

Day 9

Dharunya S

Git Commands

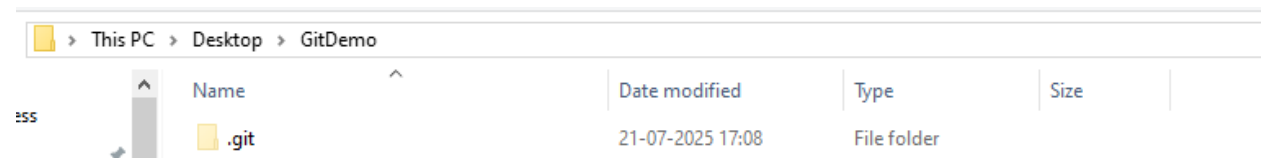
1. Open Git bash shell and create a new folder “GitDemo” and initialize as git repository

```
mkdir GitDemo  
cd GitDemo
```

2. Git bash initializes the “GitDemo” repository.

```
git init
```

Initialized empty Git repository in C:/Users/dharunya.s/Desktop/GitDemo/.git/



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

```
$ echo "Welcome to Git Demo Project" > welcome.txt
```

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

```
ls -al
```

```
-rw-r--r-- 1 dharunya.s 1049089 29 Jul 21 17:06 welcome.txt
```

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

```
cat welcome.txt
```

Welcome to Git Demo Project

6. Execute git status

```
$ git status
```

On branch master

No commits yet

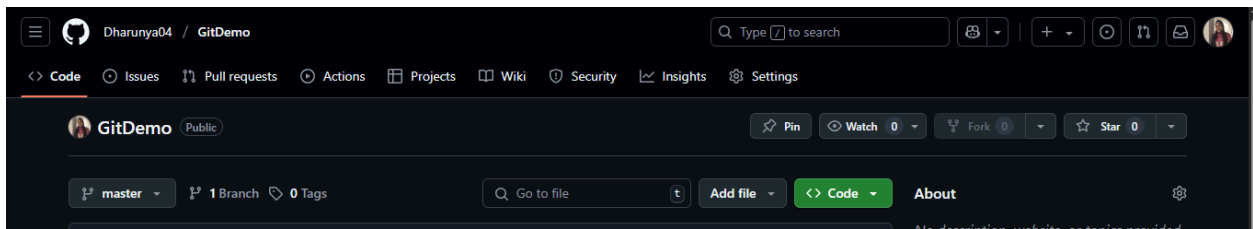
Untracked files:

(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)

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



8. Add remote origin

```
$ git remote add origin https://github.com/Dharunya04/GitDemo.git
```

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

```
dharunya.s@PTPLL567 MINGW64 ~/Desktop/GitDemo (master)
```

```
$ git commit -m "Initial commit with welcome.txt"
```

```
[master (root-commit) a927d00] Initial commit with welcome.txt
```

```
1 file changed, 1 insertion(+)
```

```
create mode 100644 welcome.txt
```

```
dharunya.s@PTPLL567 MINGW64 ~/Desktop/GitDemo (master)
```

```
$ git push -u origin master
```

```
Enumerating objects: 3, done.
```

```
Counting objects: 100% (3/3), done.
```

```
Writing objects: 100% (3/3), 258 bytes | 86.00 KiB/s, done.
```

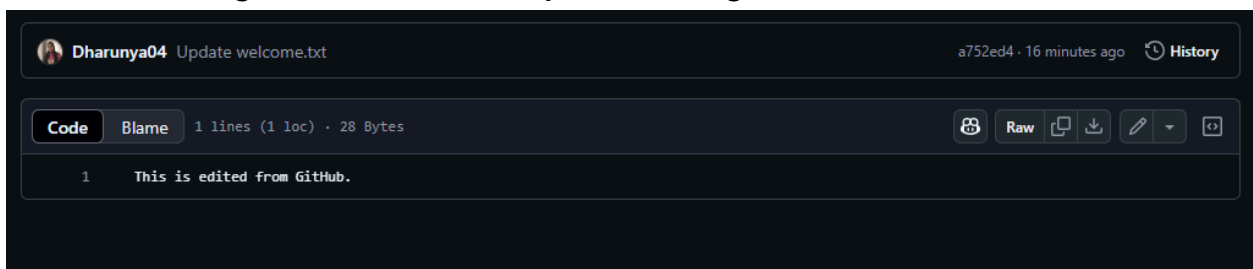
```
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
```

```
To https://github.com/Dharunya04/GitDemo.git
```

```
* [new branch]      master -> master
```

```
branch 'master' set up to track 'origin/master'.
```

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



```
$ git pull origin master
```

```
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), 913 bytes | 38.00 KiB/s, done.
```

```
From https://github.com/Dharunya04/GitDemo
```

```
* branch      master    -> FETCH_HEAD
a927d00..a752ed4 master  -> origin/master
Updating a927d00..a752ed4
Fast-forward
welcome.txt | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)
```

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

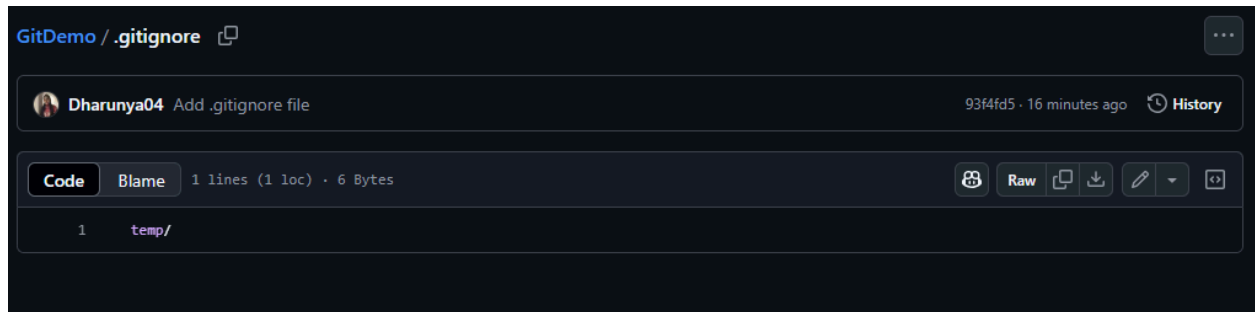
```
$ mkdir temp
```

```
dharunya.s@PTPLL567 MINGW64 ~/Desktop/GitDemo (master)
$ echo "temp/" > .gitignore
```

```
dharunya.s@PTPLL567 MINGW64 ~/Desktop/GitDemo (master)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git
touches it
```

```
dharunya.s@PTPLL567 MINGW64 ~/Desktop/GitDemo (master)
$ git commit -m "Add .gitignore file"
[master 93f4fd5] Add .gitignore file
1 file changed, 1 insertion(+)
create mode 100644 .gitignore
```

```
dharunya.s@PTPLL567 MINGW64 ~/Desktop/GitDemo (master)
$ git push
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), 293 bytes | 97.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Dharunya04/GitDemo.git
a752ed4..93f4fd5 master -> master
```



Docker

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

`docker-compose.yaml` (or `docker-compose.yml`) is a configuration file used by Docker Compose—a tool that helps manage multi-container Docker applications. Instead of running `docker run` commands separately for each container, you define services, networks, and volumes in this file, and then launch them all together.

```
version: '3'
```

```
services:
```

```
  web:
```

```
    image: nginx:latest
```

```
    ports:
```

```
      - "8080:80"
```

2. What is Docker hub?

Docker Hub is a cloud-based registry that works like an app store for Docker containers. It lets you find and pull ready-to-use images like MySQL, Node.js, or Python. You can also upload and share your own container images.