

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

```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo
$ git init
Initialized empty Git repository in C:/Users/subramaniya.k/Desktop/GitDemo/.git/
```

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

```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ echo "Welcome to GitDemo repository" > welcome.txt
```

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

```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ cat > welcome.txt
Added the text using cat command!
```

4. [To](#) verify if the file “welcome.txt” is created, execute ls-al

```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ ls -al
total 13
drwxr-xr-x 1 karthikeyan.r 1049089  0 Jul 21 16:44 ./
drwxr-xr-x 1 karthikeyan.r 1049089  0 Jul 21 16:39 ../
drwxr-xr-x 1 karthikeyan.r 1049089  0 Jul 21 16:42 .git/
-rw-r--r-- 1 karthikeyan.r 1049089 34 Jul 21 16:45 welcome.txt
```

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

```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ cat welcome.txt
Added the text using cat command!
```

6. Execute `git status`

```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/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
```

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

The screenshot shows the GitHub interface for a public repository named 'GitDemo' by user 'karthikeyan-r-payoda'. The repository has 1 branch (master) and 0 tags. The commit history shows three commits: the first commit 'Add .gitignore to ignore temp folder' by 'karthikeyan-r-payoda' 2 minutes ago, and two subsequent commits 'Update welcome.txt' also by 'karthikeyan-r-payoda' 2 and 6 minutes ago. The repository has 0 stars, 0 watching, and 0 forks. The 'About' section is empty, stating 'No description, website, or topics provided.'

## 8. Add remote origin




```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ git remote add origin https://github.com/karthikeyan-r-payoda/GitDemo.git
```

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

```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ git add .
warning: in the working copy of 'welcome.txt', LF will be replaced by CRLF the next time Git touches it

karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ git commit -m "Initial Commit"
[master (root-commit) efe5006] Initial Commit
1 file changed, 1 insertion(+)
create mode 100644 welcome.txt

karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 257 bytes | 85.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/karthikeyan-r-payoda/GitDemo.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

Name	Date modified	Type	Size
 .git	21-07-2025 17:10	File folder	
 .gitignore	21-07-2025 17:09	Git Ignore Source ...	1 KB
 welcome	21-07-2025 17:07	Text Document	1 KB

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

```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ git pull origin master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 956 bytes | 39.00 KiB/s, done.
From https://github.com/karthikeyan-r-payoda/GitDemo
* branch          master      -> FETCH_HEAD
   efe5006..068763f master     -> origin/master
Updating efe5006..068763f
Fast-forward
 welcome.txt | 2 ++
 1 file changed, 2 insertions(+)
```

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

```
karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ echo "temp/" > .gitignore

karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the ne
xt time Git touches it

karthikeyan.r@PTPLL593 MINGW64 ~/Desktop/GitDemo (master)
$ git commit -m "Add .gitignore to ignore temp folder"
[master 9c20682] Add .gitignore to ignore temp folder
 1 file changed, 1 insertion(+)
 create mode 100644 .gitignore

karthikeyan.r@PTPLL593 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), 306 bytes | 153.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/karthikeyan-r-payoda/GitDemo.git
 068763f..9c20682 master -> master
```

## DOCKER

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

Definition:

A `docker-compose.yaml` file is used to define and run multi-container Docker applications. It describes how to configure your app's services, networks, and volumes in a single file, making it easier to manage and deploy complex environments.

```
version: '3'
```

```
services:
```

```
  web:
```

```
    image: nginx:latest
```

```
    ports:
```

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

```
  app:
```

```
    image: python:3.10
```

```
    volumes:
```

```
      - ./app:/usr/src/app
```

```
    working_dir: /usr/src/app
```

```
    command: python app.py
```

## 2 . What is Docker hub?

Docker Hub is a cloud-based registry for Docker images.

- It allows you to publish, share, store, and download container images created by yourself or the community.
- You can pull pre-made official images (like Ubuntu, NGINX, MySQL).
- It streamlines development and deployment by serving as the central image repository for Docker containers.