

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/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git log --oneline -n 2
76cd068 (HEAD -> master) Express commit
c0e9999 Express commit

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git log --oneline teststatus
76cd068 (HEAD -> master) Express commit

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$
```

```
MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git log --oneline teststatus
76cd068 (HEAD -> master) Express commit

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$
```

```
MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git log --oneline
76cd068 (HEAD -> master) Express commit
c0e9999 Express commit
4fca12e first commit
3fa5d68 first commit

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$
$ git log
commit 76cd06858a2fd525d470e11e6b815b52d5e498f (HEAD -> master)
Author: khatrimeetsepm <khatrimeetsepm@gmail.com>
Date: Mon Feb 5 14:29:14 2024 +0530

    Express commit

commit c0e99995a67c4c99e2bbf6da2e2db44eead366a8
Author: khatrimeetsepm <khatrimeetsepm@gmail.com>
Date: Mon Feb 5 14:24:44 2024 +0530

    Express commit

commit 4fca12e185dd6aabe283b7fa46dedfca981fd08f
Author: khatrimeetsepm <khatrimeetsepm@gmail.com>
Date: Mon Feb 5 14:20:07 2024 +0530

    first commit

commit 3fa5d68348599bf777ea3091f465b04edab7e8b9
Author: khatrimeetsepm <khatrimeetsepm@gmail.com>
Date: Mon Feb 5 14:12:47 2024 +0530

    first commit

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$
```

```
MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git commit -m "Express commit"
[master 76cd068] Express commit
2 files changed, 12 insertions(+)
create mode 100644 teststatus
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ |

MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git add .
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (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-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$
```

```
MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ touch teststatus
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (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-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (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)
--no-progress        suppress progress reporting
--no-recurse-submodules=<checkout>
                    control recursive updating of submodules
--no-merge           force progress reporting
--no-conflict-style  perform a 3-way merge with the new branch
                    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
-j, --ours           checkout our version for unmerged files
-J, --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-must
                    with --pathspec-from-file, pathspec elements are separated with NUL character

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git add .
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git commit -am "Express commit"
[master c0e9909] Express commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 index.html
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (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-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$
```

```
MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git add .
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git commit -am "Express commit"
[master c0e9999] Express commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 index.html
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$

MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (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-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ |
```

```
MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git add .

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ |

MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ ls -al
total 4
drwxr-xr-x 1 AI&DS805 197121 0 Feb  5 14:01 ./
drwxr-xr-x 1 AI&DS805 197121 0 Feb  5 14:01 ../
drwxr-xr-x 1 AI&DS805 197121 0 Feb  5 14:01 .git/
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ |
```

```
MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$ ls -a
./ ../ .git/
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$

MINGW64/c/Users/AI&DS805/Desktop/T-1353/EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353
$ mkdir EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353
$ cd EXP-1
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1
$ git init
Initialized empty Git repository in C:/Users/AI&DS805/Desktop/T-1353/EXP-1/.git/
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353/EXP-1 (master)
$
```



```
MINGW64/c/Users/AI&DS805/Desktop/T-1353
AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~
$ cd Desktop/

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop
$ mkdir T-1353

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop
$ cd T-1353

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353
$ git config --global user.name "khatrimeetsepm"

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353
$ git config --global user.email "khatrimeetsepm@gmail.com"

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353
$ git config --global --list
fatal: unable to read config file 'C:/Users/AI&DS805/.gitconfig': No such file or directory

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353
$ git config --global user.name "khatrimeetsepm"

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353
$ git config --global user.email "khatrimeetsepm@gmail.com"

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353
$ git config --global --list
user.name=khatrimeetsepm
user.email=khatrimeetsepm@gmail.com

AI&DS805@DESKTOP-ONLHRU4 MINGW64 ~/Desktop/T-1353
$ |
```

Conclusion :

We have successfully implemented Git Commands.