

## 1. Open Github on desktop and Create a new repository

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner \*



mushruf12 ▾



Repository name \*

Demo



Great repository names are short and memorable. Need inspiration? How about [vigilant-doodle?](#)

Description (optional)



**Public**

Anyone on the internet can see this repository. You choose who can commit.



**Private**

You choose who can see and commit to this repository.

**Initialize this repository with:**

Skip this step if you're importing an existing repository.

☐ **Add a README file**

This is where you can write a long description for your project. [Learn more.](#)

**Add .gitignore**

Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: **None** ▾

**Choose a license**

A license tells others what they can and can't do with your code. [Learn more.](#)




License: **None** ▾

You are creating a public repository in your personal account.

Create repository

## 2. Click on Create repository

**Quick setup — if you've done this kind of thing before**

 Set up in Desktop or  HTTPS  SSH

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

---

**...or create a new repository on the command line**

```
echo "# Demo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/mushruf12/Demo.git
git push -u origin main
```

---

**...or push an existing repository from the command line**

```
git remote add origin https://github.com/mushruf12/Demo.git
git branch -M main
git push -u origin main
```

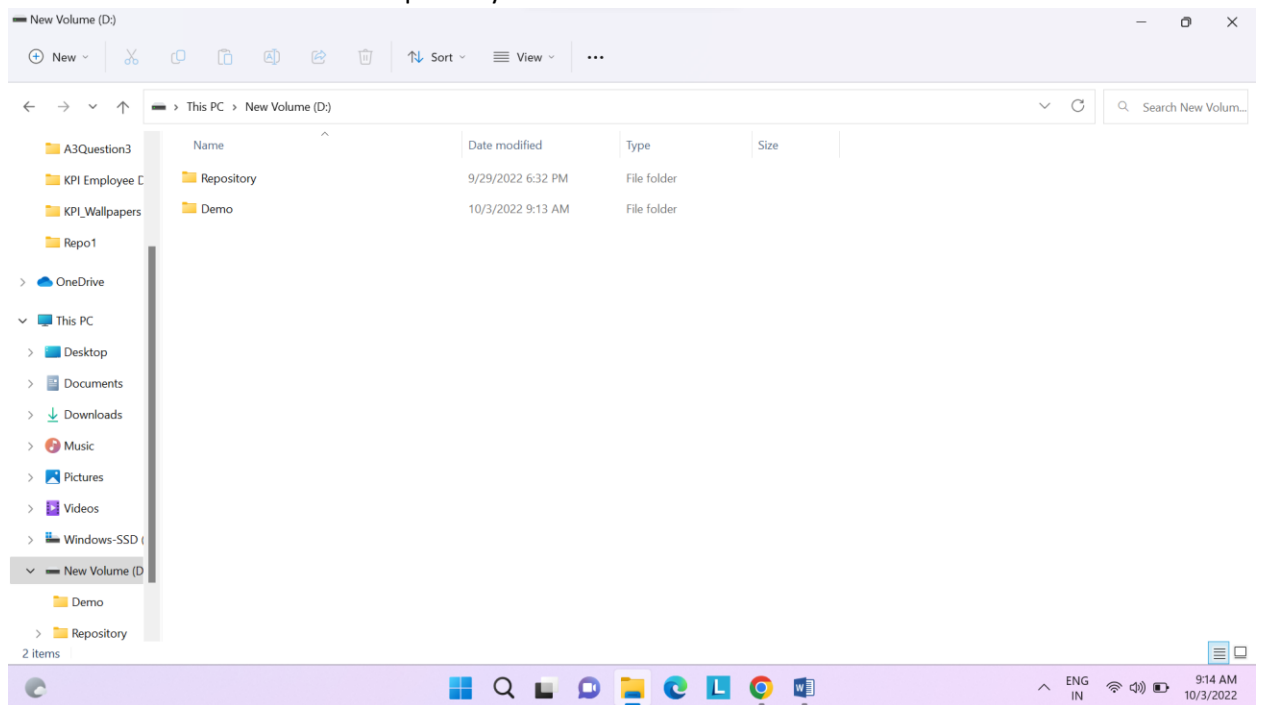
---

**...or import code from another repository**

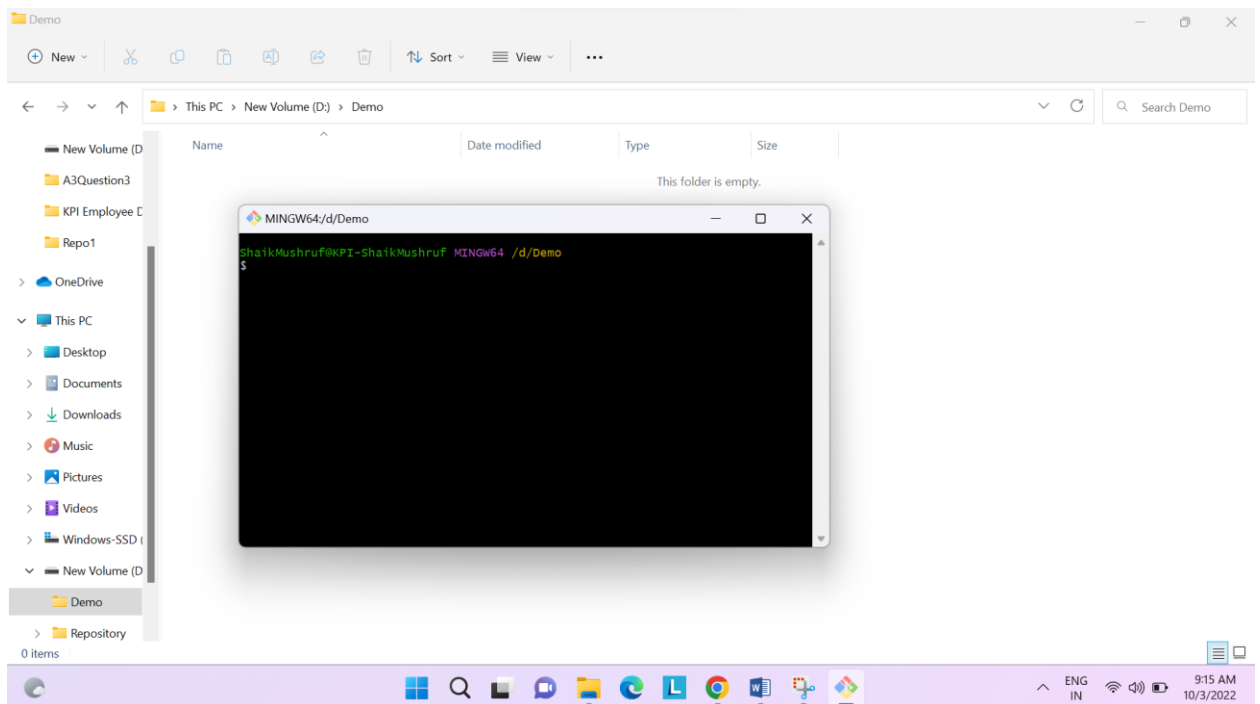
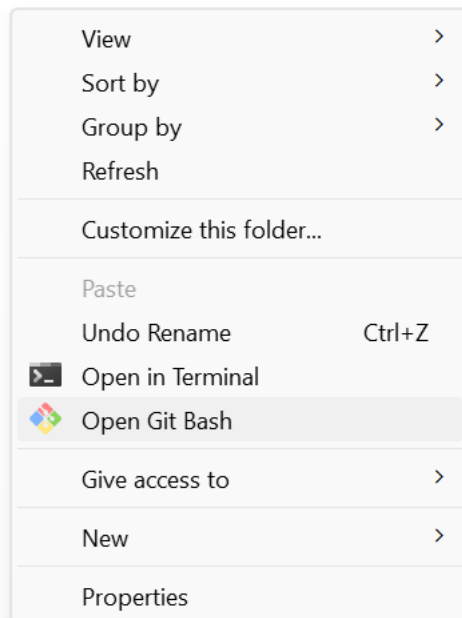
You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

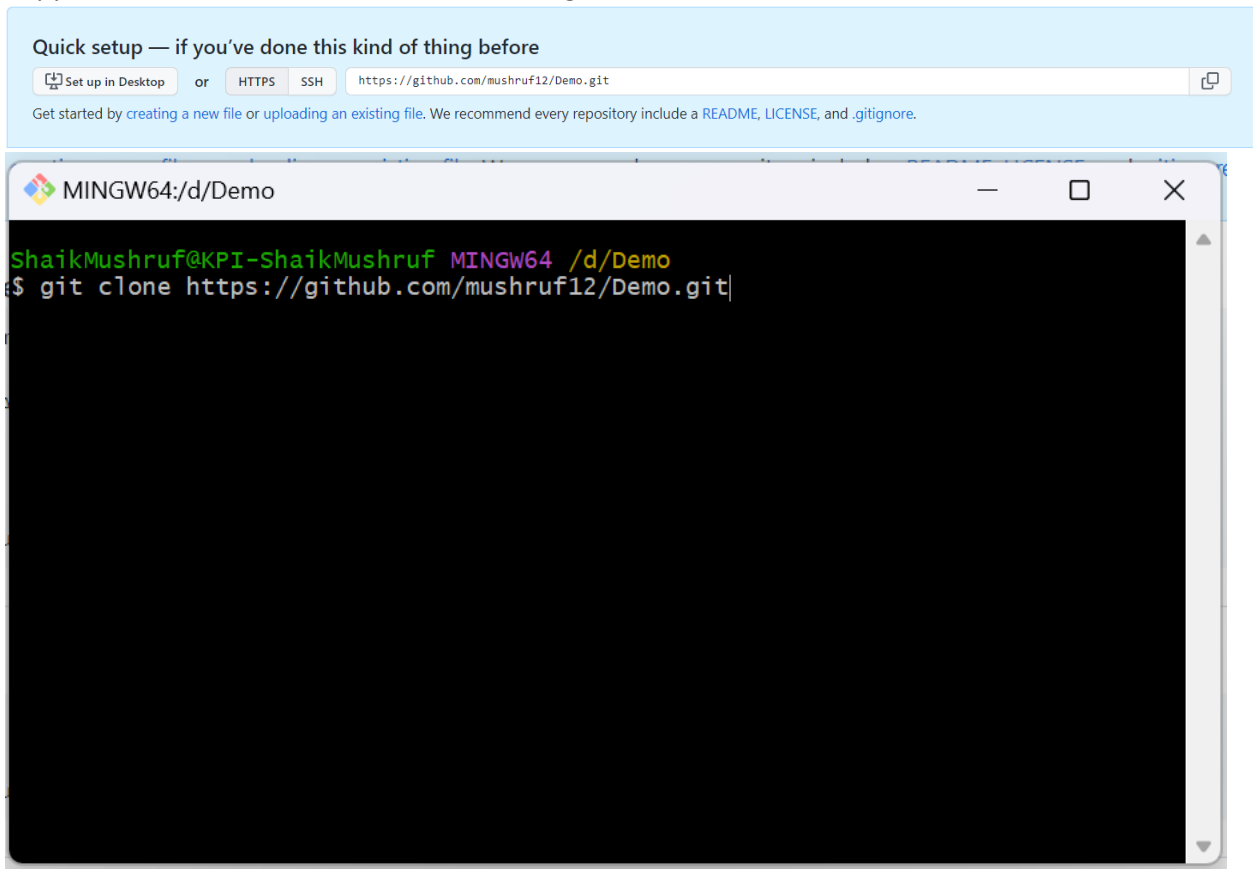
## 3. Create a new folder with same repository name



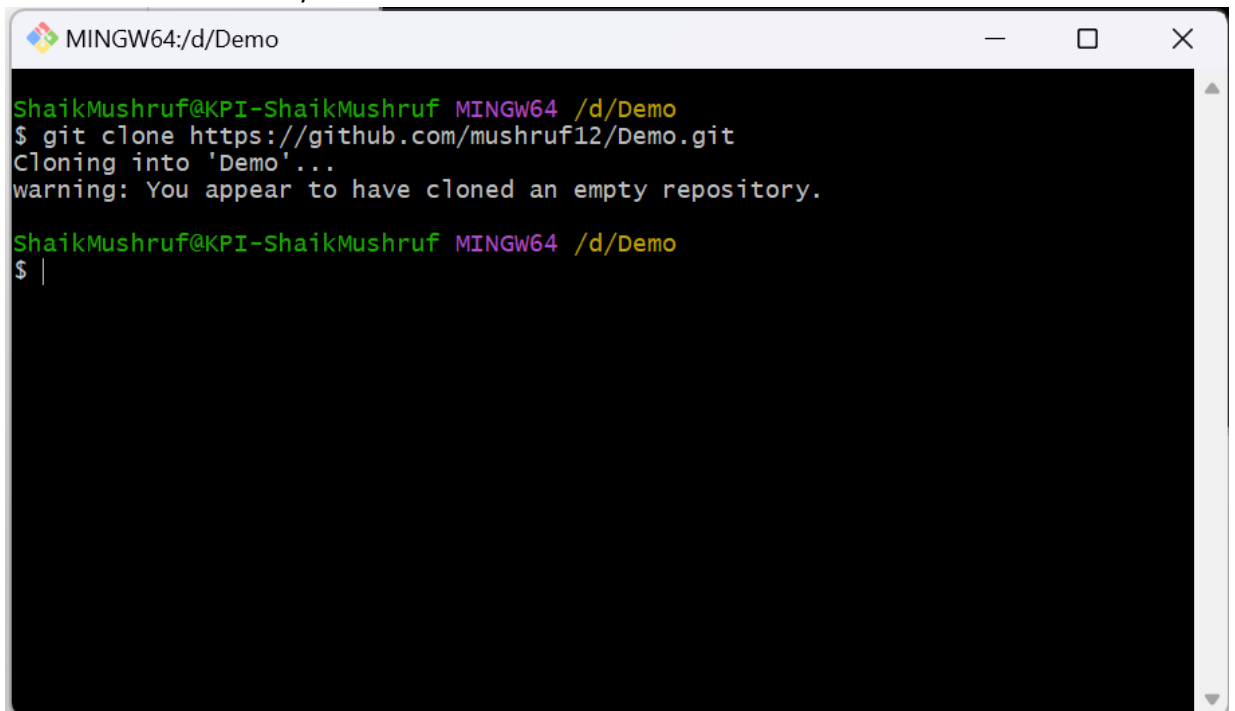
4. Open the folder and right click to open git bash



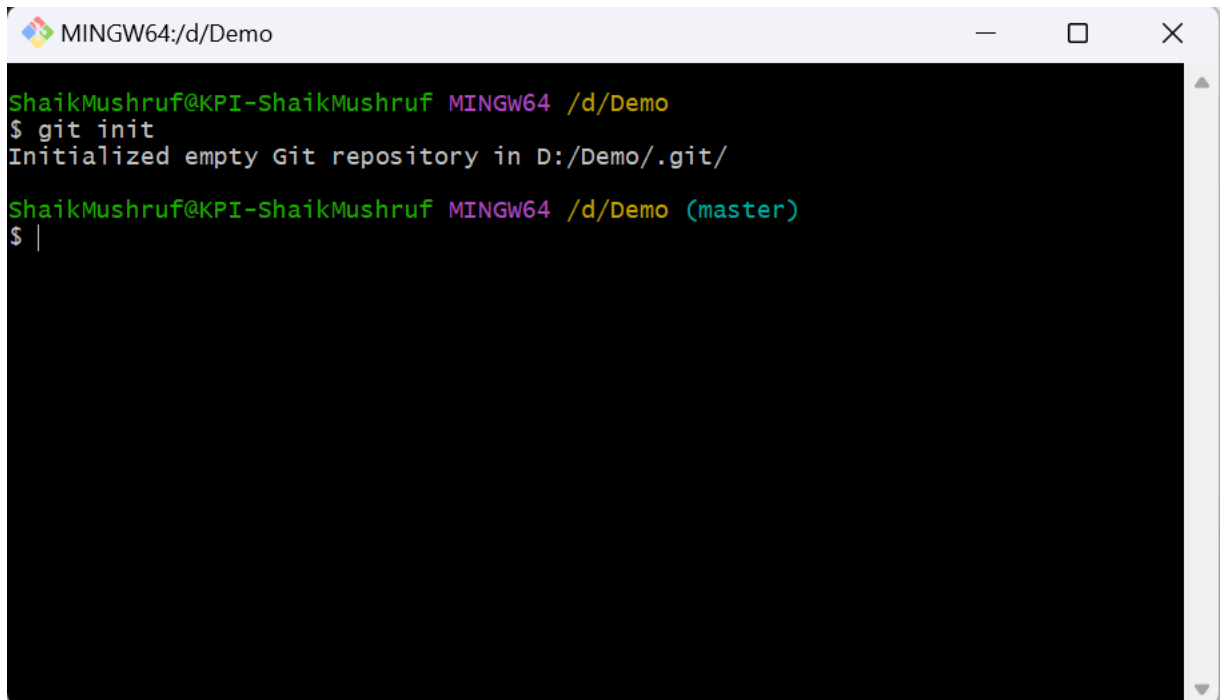
5. Copy the HTTPS Code and enter the command `git clone <HTTPS Code>`



6. Git Cloned to Successfully



7. Git init Command

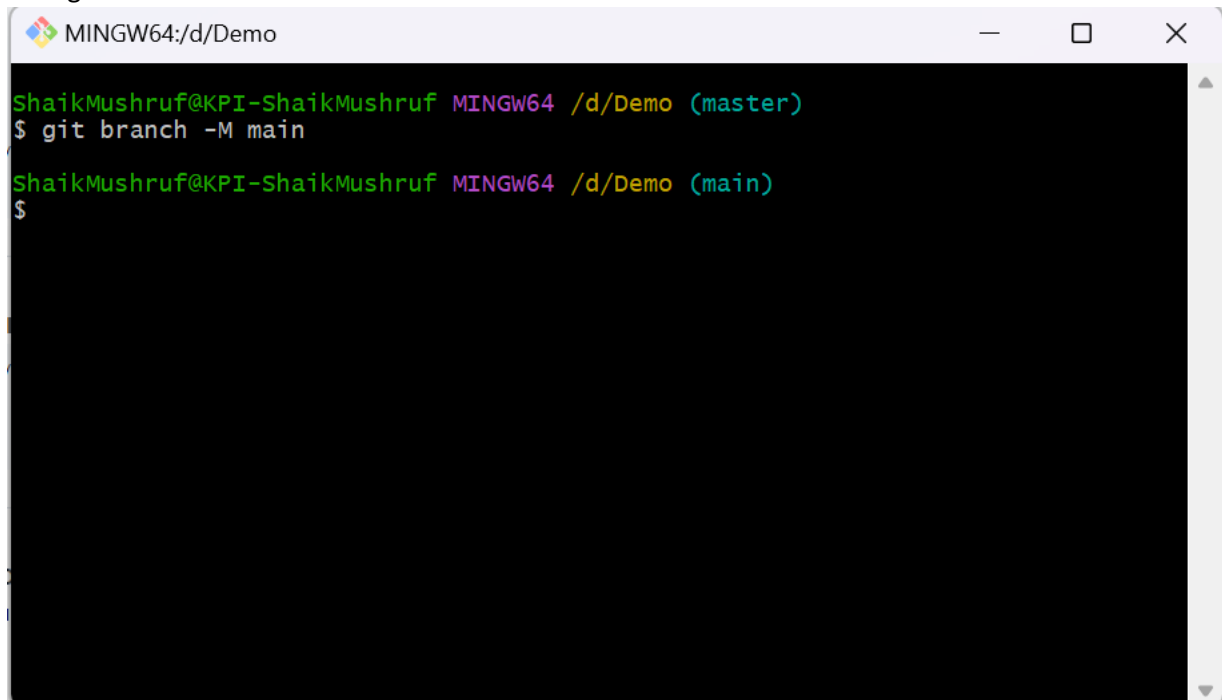


```
MINGW64:/d/Demo

ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo
$ git init
Initialized empty Git repository in D:/Demo/.git/

ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo (master)
$ |
```

8. Change master to main

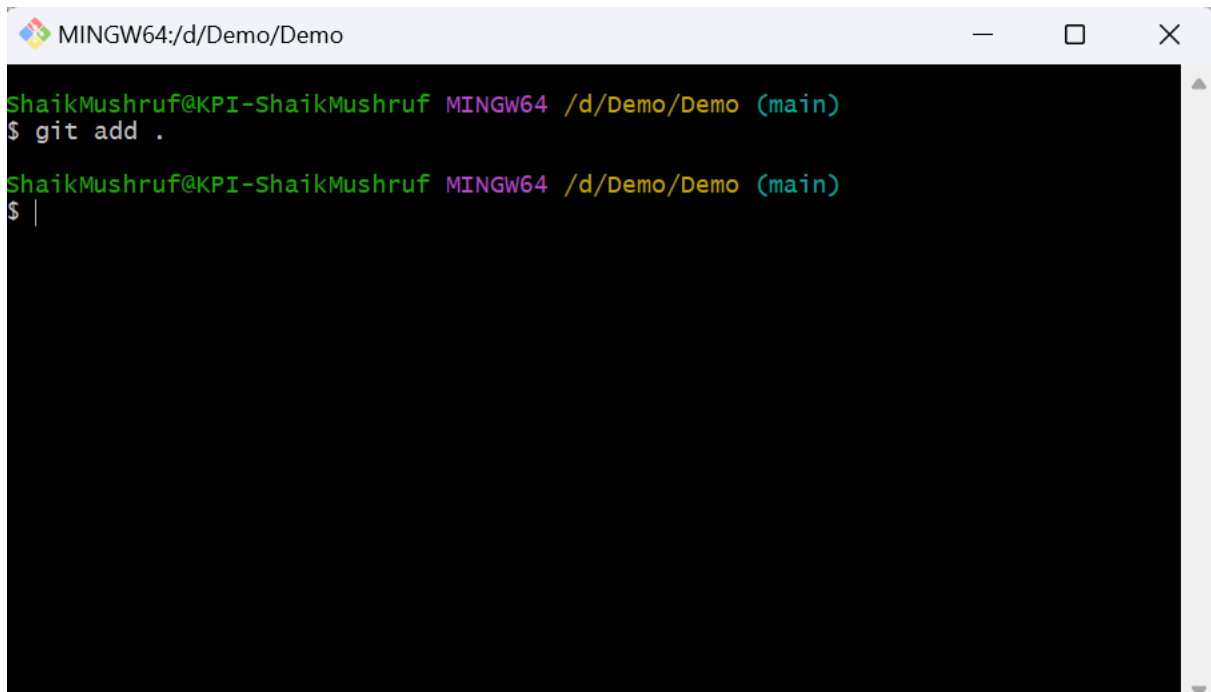


```
MINGW64:/d/Demo

ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo (master)
$ git branch -M main

ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo (main)
$
```

9. Git add

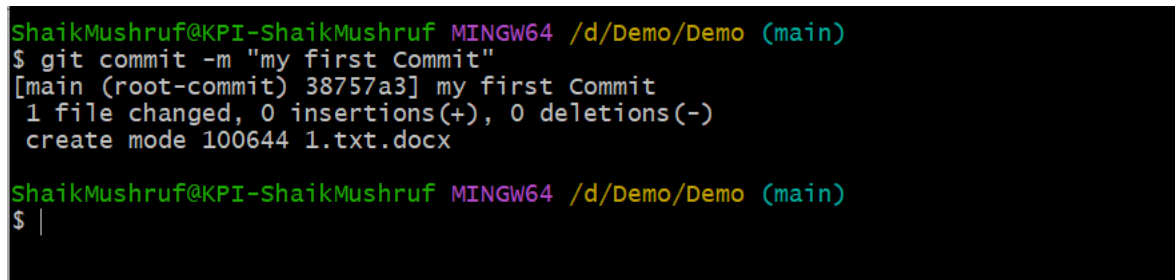


```
MINGW64:/d/Demo/Demo

ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo/Demo (main)
$ git add .

ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo/Demo (main)
$ |
```

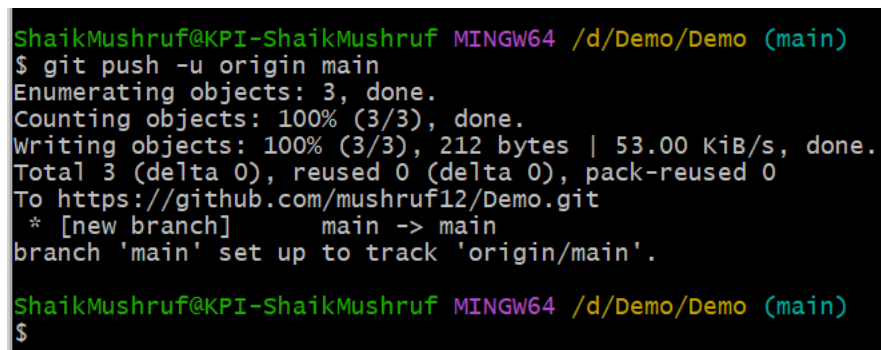
10. Git commit



```
ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo/Demo (main)
$ git commit -m "my first Commit"
[main (root-commit) 38757a3] my first Commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 1.txt.docx

ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo/Demo (main)
$ |
```

11. Git push



```
ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo/Demo (main)
$ git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 212 bytes | 53.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/mushruf12/Demo.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

ShaikMushruf@KPI-ShaikMushruf MINGW64 /d/Demo/Demo (main)
$
```

## 12. Reload the tab

