EXPERIMENT 3

AIM:- To Perform various GIT operations on local and Remote repositories using GIT Cheat-Sheet

THEORY:-

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. Git is easy to learn and has a tiny footprint with lightning fast performance. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like cheap local branching, convenient staging areas, and multiple workflows.

Some of the basic operations in Git are:

1. Initialize

2. Add

3. Commit

4. Pull

5. Push

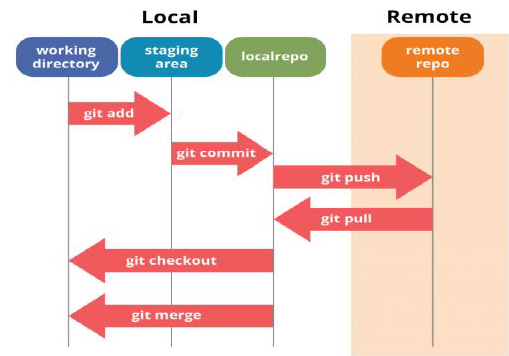
Some advanced Git operations are:

1. Branching

2. Merging

3. Rebasing

The following diagram depict the all supported operations in GIT



Installation of GIT:-

1) In windows, download GIT from https://git-scm.com/ and perform the straightforward installation.

2) In Ubuntu, install GIT using $sudo apt install git, Confirm the version after installation using command $git –version

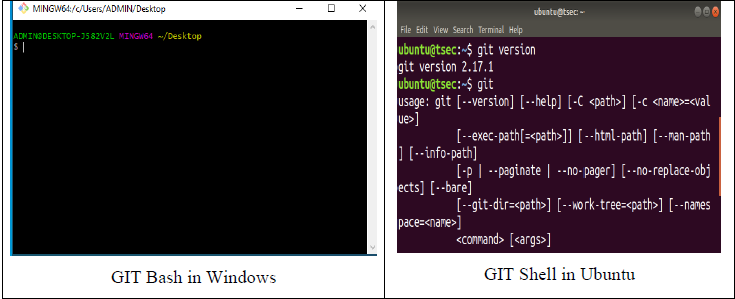




Once installation is done, open the terminal in Ubuntu and perform the following steps or in windows Right click and select Git bash here.



The output of GIT Bash in windows and GIT shell in Ubuntu is shown below



To perform version control, let us create a directory dvcs (Distributed version control system) and change directory to dvcs.

$ mkdir git-dvcs

$ cd git-dvcs/

15L@203-009 MINGW64 ~/Desktop/M-111 (master)

$ git init

Initialized empty Git repository in C:/Users/15L/Desktop/M-111/.git/

15L@203-009 MINGW64 ~/Desktop/M-111 (master)

$ git status

On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

15L@203-009 MINGW64 ~/Desktop/M-111 (master)

$ git version

git version 2.39.1.windows.1

15L@203-009 MINGW64 ~/Desktop/M-111 (master)

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

15L@203-009 MINGW64 ~/Desktop/M-111 (master)

$ git config --global user.email "Anshiet2003-122759@gmail.com"

15L@203-009 MINGW64 ~/Desktop/M-111 (master)

$ git config --global --list

user.name=anshiet

user.email=anshietu@gmail.com

color.ui=true

15L@203-009 MINGW64 ~/Desktop/M-111 (master)

$ mkdir git-demo-project

15L@203-009 MINGW64 ~/Desktop/M-111 (master)

$ cd git-demo-project/

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ ls -a

./ ../

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ touch index.html

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git add .

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git commit -m "First Commit"

[master (root-commit) bcdc1ad] First Commit

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 git-demo-project/index.html

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ touch index.css

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git add .

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git commit -an "express commit"

fatal: paths 'express commit ...' with -a does not make sense

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git commit -am "express Commit"

[master 2d333b7] express Commit

3 files changed, 1 insertion(+)

create mode 100644 git-demo-project/index.css

create mode 100644 git-demo-project/style.css

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git add .

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git commit -m "Initial setup"

On branch master

nothing to commit, working tree clean

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git log

commit 2d333b700c323d6e070a15d6350cca96a53bca1c (HEAD -> master)

Author: anshiet < anshietu@gmail.com >

Date: Tue Jan 16 14:08:25 2024 +0530

express Commit

commit bcdc1ad9164fab125e339664bf562eb03443061a

Author: anshiet < anshietu@gmail.com >

Date: Tue Jan 16 14:06:56 2024 +0530

First Commit

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git log --oneline

2d333b7 (HEAD -> master) express Commit

bcdc1ad First Commit

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git remote add origin https://github.com/Anshiet2003-122/SEPM-LAB-1.git

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (master)

$ git branch -M main

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git push -u origin main

To https://github.com/anshiet-122/SEPM-LAB-1.git

! [rejected] main -> main (fetch first)

error: failed to push some refs to 'https://github.com/Anshiet2003-122/SEPM-LAB-1.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git push -u origin main

To https://github.com/Anshiet2003-122/SEPM-LAB-1.git

! [rejected] main -> main (fetch first)

error: failed to push some refs to 'https://github.com/Anshiet2003-122/SEPM-LAB-1.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git remote add origin https://github.com/Anshiet2003-122/SEPM-LAB-1.git

error: remote origin already exists.

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git branch-M main

git: 'branch-M' is not a git command. See 'git --help'.

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git push -u origin main

To https://github.com/Anshiet2003-122/SEPM-LAB-1.git

! [rejected] main -> main (fetch first)

error: failed to push some refs to 'https://github.com/Anshiet2003-122/SEPM-LAB-1.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git branch -M MAIN

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (MAIN)

$ git branch -M main

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git push -u origin main

To https://github.com/anshiet-122/SEPM-LAB-1.git

! [rejected] main -> main (fetch first)

error: failed to push some refs to 'https://github.com/Anshiet2003-122/SEPM-LAB-1.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git remote set-url origin https://github.com/anshiet2003/sepm122.git

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git remote -v

origin https://github.com/Anshiet2003-122/sepm1.git (fetch)

origin https://github.com/Anshiet2003-122/sepm1.git (push)

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git branch -M main

15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$ git push -u origin main

Enumerating objects: 8, done.

Counting objects: 100% (8/8), done.

Delta compression using up to 20 threads

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

Writing objects: 100% (8/8), 587 bytes | 587.00 KiB/s, done.

Total 8 (delta 0), reused 0 (delta 0), pack-reused 0

To https://github.com/Anshiet2003-122/sepm1.git

\* [new branch] main -> main

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

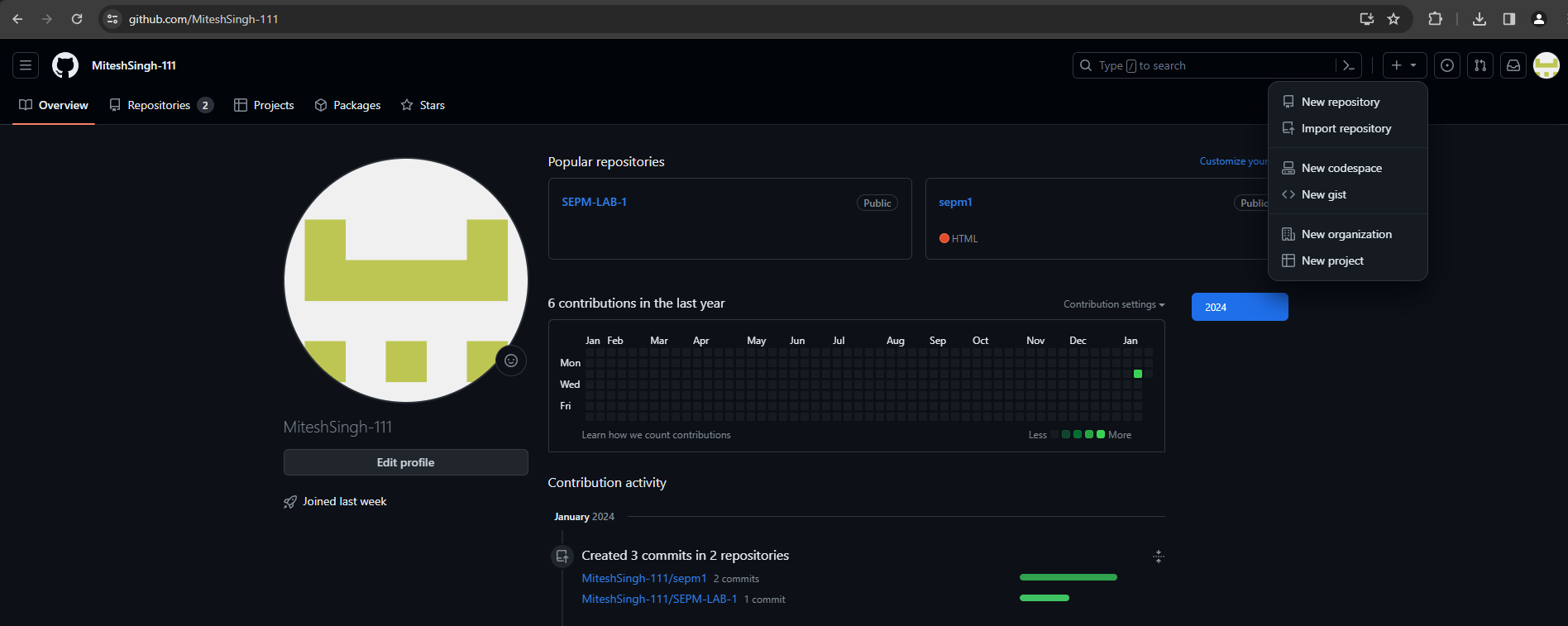
15L@203-009 MINGW64 ~/Desktop/M-111/git-demo-project (main)

$

**Example 2: Performing Version control in GITHUB with Pull and Push commands**

First open Github.com and create a new account.After verifying account through E-mail, create a Repository on github.com.

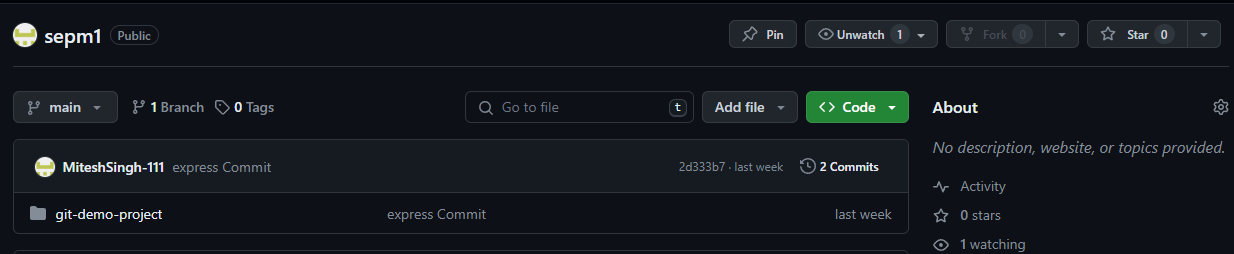
Open github.com→ create an account→After login Select New repository from the menu.



Now Specify a Name to repository and select public option followed by create repository

By default, we can create public repository in Github. So we can copy the entire public repository of any other users in to own account using “FORK” Operation. Now fork the repository (Sharing with other users who wants to contribute).

Login with another account→Copy and Paste URL of repository→then just click on fork to clone to others account. Suppose we want to fork public repository “timetracker”. So search for “timetracker” github repository on google and once its opened clicked on “Fork button” from the top of the github web page as shown below.



**Pull and Push Processes**

The pull command used to fetch the repository from github to local while push is used to commit files from local repository to Github.

Push → Push changes to Web repository

Pull → Pull changes to Local repository

The following commands are used for pull and push repositories

1. **A) Push command**

$ git remote add origin https://github.com/Anshiet2003-122/sepm1.git

$ git remote show origin

If you add remote again then will show you fatal error.

$ git remote add origin https://github.com/bhushanjadhav1/Myrepository.git

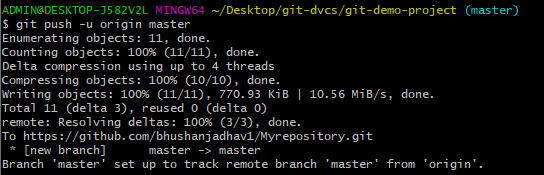
fatal: remote origin already exists.

So, to delete origin rm origin command is used

$ git remote rm origin

Now, to push the local repository to remote github following command is used

$ git push -u origin master



CONCLUSION:- Performed various GIT operations on local and Remote repositories using GIT Cheat-Sheet.