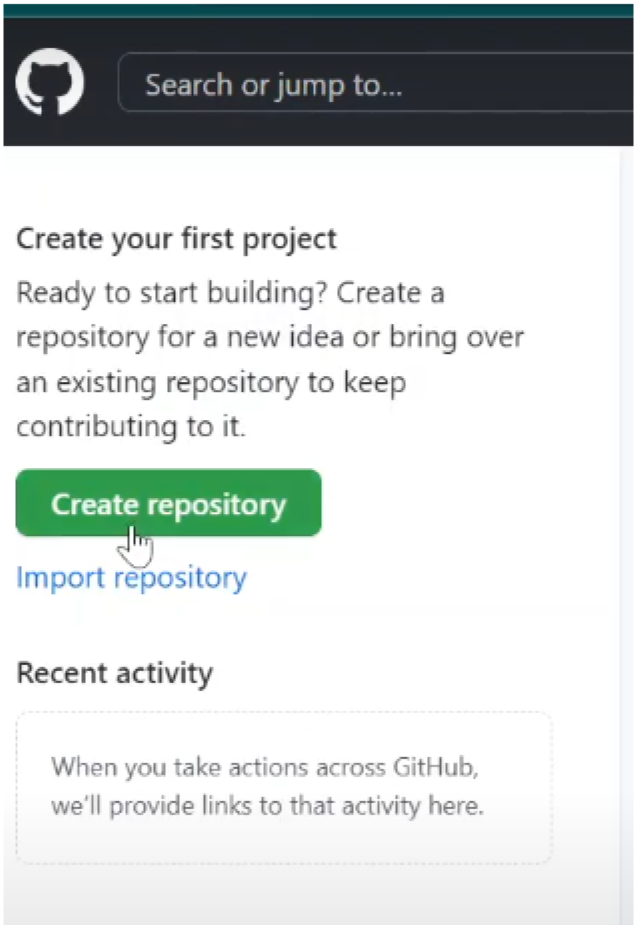
Step 1: register into github to push all our code into it:

1. Navigate to [https://github.com/](https://github.com/).
2. Click Sign up by entering email address, password, and username
3. Now we will get the verification link to our mentioned email address, verify it

Step 2: creating the  new repository in github account by clicking create repository as follows:



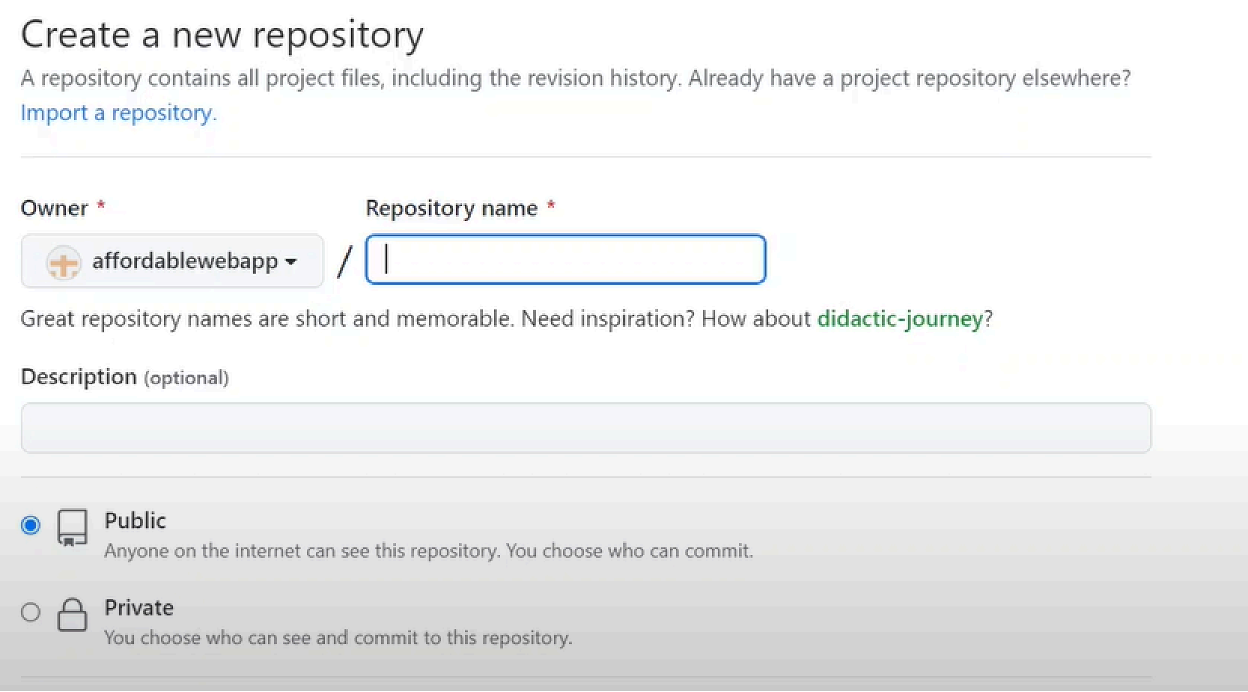
Create your first project

Ready to start building? Create a repository for a new idea or bring over an existing repository to keep contributing to it.

Recent activity

When you take actions across GitHub, we'll provide links to that activity here.

1. Enter new repository name(bash_project)
2. Select the repository should be  public or private
3. Select add a README file if you need
4. Click 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 *

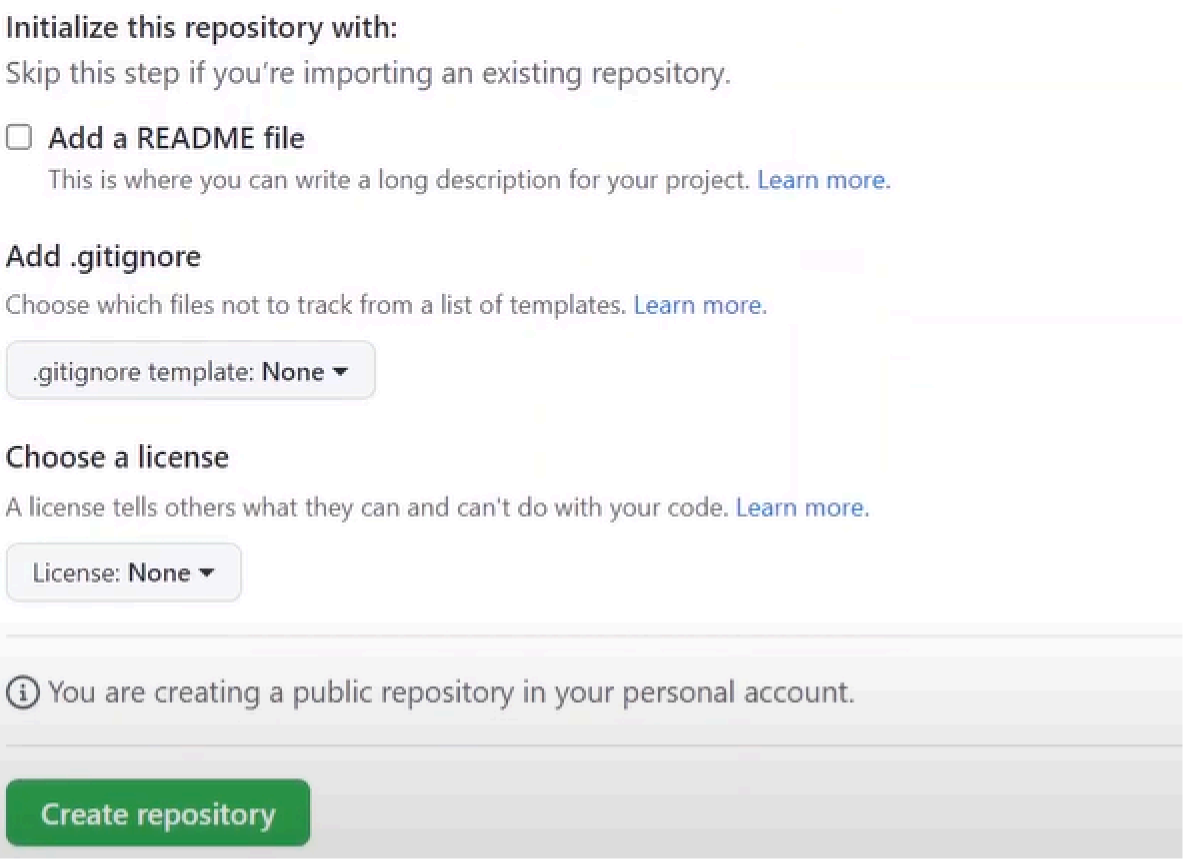
affordablewebapp ▾ /

Great repository names are short and memorable. Need inspiration? How about didactic-journey?

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

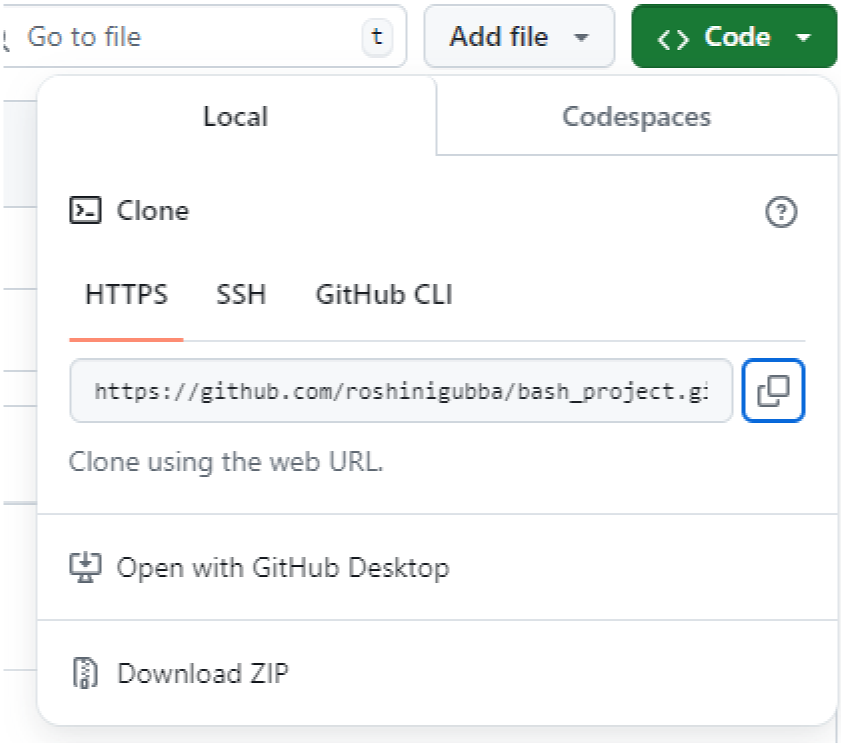
A license tells others what they can and can't do with your code. Learn more.

License: None ▾

ⓘ You are creating a public repository in your personal account.

Step 3: cloning the git repository:

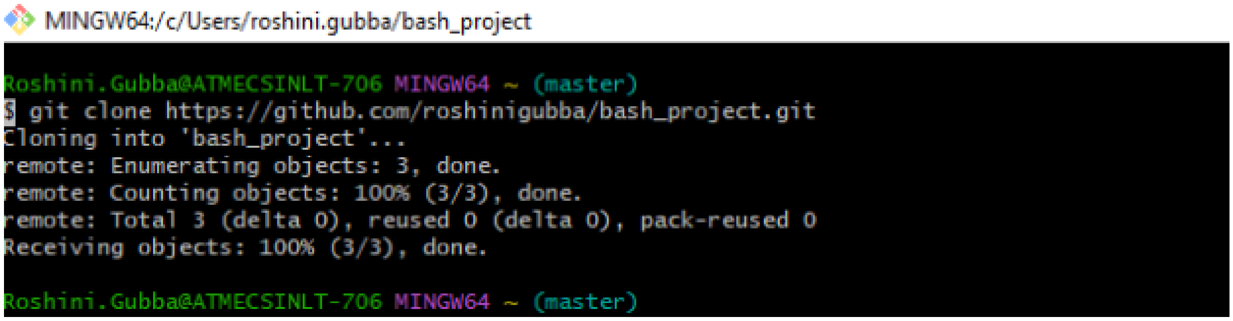
1. Copy the repository https link from github and paste it in the git bash:



```
https://github.com/roshinigubba/bash_project.gi
```

Clone using the web URL.

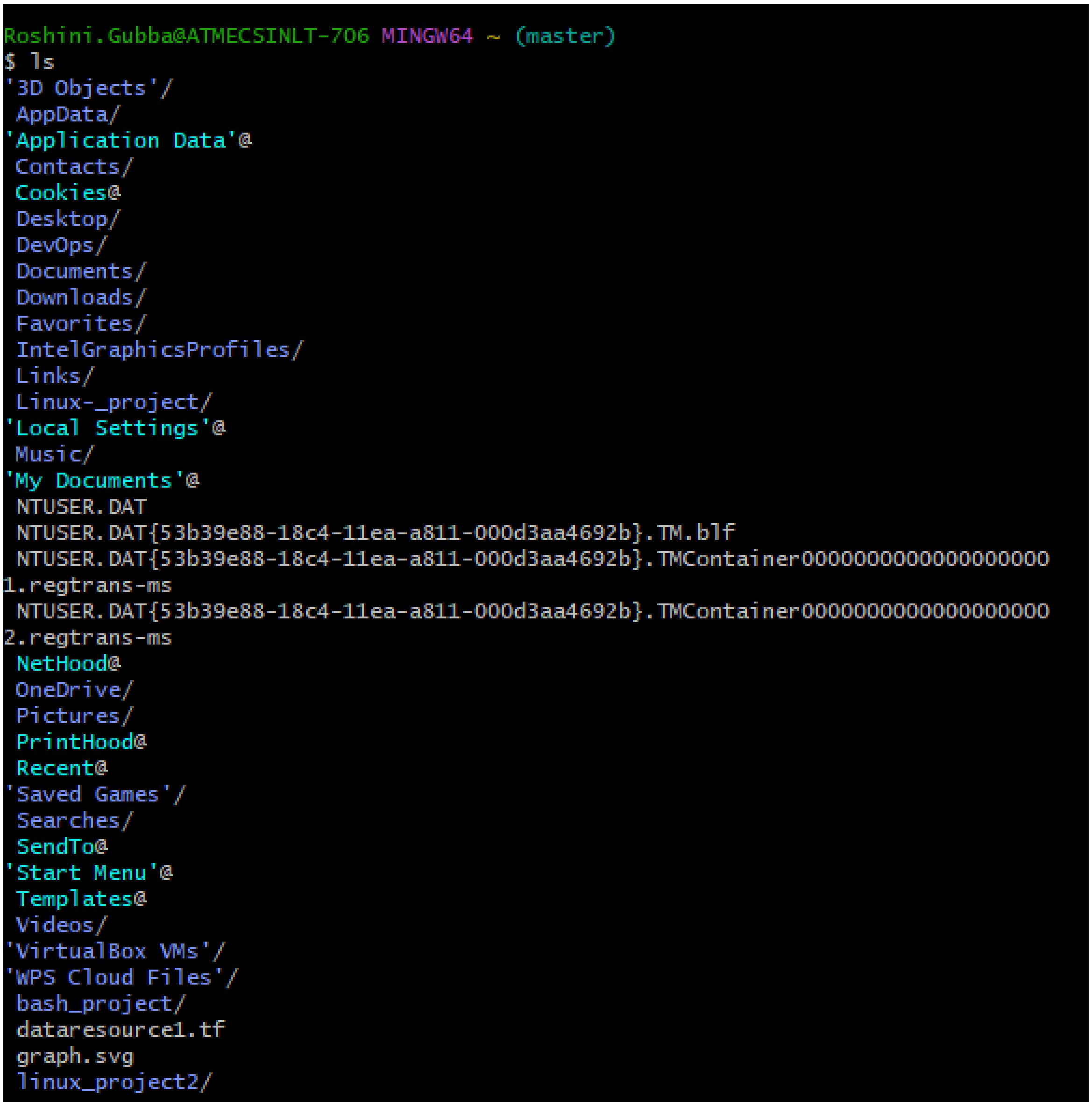
2. Copy that https link as following in git bash for cloning:



```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~ (master)
$ git clone https://github.com/roshinigubba/bash_project.git
Cloning into 'bash_project'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~ (master)
```

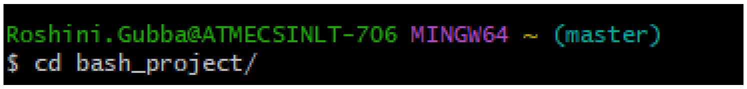
3. We will clone that repository into our git bash to update the repository from our computer by pushing code, files, creating branches through git bash

Step 4: we will list all the directories, to find our git repository in our local computer:



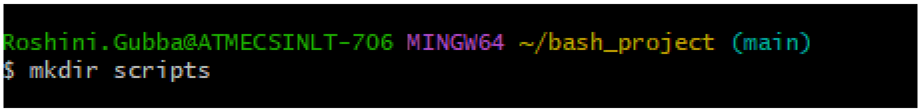
```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~ (master)
$ ls
'3D Objects'/
 AppData/
'Application Data'@
 Contacts/
 Cookies@
 Desktop/
 DevOps/
 Documents/
 Downloads/
 Favorites/
 IntelGraphicsProfiles/
 Links/
 Linux-_project/
'Local Settings'@
 Music/
'My Documents'@
 NTUSER.DAT
 NTUSER.DAT{53b39e88-18c4-11ea-a811-000d3aa4692b}.TM.blf
 NTUSER.DAT{53b39e88-18c4-11ea-a811-000d3aa4692b}.TMContainer0000000000000000000
1.regtrans-ms
 NTUSER.DAT{53b39e88-18c4-11ea-a811-000d3aa4692b}.TMContainer0000000000000000000
2.regtrans-ms
 NetHood@
 OneDrive/
 Pictures/
 PrintHood@
 Recent@
'Saved Games'/
 Searches/
 SendTo@
'Start Menu'@
 Templates@
 Videos/
'VirtualBox VMs'/
'WPS Cloud Files'/
 bash_project/
 dataresource1.tf
 graph.svg
 linux_project2/
```

Step 5: we will change directory to bash_project directory to add or update the files or code, or creating directories:



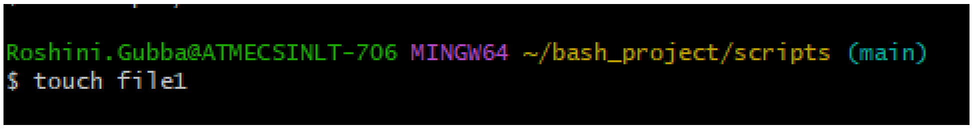
```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~ (master)
$ cd bash_project/
```

Step 6: creating a directory name as scripts to push all the scripts into it:



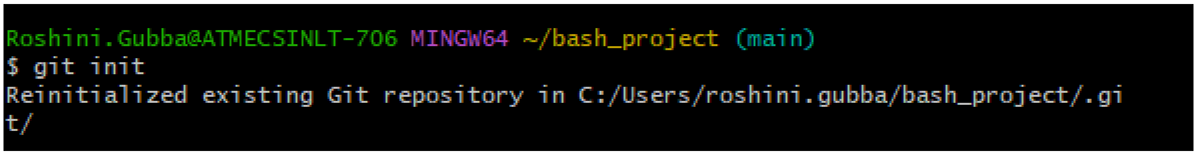
```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ mkdir scripts
```

Step 7: Go to scripts directory and create the collector file as file1:



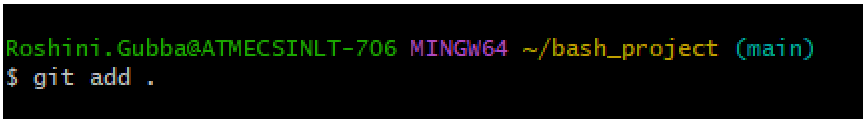
```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project/scripts (main)
$ touch file1
```

Step 8: initialize git by using git init:



```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git init
Reinitialized existing Git repository in C:/Users/roshini.gubba/bash_project/.gi
t/
```

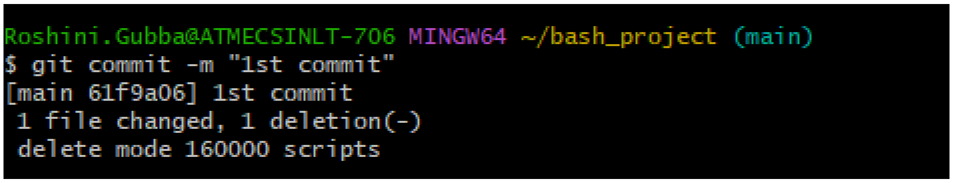
Step 9: After that we should add that directory into git by using **git add filename :**



```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git add .
```

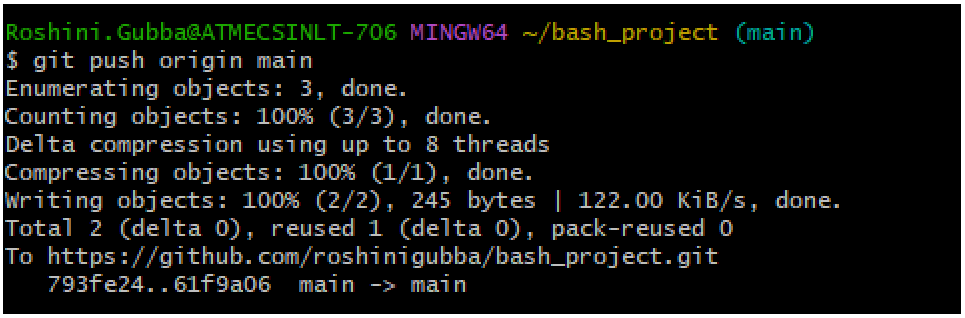
**add . : to add/update all the files at the time.**

Step 10 : After adding file, we get staged file, now we have to commit the staged file:



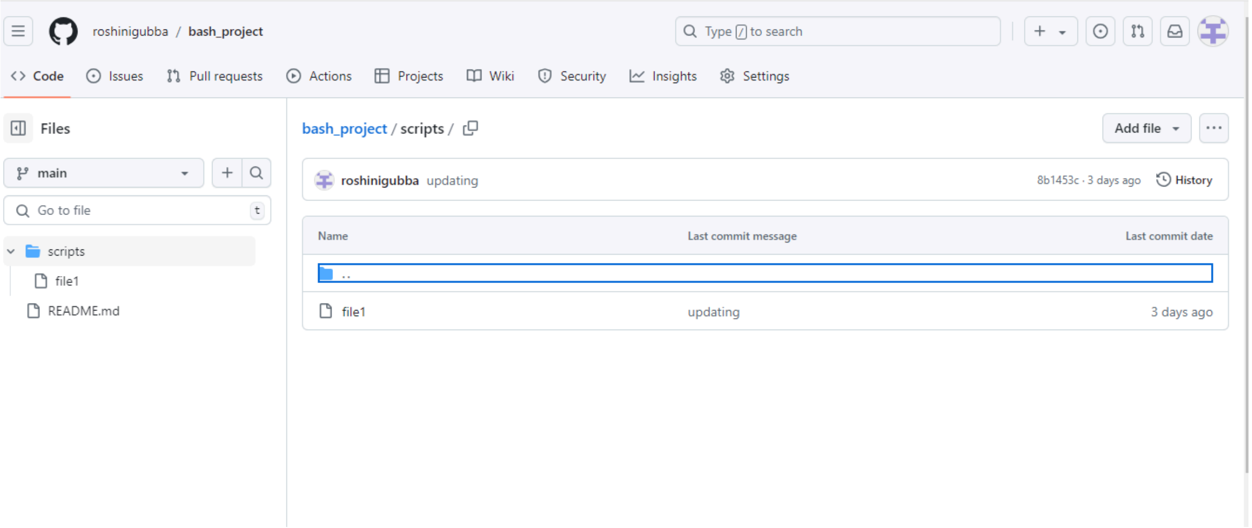
```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git commit -m "1st commit"
[main 61f9a06] 1st commit
 1 file changed, 1 deletion(-)
 delete mode 160000 scripts
```

Step 11: push that committed file/directory into git:



```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git push origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (1/1), done.
Writing objects: 100% (2/2), 245 bytes | 122.00 KiB/s, done.
Total 2 (delta 0), reused 1 (delta 0), pack-reused 0
To https://github.com/roshinigubba/bash_project.git
   793fe24..61f9a06  main -> main
```

After creating the scripts directory, creating the file inside that directory and pushing into git , it gets updated in git as follows:



bash_project / scripts /

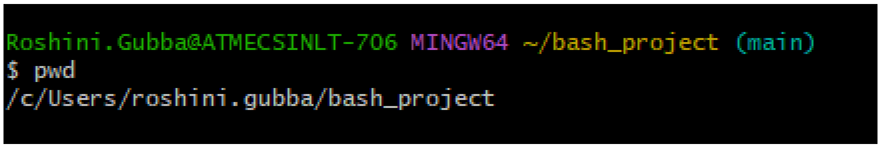
roshinigubba updating 8b1453c · 3 days ago

| Name | Last commit message | Last commit date |
|---|---|---|
| .. | | |
| file1 | updating | 3 days ago |

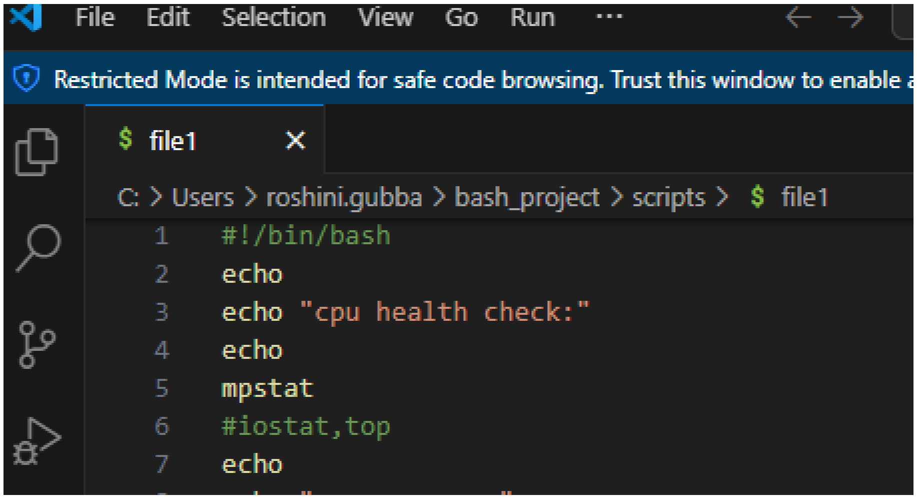
Step 12: open the file in vs code or any editor, write the script and save it, and do the following steps again after writing and saving the script:

1. git init
2. git add filename (or) git add .
3. Git commit -m “message of the file what we have updated”
4. git  push origin main

Find the location of bash_project directory in our computer and edit that file or write the script in that file by going into that location of the file:



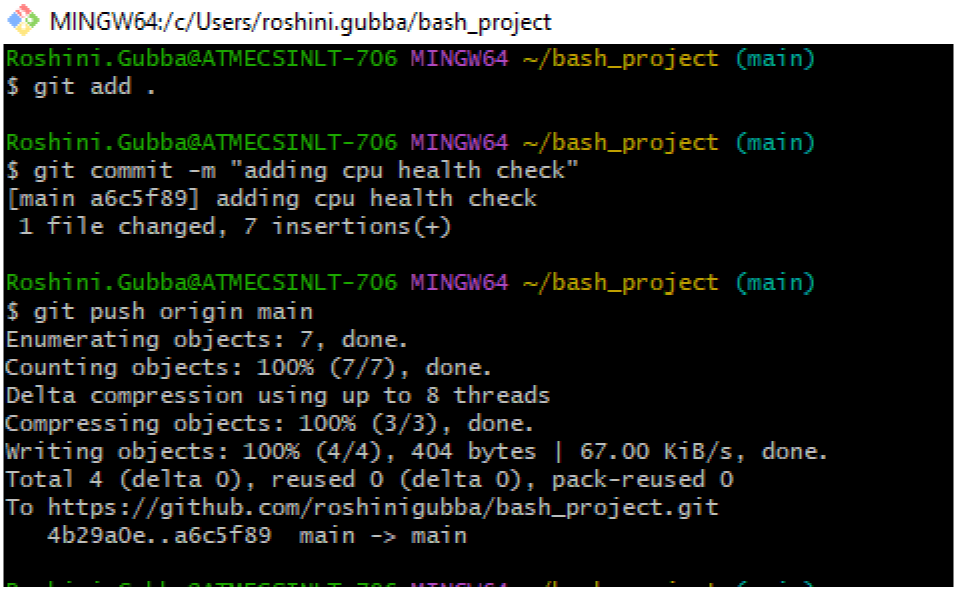
```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ pwd
/c/Users/roshini.gubba/bash_project
```



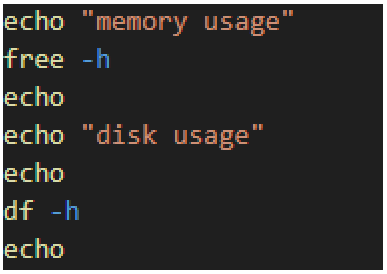
Restricted Mode is intended for safe code browsing. Trust this window to enable a

```
#!/bin/bash
echo
echo "cpu health check:"
echo
mpstat
#iostat,top
echo
```

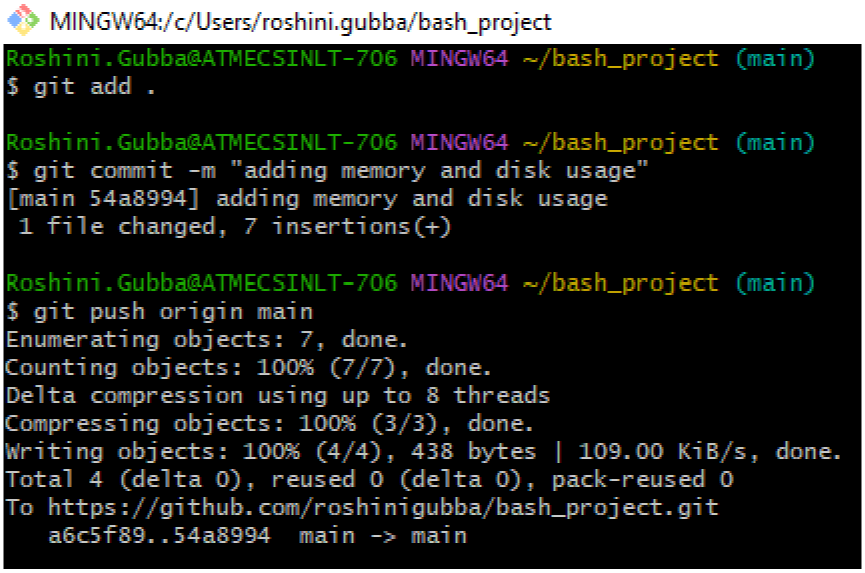
Save the file after writing the script, then follow the steps like, init, add, commit, push



```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git add .

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git commit -m "adding cpu health check"
[main a6c5f89] adding cpu health check
 1 file changed, 7 insertions(+)

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 404 bytes | 67.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/roshinigubba/bash_project.git
   4b29a0e..a6c5f89  main -> main
```

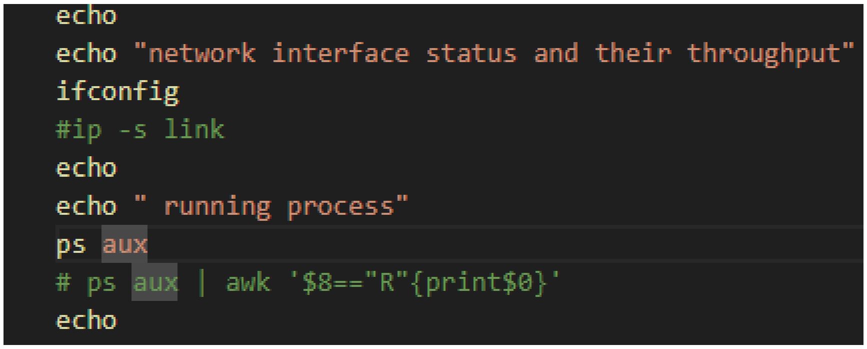


```
echo "memory usage"
free -h
echo
echo "disk usage"
echo
df -h
echo
```

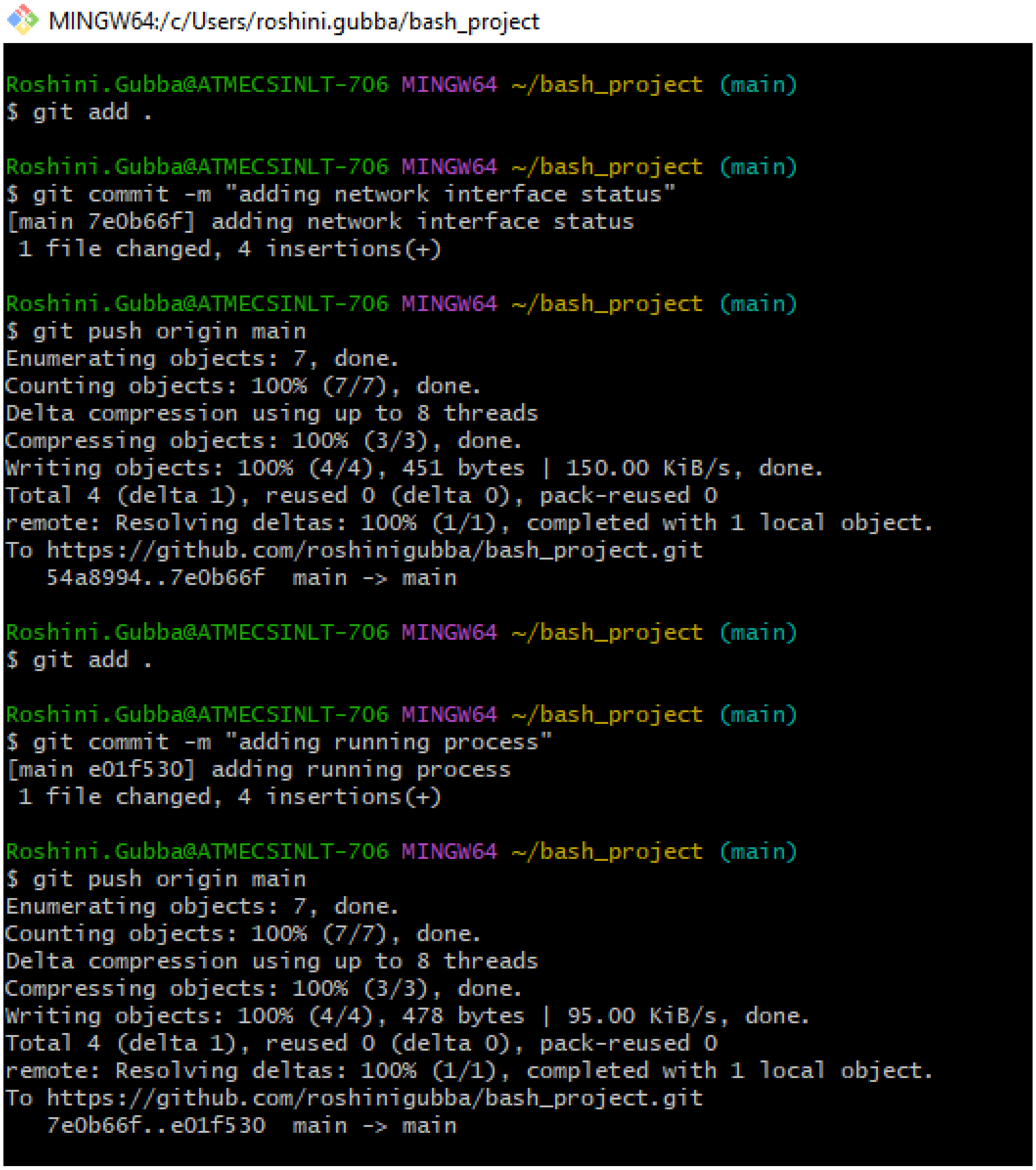


MINGW64:/c/Users/roshini.gubba/bash_project

```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git add .

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git commit -m "adding memory and disk usage"
[main 54a8994] adding memory and disk usage
 1 file changed, 7 insertions(+)

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 438 bytes | 109.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/roshinigubba/bash_project.git
   a6c5f89..54a8994  main -> main
```

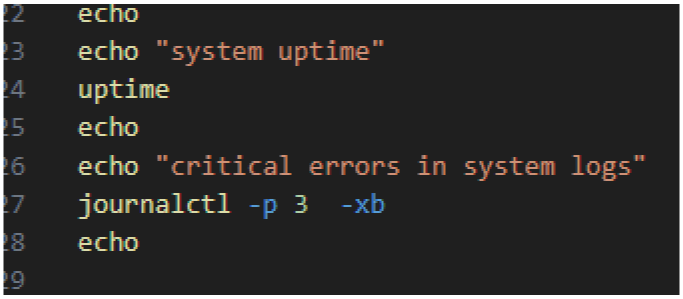


```
    echo
    echo "network interface status and their throughput"
    ifconfig
    #ip -s link
    echo
    echo " running process"
    ps aux
    # ps aux | awk '$8=="R"{print$0}'
    echo
```

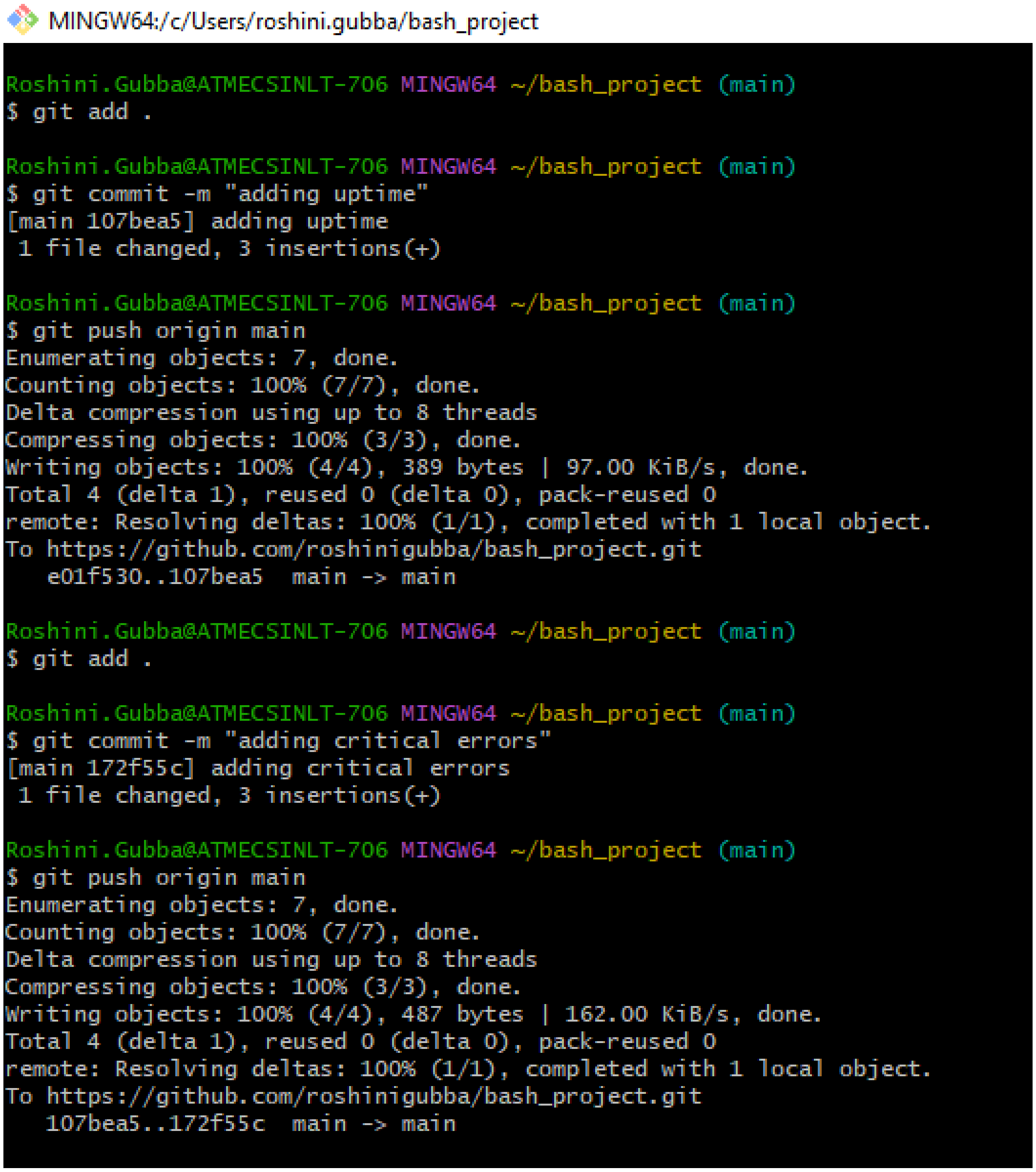


MINGW64:/c/Users/roshini.gubba/bash_project

```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git add .

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git commit -m "adding network interface status"
[main 7e0b66f] adding network interface status
 1 file changed, 4 insertions(+)

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 451 bytes | 150.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/roshinigubba/bash_project.git
   54a8994..7e0b66f  main -> main

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git add .

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git commit -m "adding running process"
[main e01f530] adding running process
 1 file changed, 4 insertions(+)

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 478 bytes | 95.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/roshinigubba/bash_project.git
   7e0b66f..e01f530  main -> main
```



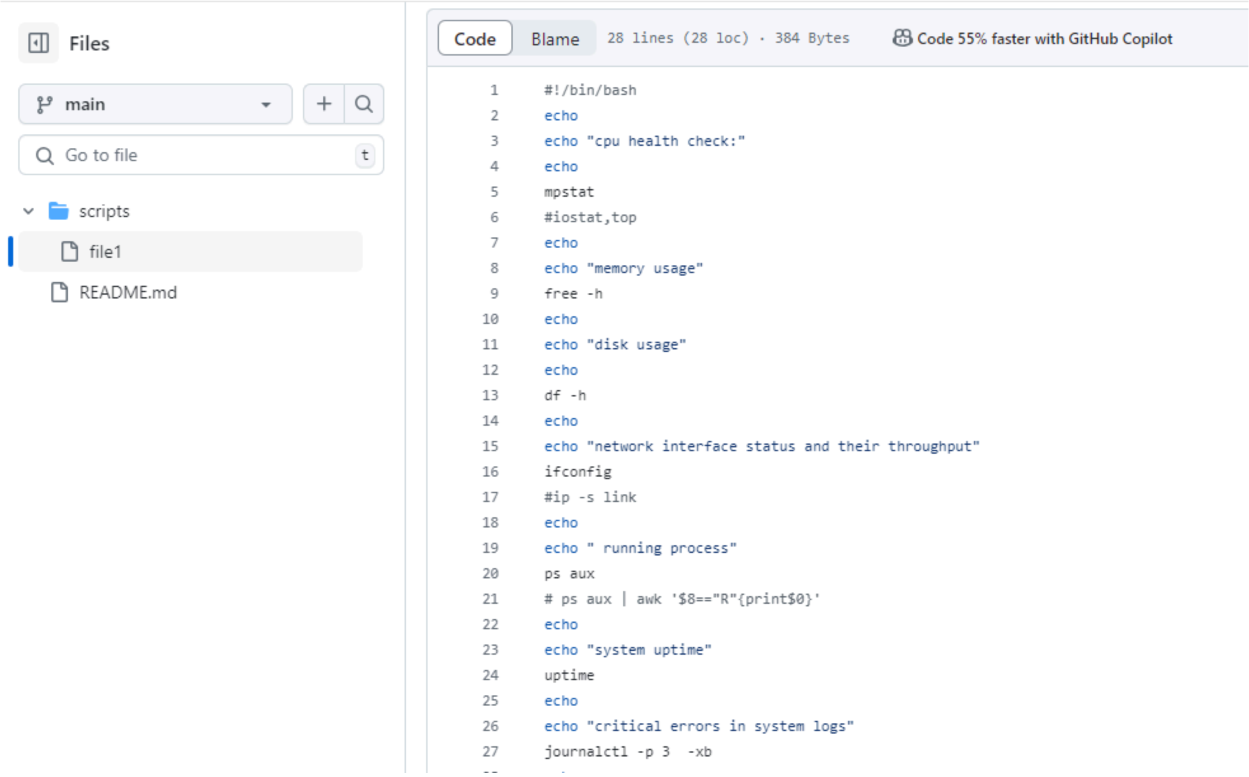
```
echo
echo "system uptime"
uptime
echo
echo "critical errors in system logs"
journalctl -p 3  -xb
echo

```



MINGW64:/c/Users/roshini.gubba/bash_project

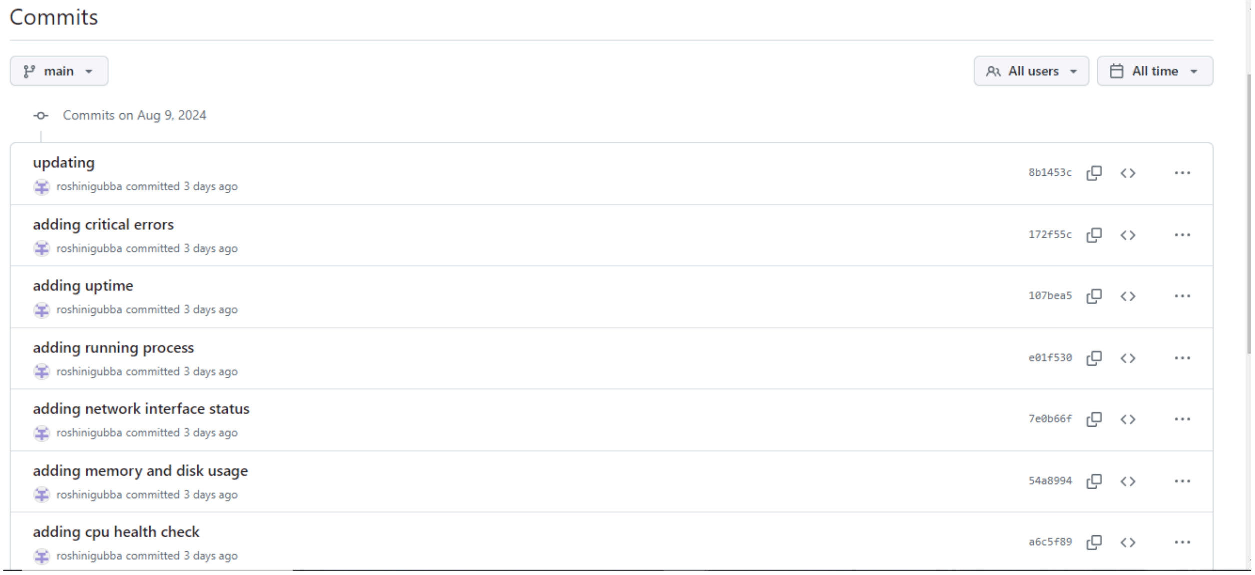
```
Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git add .

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git commit -m "adding uptime"
[main 107bea5] adding uptime
 1 file changed, 3 insertions(+)

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 389 bytes | 97.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/roshinigubba/bash_project.git
   e01f530..107bea5  main -> main

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git add .

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git commit -m "adding critical errors"
[main 172f55c] adding critical errors
 1 file changed, 3 insertions(+)

Roshini.Gubba@ATMECSINLT-706 MINGW64 ~/bash_project (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 487 bytes | 162.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/roshinigubba/bash_project.git
   107bea5..172f55c  main -> main
```

Step by step it is updated in git after pushing that code into the git repository as follows:


Files
main
Go to file
t
scripts
file1
README.md
Code
Blame
28 lines (28 loc) · 384 Bytes
Code 55% faster with GitHub Copilot

```
#!/bin/bash
echo
echo "cpu health check:"
echo
mpstat
#iostat,top
echo
echo "memory usage"
free -h
echo
echo "disk usage"
echo
df -h
echo
echo "network interface status and their throughput"
ifconfig
#ip -s link
echo
echo " running process"
ps aux
# ps aux | awk '$8=="R"{print$0}'
echo
echo "system uptime"
uptime
echo
echo "critical errors in system logs"
journalctl -p 3  -xb
```

It also list all the commits whatever we have done:



Commits on Aug 9, 2024

updating
roshinigubba committed 3 days ago
8b1453c

adding critical errors
roshinigubba committed 3 days ago
172f55c

adding uptime
roshinigubba committed 3 days ago
107bea5

adding running process
roshinigubba committed 3 days ago
e01f530

adding network interface status
roshinigubba committed 3 days ago
7e0b66f

adding memory and disk usage
roshinigubba committed 3 days ago
54a8994

adding cpu health check
roshinigubba committed 3 days ago
a6c5f89