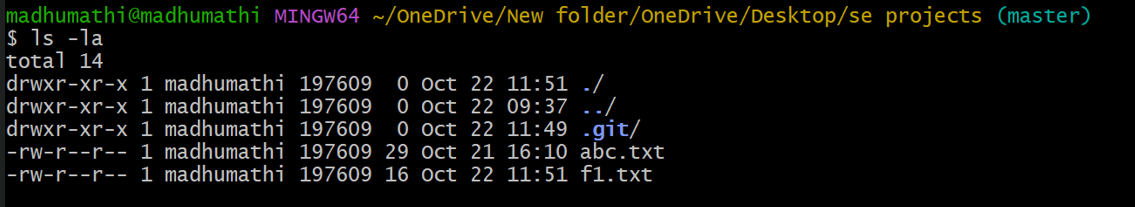


# 12.Git ls:

List out the files



# 13.Git ls-la:



# 14.Git log:

**git log:** Shows the full history of your project.

Includes:

* Commit Hash: A unique ID for each change.
* Author: Who made the change.
* Date: When the change was made.
* Message: A brief description of the change.



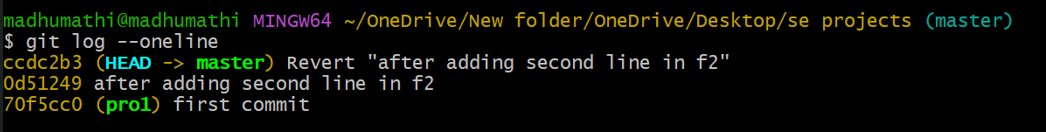
# 15.Git log—oneline:

**git log –oneline:** A simplified version of git log, showing the commit history in one line per commit.

Includes:

Shortened commit hash

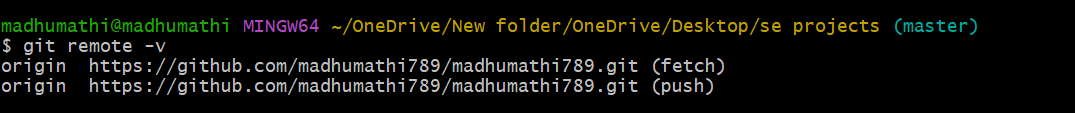
Commit message

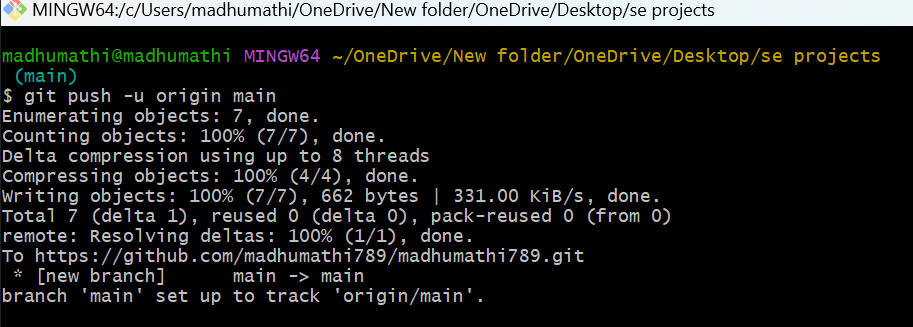


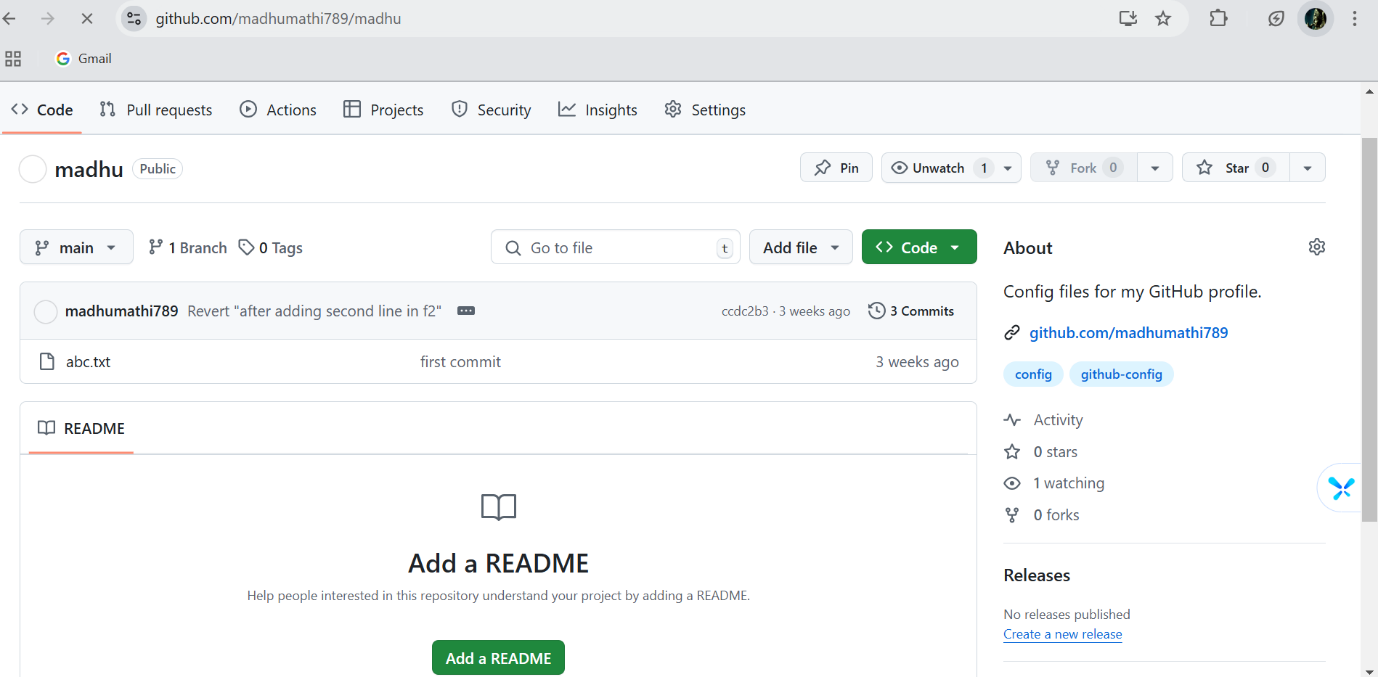
# 16.Git remote:

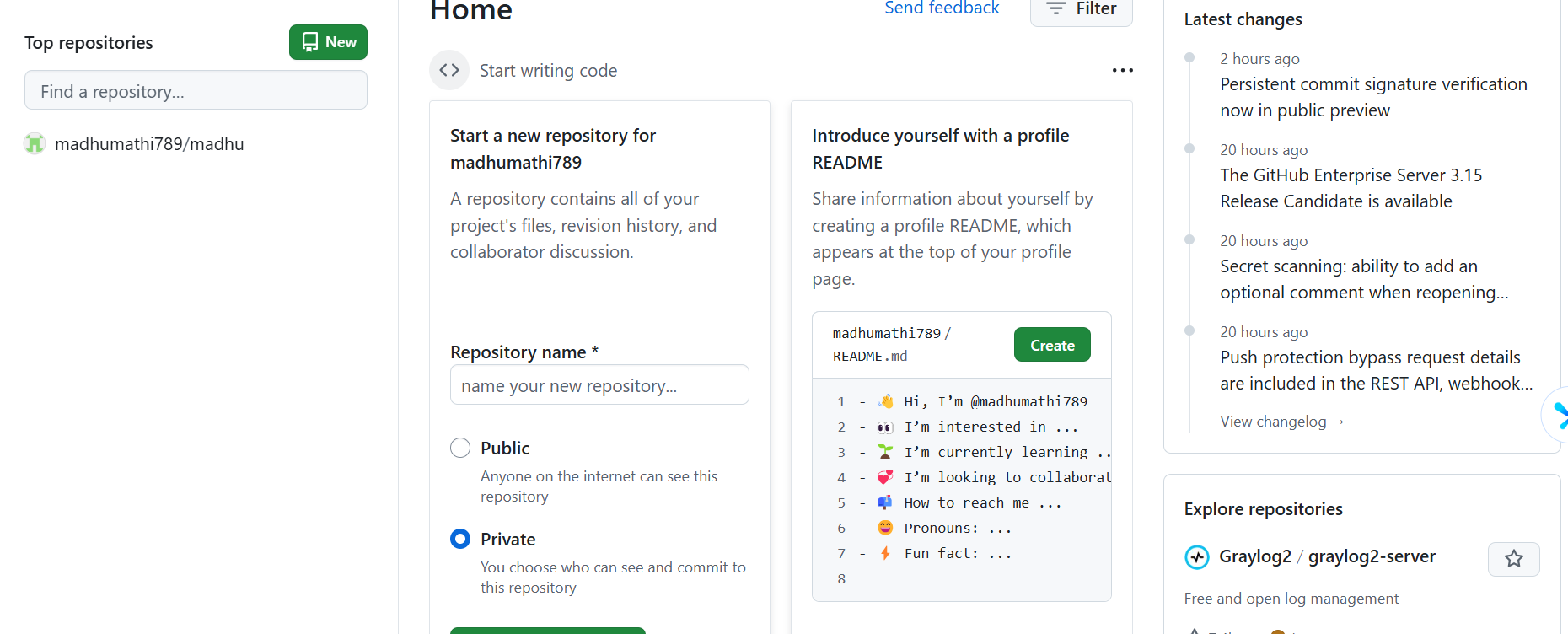
**“git remote lets you manage connections to other Git repositories stored on the internet (like GitHub or Bitbucket).”**

* In Git, remote is how you save the address of your project that’s stored online.
* You use this command to add, remove, or view the places (remotes) where your project can be stored.









# 17.git to github using SSH :

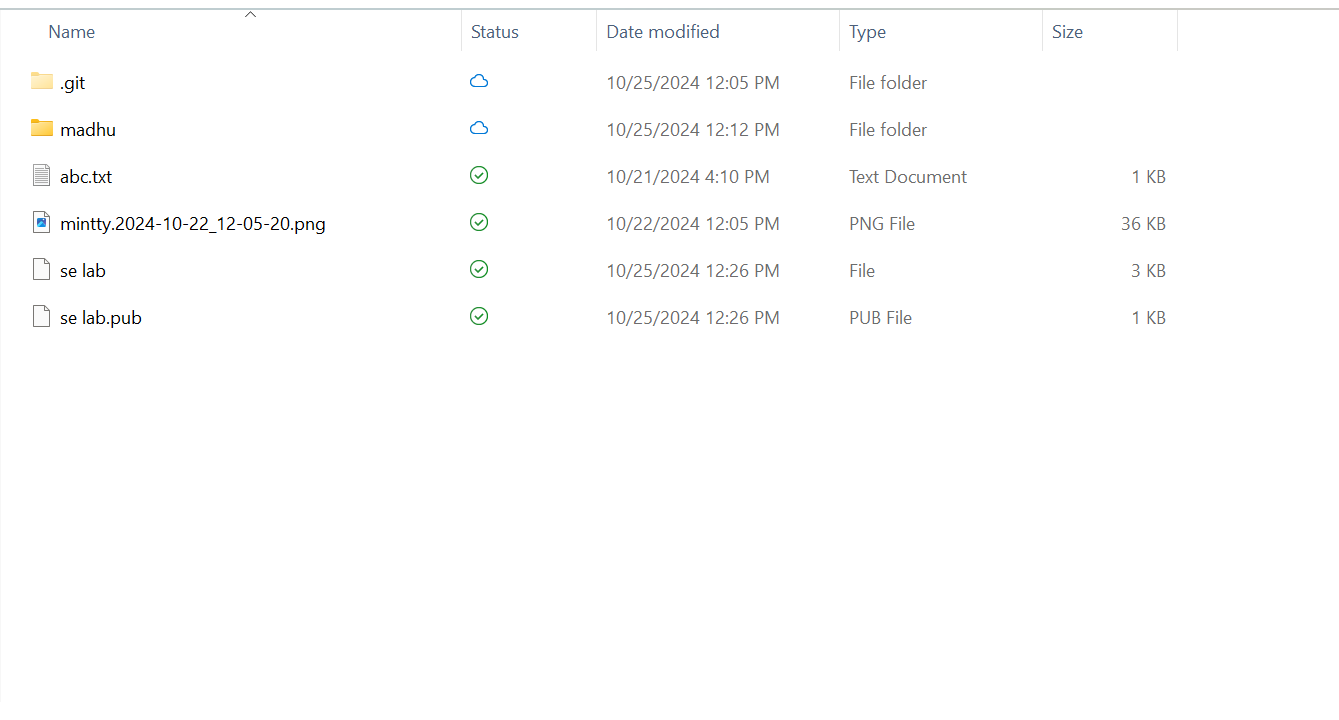
**Steps:**

**Create SSH Key:** You use a command **(ssh-keygen -t rsa)** to generate a key pair on your system.

**Copy Public Key:** You then copy the public part of the key **(the .pub file**).

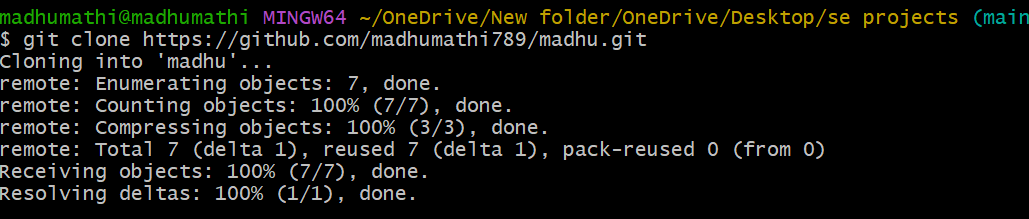
**Add to GitHub:** You go to GitHub settings, navigate to “**SSH and GPG keys**,” and paste the public key. This lets GitHub know it’s your computer connecting.

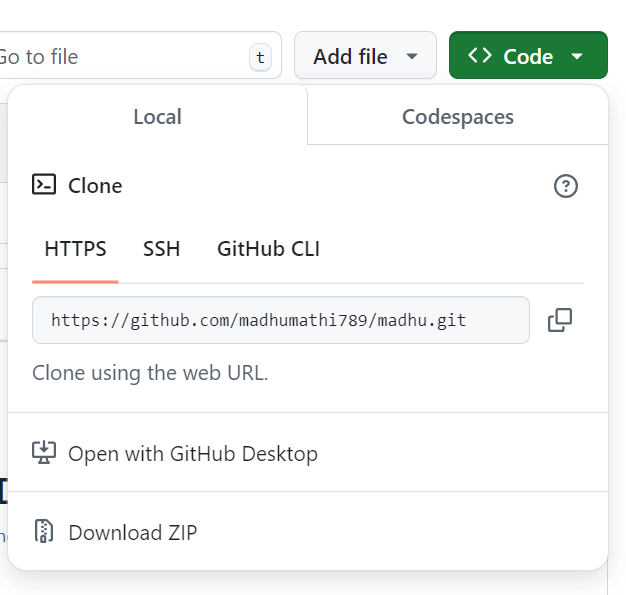
# 





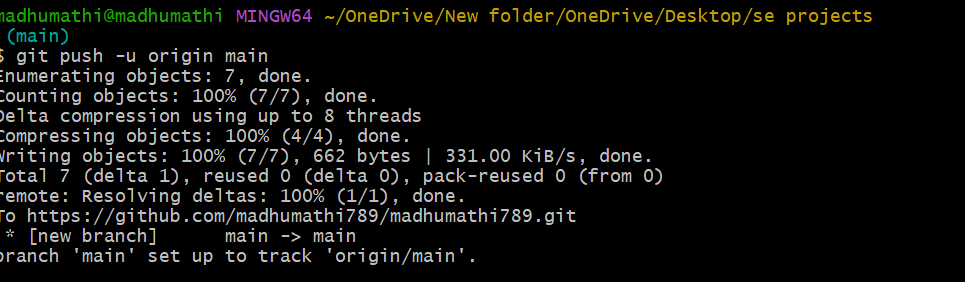
# 18.Git clone:





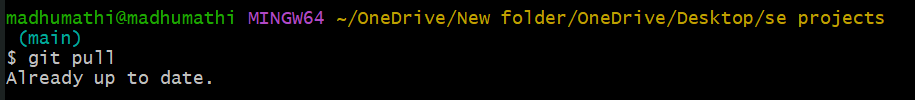
# 19.Git push:

git push is used to send your changes from your local repository (on your computer) back to the remote repository (like GitHub). This is how you share your work with others.



# 20.Git pull:

git pull brings the latest updates from a remote repository to your local repository. It fetches the changes and merges them with your current work.



# 21.Git fork:

