VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“Jnana Sangama”, Machhe, Belagavi, Karnataka-590018

Lab Experiment Record

Project Management with Git [BCSL58C]

Submitted in partial fulfillment towards AEC of 3rd semester of

Bachelor of Engineering
in
Computer Science and Engineering
(Artificial Intelligence & Machine Learning)

Submitted by

SINCHANA
4GW24CI045

DEPARTMENT OF CSE (ArtificialIntelligence & Machine Learning)

GSSS INSTITUTE OF ENGINEERING& TECHNOLOGY FOR WOMEN

(Affiliated to VTU, Belagavi, Approved by AICTE, New Delhi & Govt. of Karnataka)

K.R.S ROAD, METAGALLI, MYSURU-570016, KARNATAKA

(Accreditedby NAAC)

TABLE OF CONTENT

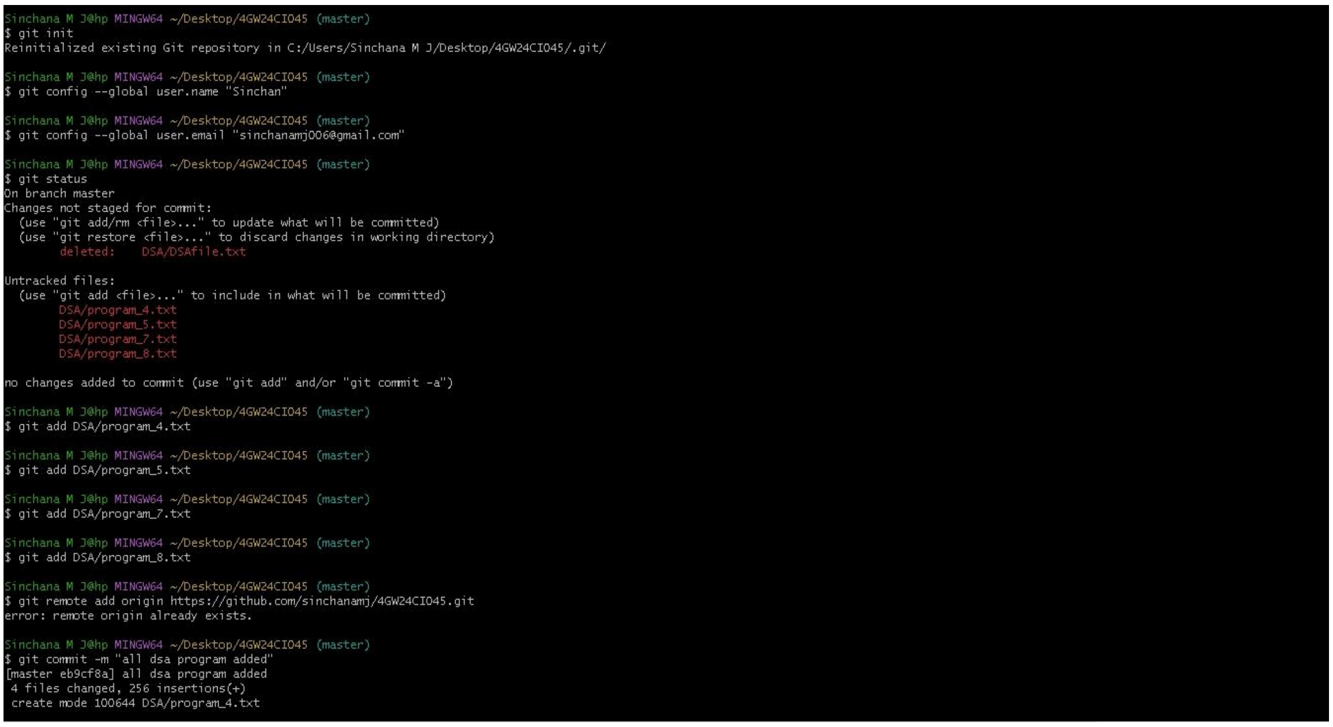
EXPERIMENT 1

SETTING UP AND BASIC COMMANDS

It is the learning and configuring basic git commands that are essential to manage projects with groups.

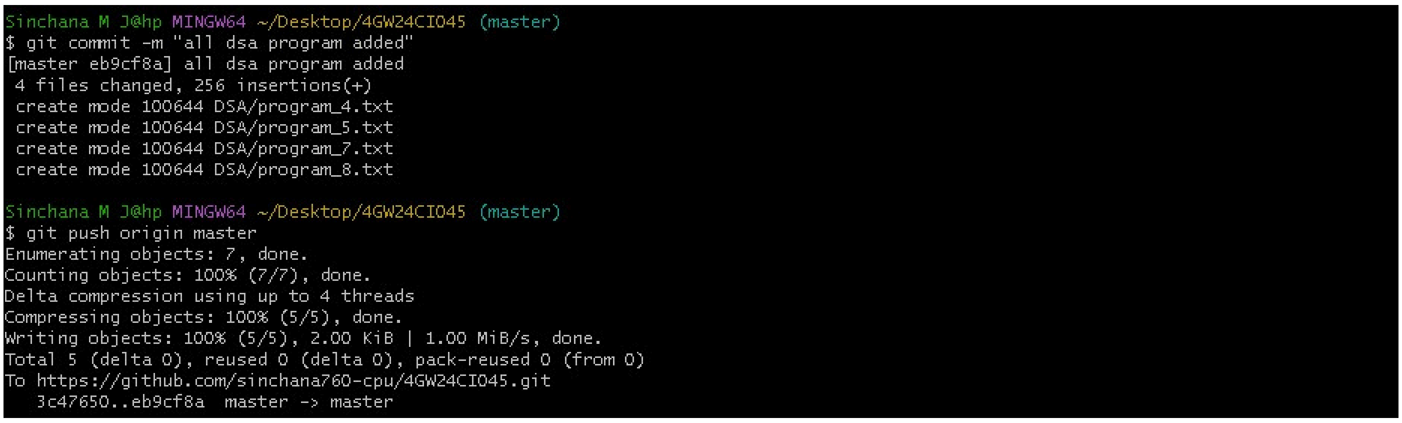
Command:

- Git init
- It initializes the new git repo
- Creates hidden. git folder
- Create new file (ex: file.txt)
- Add a file to the staging area using git add file.txt



```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git init
Reinitialized existing Git repository in C:/Users/Sinchana M J/Desktop/4GW24CI045/.git/

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git config --global user.name "Sinchan"

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git config --global user.email "sinchanamj006@gmail.com"

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git status
On branch master
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:    DSA/DSAfile.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        DSA/program_4.txt
        DSA/program_5.txt
        DSA/program_7.txt
        DSA/program_8.txt

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

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git add DSA/program_4.txt

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git add DSA/program_5.txt

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git add DSA/program_7.txt

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git add DSA/program_8.txt

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git remote add origin https://github.com/sinchanamj/4GW24CI045.git
error: remote origin already exists.

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git commit -m "all dsa program added"
[master eb9cf8a] all dsa program added
 4 files changed, 256 insertions(+)
 create mode 100644 DSA/program_4.txt
```

- Commit the changes with msg
- Git commit –m “file added”



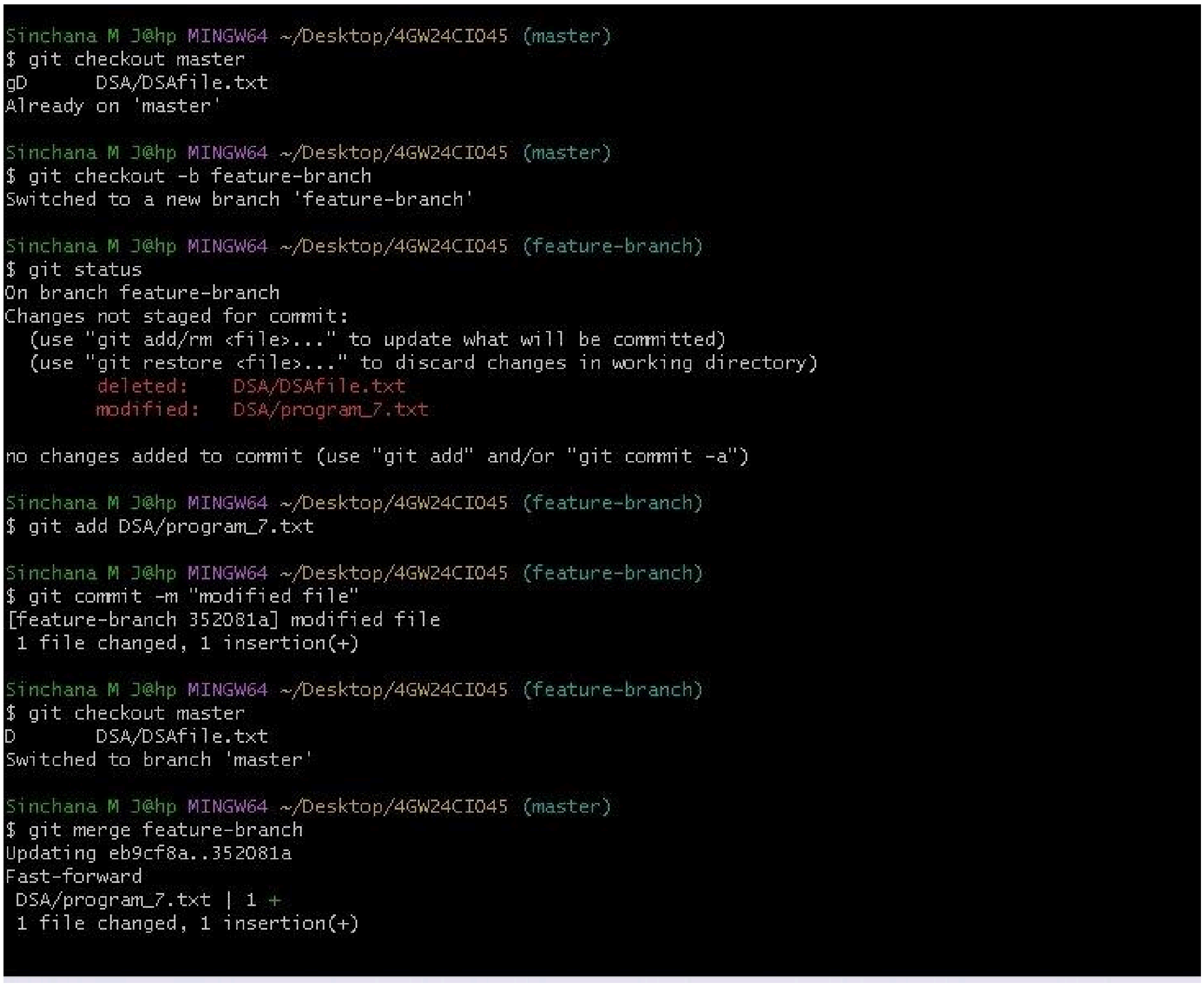
```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git commit -m "all dsa program added"
[master eb9cf8a] all dsa program added
 4 files changed, 256 insertions(+)
 create mode 100644 DSA/program_4.txt
 create mode 100644 DSA/program_5.txt
 create mode 100644 DSA/program_7.txt
 create mode 100644 DSA/program_8.txt

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git push origin master
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 2.00 KiB | 1.00 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/sinchana760-cpu/4GW24CI045.git
   3c47650..eb9cf8a  master -> master
```

EXPERIMENT 2

CREATING AND MANAGING BRANCHES

Createanewbranchnamed“feature-branch”. Switch to the “master” branch. Merge the “feature-branch” into “ ma ste r” .

- **Create a new branch**
- **Command: it** branch feature-branch
- Itstarts a copy ofthe current branch
- Switching to master branch:
- **Command: it** checkout master
- Now the file change to match whatever master contains, if any commit you make now will belong to master, not feature –branch
- Merge feature –branch into master:
- **Command: it** merge feature-branch
- Git takes all commits from feature-branch and combines them into master.



```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git checkout master
gD      DSA/DSAfile.txt
Already on 'master'

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git checkout -b feature-branch
Switched to a new branch 'feature-branch'

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (feature-branch)
$ git status
On branch feature-branch
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:    DSA/DSAfile.txt
        modified:   DSA/program_7.txt

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

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (feature-branch)
$ git add DSA/program_7.txt

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (feature-branch)
$ git commit -m "modified file"
[feature-branch 352081a] modified file
 1 file changed, 1 insertion(+)

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (feature-branch)
$ git checkout master
D       DSA/DSAfile.txt
Switched to branch 'master'

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git merge feature-branch
Updating eb9cf8a..352081a
Fast-forward
 DSA/program_7.txt | 1 +
 1 file changed, 1 insertion(+)
```

EXPERIMENT 3

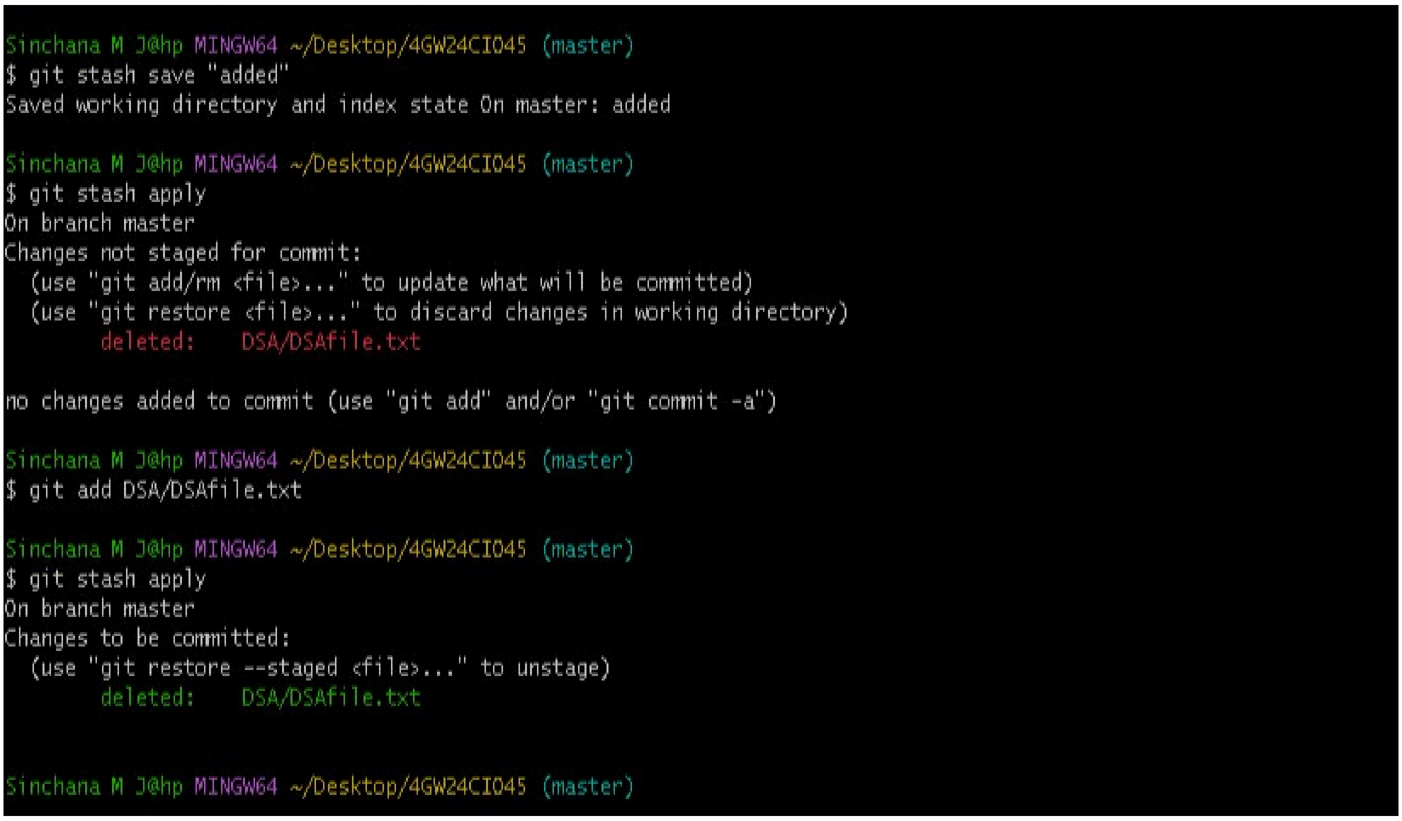
CREATING AND MANAGING BRANCHES

WriteThecommendsto.Stash your changes. Switch branches. And then apply the. Stashed changes.

Command:

- Gitstash
- Gitcheckout another branch
- Gitstashes apply

- Stashyour changes using git stash before committing.
- If youuse it after committing git, stash says, “no local changes to save”.
- Git checkout moves you to different branch
- Applybrings the stashed changes



```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git stash save "added"
Saved working directory and index state On master: added

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git stash apply
On branch master
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:    DSA/DSAfile.txt

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

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git add DSA/DSAfile.txt

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git stash apply
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    DSA/DSAfile.txt


Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
```

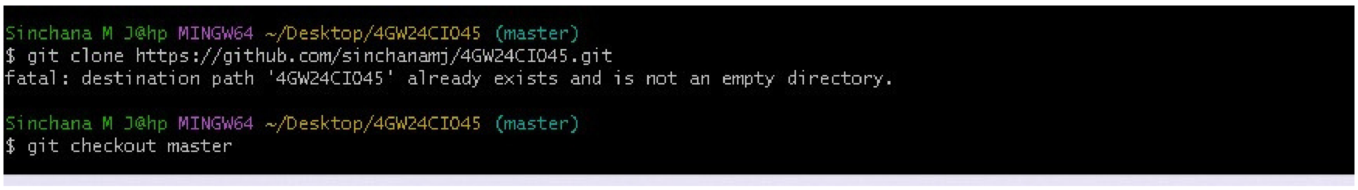
EXPERIMENT 4

COLLABORATION AND REMOTE REPOSITORIES.

Clonearemote Git repository to your local machine.

Command:

- Git clone <repo-URL>
- Clone will copy entire history and files.
- Git automatically creates folder named after the repo
- Cloning inside an existing repo is mistake
- Set up a remote called origin
- Itallows pull and push
- After cloning check status and branch



```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git clone https://github.com/sinchanamj/4GW24CI045.git
fatal: destination path '4GW24CI045' already exists and is not an empty directory.

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git checkout master
```

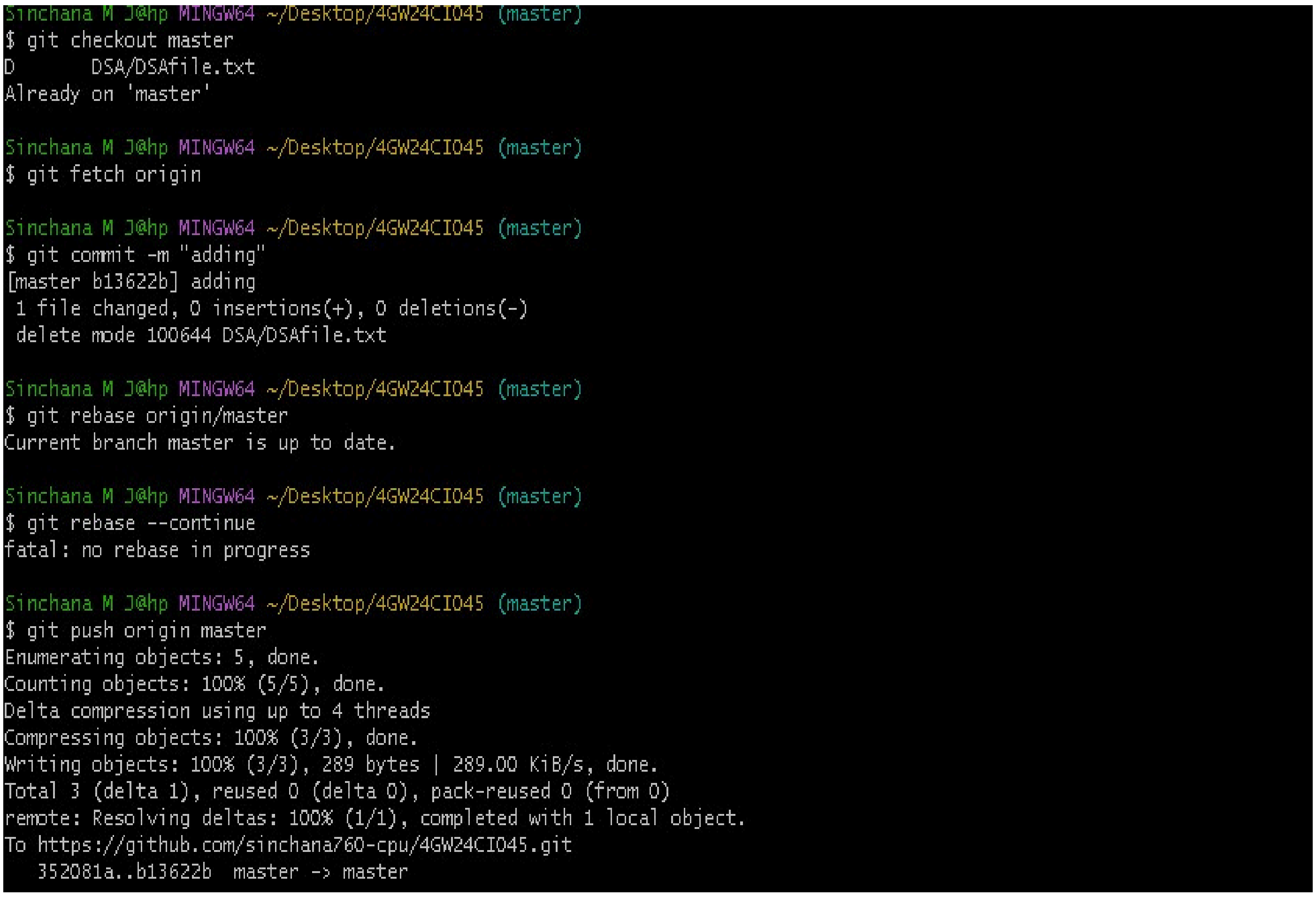
EXPERIMENT 5

COLLABORATION AND REMOTE REPOSITORIES

Fetchthe latestchangesfromaremoterepository and rebase your local branch onto the updated remote branch.

Command:

- Git fetch origin
- Git contacts the remote repo
- Save new commits
- Your local branch stays exactly where it was
- Git rebase origin/main
- Git takes local commits
- Temporarily removes them
- Moves your branch to latest remote
- Re applies your commit on top



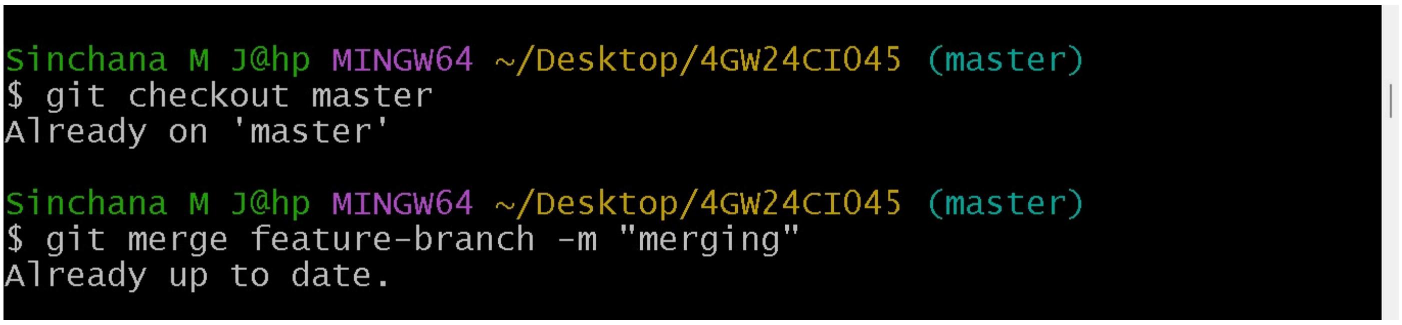
```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git checkout master
D       DSA/DSAfile.txt
Already on 'master'

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git fetch origin

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git commit -m "adding"
[master b13622b] adding
 1 file changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 DSA/DSAfile.txt

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git rebase origin/master
Current branch master is up to date.

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git rebase --continue
fatal: no rebase in progress

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 289 bytes | 289.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/sinchana760-cpu/4GW24CI045.git
   352081a..b13622b  master -> master
```

EXPERIMENT 6
COLLABORATION AND REMOTE REPOSITORIES

Write thecommandtomerge"feature-branch" into "master" while providing a custom commit message for the merge

Command:

- Gitcheckout master
- Youmove onto the master branch
- Gitalways merge into the branch you are currently on
- Gitmerge feature-branch –m “merged feature-branch into master”
- Gitcombines changes from feature-branch
- Creates merge commit
- Usesyour custom msg instead of opening an editor



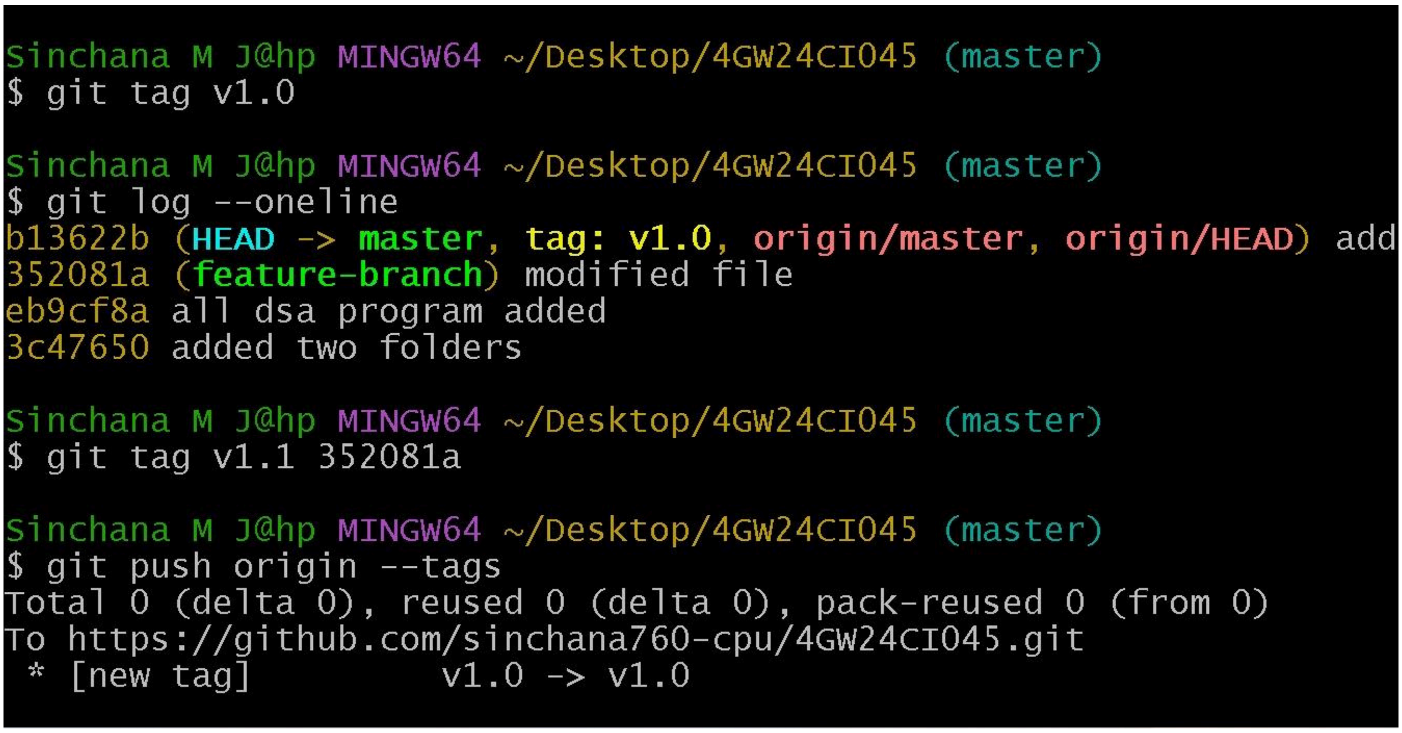
```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git checkout master
Already on 'master'

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git merge feature-branch -m "merging"
Already up to date.
```

EXPERIMENT 7
GITTAGS AND RELEASES

Writethecommandtocreatealightweight Git tag named "v1.0" for a commit in your local repository.

Command:

- Git tag v1.0
- This is for current commit
- A lightweight tag is just a name pointing to a commit
- Git tag v1.0 <commit-hash>
- Ex: git tag v1.0 a1bc3d
- Once created, a tag stays fixed on the commit it points to
- Push tag



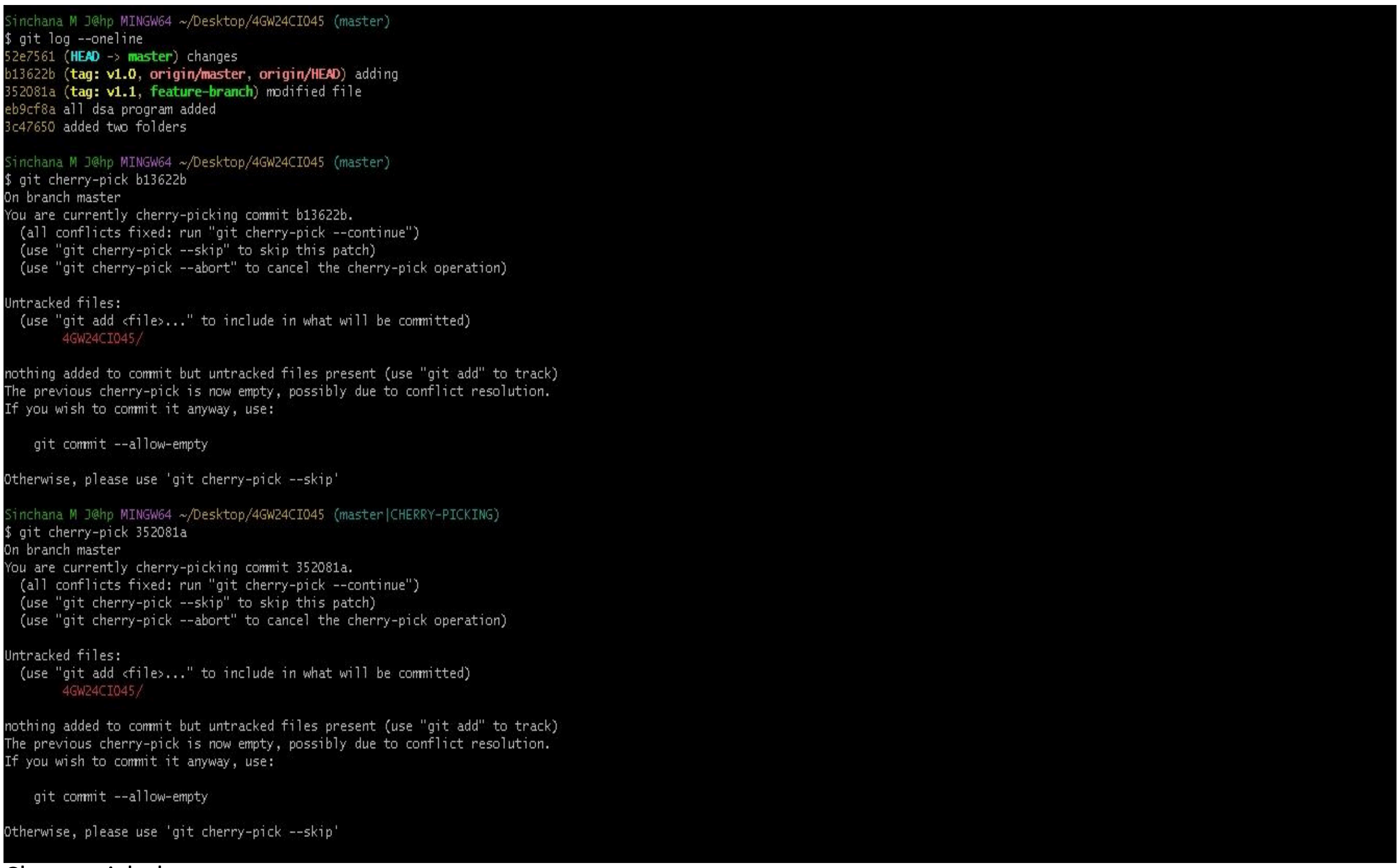
```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git tag v1.0

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git log --oneline
b13622b (HEAD -> master, tag: v1.0, origin/master, origin/HEAD) add
352081a (feature-branch) modified file
eb9cf8a all dsa program added
3c47650 added two folders

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git tag v1.1 352081a

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git push origin --tags
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/sinchana760-cpu/4GW24CI045.git
 * [new tag]         v1.0 -> v1.0
```

EXPERIMENT 8
ADVANCEDGIT OPERATIONS

Writethecommandtocherry-pickforarangeofcommits from "source-branch" to the current.

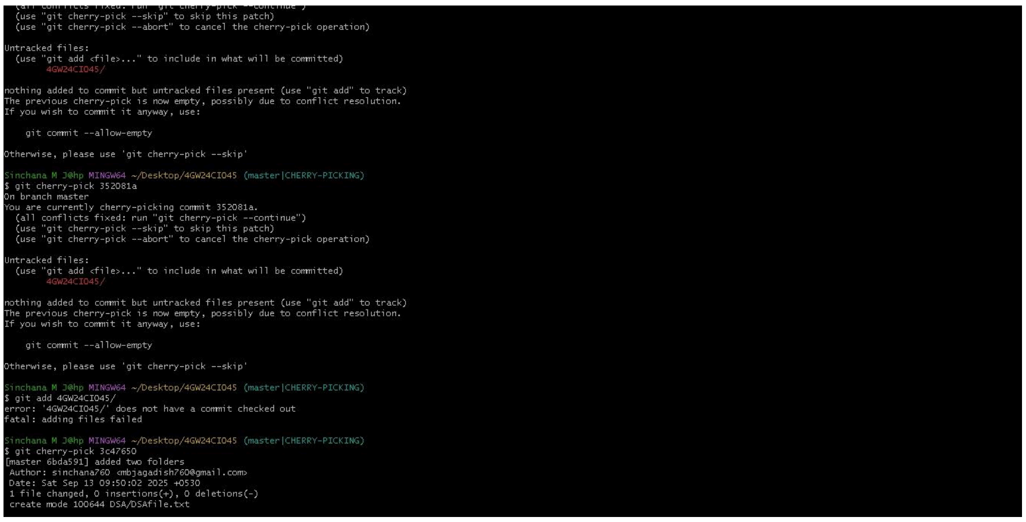
C ommand:

- Git checkout <current branch>
- <Current-branch>-->branchyouwantthecommits to go onto
- <start-commit>--thefirstcommitintherange you want to pick
- <end-commit>-->last commit inthe range
- Afterstartcommitiscritical;ittellsgittoinclude the start commit itself
- Switch to your targeted branch
- Identify the commit range



```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git log --oneline
52e7561 (HEAD -> master) changes
b13622b (tag: v1.0, origin/master, origin/HEAD) adding
352081a (tag: v1.1, feature-branch) modified file
eb9cf8a all dsa program added
3c47650 added two folders

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git cherry-pick b13622b
On branch master
You are currently cherry-picking commit b13622b.
  (all conflicts fixed: run "git cherry-pick --continue")
  (use "git cherry-pick --skip" to skip this patch)
  (use "git cherry-pick --abort" to cancel the cherry-pick operation)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        4GW24CI045/

nothing added to commit but untracked files present (use "git add" to track)
The previous cherry-pick is now empty, possibly due to conflict resolution.
If you wish to commit it anyway, use:

    git commit --allow-empty

Otherwise, please use 'git cherry-pick --skip'

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master|CHERRY-PICKING)
$ git cherry-pick 352081a
On branch master
You are currently cherry-picking commit 352081a.
  (all conflicts fixed: run "git cherry-pick --continue")
  (use "git cherry-pick --skip" to skip this patch)
  (use "git cherry-pick --abort" to cancel the cherry-pick operation)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        4GW24CI045/

nothing added to commit but untracked files present (use "git add" to track)
The previous cherry-pick is now empty, possibly due to conflict resolution.
If you wish to commit it anyway, use:

    git commit --allow-empty

Otherwise, please use 'git cherry-pick --skip'
```

Cherry-pick the



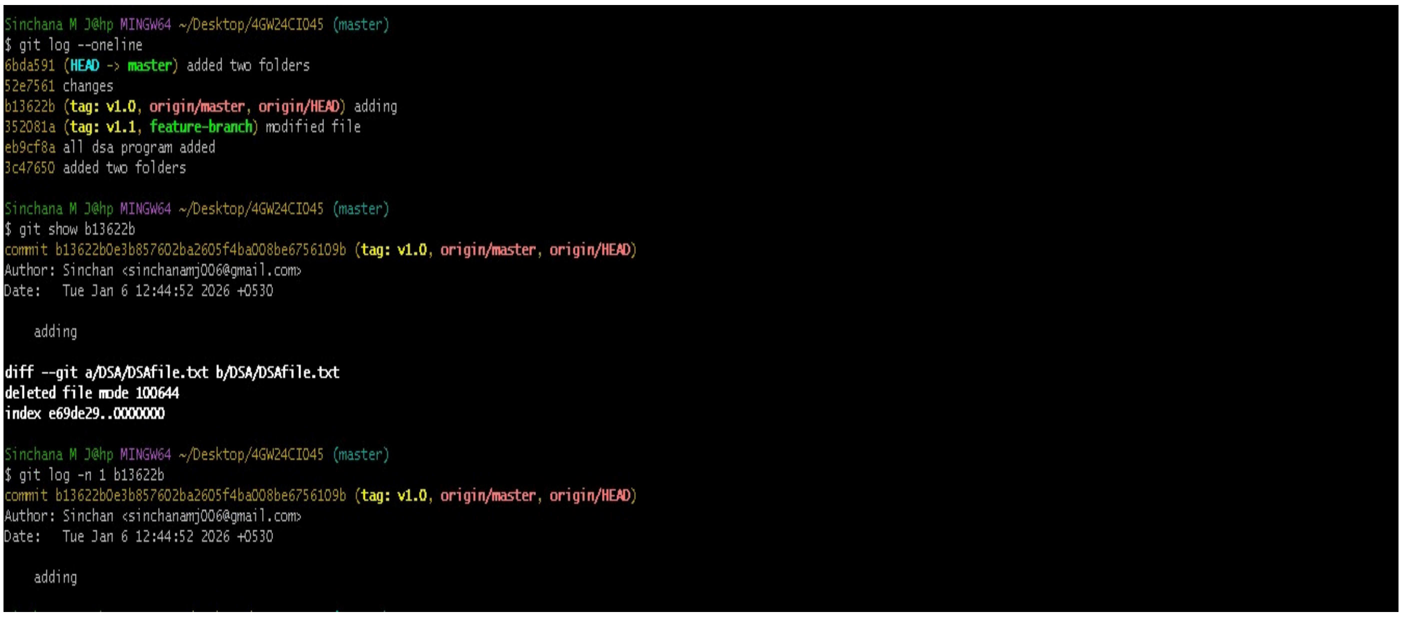
```
  (use "git cherry-pick --skip" to skip this patch)
  (use "git cherry-pick --abort" to cancel the cherry-pick operation)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        4GW24CI045/

nothing added to commit but untracked files present (use "git add" to track)
The previous cherry-pick is now empty, possibly due to conflict resolution.
If you wish to commit it anyway, use:

    git commit --allow-empty

Otherwise, please use 'git cherry-pick --skip'

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master|CHERRY-PICKING)
$ git cherry-pick 352081a
On branch master
You are currently cherry-picking commit 352081a.
  (all conflicts fixed: run "git cherry-pick --continue")
  (use "git cherry-pick --skip" to skip this patch)
  (use "git cherry-pick --abort" to cancel the cherry-pick operation)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        4GW24CI045/

nothing added to commit but untracked files present (use "git add" to track)
The previous cherry-pick is now empty, possibly due to conflict resolution.
If you wish to commit it anyway, use:

    git commit --allow-empty

Otherwise, please use 'git cherry-pick --skip'

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master|CHERRY-PICKING)
$ git add 4GW24CI045/
error: '4GW24CI045/' does not have a commit checked out
fatal: adding files failed

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master|CHERRY-PICKING)
$ git cherry-pick 3c47650
[master 6bda591] added two folders
 Author: sinchana760 <mbjagadish760@gmail.com>
 Date: Sat Sep 13 09:50:02 2025 +0530
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 DSA/DSAfile.txt
```

EXPERIMENT 9
ANALYSING AND CHANGING GIT HISTORY

Given a commit ID, how would you use Git to view the details of that specific commit, including the author, date, and commit message?

Command:

- Git show <commit-id>
- The hash of the commit you want to inspect
- Firstly, get the commit hash
- Git tag –one line
- This show hash for all commits
- View details git show a1b2c3d



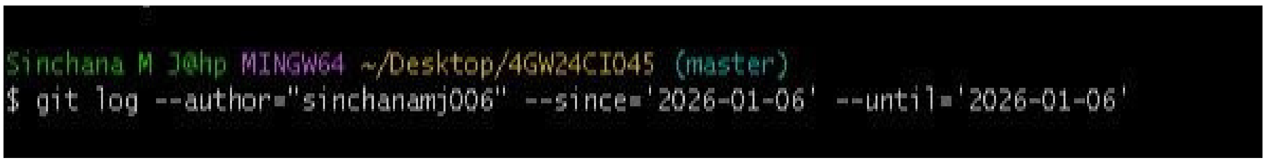
```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git log --oneline
6bda591 (HEAD -> master) added two folders
52e7561 changes
b13622b (tag: v1.0, origin/master, origin/HEAD) adding
352081a (tag: v1.1, feature-branch) modified file
eb9cf8a all dsa program added
3c47650 added two folders

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git show b13622b
commit b13622b0e3b857602ba2605f4ba008be6756109b (tag: v1.0, origin/master, origin/HEAD)
Author: Sinchan <sinchanamj006@gmail.com>
Date:   Tue Jan 6 12:44:52 2026 +0530

    adding

diff --git a/DSA/DSAfile.txt b/DSA/DSAfile.txt
deleted file mode 100644
index e69de29..0000000

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git log -n 1 b13622b
commit b13622b0e3b857602ba2605f4ba008be6756109b (tag: v1.0, origin/master, origin/HEAD)
Author: Sinchan <sinchanamj006@gmail.com>
Date:   Tue Jan 6 12:44:52 2026 +0530

    adding
```

EXPERIMENT 10
ANALYSING AND CHANGING GIT HISTORY

Write the command to list all commits made by the author "JohnDoe" between "2023-01- 01"and "2023-12-31."

Command:

- Start with branch
- Git checkout master
- Git only shows commits on the current branch by default
- Filter by author
- Use –author exactly as written in commints
- Filter by date
- Use ISO format YYYY-MM-DD to avoid ambiguity



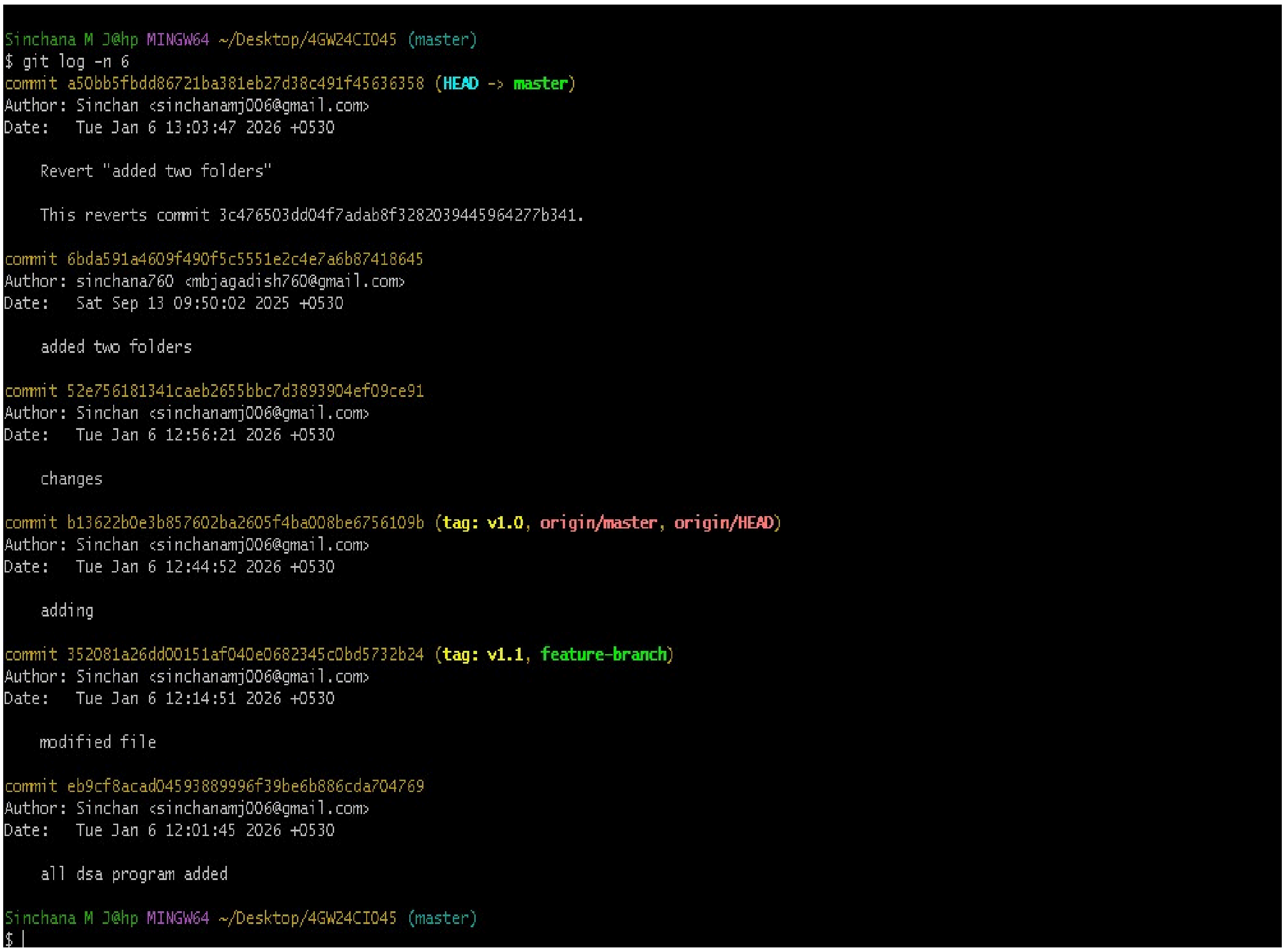
```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git log --author="sinchanamj006" --since='2026-01-06' --until='2026-01-06'
```

EXPERIMENT 11
ANALYSING AND CHANGING GIT HISTORY

Writethecommandtodisplaythelastfive commits in the repository's history

C ommand:

- Git log –5
- It shows the commit
- History Of the current branch
- -5meanslimittheoutputtothelast 5 commits only
- Commithash–uniqueIDforthecommit (SHA-1, long string)
- Bydefault,gitalsoshowsthefilechanges if you don't suppress them

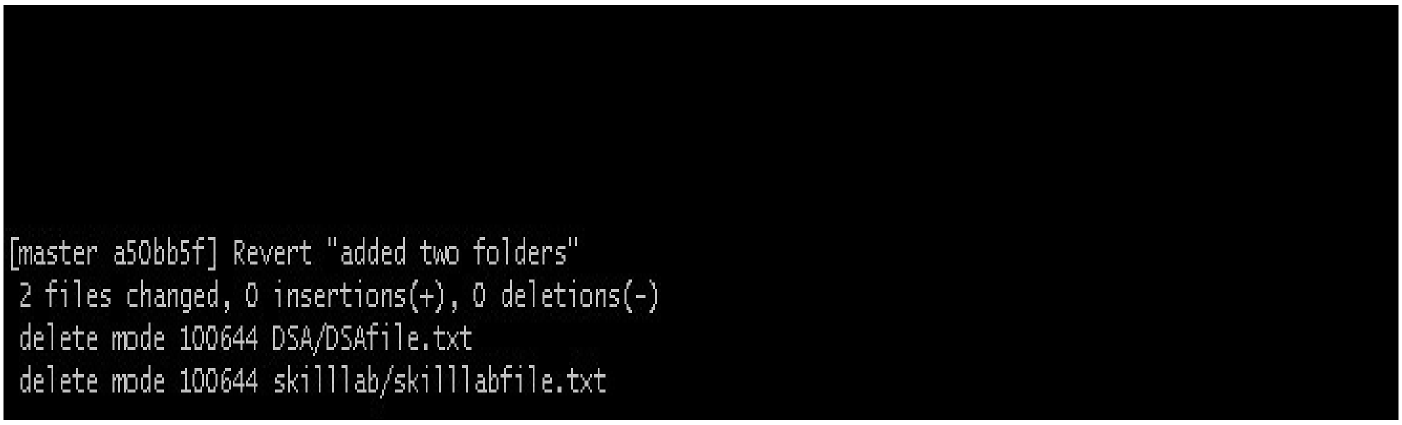


```
Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ git log -n 6
commit a50bb5fbdd86721ba381eb27d38c491f45636358 (HEAD -> master)
Author: Sinchan <sinchanamj006@gmail.com>
Date:   Tue Jan 6 13:03:47 2026 +0530

    Revert "added two folders"

    This reverts commit 3c476503dd04f7adab8f3282039445964277b341.

commit 6bda591a4609f490f5c5551e2c4e7a6b87418645
Author: sinchana760 <mbjagadish760@gmail.com>
Date:   Sat Sep 13 09:50:02 2025 +0530

    added two folders

commit 52e756181341caeb2655bbc7d3893904ef09ce91
Author: Sinchan <sinchanamj006@gmail.com>
Date:   Tue Jan 6 12:56:21 2026 +0530

    changes

commit b13622b0e3b857602ba2605f4ba008be6756109b (tag: v1.0, origin/master, origin/HEAD)
Author: Sinchan <sinchanamj006@gmail.com>
Date:   Tue Jan 6 12:44:52 2026 +0530

    adding

commit 352081a26dd00151af040e0682345c0bd5732b24 (tag: v1.1, feature-branch)
Author: Sinchan <sinchanamj006@gmail.com>
Date:   Tue Jan 6 12:14:51 2026 +0530

    modified file

commit eb9cf8acad04593889996f39be6b886cda704769
Author: Sinchan <sinchanamj006@gmail.com>
Date:   Tue Jan 6 12:01:45 2026 +0530

    all dsa program added

Sinchana M J@hp MINGW64 ~/Desktop/4GW24CI045 (master)
$ |
```

EXPERIMENT 12
ANALYSING AND CHANGING GIT HISTORY

Write the command to undo the changes introduced by the commit with the ID "abc123"

- Revert the commit
- This creates a new commit that undoes the changes from abc23
- Git revert abc123
- Reset to a previous state
- Git reset –hard abc123
- Abc123 points to commit before abc123



```
[master a50bb5f] Revert "added two folders"
 2 files changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 DSA/DSAfile.txt
 delete mode 100644 skilllab/skilllabfile.txt
```