

PUSH: UPLOAD

1. Create folder "newfile" and create file first.py

2. Go inside folder "newfile"

3. open with git bash here

4. git init

//ls -al #to check all files

//git status (to see untracked files)

5. Git add first.py

//git status (now first.py will be in green)

// error bcoz git not configured

6. git config --global user.email

"rizwantopper@gmail.com"

7. git config --global user.name

"Rizwanishere"

8. git commit -m "first commit" first.py

Code is now committed

Check below for image

```
MINGW64:/c/Users/rizwa/OneDrive/Desktop/newfile
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile
$ git init
Initialized empty Git repository in C:/users/rizwa/OneDrive/Desktop/newfile/.git/
/
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ ls -al
total 17
drwxr-xr-x 1 rizwa 197609 0 Oct  2 16:17 .
drwxr-xr-x 1 rizwa 197609 0 Oct  2 16:13 ..
drwxr-xr-x 1 rizwa 197609 0 Oct  2 16:17 .git/
-rw-r--r-- 1 rizwa 197609 20 Oct  2 16:14 first.py
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ git status
On branch main
No commits yet
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    first.py
nothing added to commit but untracked files present (use "git add" to track)
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ git add first.py
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ git status
on branch main
No commits yet
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   first.py
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ git commit -m "first commit" first.py
Author identity unknown
*** Please tell me who you are.
Run
  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"
to set your account's default identity.
omit --global to set the identity only in this repository.
fatal: unable to auto-detect email address (got 'rizwa@RIZWAN.(none)')
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
```

```
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ git config --global user.email "rizwantopper@gmail.com"
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ git config --global user.name "Rizwanishere"
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ git commit -m "first commit" first.py
[main (root-commit) 92acaa3] first commit
 1 file changed, 1 insertion(+)
  create mode 100644 first.py
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$
```

1. Now go to github

2. Create new repository

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

Copy the code

3. Now paste it on git bash

4. New terminal opens for login on github

5. Go to github>settings>developer
settings>Personal access

tokens>tokens(classic)>generate new

6. paste token on that new terminal

**Code is successfully pushed(uploaded on
github).**

Now if you make changes in the first.py file then the changes will also be made into github??

No.

So you have to first:

1. commit it by using

Git commit -m "second commit" first.py

2. Push(upload) it by using

Git push -u origin main

```
rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ git commit -m "second commit" first.py
[main 54ae78b] second commit
 1 file changed, 2 insertions(+), 1 deletion(-)

rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$ git push -u origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 319 bytes | 319.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Rizwanishere/newfile.git
 3eb77d6..54ae78b main -> main
branch 'main' set up to track 'origin/main'.

rizwa@RIZWAN MINGW64 ~/OneDrive/Desktop/newfile (main)
$
```

HOW TO PULL?

If you just pushed a file and made some changes from github ui then enter this cmd
Git pull origin main.

Otherwise...

HOW TO CLONE REPOSITORY:

If you want some other person's repository in your local computer then

1. go to github repo of that person and click on **<>Code** then click on HTTPS link and copy that link(url)
2. Open git
3. Cd <directory> in which you want that repo to be cloned
4. Go to git terminal and enter command
Git clone <url>
pull done successfully

