Placement Empowerment Program

Cloud Computing and DevOps Centre

Set Up Git Branching: Create a new branch in your Git repository for testing .
Add a new feature and merge it

Name: Mohana Priya M Department: CSE

Introduction:

In this Proof of Concept (POC), Git is utilized for version control to streamline the development process. It enables developers to work on new features in separate branches, keeping them independent from the main branch. Once the features are complete, they can be merged back, promoting structured and collaborative development

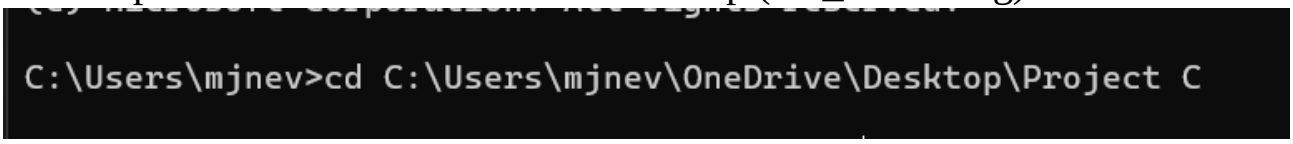
Step-by-Step Overview

Step 1:

Create a folder and name it (poc-5).

Step 2:

Set the path to the folder created in first step (Git_Branching).



Step 3:

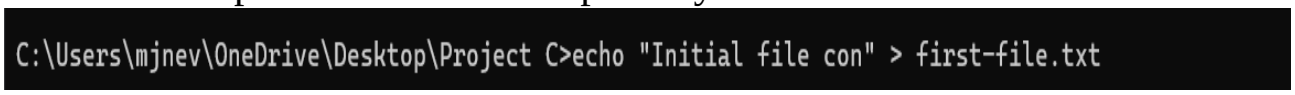
Initialize Git by typing this command:

git init

This command will create a .git folder inside your folder, which tells Git to start tracking your files.

Step 4:

Create a simple file to start the repository:



Step 5:

Add the File to Git

Tell Git to track this file:

Step 6:

Save this change in Git with a commit message.

```
C:\Users\mjnev\OneDrive\Desktop\Project C>git commit -m "Initial commit"
[master (root-commit) 523cfac] Initial commit
 1 file changed, 1 insertion(+)
 create mode 100644 first-file.txt
```

Step 7:

Create and switch to a new branch called testing-feature.

```
C:\Users\mjnev\OneDrive\Desktop\Project C>git checkout -b testing-feature
Switched to a new branch 'testing-feature'
```

Step 8:

Let’s add a new file for our feature:

```
C:\Users\mjnev\OneDrive\Desktop\Project C>echo "Initial file content" > first-file.txt
```

Step 9:

Now, stage the changes:

```
C:\Users\mjnev\OneDrive\Desktop\Project C>git add .
```

Step 10:

Commit the changes:

```
C:\Users\mjnev\OneDrive\Desktop\Project C>git commit -m "Add new Feature file"
[testing-feature ce1d460] Add new Feature file
 1 file changed, 1 insertion(+), 1 deletion(-)
```

Step 11:

Switch to the master Branch

```
C:\Users\mjnev\OneDrive\Desktop\Project C>git checkout master
Switched to branch 'master'
```

Step 12:

Merge Changes from testing-feature to master

```
C:\Users\mjnev\OneDrive\Desktop\Project C>git merge testing-feature
Updating 523cfac..ce1d460
Fast-forward
 first-file.txt | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```

Step 13:

Once the merge is done, you can delete the testing-feature branch:

```
C:\Users\mjnev\OneDrive\Desktop\Project C>git branch -d testing-feature
Deleted branch testing-feature (was ce1d460).
```

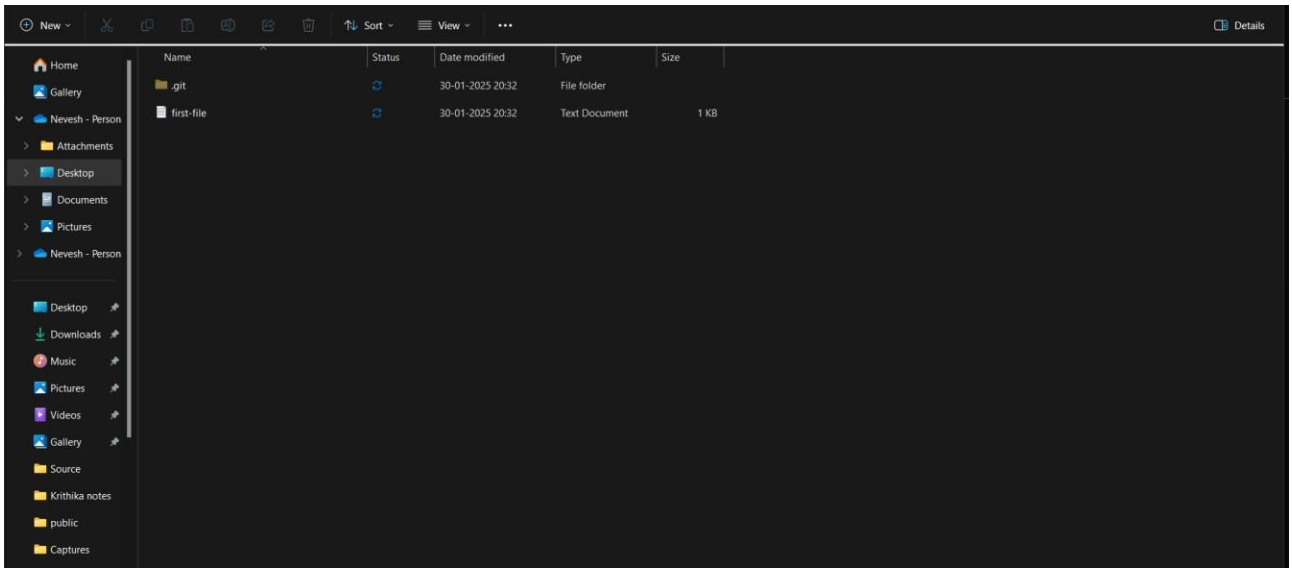
Step 14:

Now, check the files in the folder:

```
C:\Users\mjnev\OneDrive\Desktop\Project C>dir
 Volume in drive C is OS
 Volume Serial Number is 7659-4652

 Directory of C:\Users\mjnev\OneDrive\Desktop\Project C

30-01-2025  20:32    <DIR>          .
30-01-2025  20:23    <DIR>          ..
30-01-2025  20:32                25 first-file.txt
               1 File(s)             25 bytes
               2 Dir(s)  35,931,967,488 bytes free
```



Outcomes:

By completing this PoC of managing branches in Git for a local repository, you will:

1. Successfully initialize a Git repository in your local project folder.

2. Create and manage multiple branches for feature development and experimentation.

3. Track and commit changes made to files in different branches.

4. Merge feature branches back into the main branch while maintaining project integrity.

5. Gain hands-on experience with key Git commands such as git init, git add, git commit, git checkout, and git merge.