NAME-CHINMAYEE RANA
REGD NO.-2141018149

GIT TUTORIAL

GIT : Git is a version control system that lets developers track changes in their code, collaborate with others, and manage different versions of a project efficiently.

GIT COMMANDS :

git –version -> We use git –version command to check the installed Git version on a system.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt
$ git --version
git version 2.49.0.windows.1
```

ls -> We use the **ls** command in git to view the files and folders in the current directory.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt
$ ls
```

ls –a -> We use the **ls –a** command to show all files, including hidden files(like .git) in the directory.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt
$ ls -a
./  ../
```

git init -> git init is used to initialize a new Git repository in your project folder.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt
$ git init
Initialized empty Git repository in D:/gitt/.git/
```

git status -> git status is used to show the current state of the repository , including staged, unstaged, and untracked files.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

git log -> The git log command is used to view the commit history of a Git repository. It shows a list of commits made.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git log
fatal: your current branch 'master' does not have any commits yet
```

git config -> The git config command is used to set git settings, like your username, email, editor and preferences.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git config --global user.name "Chinmayee Rana"

USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git config --global user.email "chinmayeerana52@gmail.com"

USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git config --global user.name
Chinmayee Rana

USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git config --global user.email
chinmayeerana52@gmail.com
```

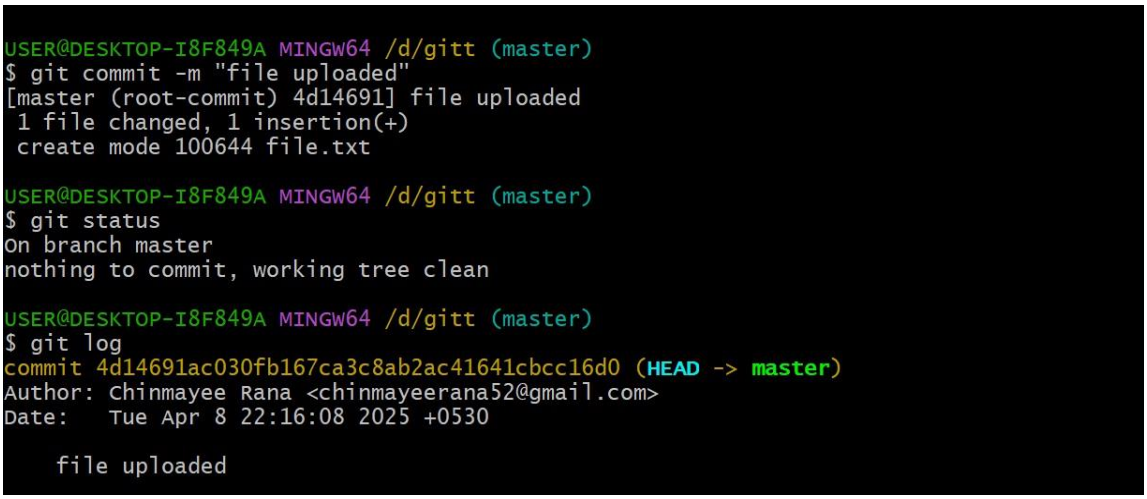
git add -> The git add command is used to stage changes in your working directory for the next commit.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ echo hello >> file.txt

USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git add file.txt
warning: in the working copy of 'file.txt', LF will be replaced by CRLF the next time Git touches it

USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git status
On branch master

No commits yet

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

git commit -> The git commit command is used to save your staged changes to the local git repository.

➔ *Example*



git checkout -> The git checkout command is used for switching brances or restoring files in your working directory.

➔ *Example*

git mv -> The git mv command is used to move or rename files in a Git repository, while also staging the change for commit.

➔ *Example*

git branch -> The git branch command is used to manage branches in a git repository.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git branch
* master

USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git branch "wipro"

USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git branch
* master
  wipro

USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git checkout wipro
D       file.txt
A       file1.txt
Switched to branch 'wipro'
```

git ls-files -> The git ls-files command is used to list all files tracked by git in your current repository.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt (wipro)
$ git ls-files
file1.txt
```

git merge -> The git merge is used to combine changes from one branch into another typically used to bring changes from a feature branch into the main branch.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /d/gitt (master)
$ git merge wipro
Already up to date.
```

git clone -> The git clone command is used to create a local copy of a remote git repository.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /c/GIT/clone (master)
$ git clone https://github.com/chinmayee11111/Wipro.git
Cloning into 'Wipro'...
remote: Enumerating objects: 9, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 9 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (9/9), done.
```

touch -> The touch command is used to create an empty file if it doesn't exist and update the access and modification timestamps of an existing file to the current time.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /c/GIT/clone1/Wipro/Chinmayee (main)
$ touch file.txt
```

git rm -> The git rm command is used to remove files from both the working directory and staging area in a git repository.

➔ *Example*

```
USER@DESKTOP-I8F849A MINGW64 /c/GIT/clone1/Wipro/Chinmayee (main)
$ git rm file.txt
rm 'Chinmayee/file.txt'
```