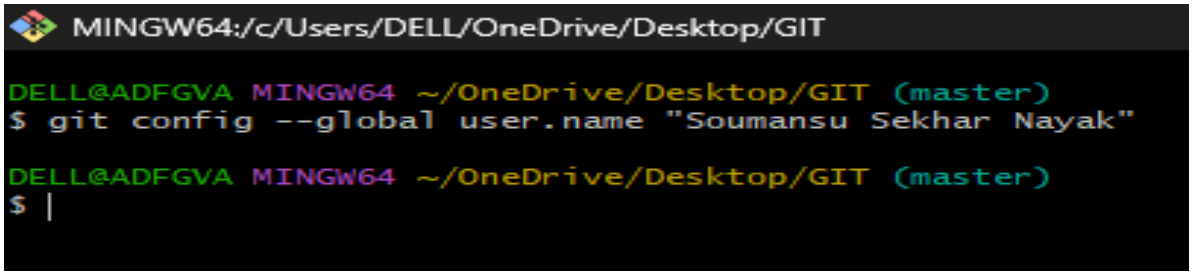
Task: Tutorial for Git and GitHub with the Commands.

1. Git Setup

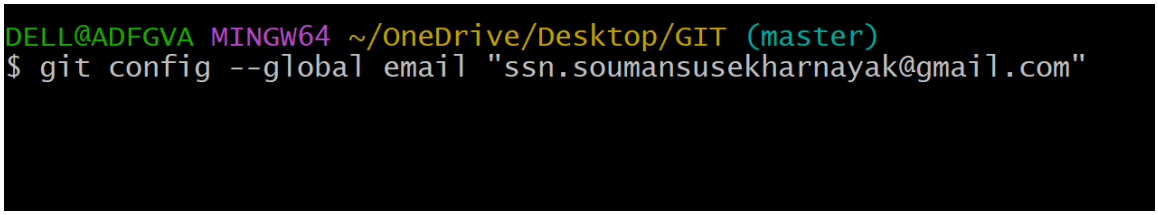
- Sets your Git username.
- **--global** makes it apply to all your repositories on your system.

git config git-global user.name "Soumansu sekhar Nayak"



- Sets your Git email globally. This is important for associating commits with your GitHub account.

git config-global user.email "ssn.soumansusekharnayak@gmail.com"



2. Initialize a Repository

- Initializes a new Git repository in the current directory, allowing version control to begin.
- Creates a **.git** folder to track changes:

git init

```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git init
Reinitialized existing Git repository in C:/Users/DELL/OneDrive/Desktop/GIT/.git/
```

3. Check Repository Status

- Displays the current status of the repository, showing which files are staged, modified, or un-tracked.

git status

4. Add Files to Staging

- Stages a specific file to be committed. Only staged files are included in the next commit.

```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git add file.txt
```

git add filename

5. Commit Changes

- Creates a snapshot of the staged changes with a descriptive message.
- Save the staged changes to the repository:

git commit -m "Your commit message"

```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git commit -m "file.txt"
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        clone/
        clone1/

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

6. View Commit History

- Displays a history of commits, including commit IDs, authors, dates, and messages.

git log

```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git log
commit 0c5af8c5559a52d7a80196874f12180bee3bc62f (HEAD -> master, Wipro1, Wipro)
Author: Soumansu sekhar nayak <ssn.soumansusekharnayak@gmail.com>
Date:   Wed Apr 2 15:51:58 2025 +0530

    file3 updated

commit aaf1c0dd7f9a8949d4adf64238cb260fd71a268f
Author: Soumansu sekhar nayak <ssn.soumansusekharnayak@gmail.com>
Date:   Wed Apr 2 15:37:48 2025 +0530

     file1 and file2 updated.

commit 0518a0e563ebfdcd9bb45052b2511604e70932a2
Author: Soumansu sekhar nayak <ssn.soumansusekharnayak@gmail.com>
Date:   Wed Apr 2 15:22:17 2025 +0530

    File updated
```

- Shows each commit in one line (useful for quick reference).

git log --oneline

```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git log --oneline
0c5af8c (HEAD -> master, Wipro1, Wipro) file3 updated
aaf1c0d  file1 and file2 updated.
0518a0e File updated
```

7. Connect to GitHub

- Links the local repository to a remote one, usually on GitHub, for pushing and pulling code.

*git remote add origin **https://github.com/username/repo-name***

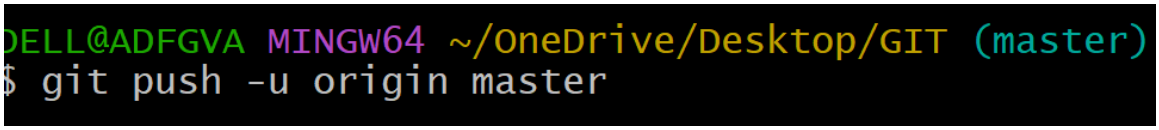
```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git remote add origin https://github.com/soumansu/Wipro.git
```

8. Push Code to GitHub

- Pushes local commits to the remote master branch and sets the upstream tracking reference

- **-u** sets upstream tracking so future **git push** commands can be used without specifying the branch.

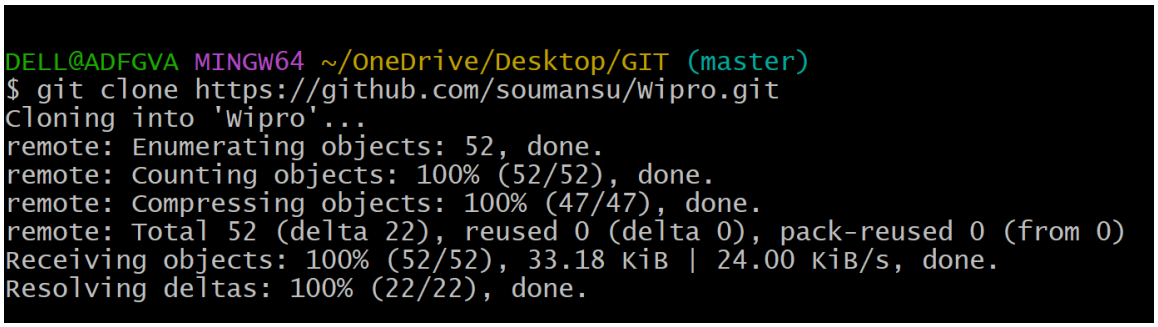
git push -u origin master



9. Clone a Repository

- Creates a local copy of a remote repository by downloading its contents and history

git clone https://github.com/username/repo-name.git



10. Pull Changes

- Fetches and integrates changes from the remote master branch into the local branch.

git pull origin master

```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git pull
remote: Enumerating objects: 52, done.
remote: Counting objects: 100% (52/52), done.
remote: Compressing objects: 100% (47/47), done.
remote: Total 52 (delta 22), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (52/52), 33.16 KiB | 12.00 KiB/s, done.
From https://github.com/soumansu/Wipro
 * [new branch]      main       -> origin/main
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

    git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

    git branch --set-upstream-to=origin/<branch> master
```

11. Create a New Branch

- Creates a new branch from the current HEAD. Useful for feature development without affecting the main code.

git branch <new-branch>

```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git branch wipro
fatal: a branch named 'wipro' already exists
```

12.Checkout

- Switches the working directory to the specified branch.

git checkout <new-branch>

```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master)
$ git chechout wipro
git: 'chechout' is not a git command. See 'git --help'.

The most similar command is
        checkout
```

13. Merge Branches

- Combines the specified branch into the current branch, integrating changes made in the other branch.

git merge new-branch

```
DELL@ADFGVA MINGW64 ~/OneDrive/Desktop/GIT (master
$ git merge wipro
Already up to date.
```