Git Practical Exam (LMS)

Parth Shah
Simform Solutions

Overview

This exercise demonstrates practical Git workflows using a combination of git-flow, feature branching, and advanced operations like **squash, rebase, reset, and cherry-pick**.

Goals

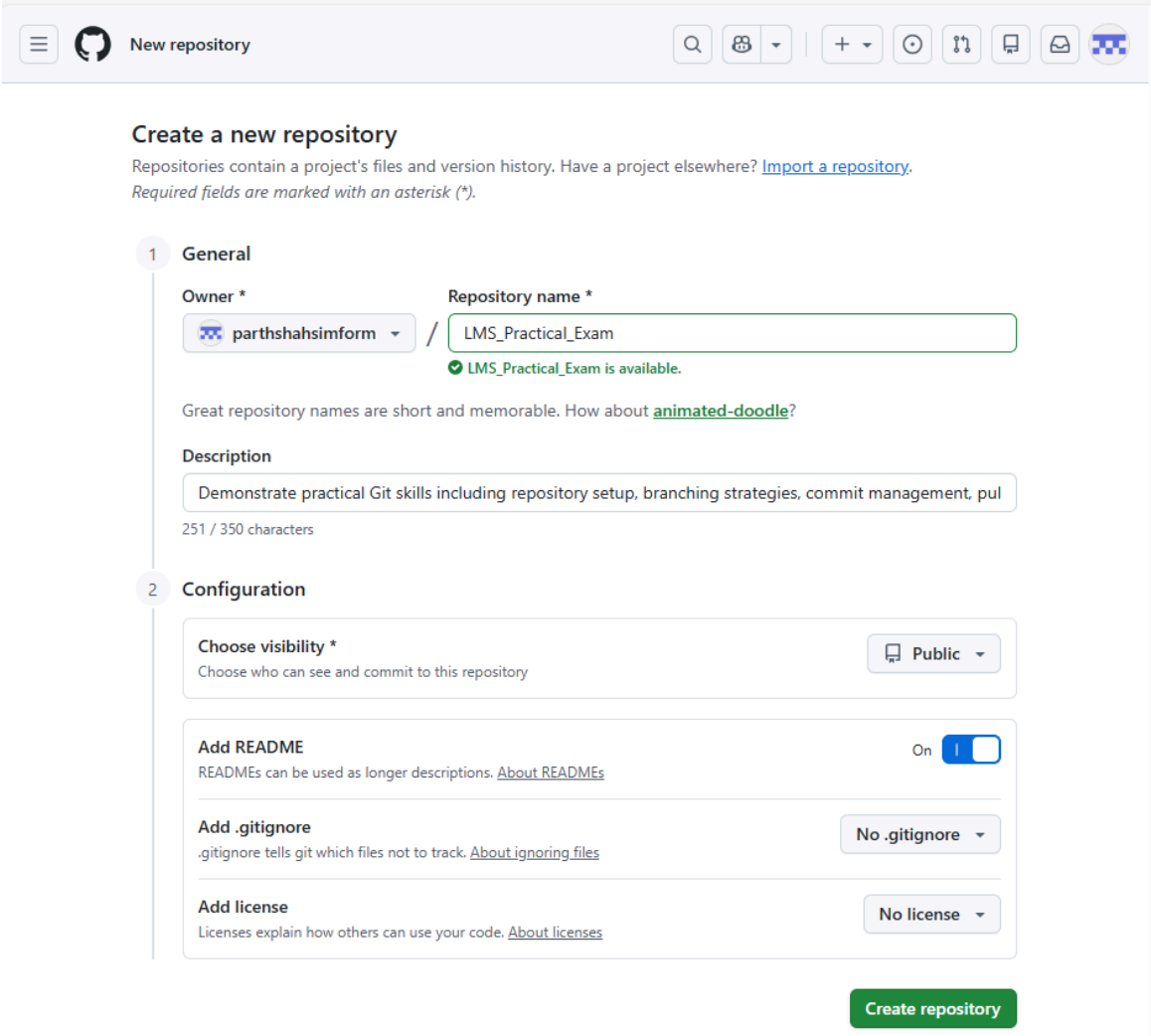
- Learn to **initialize**, **clone** and use **Git-flow** in a repository.
- Practice creating **features and sub-branches**.
- Apply advanced Git commands: **squash, rebase, reset, cherry-pick** and keep the **pull requests (PRs) clean and concise.**

- Understand **branch management** when previous PRs are under review.
- Maintain a **version history** by tagging builds.
- Perform **commit message** updates and manage **branch deletions** safely.

Specifications

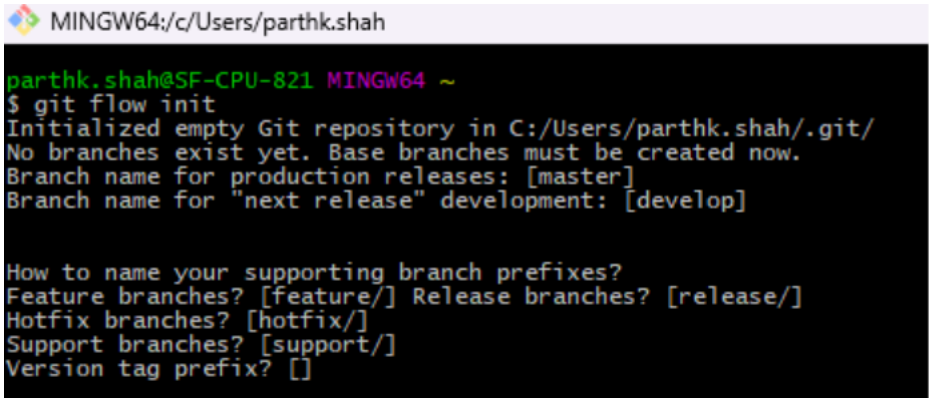
Repository setup

- Created a new repository using a template.



- Clone the Repo and Initialize the Git-flow.

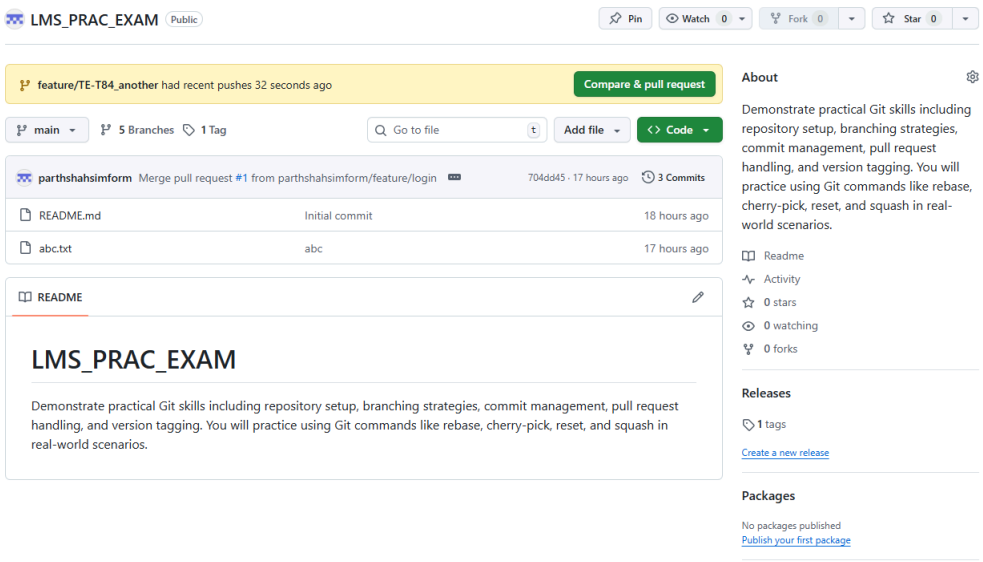
```
C:\Users\parthk.shah>cd Documents

C:\Users\parthk.shah\Documents>git clone https://github.com/parthshah02/LMS_Basics_of_Git.git
Cloning into 'LMS_Basics_of_Git'...
remote: Enumerating objects: 15, done.
remote: Counting objects: 100% (15/15), done.
remote: Compressing objects: 100% (11/11), done.
remote: Total 15 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (15/15), 5.43 KiB | 2.71 MiB/s, done.
Resolving deltas: 100% (1/1), done.
```



Feature branch workflow

Create a feature branch from develop for project setup:

```
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git checkout -b TE-T84_Project
Switched to a new branch 'TE-T84_Project'
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git commit -m "created a proper zoho-id feature branch"
On branch TE-T84_Project
```



// feature branch to develop

```
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git add bug.txt
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git commit -m "merging to develop"
[feature/TE-T84_fix e449d41] merging to develop
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 bug.txt
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git status
On branch feature/TE-T84_fix
nothing to commit, working directory clean
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git push -u origin feature/TE-T84_fix
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 334 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'feature/TE-T84_fix' on GitHub by visiting:
remote:      https://github.com/parthshahsimform/LMS_PRAC_EXAM/pull/new/feature/TE-T84_fix
remote:
To git@github.com:parthshahsimform/LMS_PRAC_EXAM.git
 * [new branch]      feature/TE-T84_fix -> feature/TE-T84_fix
Branch feature/TE-T84_fix set up to track remote branch feature/TE-T84_fix from origin.
```

Sub-branch for practice (squash, reset, rebase, cherry-pick)

RESET

In Git, reset moves your current branch (HEAD) to a different commit, and depending on the option you use

Soft reset (keeps changes staged)

Mixed reset (keeps changes unstaged)

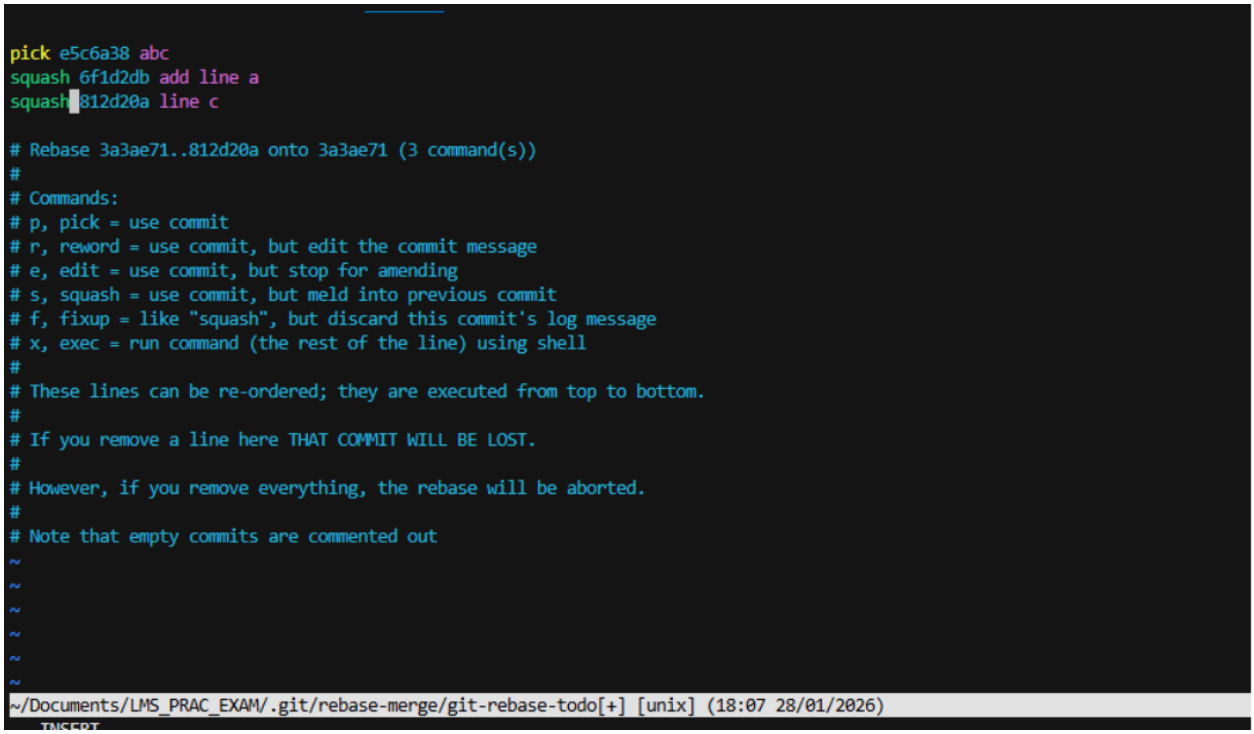
Hard reset (discards changes)

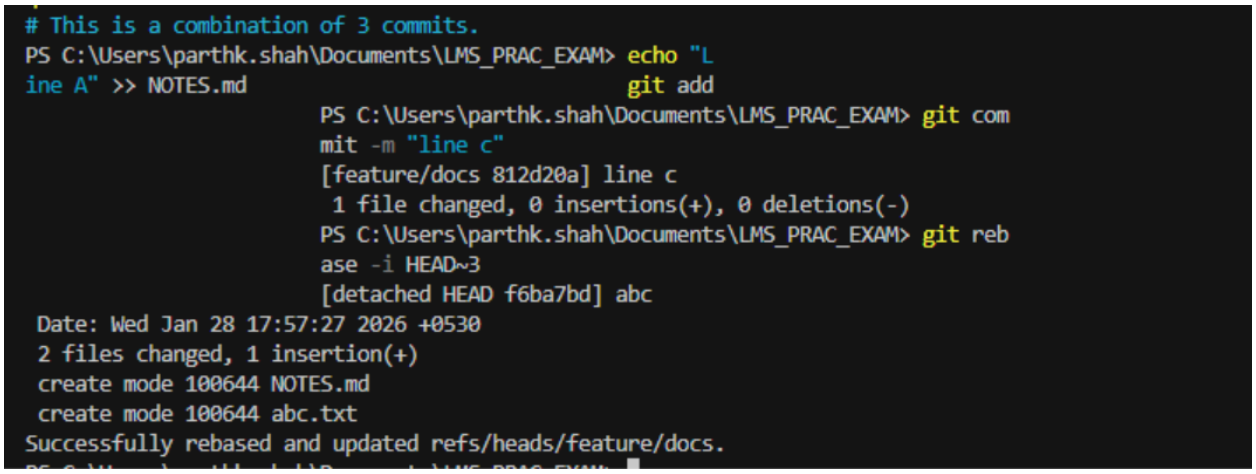
```
PS C:\Users\parthk.shah\Documents\LMS_Basics_of_Git> git log --oneline
ca8dbc9 docs: add contributing section
77d1d7b Merge pull request #1 from parthshah02/parthshah02-test-1
3d945b9 Add initial content to test_change.txt
fcf856d Add git pull command explanation
f73e8c4 Document Git commands and their usage
ec4ade7 Create basics.txt
e3090ff Updated README.md
274d6cb Initial commit
PS C:\Users\parthk.shah\Documents\LMS_Basics_of_Git>
>> git reset --soft HEAD~1
>> git commit -m "docs: contributing + standards (squashed)"
>>
[TE-T84_practical_lms 6881ffa] docs: contributing + standards (squashed)
 1 file changed, 0 insertions(+), 0 deletions(-)
 rewrite README.md (100%)
PS C:\Users\parthk.shah\Documents\LMS_Basics_of_Git> git log --oneline
6881ffa docs: contributing + standards (squashed)
77d1d7b Merge pull request #1 from parthshah02/parthshah02-test-1
3d945b9 Add initial content to test_change.txt
fcf856d Add git pull command explanation
f73e8c4 Document Git commands and their usage
ec4ade7 Create basics.txt
e3090ff Updated README.md
274d6cb Initial commit
```

SQUASH

Squash combines multiple commits into a single commit.

Mark the first commit as pick, and the rest as squash (or s) as per the requirements .





REBASE

Rebase moves or reapplies commits from one branch onto another, creating a linear history.

```
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git checkout develop
Switched to branch 'develop'
Your branch is up-to-date with 'origin/develop'.
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git pull --ff-only
Already up-to-date.
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git checkout -b another
Switched to a new branch 'another'
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> echo 'change' >> second.txt
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git add second.txt
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git commit -m 'another change'
[another ce8cd7e] another change
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 second.txt
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git push -u origin another
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 298 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'another' on GitHub by visiting:
remote:      https://github.com/parthshahsimform/LMS_PRAC_EXAM/pull/new/another
remote:
To git@github.com:parthshahsimform/LMS_PRAC_EXAM.git
 * [new branch]      another -> another
Branch another set up to track remote branch another from origin.
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git fetch origin
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git rebase origin/develop
Current branch another is up to date.
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git fetch origin
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (1/1), done.
From github.com:parthshahsimform/LMS_PRAC_EXAM
   3a3ae71..704dd45  main       -> origin/main
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git push -f origin another
Everything up-to-date
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM>
```

CHERRY-PICK

Cherry-pick lets you **apply a specific commit from one branch onto another**. Instead of merging or rebasing an entire branch, you can "pick" just one commit and replay it in your current branch.

```
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM>
>> git checkout develop && git pull
>> git checkout -b feature/TE-T84
>> echo "Minor README CHANGE" >> README.md
>> git add README.md
>> git commit -m "TE-T84: Update README"
>> git push -u origin feature/TE-T84
>>
>>
>> git checkout develop
>> git checkout -b feature/TE-T84
>>
>>
>> git log --oneline origin/feature/
>> git cherry-pick 7a7579d
>>
>>
>> git commit --amend -m "cherry-pick tset"
>>
>> echo "C1" >> misc.txt && git add misc.txt && git commit -m "TE-T84: C1"
>> echo "C2" >> misc.txt && git add misc.txt && git commit -m "TE-T84: C2"
>> echo "C3" >> misc.txt && git add misc.txt && git commit -m "TE-T84: C3"
>> git reset --soft HEAD~1
>> git revert HEAD
```

REVERT and DELETE

Revert - Creates a new commit that undoes the changes of a previous commit.

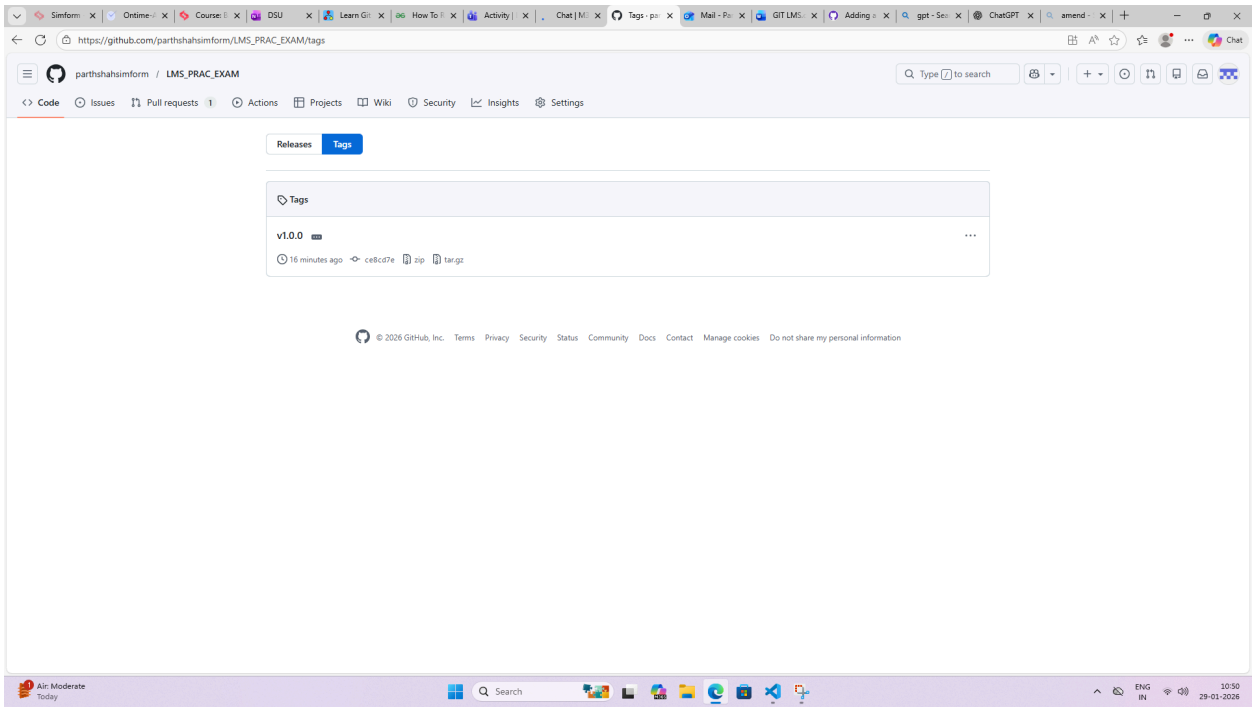
Delete - In Git, delete usually refers to removing a branch or a tag. It doesn't delete commits themselves (unless they're unreachable), but it removes the reference (pointer) to them.

```
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git push -u origin feature/TE-T84_another
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 329 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'feature/TE-T84_another' on GitHub by visiting:
remote:      https://github.com/parthshahsimform/LMS_PRAC_EXAM/pull/new/feature/TE-T84_another
remote:
To git@github.com:parthshahsimform/LMS_PRAC_EXAM.git
 * [new branch]      feature/TE-T84_another -> feature/TE-T84_another
Branch feature/TE-T84_another set up to track remote branch feature/TE-T84_another from origin.
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git push origin --delete another
To git@github.com:parthshahsimform/LMS_PRAC_EXAM.git
 - [deleted]         another
```

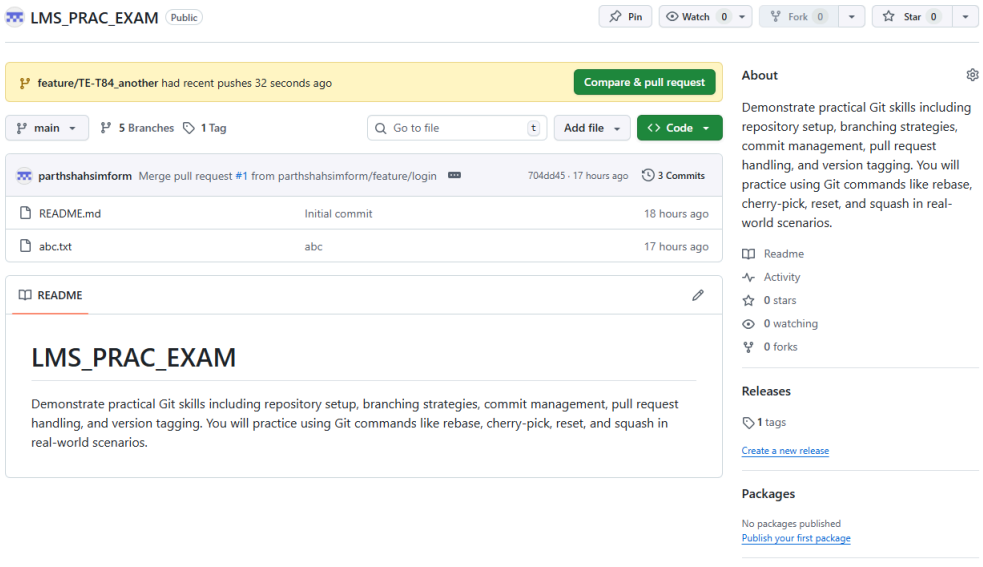
VERSION TAGS

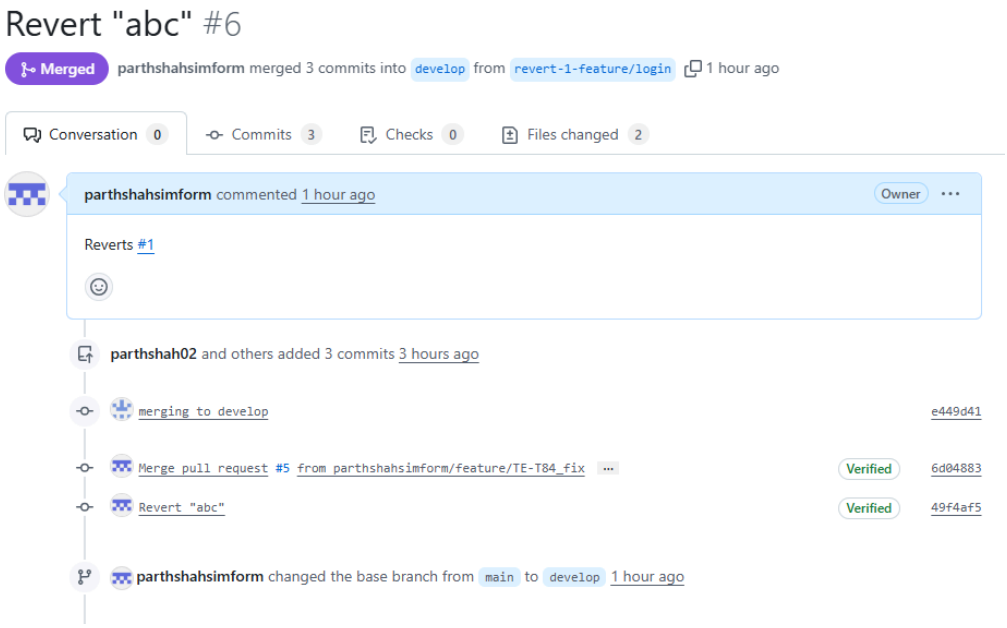
- **Version tags** are tags used to label commits that represent official release versions of your project.
- They act like permanent bookmarks in history, so you can always return to or reference that exact state of the code.

```
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git tag -a v1.0.0 -m "Release v1.0.0"
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git push origin v1.0.0
Counting objects: 1, done.
Writing objects: 100% (1/1), 166 bytes | 0 bytes/s, done.
Total 1 (delta 0), reused 0 (delta 0)
remote: To git@github.com:parthshahsimform/LMS_PRAC_EXAM.git
 * [new tag]         v1.0.0 -> v1.0.0
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM>
```



PRACTICE SCENARIO





COMMITTING MESSAGES

```
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> echo Hello > file1.txt
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git add file1.txt
PS C:\Users\parthk.shah\Documents\LMS_PRAC_EXAM> git commit -m "added file1.txt"
[another 26f787c] added file1.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 file1.txt
```