Gitlab credentials:

User Name: Padmaja.N

password: Tulasi#6

email: [padmaja.nujella@gmail.com](mailto:padmaja.nujella@gmail.com)

Github credentials:

User Name: RadhakrishnaPadma

password: ramatulasi6

email: [padmaja.nujella@gmail.com](mailto:padmaja.nujella@gmail.com)

Bright Speed Github credentials:

email: padamaja.n@brightspeed.com

pwd: ramatulasi#6

username: BRSPD-npadmaja

Getting Started with PostgreSQL on Windows with Git Bash:

<https://medium.com/@itayperry91/get-started-with-postgresql-on-windows-a-juniors-life-4adfa6dd10e>

git clone branch better site:

<https://www.freecodecamp.org/news/git-clone-branch-how-to-clone-a-specific-branch/>

to clone only particular branch from remote repository then:

git clone --branch demobranch https://gitlab.com/sample\_group4581806/sample\_project.git

to connect to remote repository from local repository, use below command in git bash shell:

$ git remote add origin <https://github.com/Padma24n/test_demo.git>

if already origin created, if we try to create origin again It will not allow you to push the file to remote repo, so execute below cmd to check the path of remote rep

$ git remote -v

main https://github.com/RadhakrishnaPadma/github\_practice.git (fetch)

main https://github.com/RadhakrishnaPadma/github\_practice.git (push)

origin https://github.com/Padma24n/github\_test\_practice.git (fetch)

origin https://github.com/Padma24n/github\_test\_practice.git (push)

$ git push -v main main –---------- then it will push all the file in that branch to remote rep

to push all file in master branch in bash dir to remote repository, the cmd is as below:

$ git push origin master

Padmaja.N@IVLH-LP17 MINGW64 /d

$ git clone https://gitlab.com/padmaja.nujella/demoproject.git

Cloning into 'demoproject'...

remote: Enumerating objects: 6, done.

remote: Counting objects: 100% (6/6), done.

remote: Compressing objects: 100% (5/5), done.

remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0

Receiving objects: 100% (6/6), done.

Padmaja.N@IVLH-LP17 MINGW64 /d

$ git status

fatal: detected dubious ownership in repository at 'D:/'

'D:/' is owned by:

'S-1-5-18'

but the current user is:

'S-1-5-21-2819083639-1916519104-3859954802-20353'

To add an exception for this directory, call:

git config --global --add safe.directory D:/

Padmaja.N@IVLH-LP17 MINGW64 /d

$ git config --global user.name "padmaja nujella"

Padmaja.N@IVLH-LP17 MINGW64 /d

$ git config --global user.email "padmaja.nujella@gmail.com"

Padmaja.N@IVLH-LP17 MINGW64 /d

$ git init .

Reinitialized existing Git repository in D:/.git/

Padmaja.N@IVLH-LP17 MINGW64 /d

$ git clone https://gitlab.com/padmaja.nujella/demoproject.git

fatal: destination path 'demoproject' already exists and is not an empty directory.

Padmaja.N@IVLH-LP17 MINGW64 /d

$ touch test.html1.txt

Padmaja.N@IVLH-LP17 MINGW64 /d

$ cd demoproject

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ touch test.html1.txt

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ cat>test.html1.txt

this is html.

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ cat test.html1.txt

this is html.

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git branch

\* main

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git checkout origin main

error: pathspec 'main' did not match any file(s) known to git

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git checkout main

Already on 'main'

Your branch is up to date with 'origin/main'.

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git add test.html1.txt

warning: in the working copy of 'test.html1.txt', LF will be replaced by CRLF the next time Git touches it

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ ll

total 9

-rw-r--r-- 1 Padmaja.N 1049089 6310 Feb 1 10:53 README.md

-rw-r--r-- 1 Padmaja.N 1049089 14 Feb 1 11:02 test.html1.txt

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git add test.html1.txt

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git commit -m "test file commit" test.html1.txt

warning: in the working copy of 'test.html1.txt', LF will be replaced by CRLF the next time Git touches it

[main 3103dfd] test file commit

1 file changed, 1 insertion(+)

create mode 100644 test.html1.txt

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git status

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

nothing to commit, working tree clean

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git commit -m "test html1 file"

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

nothing to commit, working tree clean

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git staus

git: 'staus' is not a git command. See 'git --help'.

The most similar command is

status

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git status

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

nothing to commit, working tree clean

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git log

commit 3103dfdb0059330e672ae328e8d4d5522122e9cf (HEAD -> main)

Author: padmaja nujella <padmaja.nujella@gmail.com>

Date: Wed Feb 1 11:09:02 2023 +0530

test file commit

commit 1816b48038df32b168fbeca60a6fe2a7d4d8b91c (origin/test\_branch, origin/main, origin/HEAD)

Author: Padmaja Nujella <padmaja.nujella@gmail.com>

Date: Tue Jan 31 10:26:51 2023 +0000

Configure SAST in `.gitlab-ci.yml`, creating this file if it does not already exist

commit 98745d71cd2a8f4f688e3d0adf587c7dd6b1f367

Author: Padmaja Nujella <padmaja.nujella@gmail.com>

Date: Tue Jan 31 10:26:50 2023 +0000

Initial commit

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git push -u origin main

Enumerating objects: 4, done.

Counting objects: 100% (4/4), done.

Delta compression using up to 8 threads

Compressing objects: 100% (2/2), done.

Writing objects: 100% (3/3), 333 bytes | 333.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

To https://gitlab.com/padmaja.nujella/demoproject.git

1816b48..3103dfd main -> main

branch 'main' set up to track 'origin/main'.

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ ls lt

ls: cannot access 'lt': No such file or directory

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ ls -ltr

total 8

-rw-r--r-- 1 Padmaja.N 1049089 6310 Feb 1 10:53 README.md

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git fetch

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git status

On branch main

Your branch is up to date with 'origin/main'.

Changes not staged for commit:

(use "git add/rm <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

deleted: test.html1.txt

no changes added to commit (use "git add" and/or "git commit -a")

**----- to pull the file that has not be pull with all the files from remote repo:**

git pull origin test\_branch --🡪 branch name

Padmaja.N@IVLH-LP17 MINGW64 /d/projectdir (master)

$ ll

total 1

-rw-r--r-- 1 Padmaja.N 1049089 39 Feb 6 18:05 padma6

-rw-r--r-- 1 Padmaja.N 1049089 0 Feb 1 18:56 riya2

-rw-r--r-- 1 Padmaja.N 1049089 0 Feb 1 18:56 riya4

drwxr-xr-x 1 Padmaja.N 1049089 0 Feb 3 16:18 test\_demo/

-rw-r--r-- 1 Padmaja.N 1049089 0 Feb 1 18:57 viyu1

-rw-r--r-- 1 Padmaja.N 1049089 0 Feb 1 18:57 viyu2

-rw-r--r-- 1 Padmaja.N 1049089 0 Feb 1 18:57 viyu3

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git pull origin main

From https://gitlab.com/padmaja.nujella/demoproject

\* branch main -> FETCH\_HEAD

Already up to date.

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git log

commit 3103dfdb0059330e672ae328e8d4d5522122e9cf (HEAD -> main, origin/main, origin/HEAD)

Author: padmaja nujella <padmaja.nujella@gmail.com>

Date: Wed Feb 1 11:09:02 2023 +0530

test file commit

commit 1816b48038df32b168fbeca60a6fe2a7d4d8b91c (origin/test\_branch)

Author: Padmaja Nujella <padmaja.nujella@gmail.com>

Date: Tue Jan 31 10:26:51 2023 +0000

Configure SAST in `.gitlab-ci.yml`, creating this file if it does not already exist

commit 98745d71cd2a8f4f688e3d0adf587c7dd6b1f367

Author: Padmaja Nujella <padmaja.nujella@gmail.com>

Date: Tue Jan 31 10:26:50 2023 +0000

Initial commit

**--- my own way of push commit in bash**

Padmaja.N@IVLH-LP17 MINGW64 ~ (main)

$ cd c:

Padmaja.N@IVLH-LP17 MINGW64 /c

$ cd demodir

Padmaja.N@IVLH-LP17 MINGW64 /c/demodir (master)

$ git remote add origin https://gitlab.com/project\_group4151405/test\_demo.git

Padmaja.N@IVLH-LP17 MINGW64 /c/demodir (master)

$ git push -uf origin master

Enumerating objects: 3, done.

Counting objects: 100% (3/3), done.

Delta compression using up to 8 threads

Compressing objects: 100% (2/2), done.

Writing objects: 100% (3/3), 222 bytes | 222.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

remote:

remote: To create a merge request for master, visit:

remote: https://gitlab.com/project\_group4151405/test\_demo/-/merge\_requests/new?merge\_request%5Bsource\_branch%5D=master

remote:

To https://gitlab.com/project\_group4151405/test\_demo.git

\* [new branch] master -> master

branch 'master' set up to track 'origin/master'.

Padmaja.N@IVLH-LP17 MINGW64 /c

$ cd d:

Padmaja.N@IVLH-LP17 MINGW64 /d

$ ll

total 8450

-rw-r--r-- 1 Padmaja.N 1049089 22528 Nov 13 2017 '!qhlogs.doc'

drwxr-xr-x 1 Padmaja.N 1049089 0 Feb 17 2022 '$RECYCLE.BIN'/

drwxr-xr-x 1 Padmaja.N 1049089 0 Dec 2 16:00 D/

-rw-r--r-- 1 Padmaja.N 1049089 9426 Oct 4 11:34 D.zip

drwxr-xr-x 1 Padmaja.N 1049089 0 Feb 3 09:19 'OneDrive - InfoVision, Inc'/

drwxr-xr-x 1 Padmaja.N 1049089 0 Oct 12 12:44 OneDriveTemp/

drwxr-xr-x 1 Padmaja.N 1049089 0 Jan 18 16:25 'Personal Details'/

drwxr-xr-x 1 Padmaja.N 1049089 0 Jan 30 14:51 'Practice Languages'/

-rw-r--r-- 1 Padmaja.N 1049089 7680444 Dec 1 15:35 'Practice Languages.zip'

drwxr-xr-x 1 Padmaja.N 1049089 0 Feb 17 2022 SOPHOS/

drwxr-xr-x 1 Padmaja.N 1049089 0 Jan 25 12:06 Softwares/

drwxr-xr-x 1 Padmaja.N 1049089 0 May 17 2022 'System Volume Information'/

drwxr-xr-x 1 Padmaja.N 1049089 0 Feb 3 13:59 cfrbackup-SVEPIFPK/

-rw-r--r-- 1 Padmaja.N 1049089 133 Jan 18 15:24 dump.csv

-rwxr-xr-x 1 Padmaja.N 1049089 904704 Dec 1 2006 msdia80.dll\*

-rw-r--r-- 1 Padmaja.N 1049089 237 Dec 5 15:21 person\_ddl\_script.json

drwxr-xr-x 1 Padmaja.N 1049089 0 Feb 3 14:42 projectdir/

Padmaja.N@IVLH-LP17 MINGW64 /d

$ cd projectdir

Padmaja.N@IVLH-LP17 MINGW64 /d/projectdir (master)

$ git remote add origin <https://gitlab.com/project_group4151405/test_demo.git>

git remote add origin https://gitlab.com/project\_group4151405/padma6.git

Padmaja.N@IVLH-LP17 MINGW64 /d/projectdir (master)

$ git push -uf origin master

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 8 threads

Compressing objects: 100% (4/4), done.

Writing objects: 100% (5/5), 444 bytes | 444.00 KiB/s, done.

Total 5 (delta 0), reused 0 (delta 0), pack-reused 0

remote:

remote: To create a merge request for master, visit:

remote: https://gitlab.com/project\_group4151405/test\_demo/-/merge\_requests/new?merge\_request%5Bsource\_branch%5D=master

remote:

To https://gitlab.com/project\_group4151405/test\_demo.git

+ 7a4505f...27ee3da master -> master (forced update)

branch 'master' set up to track 'origin/master'.

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git pull

Already up to date.

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ cd ~

Padmaja.N@IVLH-LP17 MINGW64 ~ (master)

$ cd main

bash: cd: main: No such file or directory

Padmaja.N@IVLH-LP17 MINGW64 ~ (master)

$ cd /d/demoproject

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ git pull

Already up to date.

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ pwd

/d/demoproject

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$ 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

credential.https://dev.azure.com.usehttppath=true

init.defaultbranch=master

gui.recentrepo=C:/Users/Padmaja.N/f1

user.name=padmaja nujella

user.email=padmaja.nujella@gmail.com

core.repositoryformatversion=0

core.filemode=false

core.bare=false

core.logallrefupdates=true

core.symlinks=false

core.ignorecase=true

remote.origin.url=https://gitlab.com/padmaja.nujella/demoproject.git

remote.origin.fetch=+refs/heads/\*:refs/remotes/origin/\*

branch.main.remote=origin

branch.main.merge=refs/heads/main

Padmaja.N@IVLH-LP17 MINGW64 /d/demoproject (main)

$

**Git Tutorial**

**git clone https://gitlab.com/padmaja.nujella/demoproject.git**

**touch test\_html1.txt --> using git bash, we can create file in repo dir**

first create user.name and user.email then directory and then init like

--- user name and email creation

git config –global user.name “padmaja.n”

git config –global user.email [padmaja.n@infovision.com](mailto:padmaja.n@infovision.com)

--- creating directory

mkdir demodir

--- initiating

git init .

--- checking status

git status

--- creating files

echo ‘ this is demo file’ > f1

(or)

touch f1

(or)

cat>f1

this is demo file. ctrl+d

--- add untrack files to track(staging repo)

git add f1

git add filename\*

git status

--- to list all directories and files in a directory then enter:

ls -al

ls -la

--- to list only files in a directory then enter:

ls -l

--- to move to home directory enter

cd ~

Git --🡪 it is a tool

Git Hub --🡪 it is a web, to store code from different source

Clone --🡪 Bring a repository that is hosted somewhere like Github into a folder on your local machine

add -🡪 Track your files and changes in Git

commit --🡪 Save your files in Git

push --🡪 Upload Git commits to a remote repo, like Github

pull -🡪 Download changes from remote repo to your local machine, the opposite of push

git --version -🡪2.39.1

--- copy cmd

pbcopy < demodir /f1

-------- for files not to tract and not to commit, we use ignoring

it will be useful when you don’t want to track some specific files then we use a file called .gitignore

* vi .gitignore

Now all the text files will be ignored.

Ex: touch murthy{1..2}

vi .gitignore

$

$

i

murthy\*

Esc

:wq

--- move a file from gitignore to untrack, use

vi .gitignore

i

murthy2

murthy3

esc

shift :wq

**Parts:**

We have 3 parts, they are

part 1 part 2 part 3

local(local system) branches central(Git Hub web)



in local we will create repository, in that we create directories and files. To move to center(Git hub web) then we have to create a branch

**GIT Branches:**

a branch represent a independent line of development.

The git branch command lets you create, list, rename and del branches.

The default branch name is Git is Master.

* To see current branch : git branch
* To add a new branch : git branch branch-name  
  To switch at a time : git checkout branch-name
* To create & switch at a time : git checkout -b branch-name
* To rename a branch : git branch -m old new
* To delete a branch : git branch -d <branch>

The -d option will delete the branch only if it has already been pushed and merged with remote branch. Use -D instead if you want to force the branch to be deleted.

git branch

o/p: \*master

1. After install the git. Open git bash cmd and enter **git init .** It will create an empty repository.
2. git config user.name “Padmaja N”.
3. git config user.email [Padmaja.N@infovision.com](mailto:Padmaja.N@infovision.com)
4. touch f1 --- it’s a directory
5. cd f1 – to will move to the f1 directory
6. touch filename1 **–** creates a file
7. cat>filename1 this is filename1.
8. enter ctrl+d
9. cat filename1
10. to remove file --- rm -f filename1
11. cd ~
12. to remove directory enter --- rm -rf **f1**
13. mkdir gitdemodir --- creates a directory
14. cd gitdemodir --- locate to directory
15. touch firstfile --- creates a file
16. ls -al --- list all the file in that directory
17. cat>firstfile
18. enter ctrl+d
19. cat>>firstfile ---- this is to append the file
20. touch secondfile --- creates second file
21. cp firstfile secondfile – same directory
22. cat secondfile
23. in first directory, I have created 2 files and one dir
24. mv secondfile ~/demo1dir --- moving to anther dir
25. ls -al
26. cd demo1dir
27. ls -al
28. giving permissions to a file chmod +rwx filename
29. cp file to another directory with other name ----- cp f2 ~/homedir/f3ls
30. git config --get user.name ------------🡪 to get user name in git
31. create html file with notepad and save as html and keep the file in gitdemodir directory now
32. if we check the status like git status, it will be untrack means not committed. To do so file needs
33. to move to Staging Environment. Staged files are files that are ready to be committed to repository u r working on.
34. cmd to move to Staging Env, write **git add HelloWorld.html**
35. to remove user and email from git config use the as below:

git config --global --unset user.name "padmaja"

git config --global --unset user.email "padmaja.n@infovision.com"

To check for all users, command is:

git config --list

git status.

git add f1

git commit -m “f1 is committed and move to staging repo”

**to move to tracking hidden files or all the files to Staging area cmd is:**

git add .

**To create multiple files, the cmd is:**

touch filename{1..4}

**make files move to untrack again. the command is:**

git restore --staged .bash\_logout -🡪 file name

**to move to untrack again cmd is:**

git restore --staged .

**to make a particular files to track then the cmd is:**

git add filename\*

git status

git commit -m “filename commits” filename\*

to check files in particular directory enter double LL:

$ ll

Append text from one file to another file the cmd is:

cat>>f2 f1

Creating Repository in D: drive:

$ cd D:

$ git init .

Initialized empty git repository in D:/.git/

$ git config --global user.name "padmajan"

$ git config --global user.email "padmaja.n@infovision.com"

$ git config --list

$ mkdir demodir

$ cd demodir

$ touch f1

$ cat>f1

This is file One. ctrl+d

$ cat>f2

This is file two. ctrl+d

$ cat f2

This is file two.

$ cat>>f2 f1

$ cat f2

This is file One.

This is file two.

$ touch filename{1..2}

$ touch Button{.js,.css,.stories.js,.test.js}

$ ls

Button.css Button.js Button.stories.js Button.test.js

$ touch filename{1..4}.txt

$ echo “hi, I am learning git” > filename.txt

$ ls -a; -🡪 it will show all the files in that Drive in horizontal

rm -r :

**The rm stands for remove**, while the -r is necessary to tell Bash that it needs to recurse (or repeat) the command through a list of all files and sub-directory within the parent directory. Thus, the newly created homeworks directory under assignments will also be removed, when assignments is deleted.

Branch:

1. a branch represent a independent line of dev.
2. the git branch command lets you create,list,rename&del branches
3. the default branch name in Git is Master.

--- to view branch, the cmd is:

git branch

--- to create a new branch, the cmd is:

git branch demo\_branch

--- to switch to newly created branch or checkout

git switch demo\_branch (or) git checkout master

--- to change to previous branch or checkout

git switch master (or) git checkout demo\_branch

if we create another branch with another name, and will view,it will show the previous files

that we have created in previous branch

steps:

git branch riya

git checkout riya

ll

touch riya{1..3}

git add riya\*

git commit -m “riya commit” riya1 riya2

below cmd is used to directly create a branch and switch without

following steps of create branch and switch branch:

git checkout -b branch-name

to get deleted files then use:

git restore f1 --🡪 filename

to rename a branch:

git branch -m lilly tulasi

to delete a branch the cmd is

git branch -d tulasi

**Git Lab:**

this is a web page to store source code& can do testing,

deployment etc.

to create SSH for gitlab

ssh-keygen

it will save in the below path:

c/users/Padmaja/.ssh/id\_rsa

cd ~

cd .ssh

cat id\_rsa.pub ---🡪 it will prompt the url, copy the path and

paste the path in gitlab.com/ssh keys🡪key box. it is for clonging

What does rm RF do?

sudo rm -rf Syntax  
  
Allows removing root-owned files and directories. rm - Linux command for removing files or directories. -r - The option indicates recursive removal and helps remove non-empty directories. -f - The option allows removal without confirmation, even if a file does not exist.

git log –-pretty oneline

git branch -v

git push origin -–delete branch\_name

**in git bash cmd:**

touch demo.txt

vi demo.txt

**enter text as:**

read name

name=”abc”

read number

number=246

come out of the file and in prompt enter

echo $name $number

:q! 🡪 quite and discard changes

w -🡪 to jump front of the next word like hello word

backward to front word 🡪 d

to jump to starting hit o

to jump to end hit $

G jump to the last one

A jump to append or insert  
to select all hit B+lower arrow

:set number

:syntax on 🡪 highlights

:set tabstop=2

**Git Command Prompt**

C:\Users\Padmaja.N>mkdir gitdemodir

C:\Users\Padmaja.N>cd gitdemodir

C:\Users\Padmaja.N\gitdemodir>git init .

Initialized empty Git repository in C:/Users/Padmaja.N/gitdemodir/.git/

Command to view content of directory with hidden folders as well :

C:\Users\Padmaja.N\gitdemodir>dir /ah

in git bash terminal, if terminal stuck, after any command then hit ctrl+z.

To find a file in directory the command is:

file:

$ find . -name while\_loop.sh

./while\_loop.sh

$ find ./ -name while\_loop.sh

./while\_loop.sh

$ find -name while\_loop.sh

./while\_loop.sh

folder:

$ find . -name gitproject

./gitproject

to list multiple file, the cmd is:

ls w\* or ls f\*