

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Part 1: Git Commit Basics

- **Objective:** Learn how to commit changes to a Git repository.
- **Assignment:** Stage and commit files, then view your commit history

```
MINGW64:/d/repo

WINDOWS 11@mr dx MINGW64 /d/repo (main)
$ pwd
/d/repo

WINDOWS 11@mr dx MINGW64 /d/repo (main)
$ ls

WINDOWS 11@mr dx MINGW64 /d/repo (main)
$ echo "This is a text file" >> README.md

WINDOWS 11@mr dx MINGW64 /d/repo (main)
$ ls
README.md

WINDOWS 11@mr dx MINGW64 /d/repo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   README.md

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:   README.md

WINDOWS 11@mr dx MINGW64 /d/repo (main)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

WINDOWS 11@mr dx MINGW64 /d/repo (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   README.md

WINDOWS 11@mr dx MINGW64 /d/repo (main)
$ git commit -m "First commit"
[main 78b7b8e] First commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

WINDOWS 11@mr dx MINGW64 /d/repo (main)
$ git log
commit 78b7b8e2beblbb06e3b9097fe15e6e89a34e3840 (HEAD -> main)
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 21:21:17 2024 +0530

    First commit
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Explanations :

1. First navigate local directory, clone a git repository.
 2. Check file by run 'ls' command, then create a file by run " echo 'content' >> 'file_name' command.
 3. Run 'git status' command to check files stages " U untracked, M modified & A added to stage".
 4. Run 'git add' command to add file in staging area in showing ready to commit and finally run "git commit -m 'msg' " command to save changes in working directory.
 5. Run 'git log' command to show commit history.
-

Part 2: Reverting a Commit

- **Objective:** Understand how to revert a commit using git revert.
- **Assignment:** Revert a commit after making changes and observe how Git handles this.

```
WINDOWS 11@mr dx MINGW64 /d
$ mv repo2 practice

WINDOWS 11@mr dx MINGW64 /d
$ ls
'$RECYCLE.BIN'/'      'Htz infotech'/'      SKITM/'      Tech/
Book1.xlsx            'Learn extra'/'      Signature.jpg  'Videos - Shortcut.lnk'*
'Google Drive (G) - Shortcut.lnk'*  Notepad/'      'System Volume Information'/'  'WF Practising File'/'

WINDOWS 11@mr dx MINGW64 /d
$ cd practice

WINDOWS 11@mr dx MINGW64 /d/practice (master)
$ ls
README.md

WINDOWS 11@mr dx MINGW64 /d/practice (master)
$ nano README.md
```

Explanations:

1. Navigate to git directory, check files by run 'ls' command, then open file in editor to be modified by run "nano file-name" command, this open a file in nano editor. press 'ctrl+s' to save changes & 'ctrl+x' to close editor.
2. Run 'git status' command after modification to check files stages, here show file in M modified stage.
3. Run 'git add' command to add file in staging area in showing ready to commit and finally run "git commit -m 'msg' " command to save changes in working directory.
4. Run 'git log' command to show commit history, copy the hash-id of latest or commit that want to revert.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ ls
README.md

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git status
On branch master
Your branch is up to date with 'origin/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:   README.md

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

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git add .
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

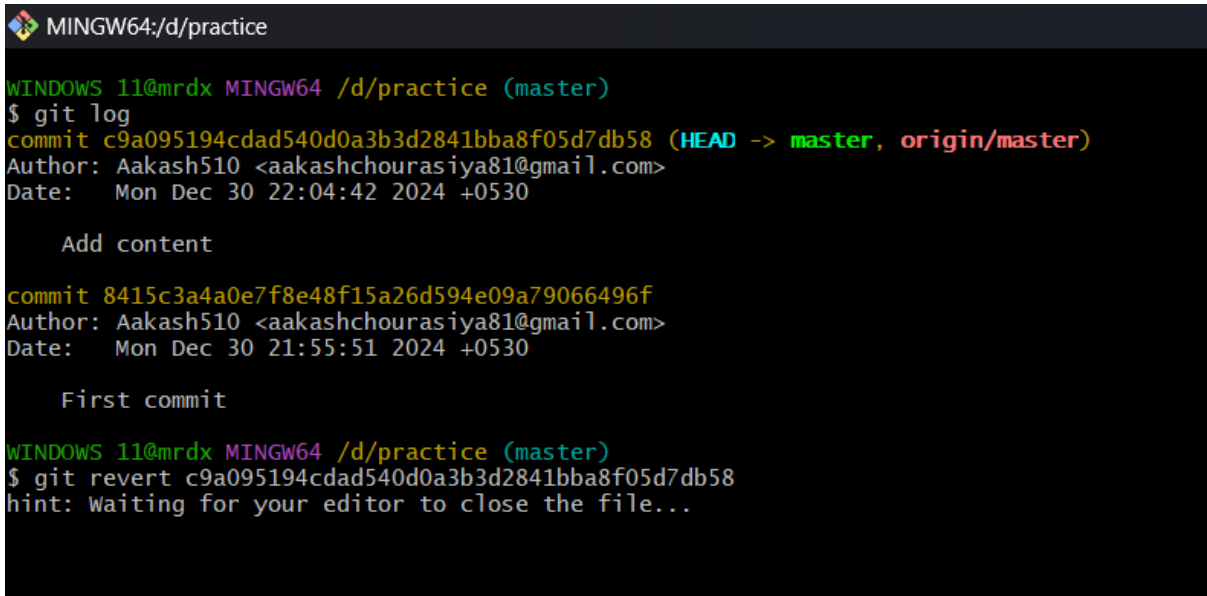
WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git commit -m "Add content"
[master c9a0951] Add content
1 file changed, 3 insertions(+), 1 deletion(-)

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 287 bytes | 287.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Aakash510/practice.git
8415c3a..c9a0951 master -> master

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$
```

A screenshot of a terminal window titled 'MINGW64:/d/practice'. The terminal shows the output of 'git log', displaying two commits: 'Add content' (hash c9a0951) and 'First commit' (hash 8415c3a). Below the log, the 'git revert' command is used to revert the 'Add content' commit, showing the hash c9a095194cdad540d0a3b3d2841bba8f05d7db58. The terminal output is as follows:

```
MINGW64:/d/practice
WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git log
commit c9a095194cdad540d0a3b3d2841bba8f05d7db58 (HEAD -> master, origin/master)
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 22:04:42 2024 +0530

    Add content

commit 8415c3a4a0e7f8e48f15a26d594e09a79066496f
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 21:55:51 2024 +0530

    First commit

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git revert c9a095194cdad540d0a3b3d2841bba8f05d7db58
hint: Waiting for your editor to close the file...
```

5. Run "git revert 'hash-id'" command to revert a commit, replace hash-id to copied.
6. Run 'git log' command to show commit history & notice that previous commit is modified.

Revert command allows you to undo changes made by a specific commit by creating a new commit that inversely applies those changes. It is a safe and collaborative way to undo changes in a shared repository, as it preserves the original commit history.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
MINGW64:/d/practice

WINDOWS 11@mr dx MINGW64 /d/practice (master)
$ git log
commit c9a095194cdad540d0a3b3d2841bba8f05d7db58 (HEAD -> master, origin/master)
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 22:04:42 2024 +0530

    Add content

commit 8415c3a4a0e7f8e48f15a26d594e09a79066496f
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 21:55:51 2024 +0530

    First commit

WINDOWS 11@mr dx MINGW64 /d/practice (master)
$ git revert c9a095194cdad540d0a3b3d2841bba8f05d7db58
[master f3794cb] Modify README.md
1 file changed, 1 insertion(+), 3 deletions(-)

WINDOWS 11@mr dx MINGW64 /d/practice (master)
$ git log

commit f3794cb3f49fee09682b08fe38a72b562479cc7e (HEAD -> master)
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 22:34:46 2024 +0530

    Modify README.md

    This reverts commit c9a095194cdad540d0a3b3d2841bba8f05d7db58.

commit c9a095194cdad540d0a3b3d2841bba8f05d7db58 (origin/master)
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 22:04:42 2024 +0530

    Add content

commit 8415c3a4a0e7f8e48f15a26d594e09a79066496f
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 21:55:51 2024 +0530

    First commit

WINDOWS 11@mr dx MINGW64 /d/practice (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
    (use "git push" to publish your local commits)

nothing to commit, working tree clean

WINDOWS 11@mr dx MINGW64 /d/practice (master)
$ |
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Part 3: Merge Conflicts

- **Objective:** Learn how merge conflicts occur and how to resolve them.
- **Assignment:** Create a merge conflict, resolve it, and complete the merge

```
MINGW64:/d/practice

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git branch
* feature
  master

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git switch feature
Switched to branch 'feature'
Your branch is up to date with 'origin/feature'.

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ ls
Into.txt  README.md

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ cat Into.txt
I am Aakash choursiya & currently exploring data fields technology & learning Python 3 & Sql.

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ nano Into.txt

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ cat Into.txt
I am Aakash choursiya & currently exploring data fields technology & currently learning python & database interaction language sql.

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ |

MINGW64:/d/practice

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ git branch
* feature
  master

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ git switch master
M    Into.txt
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 2 commits.
(use 'git push' to publish your local commits)

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ cat Into.txt
I am Aakash choursiya & currently exploring data fields technology & currently learning python & database interaction language sql.

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ nano Into.txt

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ cat Into.txt
I am Aakash choursiya & currently exploring data fields technology & currently learning python 3 & database interaction language SQL (Structured Query Language)

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git status
```

Explanation:

1. First run git branch command to view and check current branch, then run 'git switch <branch-name>' to switch another branch.
2. Check files by run 'ls' command & run 'cat <file-name>' to view its content, then open file in nano editor to be modified by run "nano <file-name>" command, this open a file in nano editor. press 'ctrl+s' to save changes & 'ctrl+x' to close editor.
3. Again run 'cat <file-name>' command to view modified content, then run 'git status' command after modification to check files stages, here show file in M modified stage.
4. Run 'git add' command to add file in staging area in showing ready to commit and finally run "git commit -m <msg>" command to save changes in working directory.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

5. Run 'git merge <file-name>' command to merge files, here unable to merge due to conflicts is occur and have to resolve them, open file in nano editor to view and solve conflicts manually.
6. After resolving conflicts run 'git add' command to add file in staging area in showing ready to commit and finally run "git commit -m <msg > " command to save changes in working directory.
7. Again run git merge command to merge files, now open file to check merges, and run git log command to view commit history, here merge commit is appear & confirm merges.

```
MINGW64/d/practice
wINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git add .

wINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git commit -m "add content"
[master 87c4e63] add content
1 file changed, 1 insertion(+), 1 deletion(-)

wINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git merge feature
Auto-merging Into.txt
CONFLICT (content): Merge conflict in Into.txt
Automatic merge failed; fix conflicts and then commit the result.

wINDOWS 11@mrdrx MINGW64 /d/practice (master|MERGING)
$ AC

wINDOWS 11@mrdrx MINGW64 /d/practice (master|MERGING)
$ nano Into.txt

MINGW64/d/practice
GNU nano 8.2 Into.txt
<<<<<< HEAD
I am Aakash choursiya & currently exploring data fields technology & currently learning python 3 & database interaction lange SEQUEL (Strued ery Laage).
=====
I am Aakash choursiya & currently exploring data fields technology & currently learning python 3 & database interaction language SQL (Structured Query Language)..
>>>>>> feature

wINDOWS 11@mrdrx MINGW64 /d/practice (master|MERGING)
$ git add .

wINDOWS 11@mrdrx MINGW64 /d/practice (master|MERGING)
$ git commit -m "Add content"
[master 3316720] Add content

wINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git merge feature
Already up to date.

wINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ cat Into.txt

I am Aakash choursiya & currently exploring data fields technology & currently learning python 3 & database interaction language SQL (Structured Query Language).

wINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git log
commit 3316720df1c977a9ee97763dbcccf1256972f231 (HEAD -> master)
Merge: 87c4e63 dc11f69
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 23:35:34 2024 +0530

    Add content

commit 87c4e63d55e16eab4d8285130d6709dea5b0ddcc
Author: Aakash510 <aakashchourasiya81@gmail.com>
Date: Mon Dec 30 23:29:48 2024 +0530

    add content
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Part 4: Cherry-Picking Commits

- **Objective:** Use git cherry-pick to apply specific commits to a different branch.
- **Assignment:** Cherry-pick commits from one branch and apply them to another branch.

```
MINGW64/d/practice
WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git log --oneline -n 4
d2c7003 (HEAD -> master, origin/master) Add education
6c7cb0d Add address
3316720 Add content
87c4e63 add content

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git checkout feature
Switched to branch 'feature'
Your branch is up to date with 'origin/feature'.

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ git log --oneline -n 2
dc11f69 (HEAD -> feature, origin/feature) add content
9c8c985 Add content

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ cat into.txt
I am Aakash chourasiya & currently exploring data fields technology & currently learning python 3 & database interaction language SQL (Structured Query Language)..

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ git cherry-pick d2c7003
Auto-merging into.txt
CONFLICT (content): Merge conflict in into.txt
error: could not apply d2c7003... Add education
hint: After resolving the conflicts, mark them with
hint: "git add/rm -p pathspec", then run
hint: "git cherry-pick --continue".
hint: You can instead skip this commit with "git cherry-pick --skip".
hint: To abort and get back to the state before "git cherry-pick",
hint: run "git cherry-pick --abort".
hint: Disable this message with "git config advice.mergeConflict false"

WINDOWS 11@mrdrx MINGW64 /d/practice (feature|CHERRY-PICKING)
$ nano into.txt

WINDOWS 11@mrdrx MINGW64 /d/practice (feature|CHERRY-PICKING)
$ git add .

WINDOWS 11@mrdrx MINGW64 /d/practice (feature|CHERRY-PICKING)
$ git cherry-pick --continue
[feature 7fa8226] Add education
Date: Tue Dec 31 00:29:36 2024 +0530
1 file changed, 1 insertion(+)

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ git log --oneline -n 4
7fa8226 (HEAD -> feature) Add education
dc11f69 (origin/feature) add content
9c8c985 Add content
44e472e Add content

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$
```

Explanation:

1. First let's see commit history in one-line of only latest 4 commit by run 'git log --oneline -n 4' command & notice that four commit is shown.
2. Checkout the feature branch and again run 'git log --oneline -n 2' command & here two commit history is show. If we want to add only latest one commit (hash-id is 'd2c7003') in this branch from master branch, then 'cherry-pick' command will be good option.
3. Run 'cat into.txt' to view content and run git '**git cherry-pick d2c7003**>' command to pick specific commit and add it in feature branch.
4. Now if any conflicts occur then open file in nano editor and resolve conflicts manually. After resolving conflicts run 'git add' command to add file in staging area or tell to git that file is ready, now run 'git cherry pick --continue' to complete the cherry-picking process in feature branch.
5. Run git log to check commit history, notice that cherry-picking is successful and new commit is add with (hash-id 7fa8226)
6. Switch to master branch view content of file and commit history and then see view content of file on feature branch & commit history below to confirm changes.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
MINGW64/d/practice
GNU nano 8.2      into.txt
<<<<<< HEAD
I am Aakash chourasiya & currently exploring data fields technology & currently learning python 3 & database interaction language SQL (Structured Query Language)..
=====
I am Aakash chourasiya & currently exploring data fields technology & currently learning python 3 & database interaction language SQL (Structured Query Language).
Address - basically i am from pipariya (m.p), but currently living in Indore city for future study.
Education - I have completed my H.S Schooling from MPBSE in 2022 & currently purshuing my graduation.
>>>>>> d2c7003 (Add education)
```

```
MINGW64/d/practice
WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git branch
  feature
* master

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ cat into.txt
I am Aakash chourasiya & currently exploring data fields technology & currently learning python 3 & database interaction language SQL (Structured Query Language).
Address - basically i am from pipariya (m.p), but currently living in Indore city for future study.
Education - I have completed my H.S Schooling from MPBSE in 2022 & currently purshuing my graduation.

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git log --oneline -n 4
d2c7003 (HEAD -> master, origin/master) Add education
6c7cb0d Add address
3316720 Add content
87c4e63 add content

WINDOWS 11@mrdrx MINGW64 /d/practice (master)
$ git checkout feature
Switched to branch 'feature'
Your branch is ahead of 'origin/feature' by 1 commit.
(Use "git push" to publish your local commits)

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ cat into.txt
I am Aakash chourasiya & currently exploring data fields technology & currently learning python 3 & database interaction language SQL (Structured Query Language)..
Education - I have completed my H.S Schooling from MPBSE in 2022 & currently purshuing my graduation.

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$ git log --oneline -n 4
7fa8226 (HEAD -> feature) Add education
dc11f69 (origin/feature) add content
9c8c985 Add content
44e472e Add content

WINDOWS 11@mrdrx MINGW64 /d/practice (feature)
$
```

Part 5: Fetching branches

- **Objective:** Learn how to fetch branches from a remote repository.
- **Assignment:** Fetch branches and inspect the contents.

1. Navigate to git directory 'new1', run 'git clone <remote repo url>' command to clone remote repo locally & have a local copy of the repository including its history and branches. next run 'ls' command to view all files and directory and enter in git repo by run cd command.
2. Set upstream to remote repo with "git remote add upstream -u <origin-remote-repo-url> command.
3. Now run git fetch upstream command to fetch data of remote origin repository locally and merge local data to origin repo using 'git merge upstream/master' command.
4. Here showing already up to date because there is no change in remote origin repository before run git fetch command & if there is any changes then here conflicts will occurs and have to resolve them.
5. Create a file using 'echo 'msg' >> <files-name>' command and run 'git add' command to add file in staging area then run "git commit -m 'msg' " command to save changes in working directory.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

6. Run 'git push' command to show local changes into remote repository.
7. Run git status command this will show
" Your branch is up to date with 'origin/feature2' "

```
MINGW64:/d/new1/python-mini-projects

WINDOWS 11@mr dx MINGW64 ~
$ cd /d/new1

WINDOWS 11@mr dx MINGW64 /d/new1
$ git clone https://github.com/Aakash510/python-mini-projects.git
Cloning into 'python-mini-projects'...
remote: Enumerating objects: 4657, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
Remote: Total 4657 (delta 1), reused 3 (delta 0), pack-reused 4651 (from 1)
Receiving objects: 100% (4657/4657), 135.22 MiB | 4.08 MiB/s, done.
Resolving deltas: 100% (2152/2152), done.
Updating files: 100% (1467/1467), done.
warning: the following paths have collided (e.g. case-sensitive paths
on a case-insensitive filesystem) and only one from the same
colliding group is in the working tree:

'projects/Fetch_and_store_tweets/img/output-2.PNG'
'projects/Fetch_and_store_tweets/img/output-2.png'
'projects/Fetch_and_store_tweets/img/output.PNG'
'projects/Fetch_and_store_tweets/img/output.png'
'projects/Fetch_and_store_tweets/img/step-1.PNG'
'projects/Fetch_and_store_tweets/img/step-1.png'
'projects/Fetch_and_store_tweets/img/step-2.PNG'
'projects/Fetch_and_store_tweets/img/step-2.png'
'projects/Fetch_and_store_tweets/img/step-3.PNG'
'projects/Fetch_and_store_tweets/img/step-3.png'
'projects/Fetch_and_store_tweets/img/step-4.PNG'
'projects/Fetch_and_store_tweets/img/step-4.png'
'projects/Fetch_and_store_tweets/img/step-5.PNG'
'projects/Fetch_and_store_tweets/img/step-5.png'
'projects/Fetch_and_store_tweets/img/step-6.PNG'
'projects/Fetch_and_store_tweets/img/step-6.png'

WINDOWS 11@mr dx MINGW64 /d/new1
$ ls
python-mini-projects/

WINDOWS 11@mr dx MINGW64 /d/new1
$ cd python-mini-projects

WINDOWS 11@mr dx MINGW64 /d/new1/python-mini-projects (master)
$ ls
CODE_OF_CONDUCT.md  README.md          note.txt
LICENSE             README_TEMPLATE.md projects/
Notebooks/          docs/              requirementsALL.txt
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
WINDOWS 11@mr dx MINGW64 /d/new1/python-mini-projects (master)
$ git remote add upstream https://github.com/Python-World/python-mini-projects.git
$ git fetch upstream
From https://github.com/Python-World/python-mini-projects
* [new branch]      master      -> upstream/master

WINDOWS 11@mr dx MINGW64 /d/new1/python-mini-projects (master)
$ git merge upstream/master
Already up to date.

WINDOWS 11@mr dx MINGW64 /d/new1/python-mini-projects (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

WINDOWS 11@mr dx MINGW64 /d/new1/python-mini-projects (master)
$ echo "This is a forked repo" >> Note1.txt

WINDOWS 11@mr dx MINGW64 /d/new1/python-mini-projects (master)
$ git add .
warning: in the working copy of 'Note1.txt', LF will be replaced by CRLF the next time Git touches it

WINDOWS 11@mr dx MINGW64 /d/new1/python-mini-projects (master)
$ git commit -m "Add note"
[master 32dd128] Add note
1 file changed, 1 insertion(+)
create mode 100644 Note1.txt

WINDOWS 11@mr dx MINGW64 /d/new1/python-mini-projects (master)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 298 bytes | 149.00 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/Aakash510/python-mini-projects.git
   374f7b9..32dd128  master -> master

WINDOWS 11@mr dx MINGW64 /d/new1/python-mini-projects (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Part 6: Syncing Forked Branch

- **Objective:** Sync your forked repository with the upstream repository.
- **Assignment:** Fetch and merge changes from the original repository into your fork.

Explanation:

1. Navigate to git directory, run 'git branch' command to check current branch and run 'git checkout feature' Command to switch branch.
2. Run 'ls' command to open file in editor to be modified using "nano file-name" command make changes and press 'ctrl+s' to save changes & 'ctrl+x' to close editor.
3. After modification run 'git add' command to add file in staging area in showing ready to commit and finally run "git commit -m 'msg' " command to save changes in working directory.
4. Run 'git fetch origin' to fetch content of remote repository, it will only fetch content locally except don't merge, we merge manually.
5. Now run git merge master to merge feature to master branch, here conflicts will occur due to another developer make changes in remote repo before fetching and we have to resolve them manually.
6. Open file in nano editor and resolve conflicts manually. After resolving conflicts run 'git add' command to add file in staging area or tell to git that file is ready, check commit history & view top commit to confirm save changes after resolve conflicts.
7. To show local changes into remote repo, run 'git push -u origin feature2' to push local change to remote repository on GITHUB.
8. Run git status command this will show
" Your branch is up to date with 'origin/feature2' "
9. After done this process, on GITHUB shows create pull request to merge feature2 branch with main branch.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

10. Create pull request and do necessary step to merge successfully on GITHUB also.

```
MINGW64:/d/my_pc/practice

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ pwd
/d/my_pc/practice

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git branch
  feature2
* master

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git checkout feature2
Switched to branch 'feature2'

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature2)
$ ls
Into.txt  README.md  data.html

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature2)
$ nano data.html

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature2)
$ git add .

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature2)
$ git commit -m "Add vyom (4th) details"
[feature2 d6ea4cb] Add vyom (4th) details
1 file changed, 9 insertions(+), 4 deletions(-)

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature2)
$
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
MINGW64:/d/my_pc/practice
1 file changed, 9 insertions(+), 4 deletions(-)

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2)
$ git fetch origin
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 8 (delta 4), reused 8 (delta 4), pack-reused 0 (from 0)
Unpacking objects: 100% (8/8), 910 bytes | 18.00 KiB/s, done.
From https://github.com/Aakash510/practice
   98aaebd..0f8806f  master    -> origin/master

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2)
$ git log --oneline -n 4
d6ea4cb (HEAD -> feature2) Add vyom (4th) details
98aaebd (master) Feature (#2)
d2c7003 Add education
6c7cb0d Add address

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2)
$ git merge origin/master
Auto-merging data.html
CONFLICT (content): Merge conflict in data.html
Automatic merge failed; fix conflicts and then commit the result.

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2|MERGING)
$ nano data.html

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2|MERGING)
$ git merge --continue
U      data.html
error: Committing is not possible because you have unmerged files.
hint: Fix them up in the work tree, and then use 'git add/rm <file>'
hint: as appropriate to mark resolution and make a commit.
fatal: Exiting because of an unresolved conflict.

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2|MERGING)
$ git add .

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2|MERGING)
$ git commit -m "resolved conflicts"
[feature2 bdef719] resolved conflicts

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2)
$ git merge --continue
fatal: There is no merge in progress (MERGE_HEAD missing).

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2)
$ git merge origin/master
Already up to date.

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2)
$ git log --oneline -n 4
bdef719 (HEAD -> feature2) resolved conflicts
d6ea4cb Add vyom (4th) details
0f8806f (origin/master, origin/HEAD) updates
9730650 Add 3rd
```

Assignment Submission

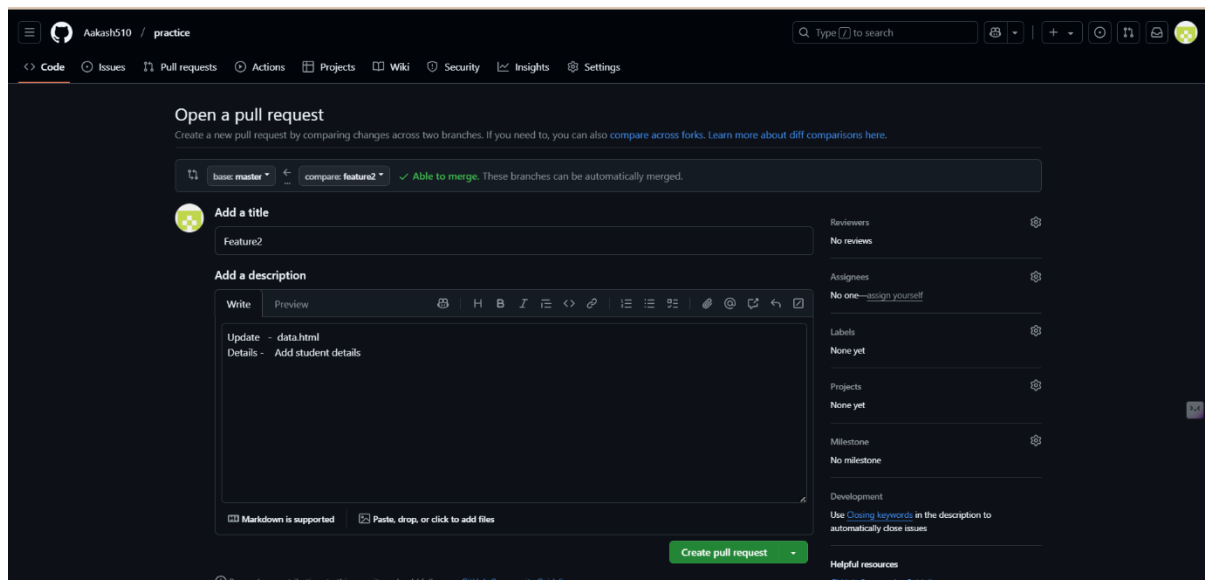
GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2)
$ git push -u origin feature2
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 648 bytes | 648.00 KiB/s, done.
Total 6 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 2 local objects.
remote:
remote: Create a pull request for 'feature2' on GitHub by visiting:
remote:   https://github.com/Aakash510/practice/pull/new/feature2
remote:
To https://github.com/Aakash510/practice.git
 * [new branch]      feature2 -> feature2
branch 'feature2' set up to track 'origin/feature2'.

WINDOWS 11@mrdrx MINGW64 /d/my_pc/practice (feature2)
$ git status
On branch feature2
Your branch is up to date with 'origin/feature2'.
```



Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

11. It is always practice to delete used branch after completion any task.
12. To do this switch to master branch and delete local feature2 branch by run 'git branch -d feature2' command and also delete remote feature branch 'using git push origin --delete feature2' command.
13. Now check status of working tree by run 'git status' command.

```
WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git branch
  feature2
* master

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git branch -d feature2
warning: deleting branch 'feature2' that has been merged to
        'refs/remotes/origin/feature2', but not yet merged to HEAD
Deleted branch feature2 (was bdef719).
```

```
WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git push origin --delete feature2
To https://github.com/Aakash510/practice.git
- [deleted]          feature2

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git branch
* master

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git status
On branch master
Your branch is behind 'origin/master' by 5 commits, and can be fast-forwarded.
  (use "git pull" to update your local branch)

nothing to commit, working tree clean
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Part 7: Merging via Command Line

- **Objective:** Merge branches using the command line.
- **Assignment:** Create branches, make changes, and merge them using the command line.

Explanation:

1. Navigate to git directory, run 'git branch' command to check current branch and run 'git checkout feature' Command to switch branch.
2. Run 'ls' command to show all files and directory, now create branch using "git branch feature1" command and switch to feature1 branch by run "git switch feaure1" command.
3. Now open file in editor to be modified using "nano file-name" command make changes and press 'ctrl+s' to save changes & 'ctrl+x' to close editor.
4. After modification run 'git add' command to add file in staging area in showing ready to commit and finally run "git commit -m 'msg' " command to save changes in working directory.
5. Switch to master branch and run 'git diff feature1' to show differences in file between two branches.
6. Run 'git merge feature1' command to merge content of files in both branches
7. Run 'git push -u origin master ' command to show local changes into remote repository in master branch.
8. Run git status command this will show
" Your branch is ahead of 'origin/master' by 1 commit "
9. Run 'git log' command to show commit history to confirm merges.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
MINGW64:/d/my_pc/practice

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ pwd
/d/my_pc/practice

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ ls
Into.txt  README.md  data.html

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git branch
* master

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git branch feature1

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git branch
feature1
* master

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git switch feature1
Switched to branch 'feature1'

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature1)
$ ls
Into.txt  README.md  data.html

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature1)
$ nano into.txt

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature1)
$ git status
On branch feature1
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:   Into.txt

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature1)
$ git add .

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature1)
$ git commit -m "Add Address"
[feature1 1f27b71] Add Address
1 file changed, 1 insertion(+)
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
MINGW64:/d/my_pc/practice

WINDOWS 11@mrdx MINGW64 /d/my_pc/practice (feature1)
$ git switch master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

WINDOWS 11@mrdx MINGW64 /d/my_pc/practice (master)
$ git diff feature1
diff --git a/Into.txt b/Into.txt
index caa8ee7..8180605 100644
--- a/Into.txt
+++ b/Into.txt
@@ -1,4 +1,3 @@

I am Aakash choursiya & currently exploring data fields technology & currently learning
Education - I have completed my H.S Schooling from MPBSE in 2022 & currently purshuing
-Add - I currently living in indore.

WINDOWS 11@mrdx MINGW64 /d/my_pc/practice (master)
$ git merge feature1
Updating e30138f..1f27b71
Fast-forward
 Into.txt | 1 +
 1 file changed, 1 insertion(+)

WINDOWS 11@mrdx MINGW64 /d/my_pc/practice (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

WINDOWS 11@mrdx MINGW64 /d/my_pc/practice (master)
$ git push -u origin main
error: src refspec main does not match any
error: failed to push some refs to 'https://github.com/Aakash510/practice.git'

WINDOWS 11@mrdx MINGW64 /d/my_pc/practice (master)
$ git push -u origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 375 bytes | 375.00 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/Aakash510/practice.git
 e30138f..1f27b71 master -> master
branch 'master' set up to track 'origin/master'.

WINDOWS 11@mrdx MINGW64 /d/my_pc/practice (master)
$ git log --oneline -n 4
1f27b71 (HEAD -> master, origin/master, origin/HEAD, feature1) Add Address
e30138f Merge pull request #3 from Aakash510/feature2
bdef719 resolved conflicts
d6ea4cb Add vyom (4th) details
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Part 8: Git Stash

- **Objective:** Learn how to temporarily store changes using git stash.
- **Assignment:** Stash changes, switch branches, and apply the stashed changes later.

```
MINGW64:/d/my_pc/practice

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ pwd
/d/my_pc/practice

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git branch
  feature
  feature1
* master
  new

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ nano data.html

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git status
On branch master
Your branch is up to date with 'origin/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:   data.html

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

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git switch feature
error: Your local changes to the following files would be overwritten by checkout:
      data.html
Please commit your changes or stash them before you switch branches.
Aborting

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git stash save "work in progress on data.html"
Saved working directory and index state On master: work in progress on data.html

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git stash list
stash@{0}: On master: work in progress on data.html
stash@{1}: On feature: work in progress on feature

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git switch feature
Switched to branch 'feature'
Your branch is up to date with 'origin/feature'.

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature)
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Explanation :

1. Navigate to git directory, run 'git branch' command to check current branch.
2. Open file in editor to be modified using "nano file-name" command make changes and press 'ctrl+s' to save changes & 'ctrl+x' to close editor.
3. Try to switch branch & notice an error occur showing that
'Your changes to the following files, will be overwritten by checkout'
4. Run "git stash save 'work is process on data.html'" command to save changes temporary in working directory. run 'git stash list' to see the all active stash.
5. Now you can easily switch branch to feature branch run 'git stash list' to see stash list, notice two stash is shown of both branches.
6. Run 'git stash pop' command to remove the stash and showing that changes not staged for commit. again run 'git stash list' to see stash list, notice that only one stash of features branch is shown.

```
WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature)
$ git stash list
stash@{0}: On master: work in progress on data.html
stash@{1}: On feature: work in progress on feature

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature)
$ git stash pop
Auto-merging data.html
On branch feature
Your branch is up to date with 'origin/feature'.

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:   data.html

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (34c28c750d345bdecb0d25c1c821be1d452b29ec)

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature)
$ git stash list
stash@{0}: On feature: work in progress on feature

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature)
$ git add .

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature)
$ git commit -m "updates"
[feature 0f77c8b] updates
1 file changed, 1 insertion(+), 1 deletion(-)
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

7. Run 'git add' command to add file in staging area then run "git commit -m 'msg'" command to save changes in working directory.
8. Switch on master branches and again run 'git stash list' to see stashes.
9. Now run 'git stash apply' command to apply the stashes and again run 'git add' command to add file in staging area then run "git commit -m 'msg'" command to save changes in working directory.

```
MINGW64:/d/my_pc/practice

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (feature)
$ git switch master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git stash list
stash@{0}: On feature: work in progress on feature

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git stash apply
Auto-merging data.html
On branch master
Your branch is up to date with 'origin/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:   data.html

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

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git stash list
stash@{0}: On feature: work in progress on feature

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git add .

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git commit -m "updates"
[master 68e5589] updates
1 file changed, 1 deletion(-)

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git stash clear

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git stash list

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git merge feature
Auto-merging Into.txt
CONFLICT (content): Merge conflict in Into.txt
Auto-merging data.html
CONFLICT (content): Merge conflict in data.html
Automatic merge failed; fix conflicts and then commit the result.
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

10. Run 'git stash clear' to remove the stash & then merge with feature branch using 'git merge feature' command here some conflicts occur resolve them.
11. After resolve run 'git add' command to add file in staging area then run "git commit -m 'msg' " command to save changes in working directory.
12. Run get push command to update remote repository with local changes.

```
WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master|MERGING)
$ git add .
WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master|MERGING)
$ git commit -m "updates"
[master 01f9150] updates

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git merge feature
Already up to date.

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git push
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Delta compression using up to 12 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (10/10), 952 bytes | 952.00 KiB/s, done.
Total 10 (delta 6), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (6/6), completed with 4 local objects.
To https://github.com/Aakash510/practice.git
   fffb290..01f9150  master -> master

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

WINDOWS 11@mr dx MINGW64 /d/my_pc/practice (master)
$ |
```

Part 9: Pushing to Remote

- **Objective:** Learn how to push changes to a remote repository.
- **Assignment:** Push changes after committing and verify on GITHUB.

Explanation:

1. Navigate to git directory using cd command, open file in editor to be modified using "nano file-name" command make changes and press 'ctrl+s' to save changes & 'ctrl+x' to close editor.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

2. Run 'git add' command to add file in staging area in showing ready to commit and finally run "git commit -m 'msg' " command to save changes in working directory.
3. Run 'git remote -v' to list the remote repositories associated with local repository.
4. Run 'git push -u origin master' command to update local changes into remote repository in master branch.

```
WINDOWS 11@mr dx MINGW64 /d/mynew1
$ ls
practice/

WINDOWS 11@mr dx MINGW64 /d/mynew1
$ cd practice

WINDOWS 11@mr dx MINGW64 /d/mynew1/practice (master)
$ ls
Into.txt  README.md  data.html

WINDOWS 11@mr dx MINGW64 /d/mynew1/practice (master)
$ nano data.html

WINDOWS 11@mr dx MINGW64 /d/mynew1/practice (master)
$ git add .

WINDOWS 11@mr dx MINGW64 /d/mynew1/practice (master)
$ git commit -m "Update table data"
[master b2a9596] Update table data
1 file changed, 1 insertion(+), 1 deletion(-)

WINDOWS 11@mr dx MINGW64 /d/mynew1/practice (master)
$ git remote -v
origin https://github.com/Aakash510/practice.git (fetch)
origin https://github.com/Aakash510/practice.git (push)

WINDOWS 11@mr dx MINGW64 /d/mynew1/practice (master)
$ git push -u origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 289 bytes | 289.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/Aakash510/practice.git
01f9150..b2a9596 master -> master
branch 'master' set up to track 'origin/master'.

WINDOWS 11@mr dx MINGW64 /d/mynew1/practice (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```


Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Part 10 : Rebasing Branches

- **Objective:** Learn how to use git rebase to maintain a clean commit history.
- **Assignment:** Rebase one branch on top of another and push changes

Explanation:

1. Navigate to git directory 'mynew2', run 'git clone <remote repo url>' command to clone remote repo locally & have a local copy of the repository including its history and branches. next run 'ls' command to view all files and directory and enter in git repo by run cd command.

```
MINGW64:/d/mynew2/practice

WINDOWS 11@mr dx MINGW64 /d
$ cd mynew2

WINDOWS 11@mr dx MINGW64 /d/mynew2
$ git clone https://github.com/Aakash510/practice.git

Cloning into 'practice'...
remote: Enumerating objects: 78, done.
remote: Counting objects: 100% (78/78), done.
remote: Compressing objects: 100% (53/53), done.
remote: Total 78 (delta 30), reused 65 (delta 21), pack-reused 0 (from 0)
Receiving objects: 100% (78/78), 10.75 KiB | 687.00 KiB/s, done.
Resolving deltas: 100% (30/30), done.
```

2. Run Cat >> "file-name" to create file and open file in editor to be modified by run "nano file-name" command, this open a file in nano editor. press 'ctrl+s' to save changes & 'ctrl+x' to close editor.
3. Run 'git add' command to add file in staging area in showing ready to commit and finally run "git commit -m 'msg'" command to save changes in working directory.

```
WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ ls
Into.txt  README.md  data.html

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ cat >> userprofile.js

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ nano userprofile.js

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ git add userprofile.js
warning: in the working copy of 'userprofile.js', LF will be replaced by CRLF the ne

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ git commit -m "Add user profile feature"
[feature1 f0c6616] Add user profile feature
1 file changed, 8 insertions(+)
create mode 100644 userprofile.js
```


Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

4. Switch master branch run 'git checkout master' and run 'git pull origin master' command to pull all the updates from remote repository & automatically locally.
5. Run 'git log --graph --oneline' command to view the commit history and with graph in oneline.

```
WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (master)
$ git pull origin master
From https://github.com/Aakash510/practice
* branch      master      -> FETCH_HEAD
Already up to date.

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (master)
$ git log --graph --oneline
* b2a9596 (HEAD -> master, origin/master, origin/HEAD) Update table data
* 01f9150 updates
| \
| * 0f77c8b updates
| * c454e51 (origin/feature) Merge branch 'master' into feature
| \
* | | 68e5589 updates
* | | fffb290 Updates
* | | 1f37171 Add table
```

6. Switch back to feature branch run 'git checkout feature' and run 'git log --graph --oneline' command to view the commit history with graph in oneline.

```
WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (master)
$ git checkout feature1
Switched to branch 'feature1'

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ git log --graph --oneline -n 6
* f0c6616 (HEAD -> feature1) Add user profile feature
* b2a9596 (origin/master, origin/HEAD, master) Update table data
* 01f9150 updates
| \
| * 0f77c8b updates
| * c454e51 (origin/feature) Merge branch 'master' into feature
| \
* | | 68e5589 updates
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

7. Run 'git rebase master' command to rebase it on top of the latest main branch.
8. Now check status of working tree by run 'git status' command.
9. To show local changes into remote repo, run 'git push -u origin feature2' to push local change to remote repository on GITHUB.
10. Create a pull request now go to GITHUB repository and create a pull request to merge feature1 back into main. be sure to describe what your feature does and any specific notes about your changes.

```
WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ git rebase master
Current branch feature1 is up to date.

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ git status
On branch feature1
nothing to commit, working tree clean

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ git push -u origin feature1
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 417 bytes | 417.00 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.
remote:
remote: Create a pull request for 'feature1' on GitHub by visiting:
remote:   https://github.com/Aakash510/practice/pull/new/feature1
remote:
To https://github.com/Aakash510/practice.git
 * [new branch]      feature1 -> feature1
branch 'feature1' set up to track 'origin/feature1'.

WINDOWS 11@mr dx MINGW64 /d/mynew2/practice (feature1)
$ git log --graph --oneline -n 6
* f0c6616 (HEAD -> feature1, origin/feature1) Add user profile feature
* b2a9596 (origin/master, origin/HEAD, master) Update table data
* 01f9150 updates
| \
| * 0f77c8b updates
| * c454e51 (origin/feature) Merge branch 'master' into feature
| | \
* | | 68e5589 updates
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Part 11: Resetting Changes

- **Objective:** Understand how to reset changes using `git reset --hard` and `git reset --soft`.

- **Assignment:** Reset changes and view the differences between hard and soft resets.

Explanation:

1. Navigate to git directory 'mynew3', run '`git clone <remote repo url>`' command to clone remote repo locally & have a local copy of the repository including its history and branches.
2. Create a branch using run '`git branch featuree`' command and switch on it Using '`git switch featuree`' command.
3. Create a file using '`echo 'msg' > <files-name>`' command and run '`git add`' command to add file in staging area then run "`git commit -m 'msg'`" command to save changes in working directory.

```
MINGW64:/d/mynew3/practice
WINDOWS 11@mr dx MINGW64 /d/mynew3
$ cd ..

WINDOWS 11@mr dx MINGW64 /d
$ cd mynew3

WINDOWS 11@mr dx MINGW64 /d/mynew3
$ git clone https://github.com/Aakash510/practice.git
Cloning into 'practice'...
remote: Enumerating objects: 82, done.
remote: Counting objects: 100% (82/82), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 82 (delta 31), reused 67 (delta 21), pack-reused 0 (from 0)
Receiving objects: 100% (82/82), 11.97 KiB | 817.00 KiB/s, done.
Resolving deltas: 100% (31/31), done.
```

```
WINDOWS 11@mr dx MINGW64 /d/mynew3
$ cd practice

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (master)
$ git branch
* master

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (master)
$ git branch featuree

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (master)
$ git checkout featuree
Switched to branch 'featuree'

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ # Add a file for the user profile feature
echo "User Profile Feature" > userProfile.js
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git add .
warning: in the working copy of 'userprofile.js', LF will be replaced by CRLF the next time you commit.

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git commit -m 'Initial commit for user profile featture'
[featuree 83670a3] Initial commit for user profile featture
1 file changed, 1 insertion(+), 8 deletions(-)

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ #Add more changes
echo "User name and email" >> userProfile.js

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git add .
warning: in the working copy of 'userprofile.js', LF will be replaced by CRLF the next time you commit.

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git commit -m "Add user name and email"
[featuree dcd09f5] Add user name and email
1 file changed, 1 insertion(+)

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ echo "User profile picture" >> userProfile.js

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git add .
warning: in the working copy of 'userprofile.js', LF will be replaced by CRLF the next time you commit.

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git commit -m "Add user profile picture"
[featuree ab08d04] Add user profile picture
1 file changed, 1 insertion(+)
```

4. Add content in file using 'echo 'msg' > <files-name>' command and run 'git add' command to add file in staging area then run "git commit -m 'msg' " command to save changes in working directory
5. Again add more content in file using 'echo 'msg' > <files-name>' command and run 'git add' command to add file in staging area then run "git commit -m 'msg' " command to save changes in working directory
6. Run 'git log --oneline -n 4' command to view the latest 4 commit history in oneline.
7. Now run 'git reset --soft head~1' to undo changes in the current branch and moves the HEAD pointer to a last commit.
Now your staging area has the changes from the "Add user profile picture" commit.
8. Run 'git status' command to view the status of your working directory, here showing file staged and ready to commit. run "git commit -m 'msg' " command to save changes in working directory.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git log --oneline -n 4
ab08d04 (HEAD -> featuree) Add user profile picture
dcd09f5 Add user name and email
83670a3 Initial commit for user profile featture
394a8b4 (origin/master, origin/HEAD, master) Merge pull request #4 from Aakash510/feature1

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ cat userprofile.js
User Profile Feature
User name and email
User profile picture

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git reset --soft HEAD~1 #Move back one commit (from "Add user profile picture")

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git log --oneline -n 4
dcd09f5 (HEAD -> featuree) Add user name and email
83670a3 Initial commit for user profile featture
394a8b4 (origin/master, origin/HEAD, master) Merge pull request #4 from Aakash510/feature1
f0c6616 (origin/feature1) Add user profile feature

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git status
On branch featuree
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   userprofile.js

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git commit -m "Add user name, email, and profile picture"
[featuree 5323402] Add user name, email, and profile picture
1 file changed, 1 insertion(+)

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git log --oneline -n 4
5323402 (HEAD -> featuree) Add user name, email, and profile picture
dcd09f5 Add user name and email
83670a3 Initial commit for user profile featture
394a8b4 (origin/master, origin/HEAD, master) Merge pull request #4 from Aakash510/feature1
```

9. Run 'git log --oneline -n 4' command to view the latest 4 commit history, notice that one commit is added.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

#git reset --hard

1. Run 'git log --oneline -n 4' command to view the latest 4 commit history.
2. Run git reset --hard HEAD~1 command to reset hard.

This will Moves the HEAD pointer to a specified commit and discards all changes in the staging area and the working directory.

This means any uncommitted changes (both staged and un-staged) will be removed and cannot be recovered with normal Git commands.

3. Run 'git status' command to view the status of your working directory, here showing nothing to commit, woking tree clean
4. Run 'git log --oneline -n 4' command to view the latest 4 commit history.

```
WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git log --oneline -n 4
5323402 (HEAD -> featuree) Add user name, email, and profile picture
dcd09f5 Add user name and email
83670a3 Initial commit for user profile featture
394a8b4 (origin/master, origin/HEAD, master) Merge pull request #4 from Aakash510/feature1

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git reset --hard HEAD~1 # This goes back to the initial commit
HEAD is now at dcd09f5 Add user name and email

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git status
On branch featuree
nothing to commit, working tree clean

WINDOWS 11@mr dx MINGW64 /d/mynew3/practice (featuree)
$ git log --oneline -n 4
dcd09f5 (HEAD -> featuree) Add user name and email
83670a3 Initial commit for user profile feature
394a8b4 (origin/master, origin/HEAD, master) Merge pull request #4 from Aakash510/feature1
f0c6616 (origin/feature1) Add user profile feature
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

Part 12: Creating an Issue and Fixing It

- **Objective:** Learn how to create and fix issues in GitHub repositories.
- **Assignment:** Create an issue, fix it on a new branch, and submit a pull request.

Explanation:

1. Create a directory run 'mkdir my1', navigate to git directory 'mynew2', run 'git clone <remote repo url>' command to clone remote repo locally & have a local copy of the repository including its history and branches.
2. next run 'ls' command to view all files and directory and enter in git repo by run cd command.

```
WINDOWS 11@mr dx MINGW64 /d
$ mkdir my1

WINDOWS 11@mr dx MINGW64 /d
$ cd my1

WINDOWS 11@mr dx MINGW64 /d/my1
$ git clone https://github.com/MasterHardik
Cloning into 'MasterHardik'...
remote: Not Found
fatal: repository 'https://github.com/MasterHardik/' not found

WINDOWS 11@mr dx MINGW64 /d/my1
$ git clone https://github.com/Aakash510/C-Programs-learning.git
Cloning into 'C-Programs-learning'...
remote: Enumerating objects: 320, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 320 (delta 0), reused 4 (delta 0), pack-reused 314 (from 1)
Receiving objects: 100% (320/320), 91.51 KiB | 574.00 KiB/s, done.
Resolving deltas: 100% (113/113), done.

WINDOWS 11@mr dx MINGW64 /d/my1
$ ls
C-Programs-learning/
```

3. Now create a branch using run 'git branch increment' command and switch on it and run 'ls' command to view all files and enter in programs file.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
WINDOWS 11@mr dx MINGW64 /d/my1
$ cd c-programs-learning

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning (main)
$ ls
'#100.c'  README.md  programs/

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning (main)
$ git branch increment

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning (main)
$ git checkout increment
Switched to branch 'increment'

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning (increment)
$ ls
'#100.c'  README.md  programs/

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning (increment)
$ cd programs

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning/programs (increment)
$ ls
'#1.c'    '#13.c'   '#18.c'   '#22.c'   '#27.c'   '#31.c'   '#36.c'   '#40.c'   '#45.c'   '#5.c'
'#10.c'   '#14.c'   '#19.c'   '#23.c'   '#28.c'   '#32.c'   '#37.c'   '#41.c'   '#46.c'   '#50.c'
'#100.c'  '#15.c'   '#2.c'    '#24.c'   '#29.c'   '#33.c'   '#38.c'   '#42.c'   '#47.c'   '#51.c'
'#11.c'   '#16.c'   '#20.c'   '#25.c'   '#3.c'    '#34.c'   '#39.c'   '#43.c'   '#48.c'   '#52.c'
'#12.c'   '#17.c'   '#21.c'   '#26.c'   '#30.c'   '#35.c'   '#4.c'    '#44.c'   '#49.c'   '#53.c'
```

4. Run cat To view the file content then run “nano ‘#1.c’” command, this open a file in nano editor make changes & press ‘ctrl+s’ to save changes & ‘ctrl+x’ to close editor.
5. Run ‘git add’ command to add file in staging area in showing ready to commit and finally run “git commit -m ‘msg’ ” command to save changes in working directory.
6. Run ‘git push -u origin master’ command to update local changes into remote repository in master branch.
7. Run ‘git log --graph --oneline -n 5’ command to view the latest 5 commit history with graph in oneline.

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

```
WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning/programs (increment)
$ cat '#1.c'
//1. Program to print "Hello, Future Coders !!".

#include<stdio.h>

void main()
{
    printf("Hello, Future Coders !!");
}

// Join telegram here : https://t.me/+352cNILEGdJmNGJl

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning/programs (increment)
$ nano '#1.c'

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning/programs (increment)
$ cat '#1.c'
//1. Program to print "Hello, Future Coders !!".

#include<stdio.h>

void main()
{
    printf("Hello, Future Coders !!");
}

// Join telegram here : https://t.me/+352cNILEGdJmNGJl
// Join linkedin here : https://www.linkedin.com/in/hardik-chouhan-870785236
```

```
WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning/programs (increment)
$ git add '#1.c'

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning/programs (increment)
$ git commit -m "Add linkedin URL"
[increment 827c19b] Add linkedin URL
1 file changed, 1 insertion(+)

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning/programs (increment)
$ git push origin increment
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 433 bytes | 433.00 KiB/s, done.
Total 4 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
remote:
remote: Create a pull request for 'increment' on GitHub by visiting:
remote:      https://github.com/Aakash510/C-Programs-learning/pull/new/increment
remote:
To https://github.com/Aakash510/C-Programs-learning.git
 * [new branch]      increment -> increment

WINDOWS 11@mr dx MINGW64 /d/my1/c-programs-learning/programs (increment)
$ git log --oneline -n 5
827c19b (HEAD -> increment, origin/increment) Add linkedin URL
4a65494 (origin/main, origin/HEAD, main) Updated: programs directory, Added: README.md
76594ad Create #100.c
489d3bb Create #99.c
12ff22a Create #98.c
```

Assignment Submission

GIT Part-3

Submitted by: Aakash Chourasiya

Submitted to: **HotWax Commerce**

8. Create a pull request now go to GITHUB repository and create a pull request to merge feature1 back into main. be sure to describe what your feature does and any specific notes about your changes.

