



**Academic Year: 2025-26**

**Name : Siddhi Tangsali**

**Semester: V**

**Student Id : 23104069**

**Date of Submission: 23/7/25**

**Class / Branch: TEIT**

**Subject: DevOps Lab**

**Name of Instructor: Ms. Sujata Oak**

---

### **Experiment No. 3**

**Aim: To understand and perform version control system / source code management using Git Branch.**

#### **Theory:**

**Branching** means creating a separate line of work in your project so you can make changes without affecting the main version.

It's like making a copy of your project to try new things (like adding a feature or fixing a bug), and then later you can combine your changes back into the main project.

When a repository is created, the files are automatically put in a branch called **main**. Whenever possible it is recommended to use branches rather than directly updating the main branch. Branching is used so that you can make changes in another area without affecting the main branch. This is done to help prevent accidental updates that might overwrite existing code.

In this lab, students will implement:

- **create a new branch,**
- **checkout the branch,**
- **make changes in the branch,**
- **stage**
- **commit** the branch
- **merge** the branch changes to the main branch, and
- **delete** the branch.

#### **Implementation:**

**STEP1:**–Goto your working directory

Compiled By Prof. Sujata Oak



PARSHVANATH CHARITABLE TRUST'S

# A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
apsit@apsit-HP-ProDesk-600-G4-PCI-MT:~$ cd Desktop
apsit@apsit-HP-ProDesk-600-G4-PCI-MT:~/Desktop$ cd git-SiddhiT
apsit@apsit-HP-ProDesk-600-G4-PCI-MT:~/Desktop/git-SiddhiT$ ls
exp2.c  gitdemo16072025  token
apsit@apsit-HP-ProDesk-600-G4-PCI-MT:~/Desktop/git-SiddhiT$ sudo su
[sudo] password for apsit:
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT# pwd
/home/apsit/Desktop/git-SiddhiT
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT# ls
exp2.c  gitdemo16072025  token
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT#
```

List the files in your working directory:

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT# ls -altr
total 24
drwxr-xr-x  7 root  root  4096 Jul 16 10:51 .git
-rw-rw-r--  1 apsit apsit   41 Jul 16 11:19 token
drwxr-xr-x  3 root  root  4096 Jul 16 11:54 gitdemo16072025
-rw-rw-r--  1 apsit apsit  1580 Jul 16 15:31 exp2.c
drwxrwxr-x  4 apsit apsit  4096 Jul 16 15:32 .
drwxr-xr-x 37 apsit apsit  4096 Jul 23 09:56 ..
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT#
```

Change the directory to your git remote repository:

```
root@labvm:/home/devasc/Desktop/git-sujata# cd gitdemo11072025/
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# ls
factorial.py  mynewfile  name.html
```

**Step2:** Verify Current branch : Use **git branch** command without a branch-name to display all the branches for this repository. Git branch command tells you how many branches you have and you are working on which branch(\*)

The "\*" next to the main branch indicates that this is the current branch

Compiled By Prof. Sujata Oak



PARSHVANATH CHARITABLE TRUST'S

**A. P. SHAH INSTITUTE OF TECHNOLOGY**

**Department of Information Technology**

**(NBA Accredited)**



```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch
* main
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch feature
```

**Step3:** Create a new branch feature: **Feature Branch:** Allows you to develop features, fix bugs, or safely experiment with new ideas in a contained area of your repository

Create a new branch called **featureA** using the **git branch** <branch-name>



```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch
* main
```

**Step 4:** Checkout the new branch

Use the **git checkout** <branch-name> command to switch to the *feature* branch.

**Step 5:** Verify the current branch:

- Verify you have switched to the *feature* branch using the **git branch** command. Note the "\*" next to the *feature* branch. This is now the *working branch*.

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch featurec
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch
featurec
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch
featurec
* main
```

Create a file add.py

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# touch add.py
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# gedit add.py
```

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# cat add.py
n1=10
n2=20
add=(n1+n2)
print(add)

root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

**Step 6:** Stage the modified file in the feature branch

A) Stage the updated file to the current *feature* branch.

#git status



```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git status
On branch featurec
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    add.py

nothing added to commit but untracked files present (use "git add" to track)
```

git add add.py

B] Use the **git status** command and notice the modified file **add.py** is staged in the *feature* branch  
#git status

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git add add.py
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git status
On branch featurec
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   add.py

root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

### Step 7: Commit the staged file in the feature branch

A] Commit the staged file using the **git commit** command. Notice the new commit ID and your message.

```
new file:   add.py
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git commit -m "Added add.py file for commit on 23072025"
[featurec 202cde0] Added add.py file for commit on 23072025
1 file changed, 5 insertions(+)
create mode 100644 add.py
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

**Step 8:** Use the **git log** command to show all commits including the commit you just did to the *feature* branch.

The prior commit was done within the *main* branch.





PARSHVANATH CHARITABLE TRUST'S

# A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git log
commit 202cde0d860ca28697bbc732c7dc6cb3339fe350 (HEAD -> featurec)
Author: itsSIDDHEE <shankk905@gmail.com>
Date: Wed Jul 23 10:50:08 2025 +0530

    Added add.py file for commit on 23072025

commit 6be7f0e36164610e946882347f026da237e36ff8 (main)
Author: itsSIDDHEE <shankk905@gmail.com>
Date: Wed Jul 16 11:54:02 2025 +0530

    Revert "Added feedback"

    This reverts commit 4b7713cb5330f33494eb6cfe5bea2f796e4e15ac.

commit 4b7713cb5330f33494eb6cfe5bea2f796e4e15ac
Author: itsSIDDHEE <shankk905@gmail.com>
Date: Wed Jul 16 11:47:45 2025 +0530

    Added feedback

commit c20a3c3bd878118a8321a81114fcf6a6528443e7 (origin/main, origin/HEAD)
Author: Techiee <shankk905@gmail.com>
Date: Wed Jul 16 11:35:47 2025 +0530

    Create mynewfile1

    I am committing mynewfile 1

commit cc74c3c9fedfcc8847f6bb5b48d0bf4f2c54c5ab
Author: itsSIDDHEE <shankk905@gmail.com>
Date: Wed Jul 16 11:08:18 2025 +0530

    first hello.py file

commit 6bf380a150a4cac0abe88b3323a0674ef179db49
Author: Techiee <shankk905@gmail.com>
Date: Wed Jul 16 10:50:08 2025 +0530

    Initial commit
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

Press a **key q** to exit from git log screen

**Step9:** How to push a **featureA** branch from local repository to remote repository?

#git push --set-upstream origin featureA



PARSHVANATH CHARITABLE TRUST'S  
**A. P. SHAH INSTITUTE OF TECHNOLOGY**  
Department of Information Technology  
(NBA Accredited)



```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git push --set-upstream origin featurec
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 894 bytes | 894.00 KiB/s, done.
Total 7 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'featurec' on GitHub by visiting:
remote:   https://github.com/itsSIDDHEE/gitdemo16072025/pull/new/featurec
remote:
To https://github.com/itsSIDDHEE/gitdemo16072025
 * [new branch]      featurec -> featurec
Branch 'featurec' set up to track remote branch 'featurec' from 'origin'.
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

Goto [github.com](https://github.com/itsSIDDHEE/gitdemo16072025) account Refresh the page



PARSHVANATH CHARITABLE TRUST'S  
**A. P. SHAH INSTITUTE OF TECHNOLOGY**  
Department of Information Technology  
(NBA Accredited)



The screenshot shows a GitHub repository page for 'gitdemo16072025'. The repository is public and has 2 branches and 0 tags. A notification from 'featureec' indicates recent pushes 2 minutes ago. The repository contains three files: README.md (Initial commit, last week), hi.py (first hello.py file, last week), and mynewfile (Create mynewfile1, last week). The README file is selected, showing the title 'gitdemo16072025' and the content 'setup git demo on remote repository'. The right sidebar shows the repository's activity, including 0 stars, 0 watching, and 0 forks. The 'Releases' section shows no published releases, and the 'Packages' section shows no published packages. The 'Languages' section shows Python as the primary language at 100.0%.

**Select featureA branch. You see the same contents in featureA branch as was in main branch.**





PARSHVANATH CHARITABLE TRUST'S  
**A. P. SHAH INSTITUTE OF TECHNOLOGY**  
Department of Information Technology  
(NBA Accredited)



gitdemo16072025 Public

Pin Watch 0 Fork

featureec had recent pushes 3 minutes ago Compare & pull request

featureec 2 Branches 0 Tags Go to file Add file <> Code

This branch is 3 commits ahead of `main`. Contribute

itsSIDDHEE Added add.py file for commit on 23072025 202cde0 · 5 minutes ago 6 Commits

README.md	Initial commit	last week
add.py	Added add.py file for commit on 23072025	5 minutes ago
hi.py	first hello.py file	last week
mynewfile	Revert "Added feedback"	last week

README

gitdemo16072025

About

setup git demo

Readme Activity 0 stars 0 watching 0 forks

Releases

No releases published [Create a new release](#)

Packages

No packages published [Publish your first package](#)

Languages

Python 100.0

Click on view all branch:

Compiled By Prof. Sujata Oak

Department of Information Technology | APSIT



PARSHVANATH CHARITABLE TRUST'S  
**A. P. SHAH INSTITUTE OF TECHNOLOGY**  
Department of Information Technology  
(NBA Accredited)



GitHub repository view for `gitdemo11072025`. The interface shows the **Branches** tab with a search bar and a **New branch** button. Below the search bar, there are three sections: **Default**, **Your branches**, and **Active branches**. Each section contains a table of branches with columns for Branch, Updated, Check status, Behind/Ahead, and Pull request.

Branch	Updated	Check status	Behind	Ahead	Pull request
main	5 days ago			Default	

Branch	Updated	Check status	Behind	Ahead	Pull request
featureA	5 minutes ago		0	4	

Branch	Updated	Check status	Behind	Ahead	Pull request
featureA	5 minutes ago		0	4	

Click on add.py file to see the contents of add.py

GitHub file view for `gitdemo16072025 / add.py`. The interface shows the **Files** tab with a search bar and a list of files. The file `add.py` is selected, and its contents are displayed in the **Code** tab.

itsSIDDHEE Added add.py file for commit on 23072025

```
1 n1=10
2 n2=20
3 add=(n1+n2)
4 print(add)
```

Now in local repository in featureA branch create 3 files:



```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch
* featurec
main
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# touch feature-1
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# touch feature-2
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# touch feature-3
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# ls
add.py feature-1 feature-2 feature-3 hi.py mynewfile README.md
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

Suppose the developers, they are done with their coding.  
Now they want to **merge** back the content which you have put it in the **featureA** branch to the **main** branch, and then they can delete that branch.

For that they have to **raise the pull request. That is PR.**

#git status

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git status
On branch featurec
Your branch is up to date with 'origin/featurec'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    feature-1
    feature-2
    feature-3

nothing added to commit but untracked files present (use "git add" to track)
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

#git add .

#git status



PARSHVANATH CHARITABLE TRUST'S

**A. P. SHAH INSTITUTE OF TECHNOLOGY**

**Department of Information Technology**

**(NBA Accredited)**



```
oot@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git add .
oot@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git status
On branch featurec
Your branch is up to date with 'origin/featurec'.
```

```
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   feature-1
    new file:   feature-2
    new file:   feature-3
```

```
oot@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

# git commit -m " Added 3 files in featureA branch"



PARSHVANATH CHARITABLE TRUST'S

# A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git commit -m "Added 3 files in featurec branch"
[featurec 55040ff] Added 3 files in featurec branch
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 feature-1
create mode 100644 feature-2
create mode 100644 feature-3
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

# git push -u origin featureA

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git push -u origin featurec
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), 395 bytes | 395.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/itsSIDDHEE/gitdemo16072025
 202cde0..55040ff featurec -> featurec
Branch 'featurec' set up to track remote branch 'featurec' from 'origin'.
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

**Goto github.com account**





PARSHVANATH CHARITABLE TRUST'S  
**A. P. SHAH INSTITUTE OF TECHNOLOGY**  
Department of Information Technology  
(NBA Accredited)



gitdemo16072025 Public

featurec had recent pushes 53 seconds ago [Compare & pull request](#)

featurec 2 Branches 0 Tags

This branch is 4 commits ahead of main.

File	Commit Message	Time
README.md	Initial commit	last week
add.py	Added add.py file for commit on 23072025	15 minutes ago
feature-1	Added 3 files in featurec branch	2 minutes ago
feature-2	Added 3 files in featurec branch	2 minutes ago
feature-3	Added 3 files in featurec branch	2 minutes ago
hi.py	first hello.py file	last week
mynewfile	Revert "Added feedback"	last week

[README](#)

**About**  
setup git demo on remote repository  
Readme  
Activity  
0 stars  
0 watching  
0 forks

**Releases**  
No releases published  
[Create a new release](#)

**Packages**  
No packages published  
[Publish your first package](#)

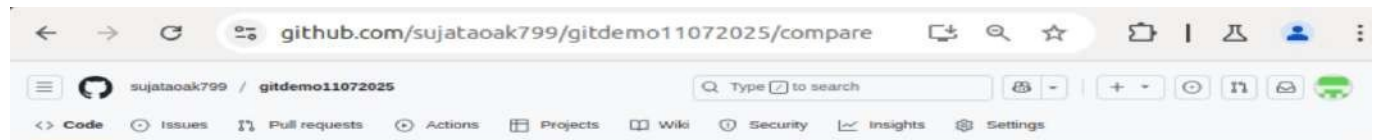
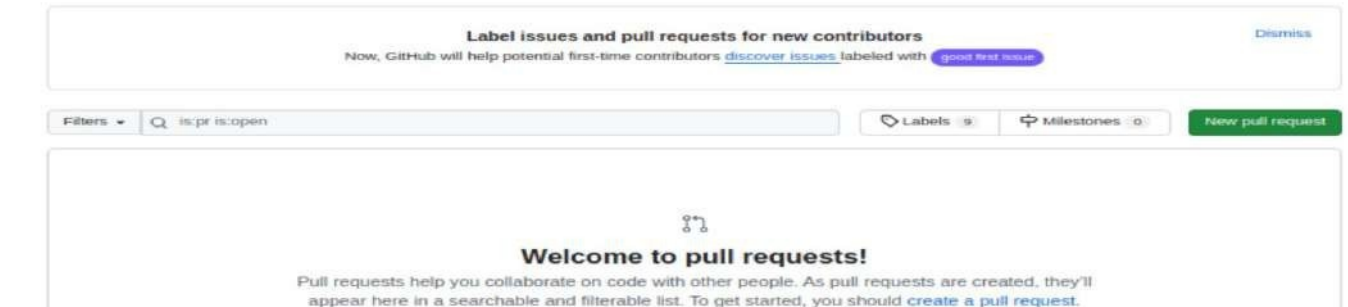
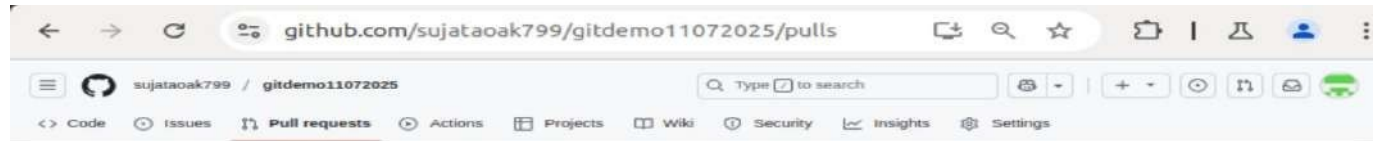
**Languages**  
Python 100.0%

If you go to **main** branch , files of **feature-A** branch will not be seen. For that we need to do **merging**

**For that goto feature-A branch → pull requests tab → click on New Pull Requests**



PARSHVANATH CHARITABLE TRUST'S  
**A. P. SHAH INSTITUTE OF TECHNOLOGY**  
Department of Information Technology  
(NBA Accredited)



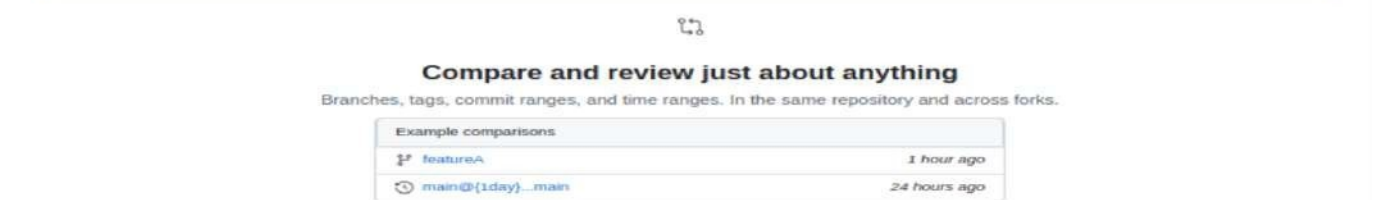
### Compare changes

Compare changes across branches, commits, tags, and more below. If you need to, you can also [compare across forks](#).

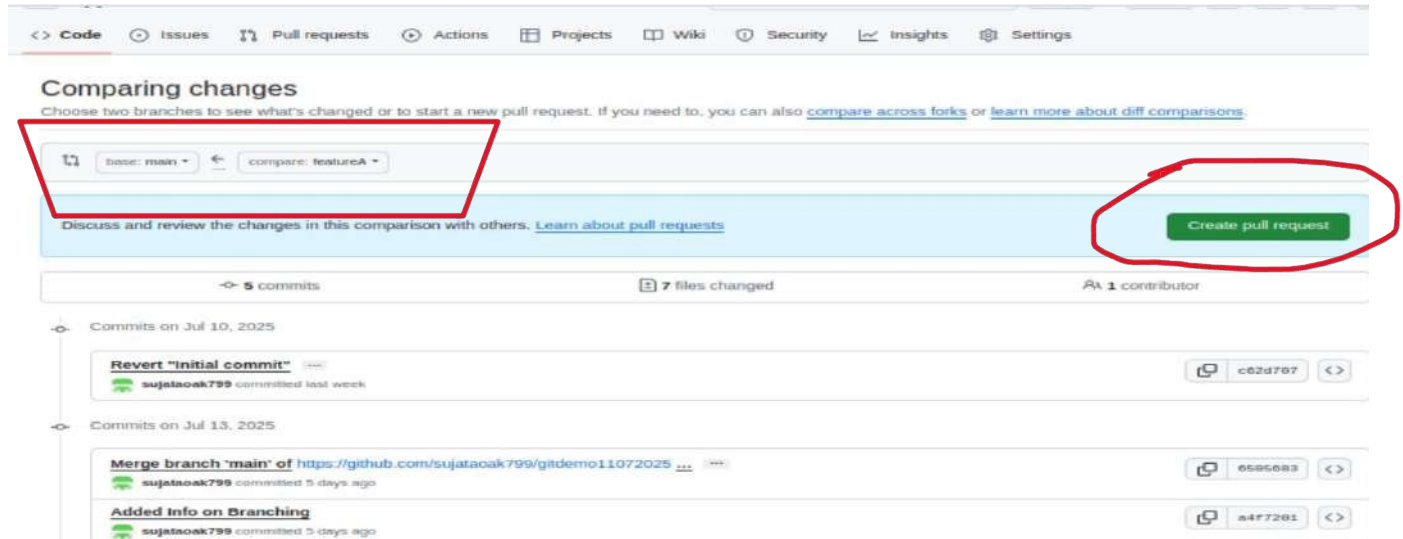
base: main ← compare: main

Choose different branches or forks above to discuss and review changes. [Learn about pull requests](#)

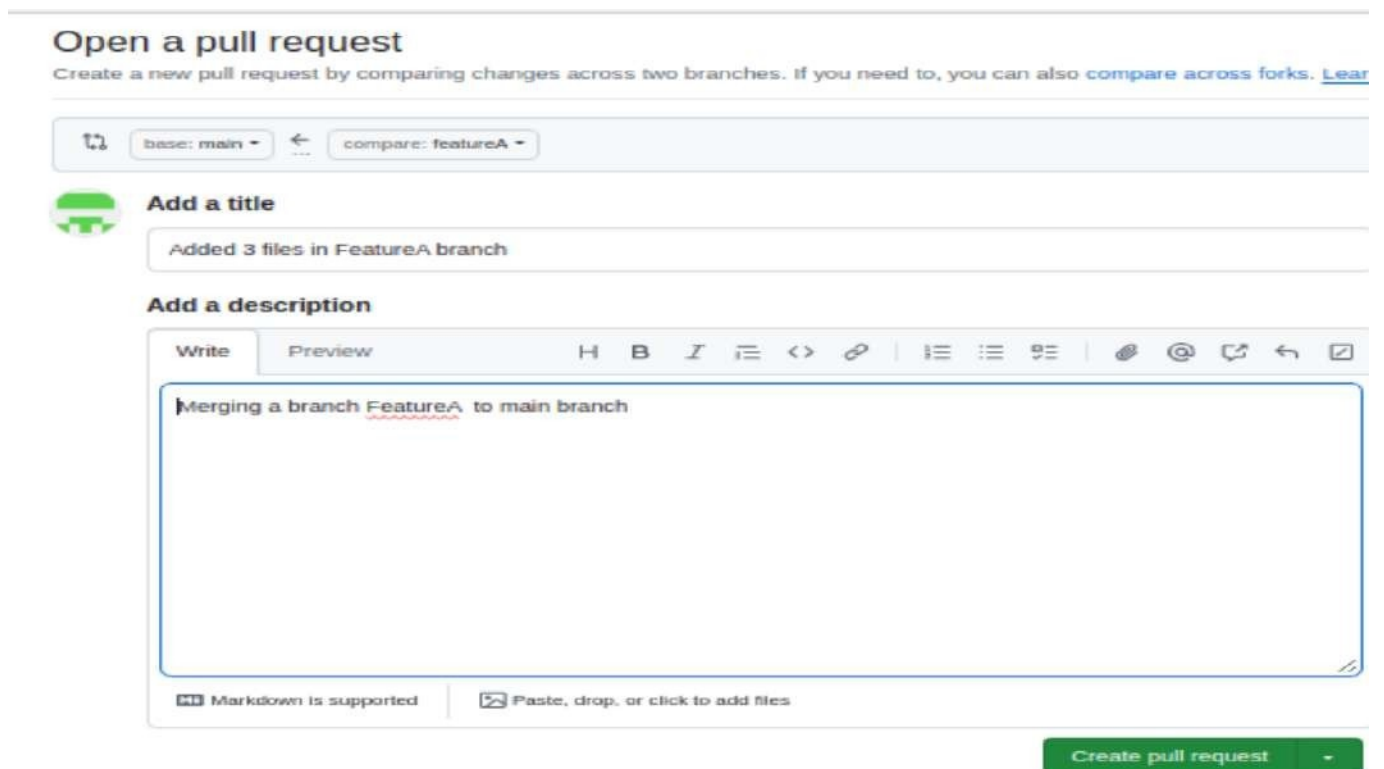
Create pull request



**Compare changes from main branch and featureA branch**

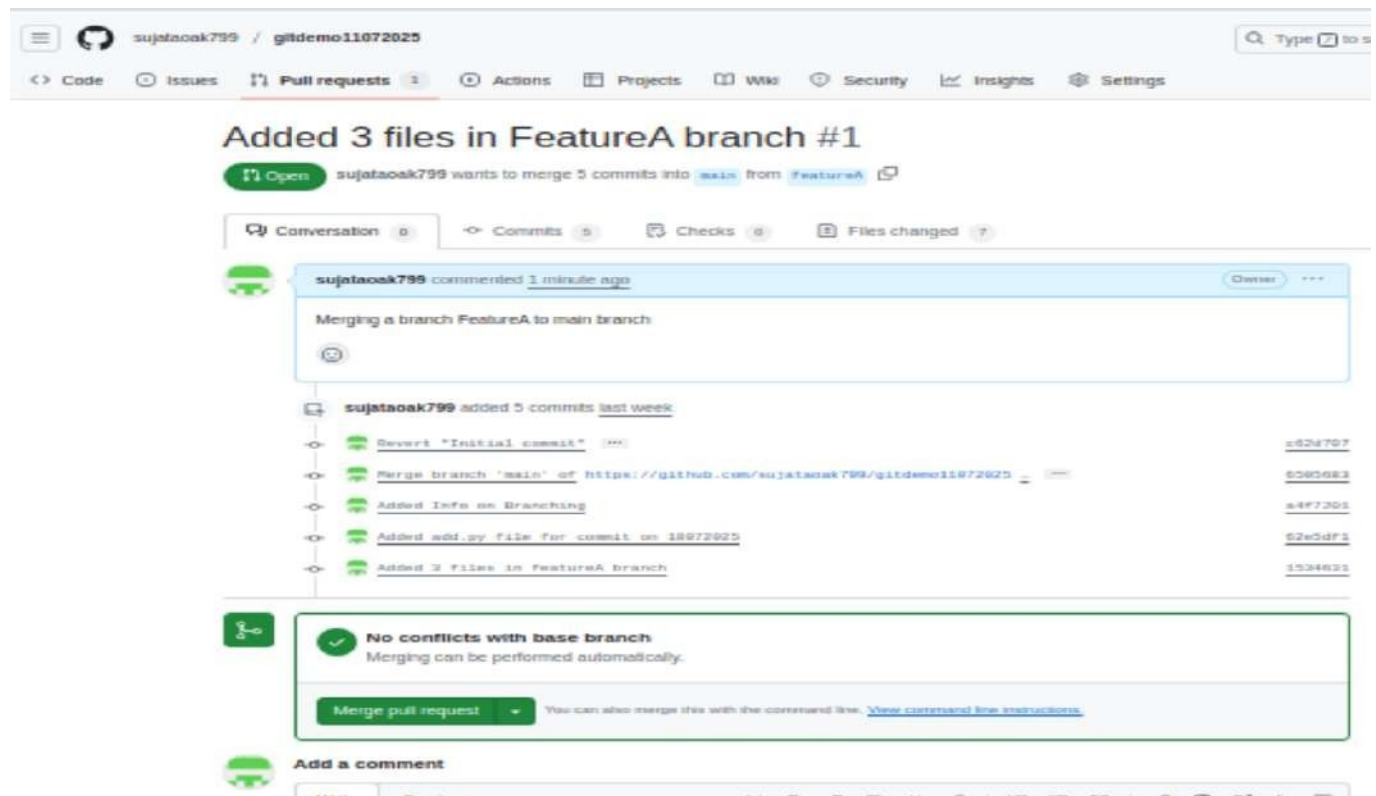


## Click on create pull requests

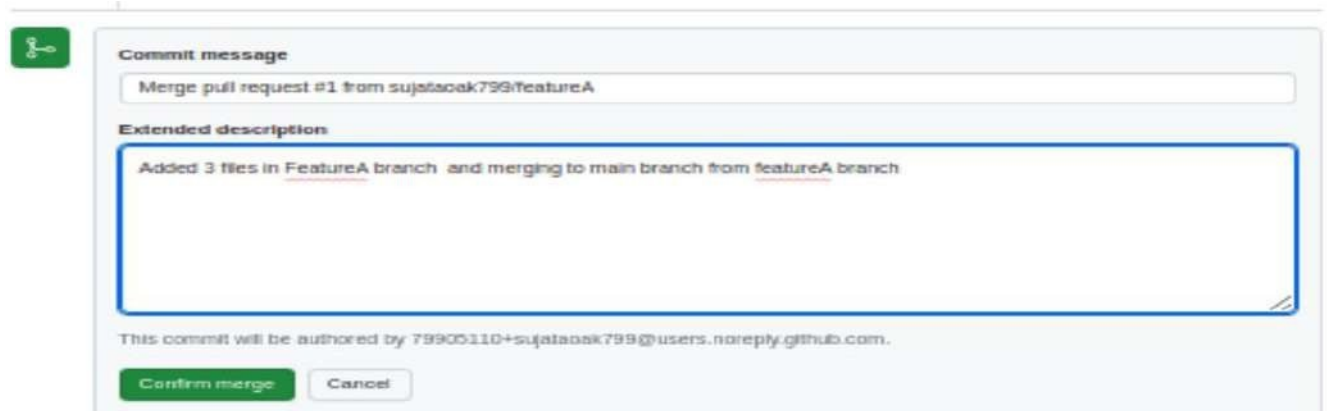




**Click on Create pull request**



**Click on Merge pull request**



**Click on Confirm merge**



PARSHVANATH CHARITABLE TRUST'S  
**A. P. SHAH INSTITUTE OF TECHNOLOGY**  
Department of Information Technology  
(NBA Accredited)



The screenshot shows a GitHub pull request interface. At the top, there are tabs for Conversation (0), Commits (4), Checks (0), and Files changed (4). The main content area shows a pull request by itsSIDHHEE, titled "Merging a branch featurec to main branch". The commit history shows four commits: "Added feedback", "Revert 'Added feedback'", "Added add.py file for commit on 23072025", and "Added 3 files in featurec branch". The pull request is successfully merged and closed, with a message: "Pull request successfully merged and closed. You're all set — the featurec branch can be safely deleted." The right sidebar shows the pull request details, including reviewers, assignees, labels, projects, milestones, and notifications.

**Goto code → main branch → All files of feature-A branch merges with main branch.**

The screenshot shows a GitHub repository page for "gitdemo16072025". The repository is public and has 2 branches and 0 tags. The main branch is selected. The file list shows the following files and their commit history:

File	Commit	Time
README.md	Initial commit	last week
add.py	Added add.py file for commit on 23072025	22 minutes ago
feature-1	Added 3 files in featurec branch	8 minutes ago
feature-2	Added 3 files in featurec branch	8 minutes ago
feature-3	Added 3 files in featurec branch	8 minutes ago
hi.py	first hello.py file	last week
mynewfile	Create mynewfile1	last week

The repository also has a README file and a "Code" button. The right sidebar shows the repository details, including the "About" section, "Releases", "Packages", and "Languages".





**Step 10:** And finally, once it has been merged, I'm going to delete the featureA branch.

Switch to the main branch using the **git checkout main** command and verify the current working branch using the **git branch** command.

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch
* featurec
  main
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 2 commits.
(use "git push" to publish your local commits)
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch
featurec
* main
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

**Step 11: Merge file contents from featureA to main branch in local repository.**

Merge the contents (known as the history) from the feature branch into the main branch using the **git merge**

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# ls
hi.py mynewfile README.md
```

<branch-name> command. The branch-name is the branch that histories are pulled from into the current branch. The output displays that one file was changed with one line inserted.

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git merge featurec
Updating 6be7f0e..55040ff
Fast-forward
 add.py      | 5 +++++
 feature-1  | 0
 feature-2  | 0
 feature-3  | 0
4 files changed, 5 insertions(+)
create mode 100644 add.py
create mode 100644 feature-1
create mode 100644 feature-2
create mode 100644 feature-3
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```



PARSHVANATH CHARITABLE TRUST'S

**A. P. SHAH INSTITUTE OF TECHNOLOGY**

**Department of Information Technology**

**(NBA Accredited)**



```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# ls
add.py feature-1 feature-2 feature-3 hi.py mynewfile README.md
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

### Step 12: Deleting a branch

A] Verify the **feature** branch is still available using the **git branch** command.

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch
featurec
* main
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

B] Delete the **featureA** branch using the **git branch -d <branch-name>** command

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch -d featurec
Deleted branch featurec (was 55040ff).
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

C] Verify the feature branch is no longer available using the **git branch** command

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git branch
* main
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

D] Note the branch featureA is not deleted from remote repository/github

For this we need to :

#git push --delete origin featureA

```
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025# git push --delete origin featurec
To https://github.com/itsSIDDHEE/gitdemo16072025
- [deleted]          featurec
root@apsit-HP-ProDesk-600-G4-PCI-MT:/home/apsit/Desktop/git-SiddhiT/gitdemo16072025#
```

Refresh your github page



PARSHVANATH CHARITABLE TRUST'S  
**A. P. SHAH INSTITUTE OF TECHNOLOGY**  
Department of Information Technology  
(NBA Accredited)



The screenshot displays a GitHub repository interface for 'gitdemo16072025'. The repository is public and has 1 branch and 0 tags. The commit history shows 8 commits, with the most recent being '9010e92' from 9 minutes ago. The commit list includes 'Initial commit', 'Added add.py file for commit on 23072025', and 'Added 3 files in featurec branch'. The README file is titled 'gitdemo16072025' and contains the text 'setup git demo on remote repository'. The sidebar on the right shows repository statistics: 0 stars, 0 watching, and 0 forks. It also includes sections for Releases, Packages, and Languages, with Python being the primary language at 100.0%.

### Conclusion:

In this experiment, we understood the use case of branching and merging. Also, different commands were used for the same such as checkout, branch and merge for displaying the changes between the initial and latter commits.