Experiment No. 1

AIM:-

To perform various GIT operations on local and remote repositories using GIT cheat-sheet.

Theory:-

Git add:

Moves changes from the working directory to the staging area. This gives you the opportunity to prepare a snapshot before committing it to the official history.

Git branch:

This command is your general-purpose branch administration tool. It lets you create isolated development environments within a single repository.

Git clone:

Creates a copy of an existing Git repository. Cloning is the most common way for developers to obtain a working copy of a central repository.

Git commit:

Takes the staged snapshot and commits it to the project history. Combined with git add, this defines the basic workflow for all Git users.

git config:

A convenient way to set configuration options for your Git installation. You'll typically only need to use this immediately after installing Git on a new development machine.

git init:

Initializes a new Git repository. If you want to place a project under revision control, this is the first command you need to learn.

git pull:

Pulling is the automated version of git fetch. It downloads a branch from a remote repository, then immediately merges it into the current branch. This is the Git equivalent of svn update.

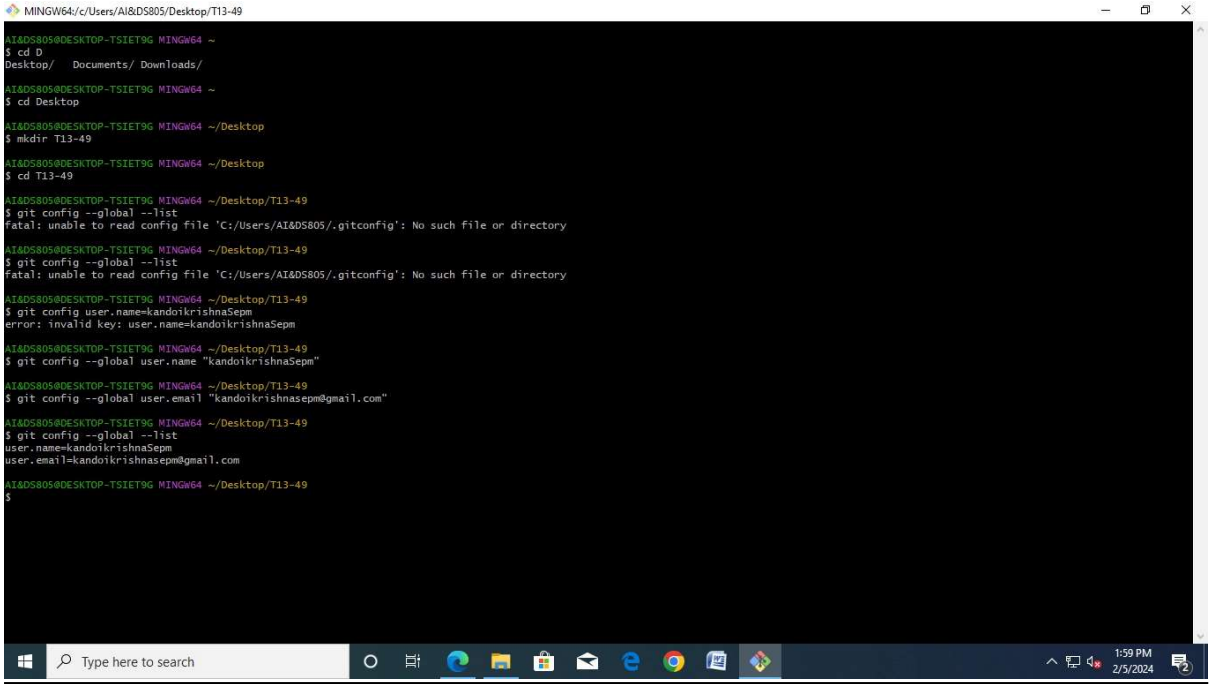
git push:

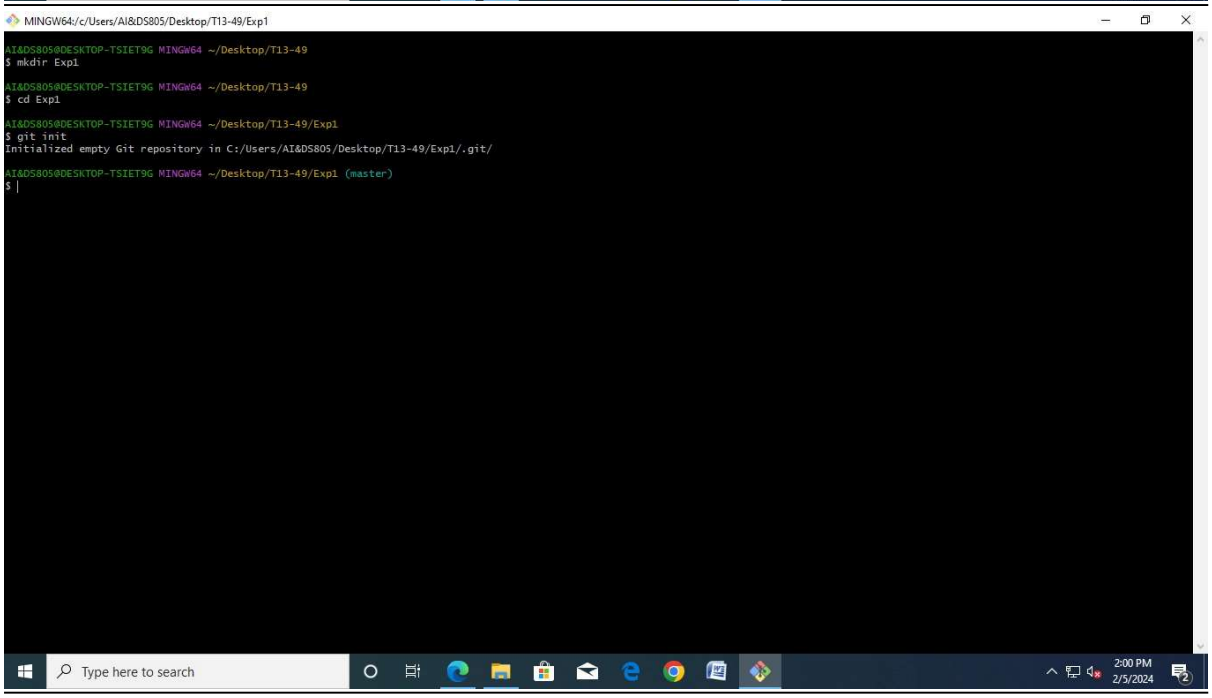
Pushing is the opposite of fetching (with a few caveats). It lets you move a local branch to another repository, which serves as a convenient way to publish contributions. This is like svn commit, but it sends a series of commits instead of a single changeset.

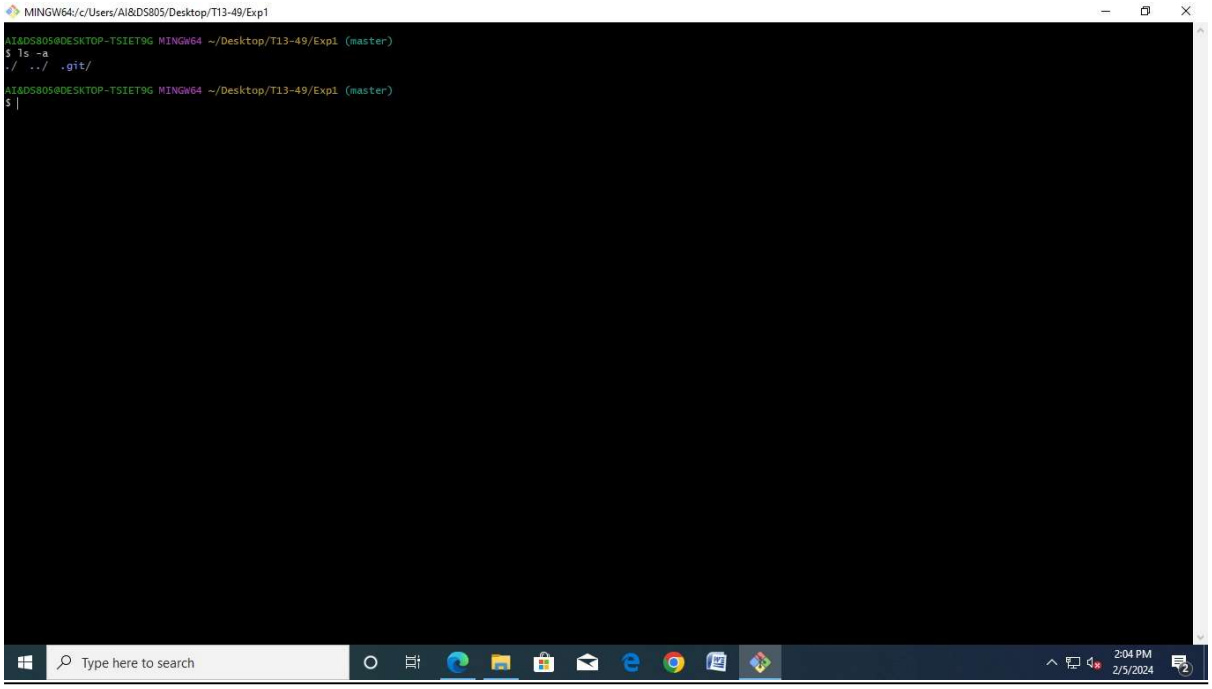
git status:

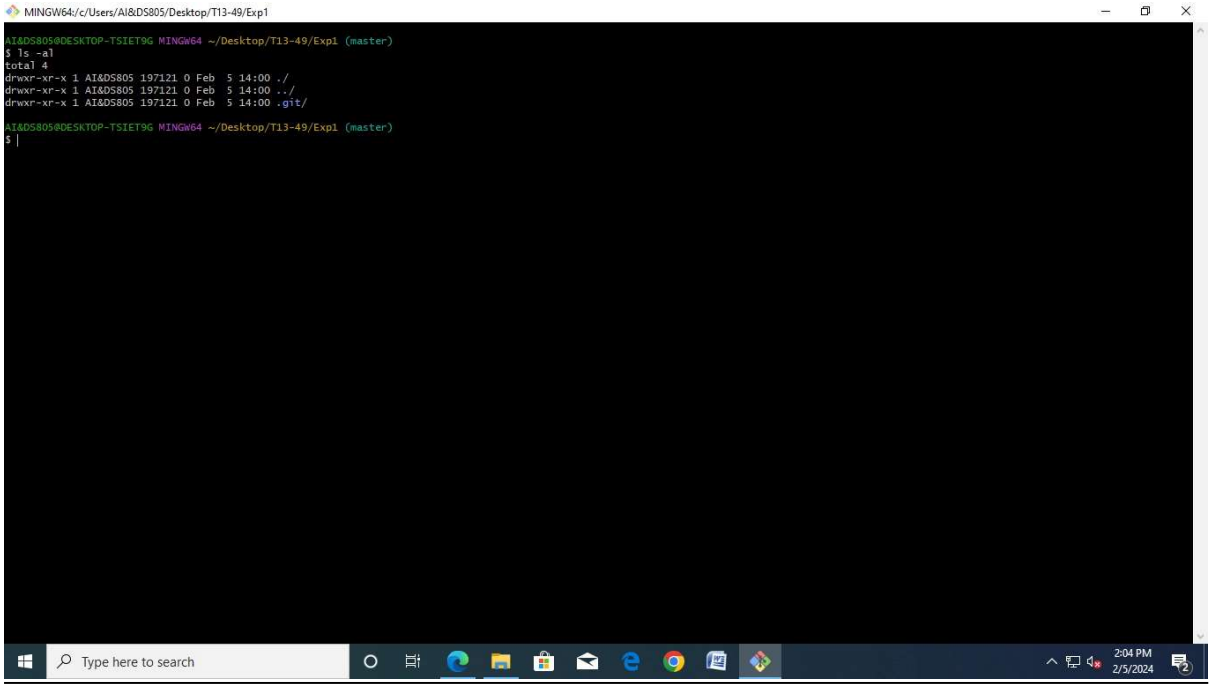
Displays the state of the working directory and the staged snapshot. You'll want to run this in conjunction with git add and git commit to see exactly what's being included in the next snapshot.

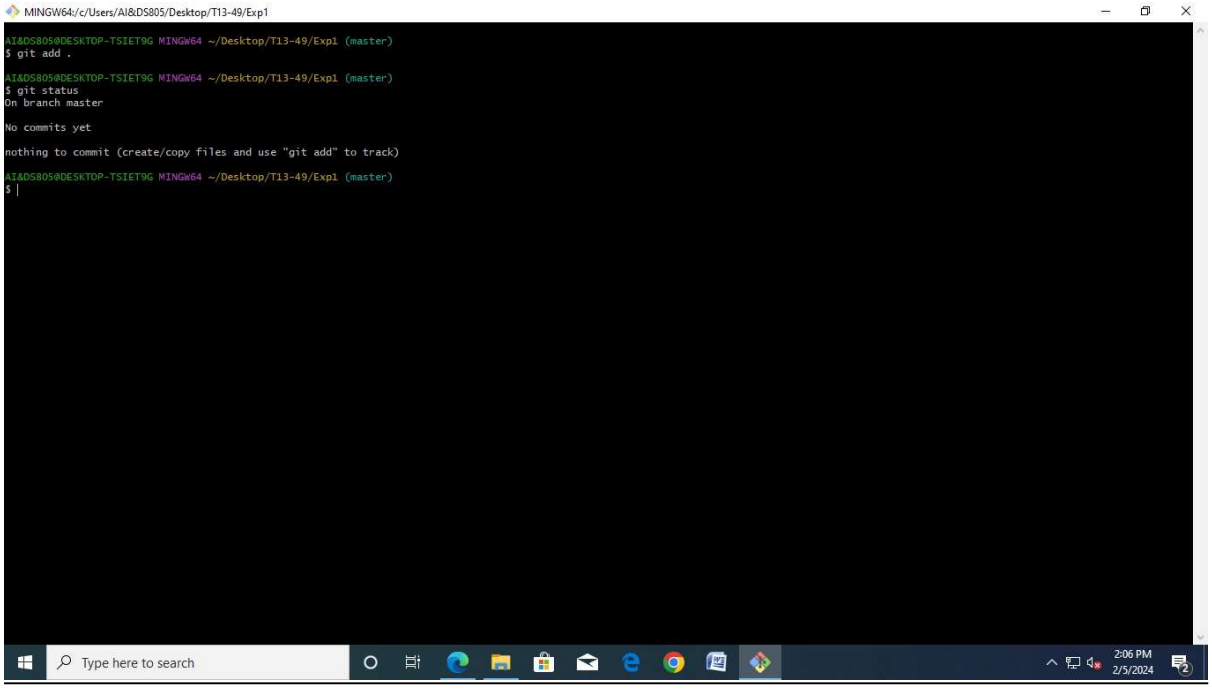
Commands

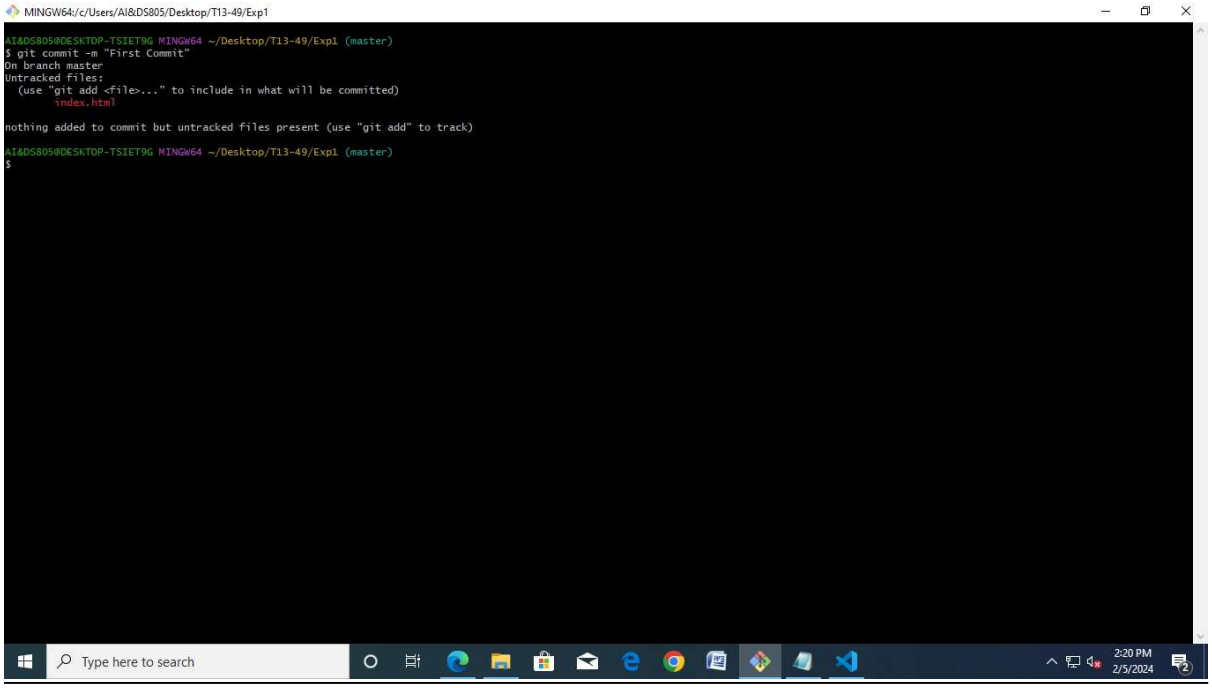

MINGW64:/c/Users/AI&DS805/Desktop/T13-49
AI&DS805@DESKTOP-TSIET9G MINGW64 ~
$ cd D
Desktop/ Documents/ Downloads/
AI&DS805@DESKTOP-TSIET9G MINGW64 ~
$ cd Desktop
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop
$ mkdir T13-49
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop
$ cd T13-49
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49
$ git config --global --list
fatal: unable to read config file 'C:/Users/AI&DS805/.gitconfig': No such file or directory
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49
$ git config --global --list
fatal: unable to read config file 'C:/Users/AI&DS805/.gitconfig': No such file or directory
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49
$ git config user.name=kandoikrishnaSepm
error: invalid key: user.name=kandoikrishnaSepm
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49
$ git config --global user.name "kandoikrishnaSepm"
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49
$ git config --global user.email "kandoikrishnasepm@gmail.com"
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49
$ git config --global --list
user.name=kandoikrishnaSepm
user.email=kandoikrishnasepm@gmail.com
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49
$
Type here to search
1:59 PM
2/5/2024

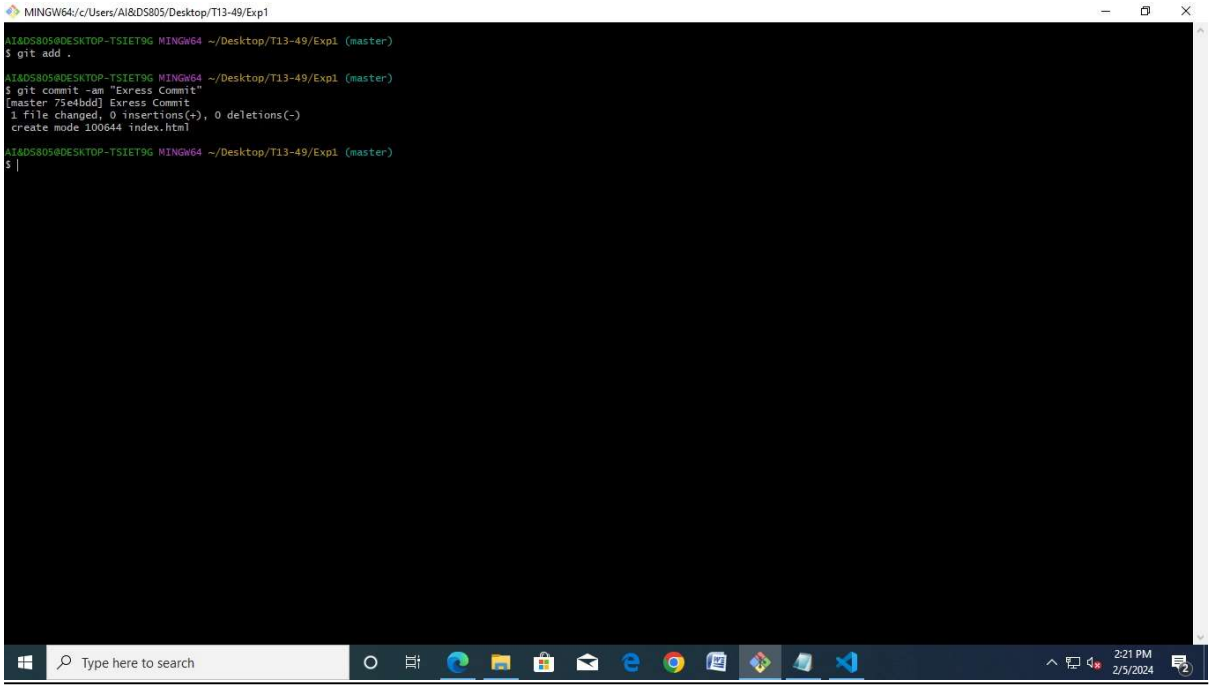

MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49
$ mkdir Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49
$ cd Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1
$ git init
Initialized empty Git repository in C:/Users/AI&DS805/Desktop/T13-49/Exp1/.git/
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:00 PM
2/5/2024

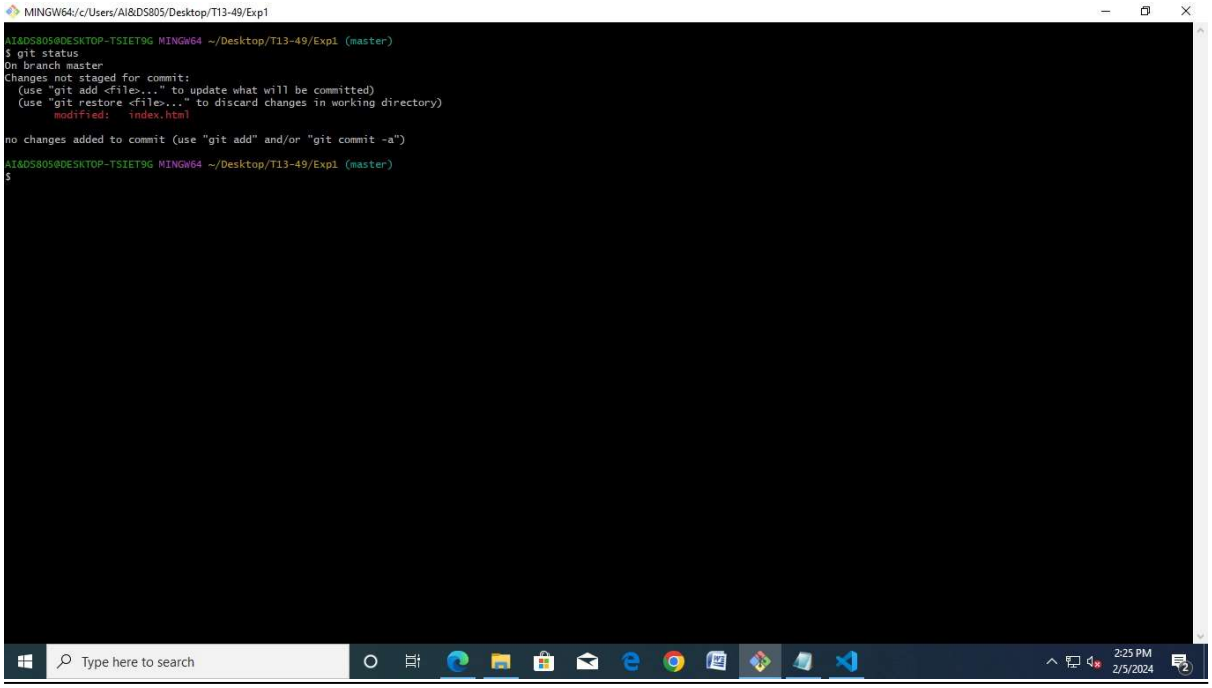

MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ ls -a
./ ../ .git/
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:04 PM
2/5/2024

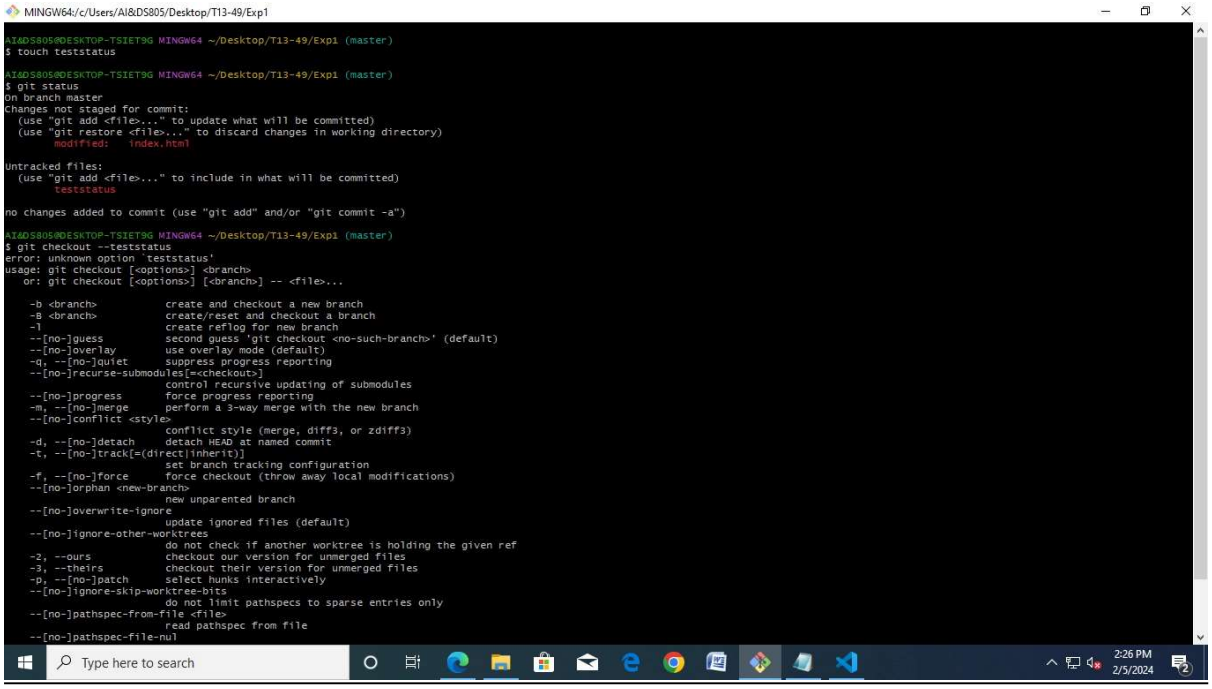

MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ ls -al
total 4
drwxr-xr-x 1 AI&DS805 197121 0 Feb 5 14:00 ./
drwxr-xr-x 1 AI&DS805 197121 0 Feb 5 14:00 ../
drwxr-xr-x 1 AI&DS805 197121 0 Feb 5 14:00 .git/
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:04 PM
2/5/2024

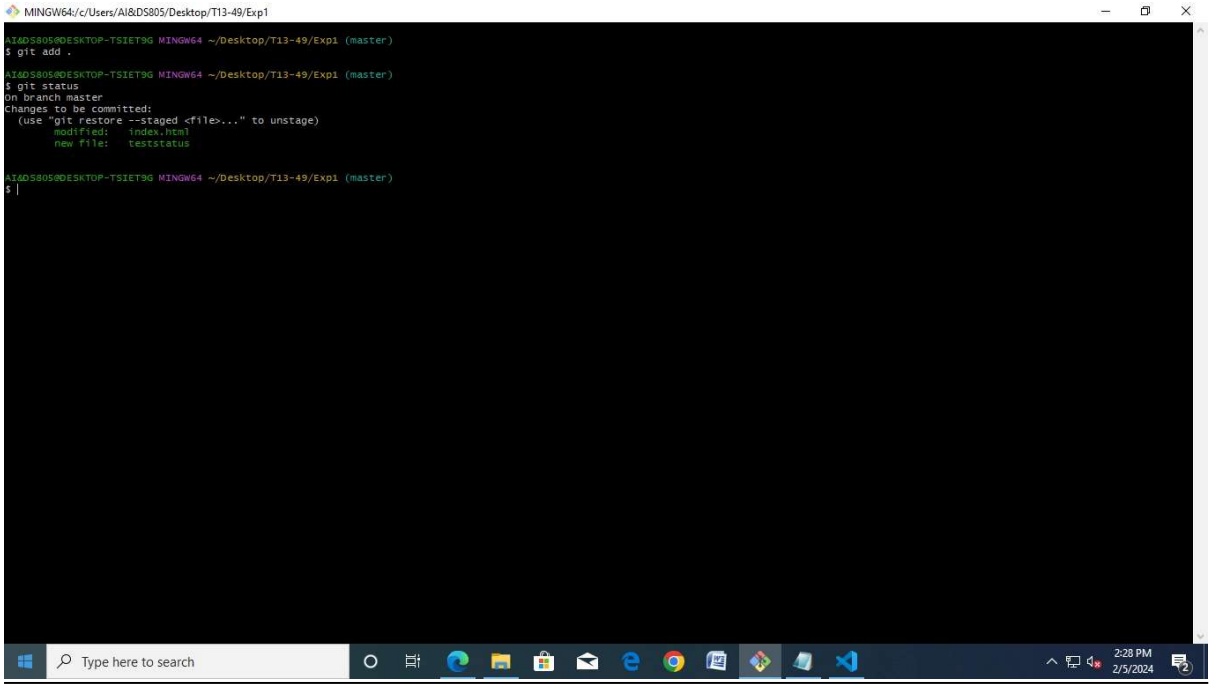

MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git add .
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git status
On branch master
No commits yet
nothing to commit (create/copy files and use "git add" to track)
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:06 PM
2/5/2024

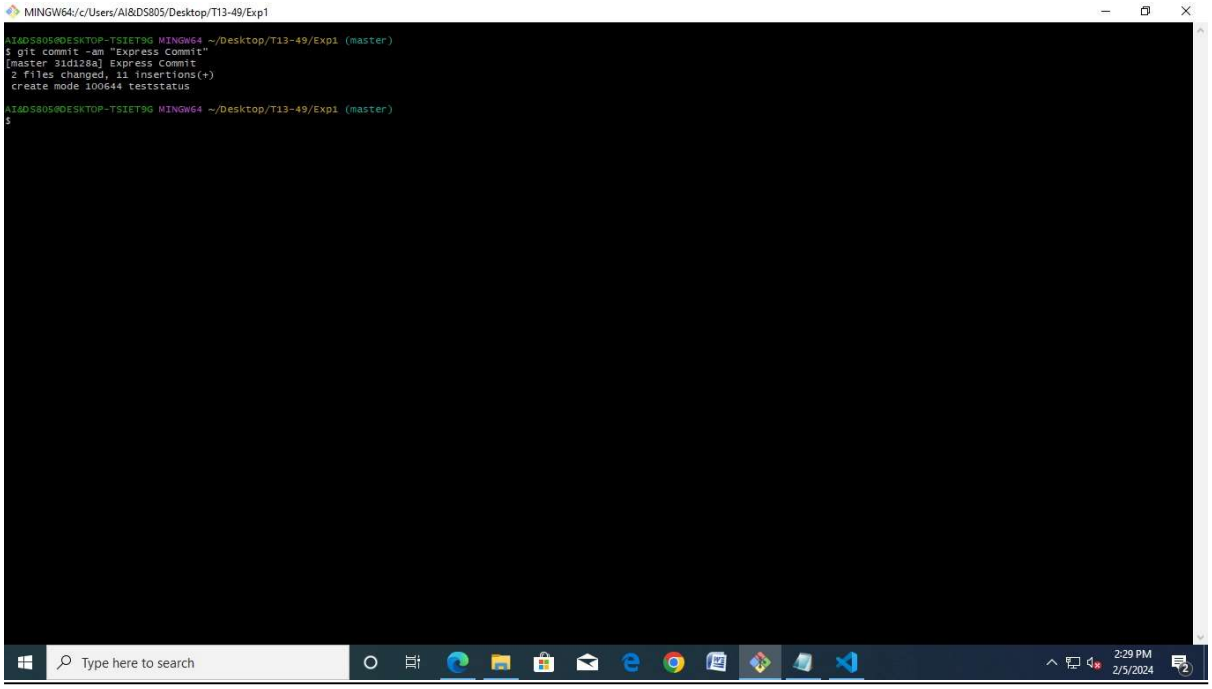

MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git commit -m "First Commit"
On branch master
Untracked files:
(use "git add <file>..." to include in what will be committed)
index.html
nothing added to commit but untracked files present (use "git add" to track)
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:20 PM
2/5/2024

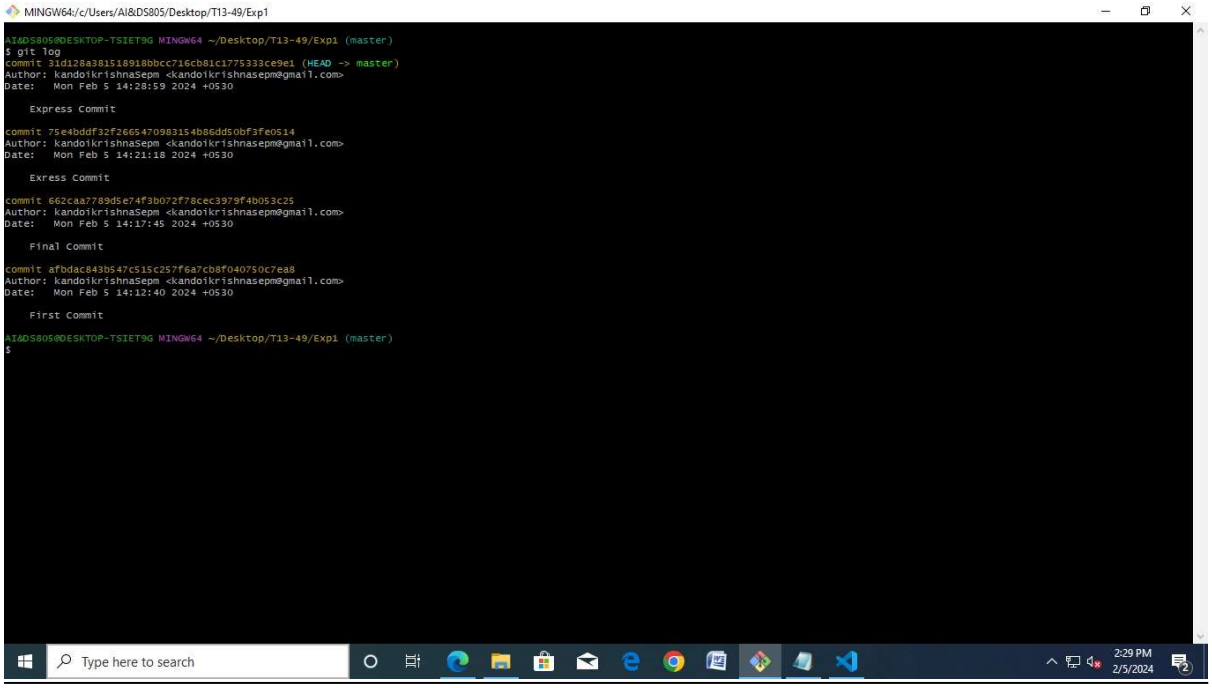

MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git add .
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git commit -am "Exress Commit"
[master 75e4bdd] Exress Commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 index.html
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:21 PM
2/5/2024

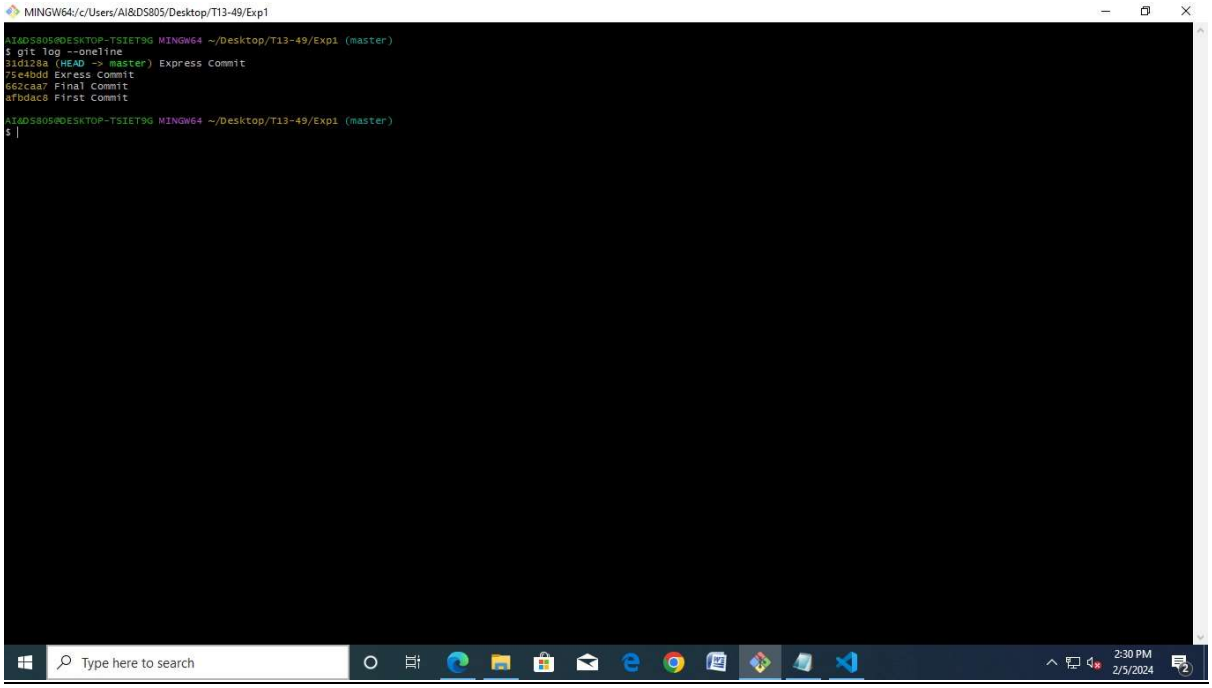

MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git status
On branch master
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified: index.html
no changes added to commit (use "git add" and/or "git commit -a")
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:25 PM
2/5/2024

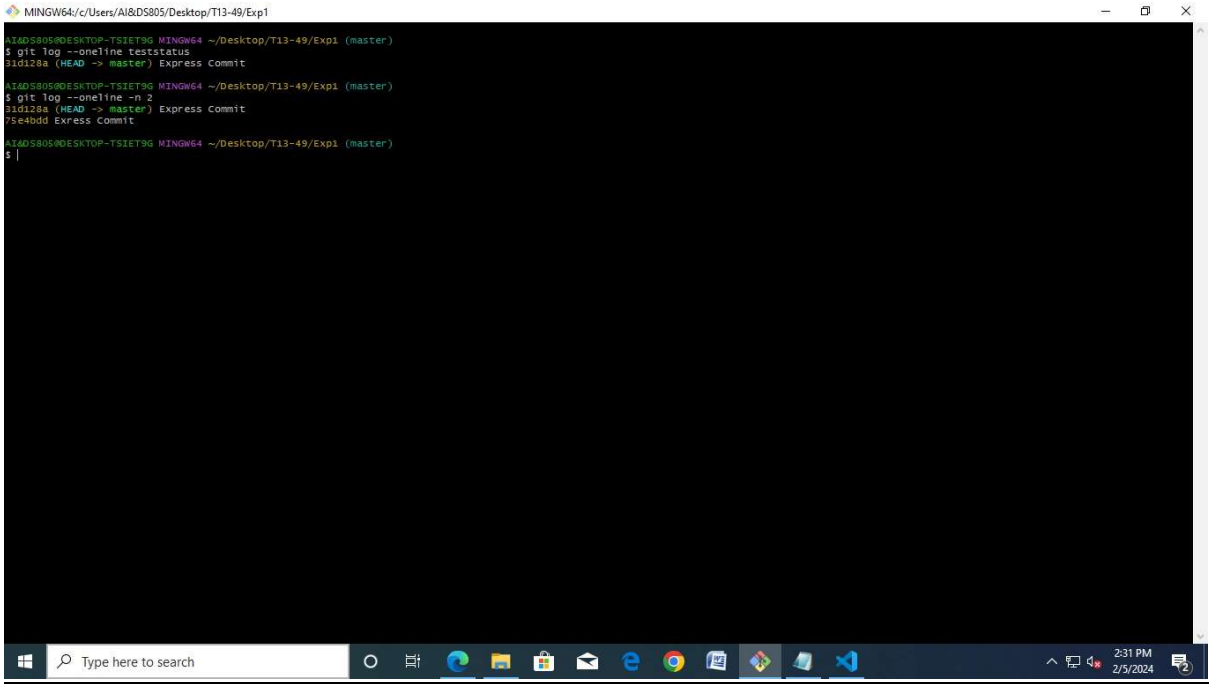

MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ touch teststatus
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git status
On branch master
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified: index.html
Untracked files:
(use "git add <file>..." to include in what will be committed)
teststatus
no changes added to commit (use "git add" and/or "git commit -a")
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git checkout --teststatus
error: unknown option `teststatus'
usage: git checkout [<options>] <branch>
or: git checkout [<options>] [<branch>] -- <file>...
-b <branch> create and checkout a new branch
-B <branch> create/reset and checkout a branch
-l create reflog for new branch
--[no-]guess second guess 'git checkout <no-such-branch>' (default)
--[no-]overlay use overlay mode (default)
-q, --[no-]quiet suppress progress reporting
--[no-]recurse-submodules[=<checkout>]
control recursive updating of submodules
--[no-]progress force progress reporting
-m, --[no-]merge perform a 3-way merge with the new branch
--[no-]conflict <style>
conflict style (merge, diff3, or zdiff3)
-d, --[no-]detach detach HEAD at named commit
-t, --[no-]track[=(direct|inherit)]
set branch tracking configuration
-f, --[no-]force force checkout (throw away local modifications)
--[no-]orphan <new-branch>
new unparented branch
--[no-]overwrite-ignore
update ignored files (default)
--[no-]ignore-other-worktrees
do not check if another worktree is holding the given ref
-2, --ours checkout our version for unmerged files
-3, --theirs checkout their version for unmerged files
-p, --[no-]patch select hunks interactively
--[no-]ignore-skip-worktree-bits
do not limit pathspecs to sparse entries only
--[no-]pathspec-from-file <file>
read pathspec from file
--[no-]pathspec-file-nul
Type here to search
2:26 PM
2/5/2024


MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git add .
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git status
On branch master
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
modified: index.html
new file: teststatus
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:28 PM
2/5/2024


MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git commit -am "Express Commit"
[master 31d128a] Express Commit
2 files changed, 11 insertions(+)
create mode 100644 teststatus
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:29 PM
2/5/2024


MINGW64:/c/Users/AI&DS805/Desktop/T13-49/Exp1
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$ git log
commit 31d128a381518918bbcc716cb81c1775333ce9e1 (HEAD -> master)
Author: kandoikrishnaSepm <kandoikrishnasepm@gmail.com>
Date: Mon Feb 5 14:28:59 2024 +0530
Express Commit
commit 75e4bddf32f26654709831546b86dd50bf3fe514
Author: kandoikrishnaSepm <kandoikrishnasepm@gmail.com>
Date: Mon Feb 5 14:21:18 2024 +0530
Exress Commit
commit 662caa7789d5e74f3b072f78cec3979f4b053c25
Author: kandoikrishnaSepm <kandoikrishnasepm@gmail.com>
Date: Mon Feb 5 14:17:45 2024 +0530
Final Commit
commit afbdac843b547c515c257f6a7cb8f040750c7ea8
Author: kandoikrishnaSepm <kandoikrishnasepm@gmail.com>
Date: Mon Feb 5 14:12:40 2024 +0530
First Commit
AI&DS805@DESKTOP-TSIET9G MINGW64 ~/Desktop/T13-49/Exp1 (master)
$
Type here to search
2:29 PM
2/5/2024





Conclusion :

We have successfully implemented Git Commands.