

## DAY - 9

Date: 21.07.2025

Name: Nancy M

### Git Commands

1. **Open Git bash shell and create a new folder “GitDemo” and initialize as git repository**

```
mkdir GitDemo  
cd GitDemo
```

2. **Git bash initializes the “GitDemo” repository.**

```
Git init
```

3. **Create a file “welcome.txt” and add content to the file (use echo or cat command)**

```
Echo "Welcome to Git" > welcome.txt
```

4. **To verify if the file “welcome.txt” is created, execute ls-al**

```
ls -al
```

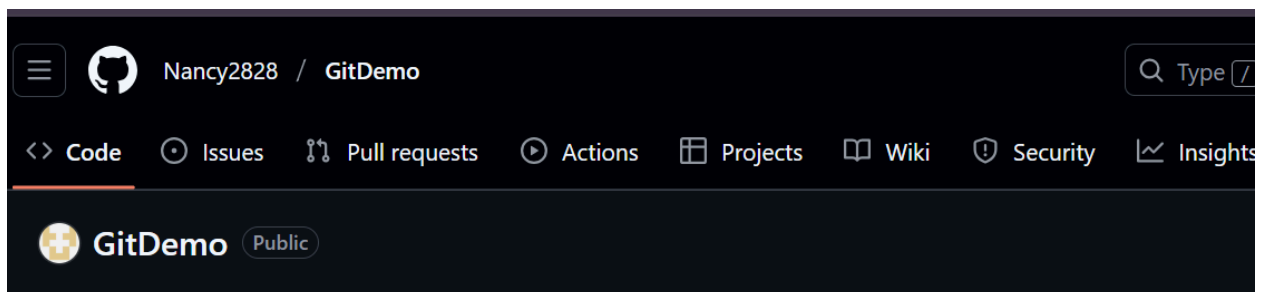
5. **To verify the content, execute the command cat filename**

```
cat welcome.txt
```

6. **Execute git status**

```
git status
```

7. **Signup with GitHub and create a remote repository “GitDemo”**



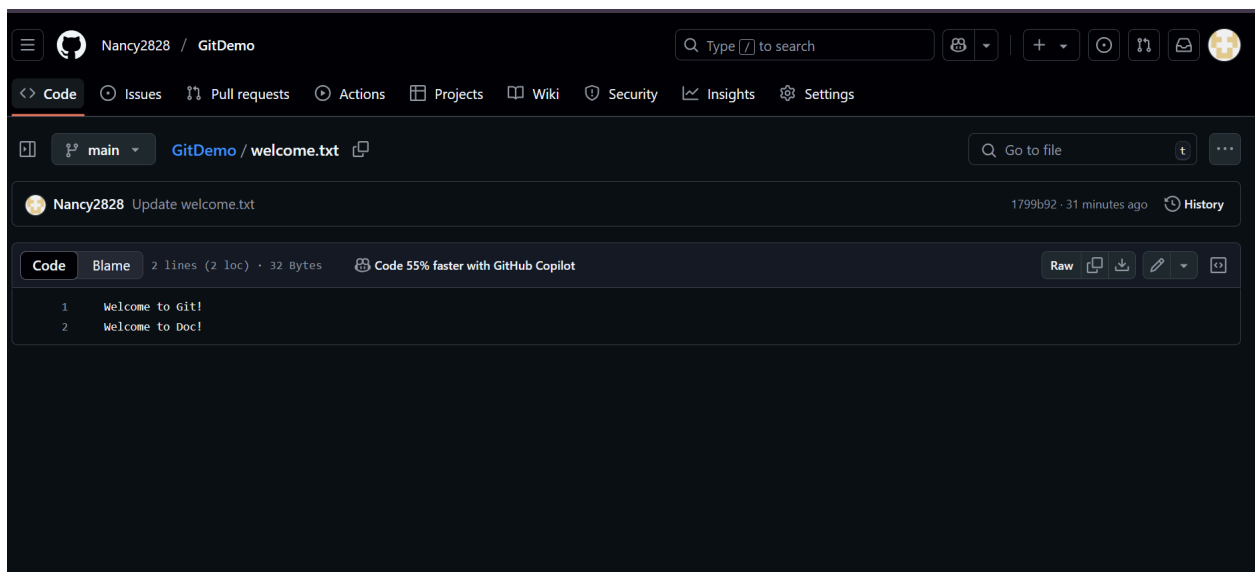
8. **Add remote origin**

```
git remote add origin https://github.com/Nancy2828/GitDemo.git
```

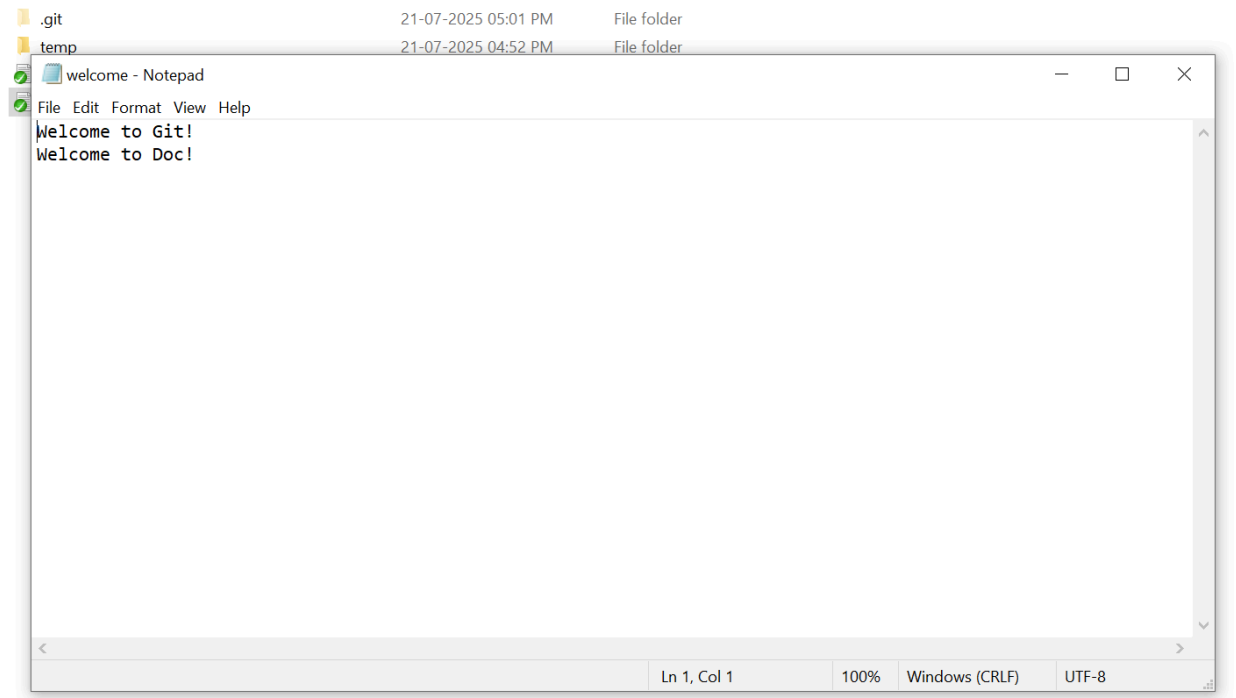
**9. Create files and push the local to remote repository.**

```
git add .  
git commit -m "sample"  
git branch -m main  
git push -u origin main
```

**10. Make changes in the remote and pull the changes to the local.**



```
git pull origin main
```



## 11. Create a .gitignore file to ignore contents of temp folder

```
mkdir temp  
echo "temp/" > .gitignore  
git add .  
git commit -m "add gitignore"  
git push -u origin main
```

```
MINGW64:/c/Users/nancy.m/GitDemo

nancy.m@PTPLL466 MINGW64 ~
$ mkdir GitDemo

nancy.m@PTPLL466 MINGW64 ~
$ cd GitDemo

nancy.m@PTPLL466 MINGW64 ~/GitDemo
$ git init
Initialized empty Git repository in C:/Users/nancy.m/GitDemo/.git/

nancy.m@PTPLL466 MINGW64 ~/GitDemo (master)
$ echo "Welcome to Git!" > welcome.txt

nancy.m@PTPLL466 MINGW64 ~/GitDemo (master)
$ ls -al
total 25
drwxr-xr-x 1 nancy.m 1049089  0 Jul 21 16:39 ./
drwxr-xr-x 1 nancy.m 1049089  0 Jul 21 16:35 ../
drwxr-xr-x 1 nancy.m 1049089  0 Jul 21 16:36 .git/
-rw-r--r-- 1 nancy.m 1049089 16 Jul 21 16:39 welcome.txt

nancy.m@PTPLL466 MINGW64 ~/GitDemo (master)
$ cat welcome.txt
Welcome to Git!
```

```
MINGW64:/c/Users/nancy.m/GitDemo

Welcome to Git!

nancy.m@PTPLL466 MINGW64 ~/GitDemo (master)
$ git status
On branch master

No commits yet

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

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

nancy.m@PTPLL466 MINGW64 ~/GitDemo (master)
$ git status
On branch master

No commits yet

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

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

```
MINGW64:/c/Users/nancy.m/GitDemo
nothing added to commit but untracked files present (use "git add" to track)

nancy.m@PTPLL466 MINGW64 ~/GitDemo (master)
$ git remote add origin https://github.com/Nancy2828/GitDemo.git

nancy.m@PTPLL466 MINGW64 ~/GitDemo (master)
$ git add .
warning: in the working copy of 'welcome.txt', LF will be replaced by CRLF the next time Git touches it

nancy.m@PTPLL466 MINGW64 ~/GitDemo (master)
$ git commit -m "sample"
[master (root-commit) 232f808] sample
1 file changed, 1 insertion(+)
create mode 100644 welcome.txt

nancy.m@PTPLL466 MINGW64 ~/GitDemo (master)
$ git branch -m main

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 232 bytes | 77.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
```

```
MINGW64:/c/Users/nancy.m/GitDemo

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ git pull origin main
From https://github.com/Nancy2828/GitDemo
* branch      main      -> FETCH_HEAD
Already up to date.

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ git pull origin main
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 919 bytes | 39.00 KiB/s, done.
From https://github.com/Nancy2828/GitDemo
* branch      main      -> FETCH_HEAD
  232f808..1799b92  main      -> origin/main
Updating 232f808..1799b92
Fast-forward
 welcome.txt | 1 +
1 file changed, 1 insertion(+)

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ mkdir temp

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
```

MINGW64:/c/Users/nancy.m/GitDemo

```
nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ mkdir temp

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ echo "temp/" > .gitignore

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ git commit -m "Add gitignore"
On branch main
Your branch is up to date with 'origin/main'.

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

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

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ git commit -m "Add gitignore"
```

MINGW64:/c/Users/nancy.m/GitDemo

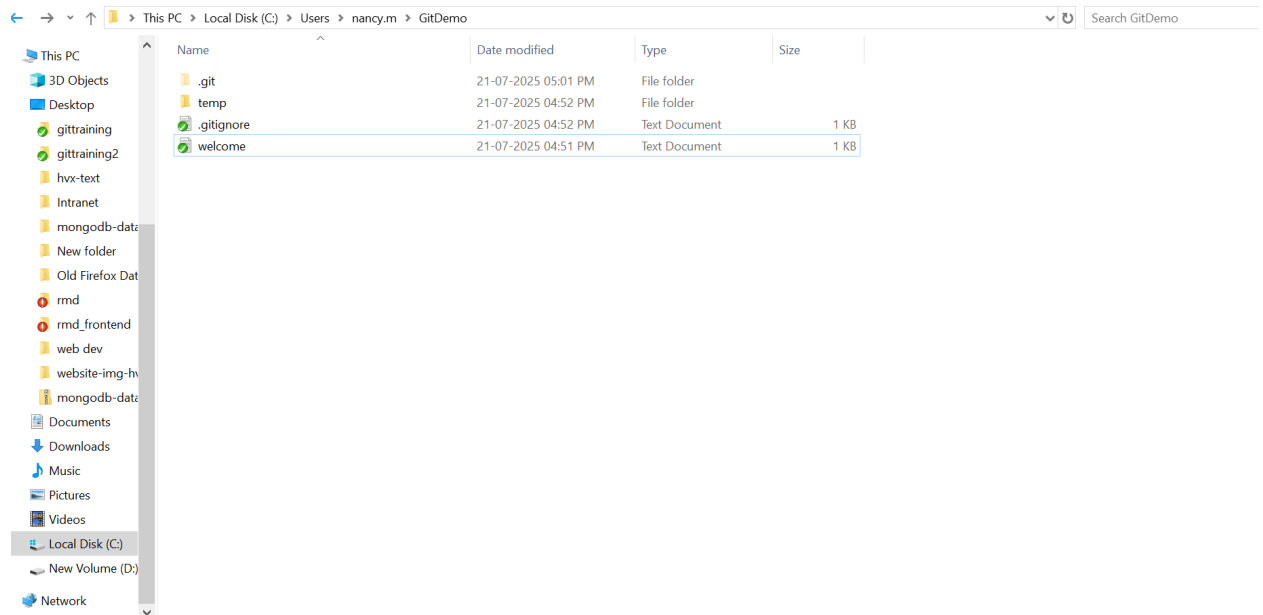
```
[main a527a68] Add gitignore
1 file changed, 1 insertion(+)
create mode 100644 .gitignore

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 289 bytes | 96.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Nancy2828/GitDemo.git
   1799b92..a527a68  main -> main
branch 'main' set up to track 'origin/main'.

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

nancy.m@PTPLL466 MINGW64 ~/GitDemo (main)
$
```



## Docker

### 1. What is Docker-compose.yaml file? Give sample docker-compose file.

The `docker-compose.yaml` file is a configuration file used by Docker Compose to define and run multi-container Docker applications.

Instead of running each container with long `docker run` commands, you define all your services (containers), volumes, networks, and configurations in a single `.yaml` file and run them with one command

```
version: '3.8'
```

```
services:
```

```
  app:
```

```
    image: node:18
```

```
    volumes:
```

```
      - ./app
```

```
    ports:
```

```
      - 3000:3000
```

command: npm start

depends\_on:

- mongo

mongo:

image: mongo

volumes:

- mongo-data:/data/db

volumes:

mongo-data:

### **What is Docker hub?**

Docker Hub is a cloud-based registry where you can:

- Find, share, and store Docker container images
- Pull images like nginx, mysql, node, etc.
- Push your own Docker images for others or for use in your projects