Full Stack Web Development Internship

Setting up Git Bash and Pushing files to a GitHub Repository

edureka!

Setting up Git Bash and Pushing Files to a GitHub Repository

1. First, you need to download and install gitbash app from the link https://git-scm.com/downloads. (Ignore this step if you have already installed gitbash onto you system. If you have not installed, you can refer to the **Git Installation Guide**).

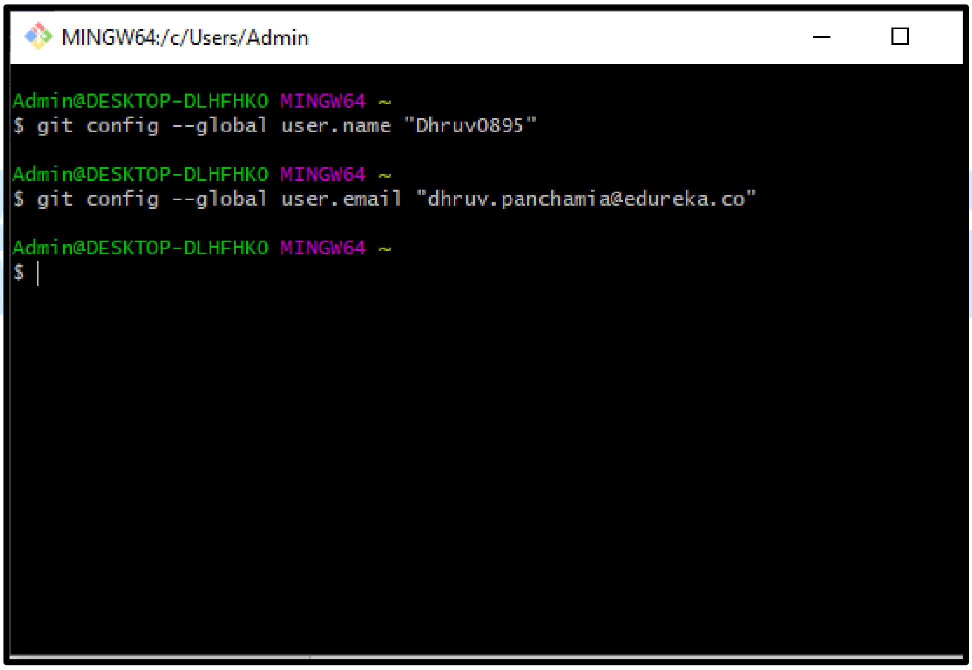
2. After finishing installation, open the '**Git Bash App'**, and in the terminal type the following commands in sequence to configure gitbash with your GitHub **username** and **email address**.

```
git config - -global user.name "GitHub_user_name_here"
```

'Press enter'

```
git config - -global user.email "GitHub_email_address_here"
```

'Press enter'

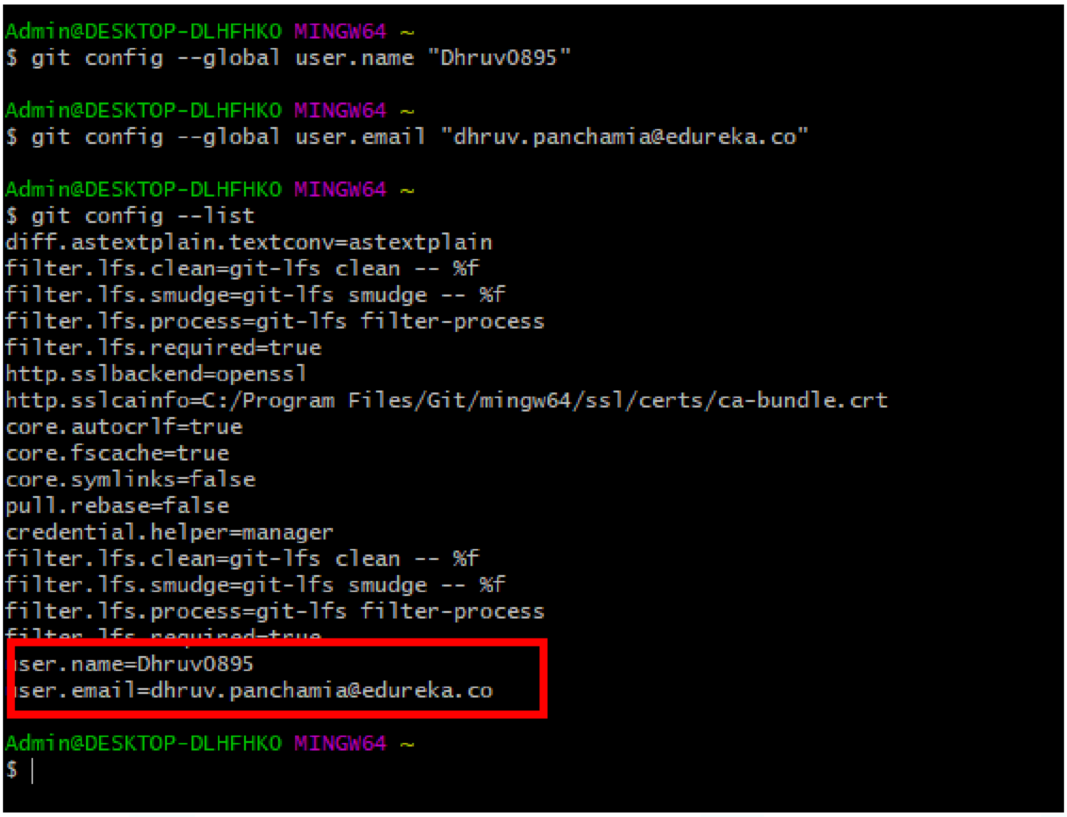


```
Admin@DESKTOP-DLHFHK0 MINGW64 ~
$ git config --global user.name "Dhruv0895"

Admin@DESKTOP-DLHFHK0 MINGW64 ~
$ git config --global user.email "dhruv.panchamia@edureka.co"

Admin@DESKTOP-DLHFHK0 MINGW64 ~
$ |
```

3. You can check whether you have configured with the correct details by typing the below command:

```
git config -list
```

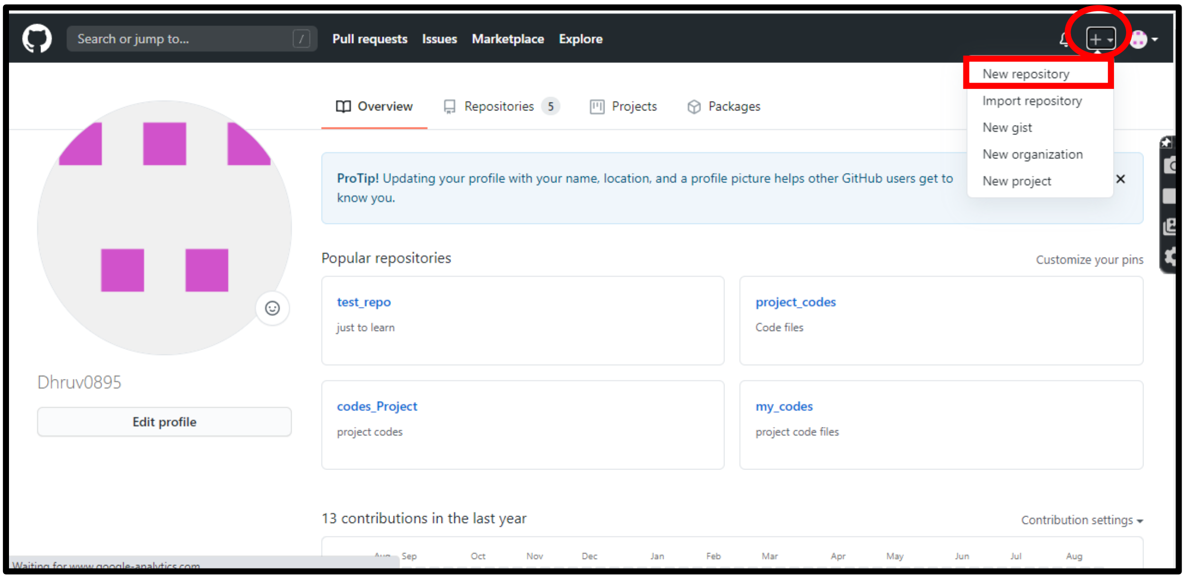
'Press Enter'



```
Admin@DESKTOP-DLHFHKO MINGW64 ~
$ git config --global user.name "Dhruv0895"

Admin@DESKTOP-DLHFHKO MINGW64 ~
$ git config --global user.email "dhruv.panchamia@edureka.co"

Admin@DESKTOP-DLHFHKO MINGW64 ~
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
ser.name=Dhruv0895
ser.email=dhruv.panchamia@edureka.co

Admin@DESKTOP-DLHFHKO MINGW64 ~
$ |
```

You can now close the terminal.

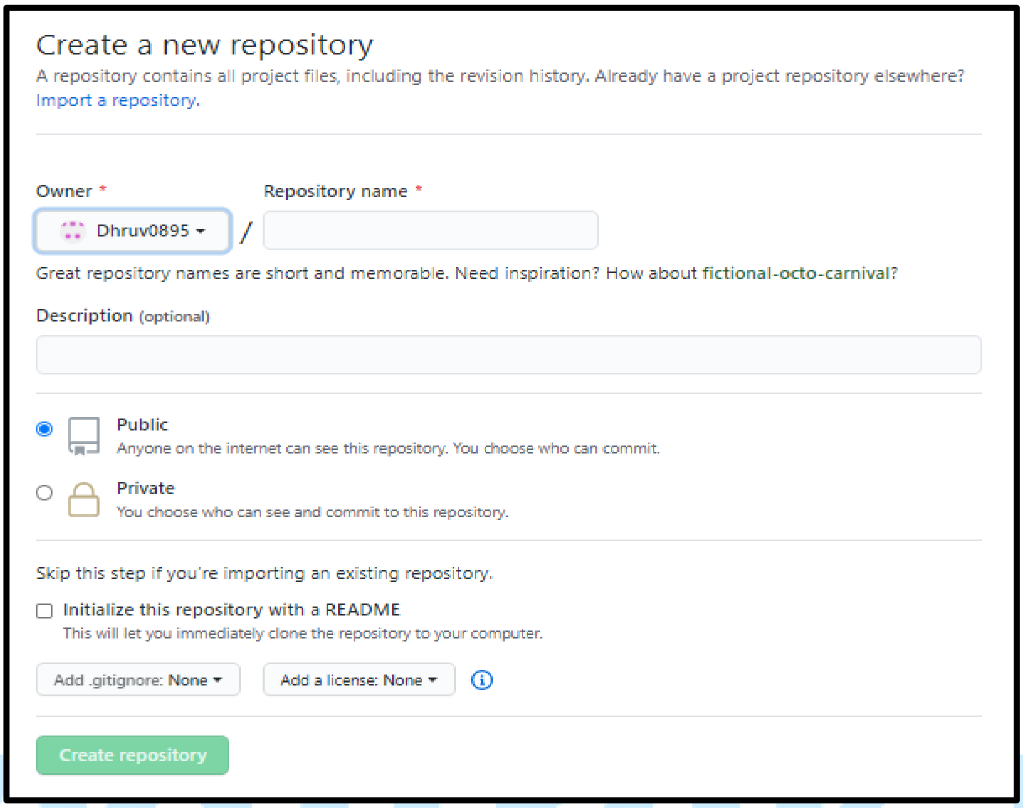
4. Next, let us learn how to create a public repository. Log in to GitHub and then click on the **'+'** symbol on the top right corner of the screen. Then click on '**New Repository**'.



ProTip! Updating your profile with your name, location, and a profile picture helps other GitHub users get to know you.

13 contributions in the last year

5. Enter a **repository name** of your choice, an optional **Description,** choose **Public** option, and then click on **Create 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 * / Repository name *

Dhruv0895 ▾ /

Great repository names are short and memorable. Need inspiration? How about fictional-octo-carnival?

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.

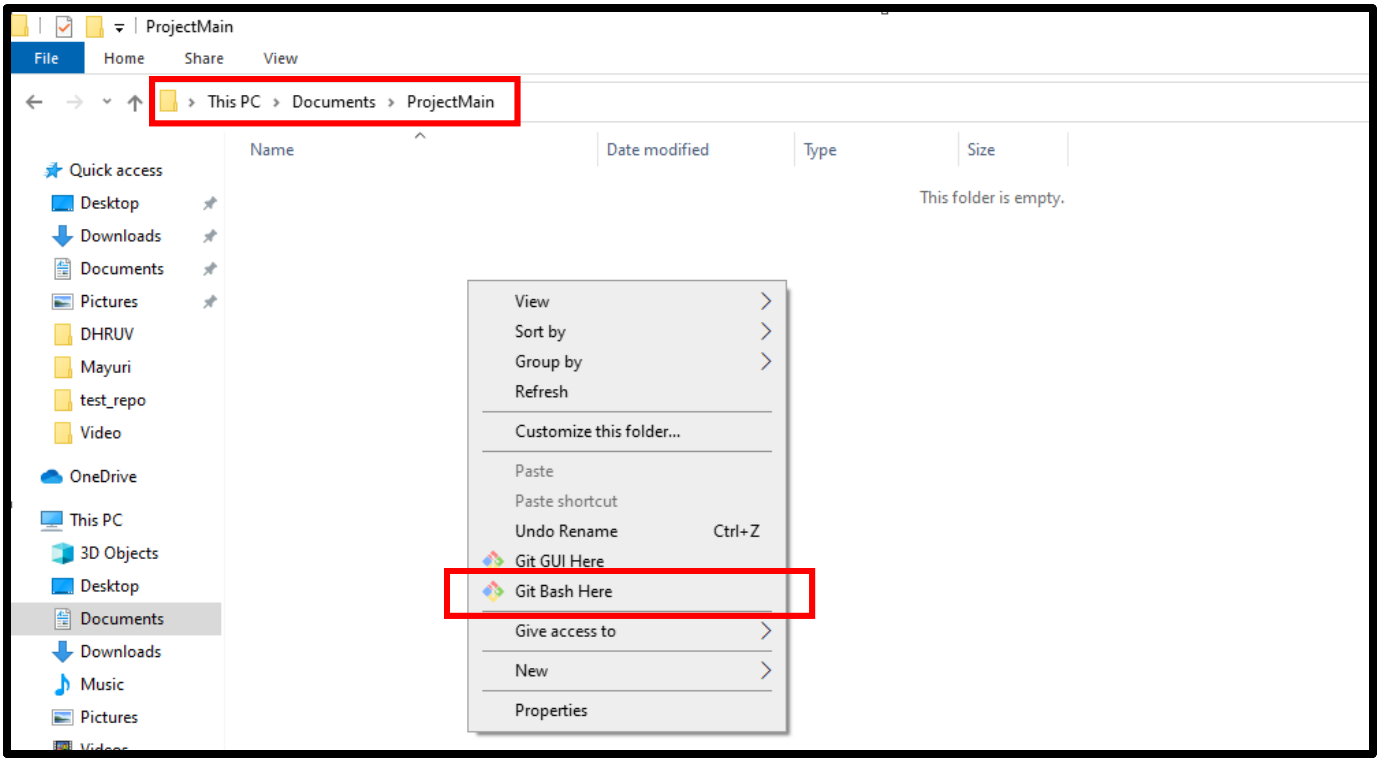
Skip this step if you're importing an existing repository.

□ Initialize this repository with a README
This will let you immediately clone the repository to your computer.

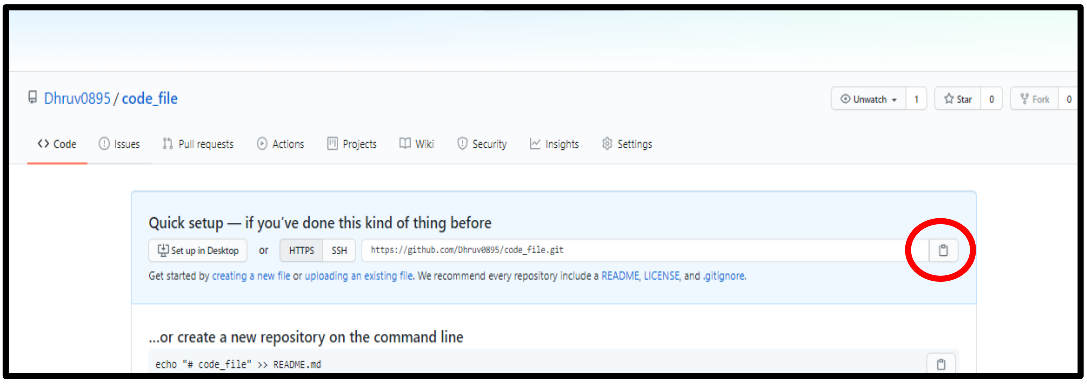
Add .gitignore: None ▾ Add a license: None ▾

Create repository

6. In a location of your choice in your PC, create a new folder where you will be adding your codes. (I have created a folder named "**ProjectMain**" inside "**Documents**" folder for demo purpose. Open the folder and right click anywhere on the screen. Then click on "**Git Bash Here**" to open the terminal.



7. Open the newly created repository from your GitHub account and click on the ‘copy’ symbol to copy the HTTP link (which will be used while cloning).



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

```
https://github.com/Dhruv0895/code_file.git
```

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

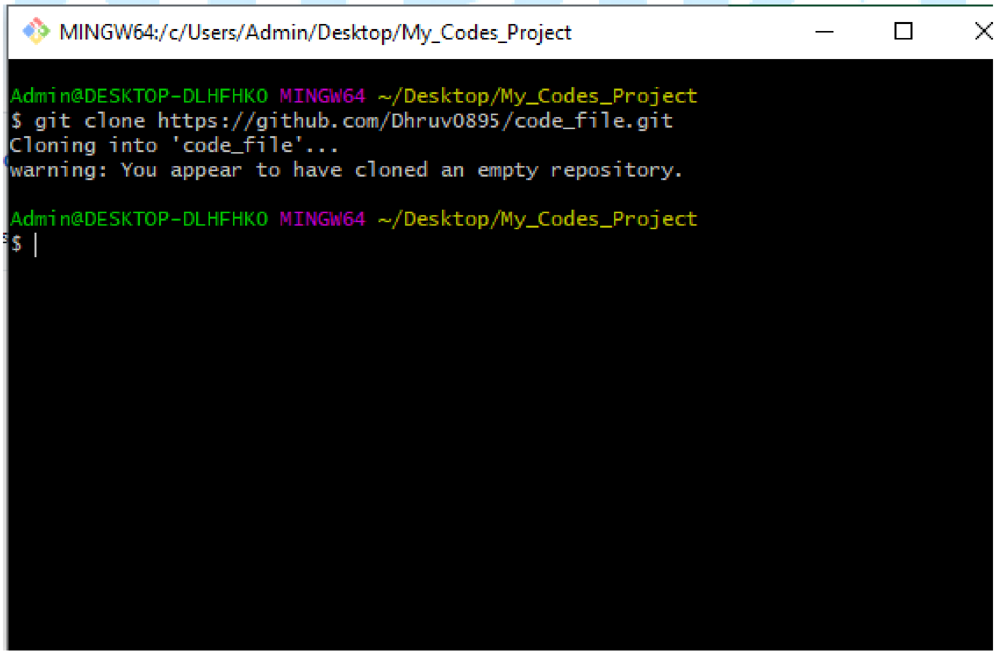
```
echo "# code_file" >> README.md
```

8. Open the terminal and type the command below:

`git clone (right click and *Paste* the copied link)`

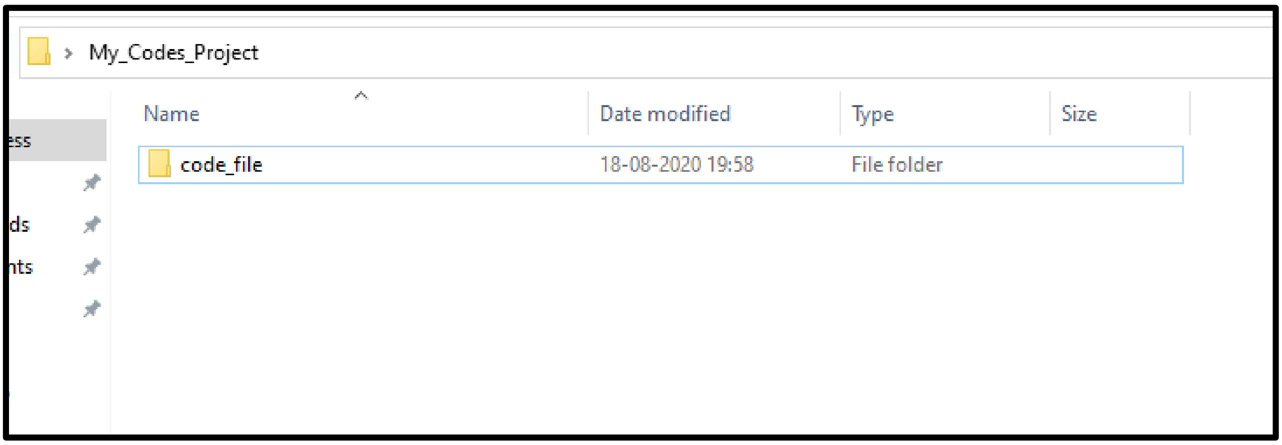
`‘Press Enter’`

(For demo purpose, I have used the link of my own repository, but the process will be the same).



```
Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project
$ git clone https://github.com/Dhruv0895/code_file.git
Cloning into 'code_file'...
warning: You appear to have cloned an empty repository.

Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project
$ |
```

9. You fill now find a folder with the same name as the name of the repository, inside the folder you have created (**code_file** in this case).

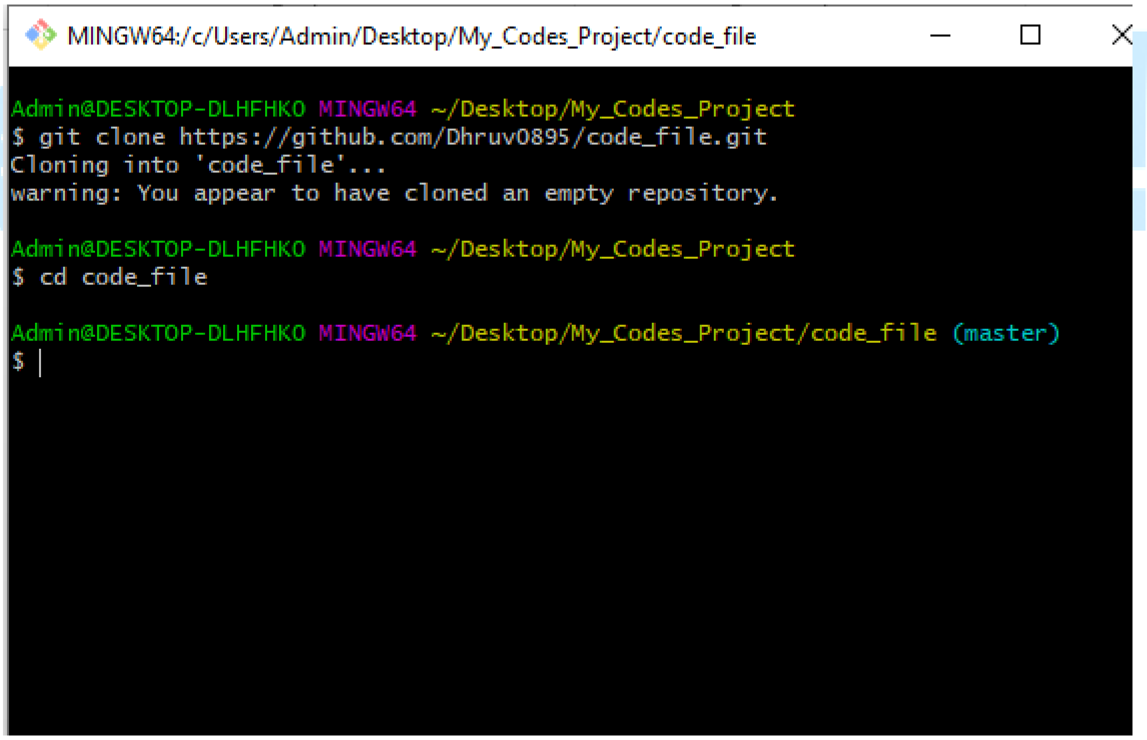


Name	Date modified	Type	Size
code_file	18-08-2020 19:58	File folder	

10. Change the directory in the terminal to get inside **code_file** folder by typing the below command:

```
cd code_file
```

'Press Enter'

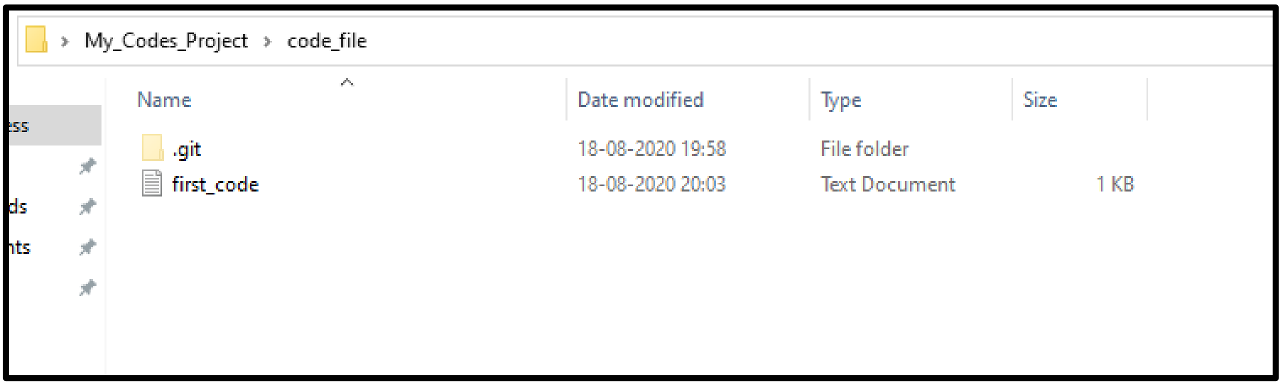


MINGW64:/c/Users/Admin/Desktop/My_Codes_Project/code_file

```
Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project
$ git clone https://github.com/Dhruv0895/code_file.git
Cloning into 'code_file'...
warning: You appear to have cloned an empty repository.

Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project
$ cd code_file

Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project/code_file (master)
$ |
```

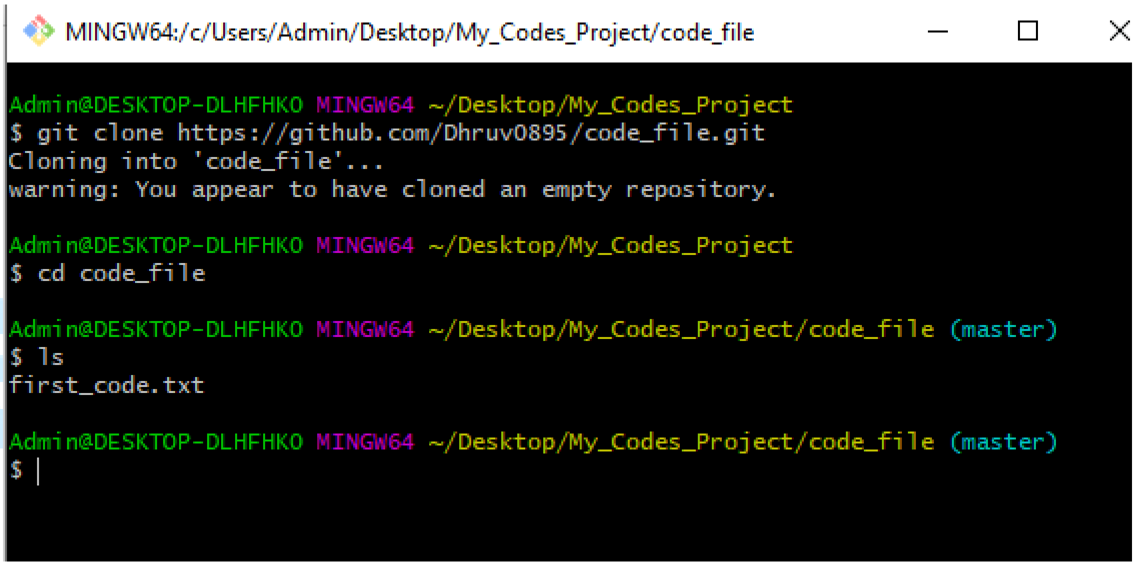
11. Open any text editor where you have written your code and save it with any suitable name inside the same folder 'code_file' (**Note**: The name of the folder will be different in your case. It will be the same as the name of the repository).

(I have used Notepad for demo purpose, and saved the file with the name '**first_code**' inside '**code_file**' folder).



Name	Date modified	Type	Size
.git	18-08-2020 19:58	File folder	
first_code	18-08-2020 20:03	Text Document	1 KB

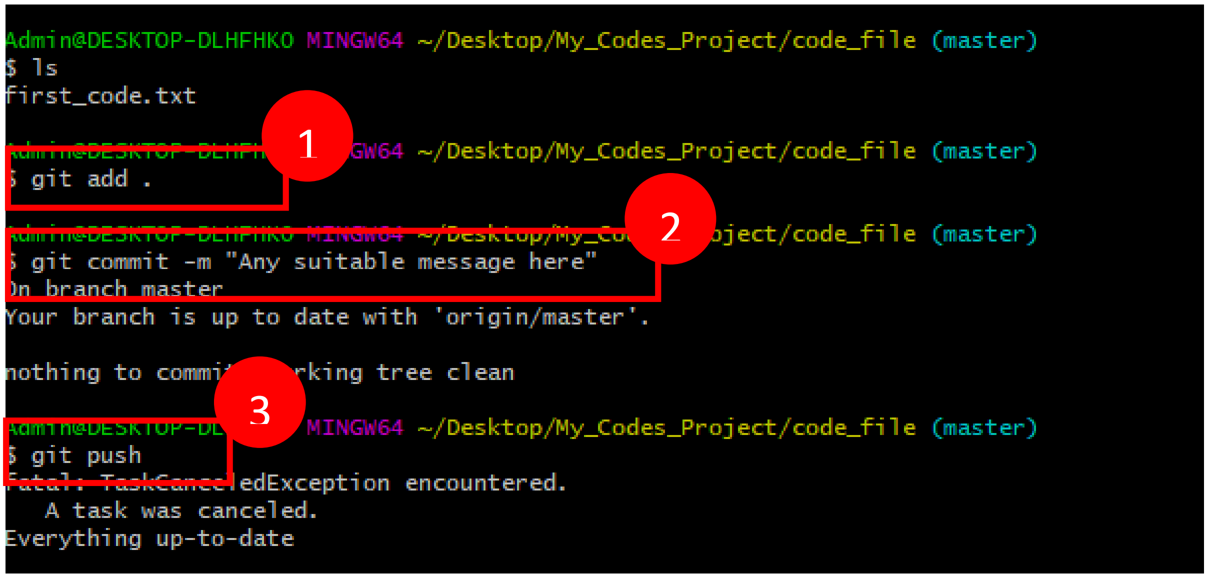
12. In the terminal, type **ls** and press enter, to check the list of files in the folder. You will be able to find the new file in the list of files (**first_code** in this case).



```
Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project
$ git clone https://github.com/Dhruv0895/code_file.git
Cloning into 'code_file'...
warning: You appear to have cloned an empty repository.

Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project
$ cd code_file

Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project/code_file (master)
$ ls
first_code.txt

Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project/code_file (master)
$ |
```

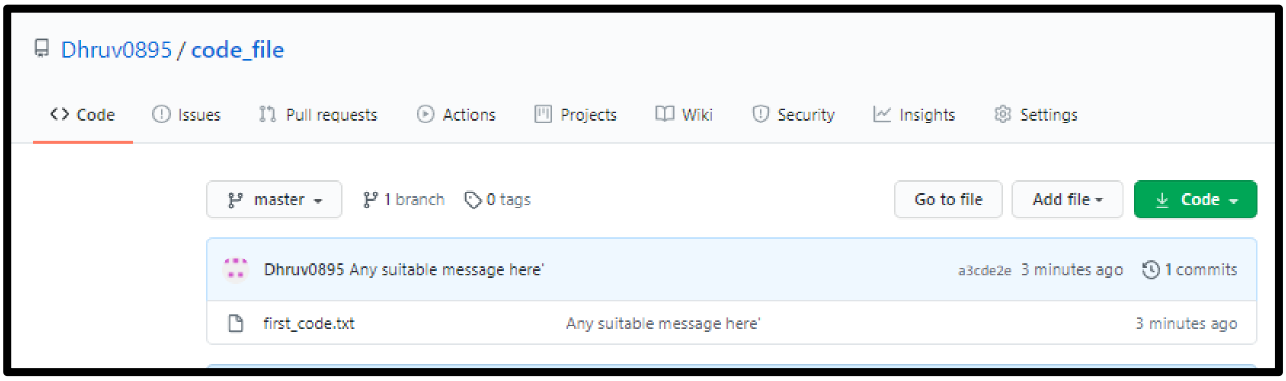
13. You now need to write the following commands to **add, commit** and **push** the file to the repository.

```
git add .

'Press Enter'

git commit -m "Any suitable message"

'Press Enter'

git push

'Press Enter'
```



```
Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project/code_file (master)
$ ls
first_code.txt

Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project/code_file (master)
$ git add .

Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project/code_file (master)
$ git commit -m "Any suitable message here"
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

Admin@DESKTOP-DLHFHKO MINGW64 ~/Desktop/My_Codes_Project/code_file (master)
$ git push
fatal: TaskCanceledException encountered.
   A task was canceled.
Everything up-to-date
```

14. If you now open the repository in your browser and refresh, you will be able to find the newly added file there.

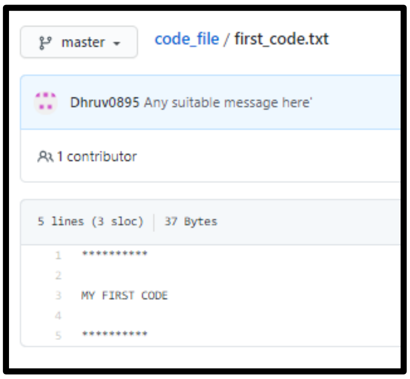


Dhruv0895 / code_file

Dhruv0895 Any suitable message here' a3cde2e 3 minutes ago 1 commits

first_code.txt Any suitable message here' 3 minutes ago

15. Click on the file to see the content inside it.



code_file / first_code.txt

Dhruv0895 Any suitable message here'

1 contributor

5 lines (3 sloc) | 37 Bytes

```
**********

MY FIRST CODE

**********
```