

EXPERIMENT – 3

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

```
15L@203-06 MINGW64 ~/Desktop/varad-git (master)
```

```
$ git config --global user.name "varad"
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git (master)
```

```
$ git config --global user.email varad232004@gmail.com
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git (master)
```

```
$ git config --global --list
```

```
user.name=shrey
```

```
user.email=varad232004@gmail.com
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git (master)
```

```
$ mkdir git-demo-project
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git (master)
```

```
$ cd git-demo-project
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
```

```
$ git init
```

```
Initialized empty Git repository in C:/Users/15L/Desktop/varad-git/git-demo-project/.git/
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
```

```
$ ls -a
```

```
./ ../ .git/
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
```

```
$ rm -rf .git/
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
```

```
$ rm -rf .git
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
```

```
$ ls -al total 0 drwxr-xr-x 1 15L 197121 0
```

```
Jan 23 13:40 ./ drwxr-xr-x 1 15L 197121 0
```

```
Jan 23 13:39 ../
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
```

```
$ git init
```

```
Initialized empty Git repository in C:/Users/15L/Desktop/varad-git/git-demo-project/.git/
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git add .
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git status
On branch master
```

```
No commits yet
```

```
Changes to be committed:
  (use "git rm --cached <file>..." to unstage) new
       file: sample.txt
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git commit -m "first commit"
[master (root-commit) d489bcf] first commit 1
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 sample.txt
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git status
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)
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git add .
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage) new
       file: index.html
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git commit -am "express commit"
[master bef012b] express commit
 1 file changed, 0 insertions(+), 0 deletions(-) create
 mode 100644 index.html
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git status
On branch master
nothing to commit, working tree clean
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
```

```
$ git log
commit bef012be210dc3856730420ce2228425d43e58f5 (HEAD -> master)
Author: shrey <sshrey30.shah@gmail.com>
Date: Tue Jan 23 13:43:06 2024 +0530

    express commit
    commit
    d489bcf61da3943db131c6d0274d9eac00737319
Author: shrey <sshrey30.shah@gmail.com>
Date: Tue Jan 23 13:41:06 2024 +0530

    first commit
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ nano index.html
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (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")

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git log
commit bef012be210dc3856730420ce2228425d43e58f5 (HEAD -> master)
Author: shrey <sshrey30.shah@gmail.com>
Date: Tue Jan 23 13:43:06 2024 +0530

    express commit
    commit
    d489bcf61da3943db131c6d0274d9eac00737319
Author: shrey <sshrey30.shah@gmail.com>
Date: Tue Jan 23 13:41:06 2024 +0530

    first commit
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ touch teststatus
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (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")

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)

```
$ git checkout -- teststatus error: pathspec 'teststatus' did not
match any file(s) known to git
```

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)

```
$ git checkout --index.html error: unknown option
`index.html' 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
--guess	second guess 'git checkout <no-such-branch>' (default)
--overlay	use overlay mode (default)
-q, --quiet	suppress progress reporting
--recurse-submodules[=<checkout>]	control recursive updating of submodules
--progress	force progress reporting
-m, --merge	perform a 3-way merge with the new branch
--conflict <style>	conflict style (merge, diff3, or zdiff3)
-d, --detach	detach HEAD at named commit -t, --
track[=(direct inherit)]	set branch tracking configuration
-f, --force	force checkout (throw away local modifications) --
orphan <new-branch>	new unparented branch
--overwrite-ignore	update ignored files (default) --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, --patch	select hunks interactively
--ignore-skip-worktree-bits	do not limit pathspecs to sparse entries
	only
--pathspec-from-file <file>	read pathspec
	from file
--pathspec-file-nul	with --pathspec-from-file, pathspec elements are separated with
NUL character	

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)

```
$ git add index.html warning: in the working copy of 'index.html', LF will be
replaced by CRLF the next time Git touches it
```

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)

```
$ git status
On branch master
Changes to be committed:
```

(use "git restore --staged <file>..." to unstage) modified:
index.html

Untracked files:

(use "git add <file>..." to include in what will be committed) teststatus

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)

\$ git add teststatus

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)

\$ git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to
unstage) modified: index.html new file:
teststatus

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)

\$ git commit -am "express commit"

[master 298a0e6] express commit

2 files changed, 1 insertion(+)

create mode 100644 teststatus

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)

\$ git status On branch master nothing

to commit, working tree clean

15L@203-06 MINGW64 ~/Desktop/varad-
git/git-demo-project (master)

\$ git log

commit 298a0e6689dd8b9d98ebc4e5413441e244b1036d (HEAD -> master)

Author: shrey <sshrey30.shah@gmail.com>

Date: Tue Jan 23 13:48:37 2024 +0530

express commit

commit
bef012be210dc3856730420ce2228425d43e58f5

Author: shrey <sshrey30.shah@gmail.com>

Date: Tue Jan 23 13:43:06 2024 +0530

express commit

commit
d489bcf61da3943db131c6d0274d9eac00737319

Author: shrey <sshrey30.shah@gmail.com>

Date: Tue Jan 23 13:41:06 2024 +0530

first commit

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)

\$ git log --oneline

```
298a0e6 (HEAD -> master) express commit
bef012b express commit d489bcf first
commit
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git remote add origin https://github.com/shrey-tsec/SEPM.git
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ rm -rf ^C
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ rm -rf https://github.com/shrey-tsec/SEPM.git
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git remote add origin https://github.com/shrey-tsec/EXP3.git
error: remote origin already exists.
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git origin git: 'origin' is not a git command. See
'git --help'.
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git show origin fatal: ambiguous argument 'origin': unknown revision or path not in
the working tree.
Use '--' to separate paths from revisions, like this:
'git <command> [<revision>...] -- [<file>...]'
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git remote show origin
* remote origin
  Fetch URL: https://github.com/shrey-tsec/SEPM.git
  Push URL: https://github.com/shrey-tsec/SEPM.git HEAD
  branch: (unknown)
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git remote remove origin
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git remote show origin fatal: 'origin' does not
appear to be a git repository fatal: Could not read
from remote repository.
```

Please make sure you have the correct access rights and the repository exists.

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git remote add origin https://github.com/shrey-tsec/EXP3.git
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git remote show origin
```

```
* remote origin
Fetch URL: https://github.com/shrey-tsec/EXP3.git
Push URL: https://github.com/shrey-tsec/EXP3.git HEAD
branch: (unknown)
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (master)
$ git branch -M main
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/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% (5/5), done.
Writing objects: 100% (8/8), 691 bytes | 691.00 KiB/s, done.
Total 8 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/shrey-tsec/EXP3.git
 * [new branch]    main -> main branch
'main' set up to track 'origin/main'.
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (main)
$ git pull Already
up to date.
15L@203-06 MINGW64
~/Desktop/varad-
git/git-demo-
project (main)

$ git pull remote: Enumerating objects: 5,
done. remote: Counting objects: 100% (5/5),
done. remote: Compressing objects: 100% (2/2),
done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 957 bytes | 239.00 KiB/s, done.
From https://github.com/shrey-tsec/EXP3
   298a0e6..7242627 main       -> origin/main
Updating 298a0e6..7242627
Fast-forward index.html
| 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (main)
$ git log --oneline origin/main
7242627 (HEAD -> main, origin/main) Update index.html
298a0e6 express commit
bef012b express commit
d489bcf first commit
```

```
15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (main)
$ $ git fetch bash: $:
command not found
```

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (main)

\$ git fetch

15L@203-06 MINGW64 ~/Desktop/varad-git/git-demo-project (main)

\$ git merge origin/main Already
up to date.

```
MINGW64/c/Users/HP/15L/git-dvcs/git-demo-project
HP 15L@203-06 MINGW64 ~
$ git version
git version 2.47.1.windows.2
HP 15L@203-06 MINGW64 ~
$ git
usage: git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
           [--exec-path=<path>] [--html-path] [--man-path] [--info-path]
           [-p | --paginate] | -p | --no-pager] [--no-replace-objects] [--no-lazy-fetch]
           [--no-optionallocks] [--no-advice] [--bare] [--git-dir=<path>]
           [--work-tree=<path>] [--namespace=<name>] [--config-env=<name>=<envvar>]
           <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  restore    Restore working tree files
  rm         Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  diff       Show changes between commits, commit and working tree, etc
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status

grow, mark and tweak your common history
  backfill   Download missing objects in a partial clone
  branch     List, create, or delete branches
  commit     Record changes to the repository
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  reset      Reset current HEAD to the specified state
  switch     Switch branches
  tag        Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
  fetch      Download objects and refs from another repository
  pull       Fetch from and integrate with another repository or a local branch
  push       Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.
HP 15L@203-06 MINGW64 ~
$ mkdir git-dvcs
HP 15L@203-06 MINGW64 ~
$ cd git-dvcs/
```

```
MINGW64/c/Users/HP/15L/git-dvcs/git-demo-project
HP 15L@203-06 MINGW64 ~/git-dvcs
$ git config --global --list
errors: unknown option 'global--list'
usage: git config list [<file-options>] [<display-options>] [--includes]
or: git config get [<file-options>] [<display-options>] [--includes] [--all] [--regex] [--value=<value>] [--fixed-value] [--default=<defaults>] <name>
or: git config set [<file-options>] [--types=<type>] [--all] [--value=<value>] [--fixed-value] <name> <value>
or: git config unset [<file-options>] [--all] [--value=<value>] [--fixed-value] <name>
or: git config rename-section [<file-options>] <old-name> <new-name>
or: git config remove-section [<file-options>] <name>
or: git config edit [<file-options>]
or: git config [<file-options>] --get-colorbool <name> [<stdout-is-atty>]

Config file location
--[no-]global      use global config file
--[no-]system      use system config file
--[no-]local       use repository config file
--[no-]worktree    use per-worktree config file
-f, --[no-]file <file>
                  use given config file
--[no-]blob <blob-id>
                  read config from given blob object

Action
--get              get value: name [<value-pattern>]
--get-all         get all values: key [<value-pattern>]
--get-regex       get values for regex: name-regex [<value-pattern>]
--get-urlmatch    get value specific for the URL: section[.var] URL
--replace-all     replace all matching variables: name value [<value-pattern>]
--add             add a new variable: name value
--unset           remove a variable: name [<value-pattern>]
--unset-all       remove all matches: name [<value-pattern>]
--rename-section  rename section: old-name new-name
--remove-section  remove a section: name
-l, --list        list all
-e, --edit        open an editor
--get-color       find the color configured: slot [<default>]
--get-colorbool   find the color setting: slot [<stdout-is-atty>]

Display options
-z, --[no-]null    terminate values with NUL byte
--[no-]name-only   show variable names only
--[no-]show-origin show origin of config (file, standard input, blob, command line)
--[no-]show-scope  show scope of config (worktree, local, global, system, command)
--[no-]show-names  show config keys in addition to their values

Type
-t, --[no-]type <type>
                  value is given this type
--bool            value is "true" or "false"
--int            value is decimal number
--bool-or-int     value is --bool or --int
--bool-or-str     value is --bool or string
--path           value is a path (file or directory name)
--expiry-date     value is an expiry date

Other
--[no-]default <value> with --get, use default value when missing entry
--[no-]comment <value>
```



```
MINGW64/c/Users/HP 15L/git-dvcs/git-demo-project
HP 15L0805-38 MINGW64 ~
$ mkdir git-dvcs
HP 15L0805-38 MINGW64 ~
$ cd git-dvcs/
HP 15L0805-38 MINGW64 ~/git-dvcs
$ git config --global
error: did you mean '--global' (with two dashes)?
HP 15L0805-38 MINGW64 ~/git-dvcs
$ git config --global
error: no action specified
HP 15L0805-38 MINGW64 ~/git-dvcs
$ git config --global user.name = "Varad"
HP 15L0805-38 MINGW64 ~/git-dvcs
$ git config --global user.email = "Varad232004@gmail.com"
HP 15L0805-38 MINGW64 ~/git-dvcs
$ git config --global --list
user.name=
user.email=
HP 15L0805-38 MINGW64 ~/git-dvcs
$ cat ~/.gitconfig
[user]
    name =
    email =
HP 15L0805-38 MINGW64 ~/git-dvcs
$ git config --global user.name "Varad"
HP 15L0805-38 MINGW64 ~/git-dvcs
$ git config --global user.email "Varad232004@gmail.com"
HP 15L0805-38 MINGW64 ~/git-dvcs
$ cat ~/.gitconfig
[user]
    name = Varad
    email = Varad232004@gmail.com
HP 15L0805-38 MINGW64 ~/git-dvcs
$ mkdir git-demo-project
HP 15L0805-38 MINGW64 ~/git-dvcs
$ cd git-demo-project/
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project
$ git init
Initialized empty Git repository in C:/Users/HP 15L/git-dvcs/git-demo-project/.git/
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$
```

BAJFINANCE
+1.01%

Search



ENG IN 13:45 30-01-2025

```
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ ls -al
total 4
drwxr-xr-x 1 HP 15L 197121 0 Jan 30 13:45 ./
drwxr-xr-x 1 HP 15L 197121 0 Jan 30 13:43 ../
drwxr-xr-x 1 HP 15L 197121 0 Jan 30 13:45 .git/
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git add .
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   index.html
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git commit -m "First Commit"
[master (root-commit) e32e3d3] First Commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index.html
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ nano index.html
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git add .
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git commit -am "Express Commit"
[master 79a0fa0] Express Commit
 1 file changed, 35 insertions(+)
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ touch teststatus
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        teststatus

nothing added to commit but untracked files present (use "git add" to track)
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git checkout -- teststatus
error: pathspec 'teststatus' did not match any file(s) known to git
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$

MINGW64/c/Users/HP 15L/git-dvcs/git-demo-project
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git add .
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git status
On branch master

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   index.html
        new file:   teststatus
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git commit -m "Third Commit"
[master 35a3bad] Third Commit
 2 files changed, 1 insertion(+), 1 deletion(-)
 create mode 100644 teststatus
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git log
commit 35a3bad0eccc8ba6c4852ed9e10ca368fbf2dec (HEAD -> master)
Author: Varad <Varad232004@gmail.com>
Date:   Thu Jan 30 14:04:37 2025 +0530

    Third Commit

commit 79a0fa0bbde5e63e326e3af01af9162adfded5c3
Author: Varad <Varad232004@gmail.com>
Date:   Thu Jan 30 13:56:15 2025 +0530

    Express Commit

commit e32e3d3316bd4d713d65c9c59d1d8bf6b4d3eb35
Author: Varad <Varad232004@gmail.com>
Date:   Thu Jan 30 13:52:00 2025 +0530

    First Commit
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git log --oneline
35a3bad (HEAD -> master) Third Commit
79a0fa0 Express Commit
e32e3d3 First Commit
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git log --oneline teststatus
35a3bad (HEAD -> master) Third Commit
79a0fa0 Express Commit
e32e3d3 First Commit
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git log --oneline 79a0fa0..e32e3d3
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ git log --oneline -n 2
35a3bad (HEAD -> master) Third Commit
79a0fa0 Express Commit
HP 15L0805-38 MINGW64 ~/git-dvcs/git-demo-project (master)
$ |
```

CONCLUSION:

Thus, we have successfully studied and performed various GIT operations on local and Remote repositories using GIT Cheat-Sheet.