PURBANCHAL UNIVERSITY

KHWOPA ENGINEERING COLLEGE

LIBALI-08, BHAKTAPUR

LAB REPORT ON .NET

LAB NO. 01

SUBMITTED BY:

Name: Nabraj Joshi

Roll No. : 770320

Group: A

SUBMITTED TO:

Department of Computer Engineering

Submission: 2081/12/09

Theory:

1. **Git:**
   Git is a distributed version control system used for tracking the changes in the source code during software development. It allows multiple developers to collaborate efficiently by managing different version of project. Git enables branching, merging and reverting changes, making code management easier. It is widely used open-source and commercial projects. Popular platform like GitHub, GitLab, and Bitbucket provide remote repositories for Git-based collaboration.

2. **GitHub**
   GitHub is a web-based platform for version control and collaboration using Git. It allows developers to store, manage, and share code repositories efficiently. GitHub supports features like branching, pull requests, issue tracking, and CI/CD integration. It is widely used for open-source and private projects, enabling seamless teamwork. GitHub also provides cloud-based hosting, making it accessible from anywhere.

General Git and GitHub Commands:

Git Configuration

```
git config --global user.name “Your Name”
```

This command sets the global username for the Git commits.

```
git config --global user.email “your_email@example.com”
```

This command sets the global email associated with Git commits.

Initializing

```
git init
```

initializes a new Git repository in the current directory.

Staging and Commits

```
git add .
```

It stages all changes and new files for commit.

```
git commit -m “Your commit message”
```

Saves the staged changes with a descriptive message.

Branching and Merging

```
git branch
```

Lists all the branches in the repository.

```
git branch <branch_name>
```

Creates a new branch for separate development.

```
git checkout <branch_name>  /  Git switch <branch_name>
```

Switches to the specified branch

git merge <branch_name>
Merges changes from the specified branch into the current branch.

Pushing and Pulling

git push -u origin <branch_name>
Uploads the local changes to the remote repository.
git pull origin <branch_name>
Fetches and merge the latest changes from the remote repository.

Status and Logs

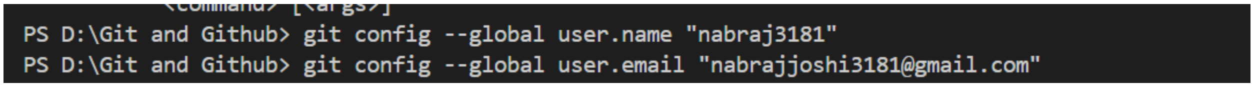
git status
Show the current state of the files in the working directory (modified, staged or untracked).
git log
Displays the commit history of the repository.

GitHub Specific

git remote add origin <repo_url>
Links the local repository to a remote repository on GitHub.

Lab Works

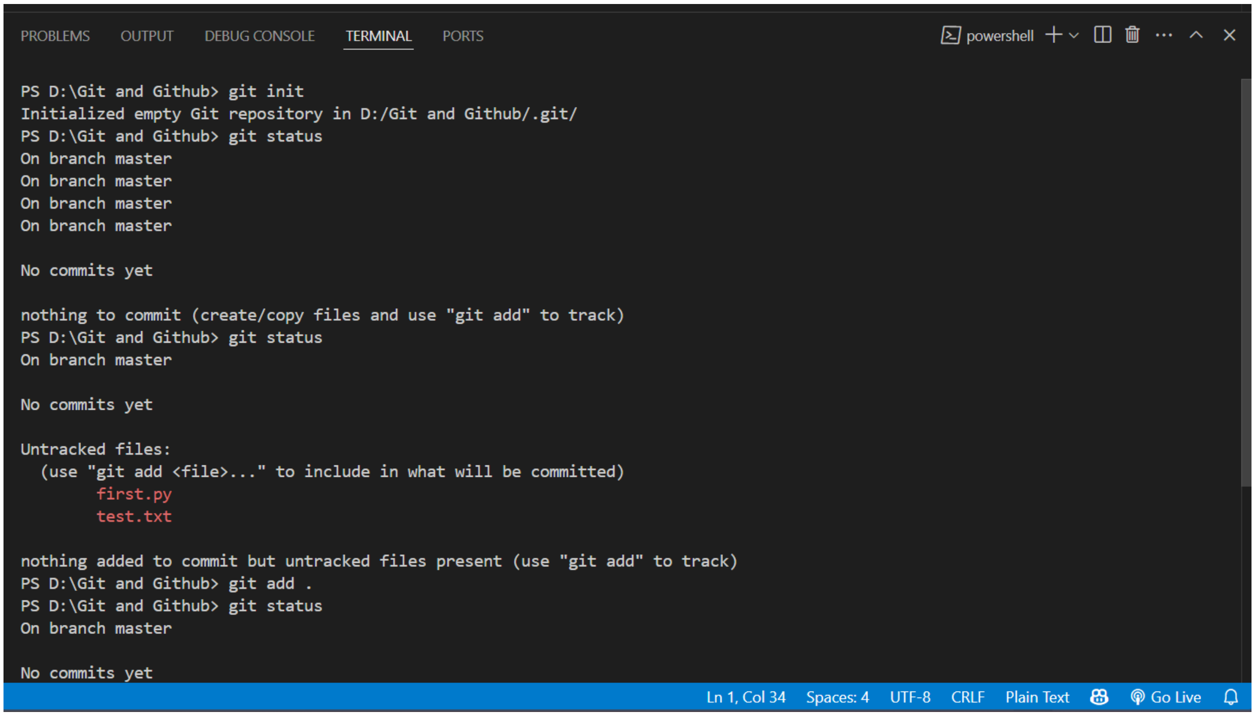
Let us first set the global username and email of the GitHub.



```
           <command> [<args>]
PS D:\Git and Github> git config --global user.name "nabraj3181"
PS D:\Git and Github> git config --global user.email "nabrajjoshi3181@gmail.com"
```

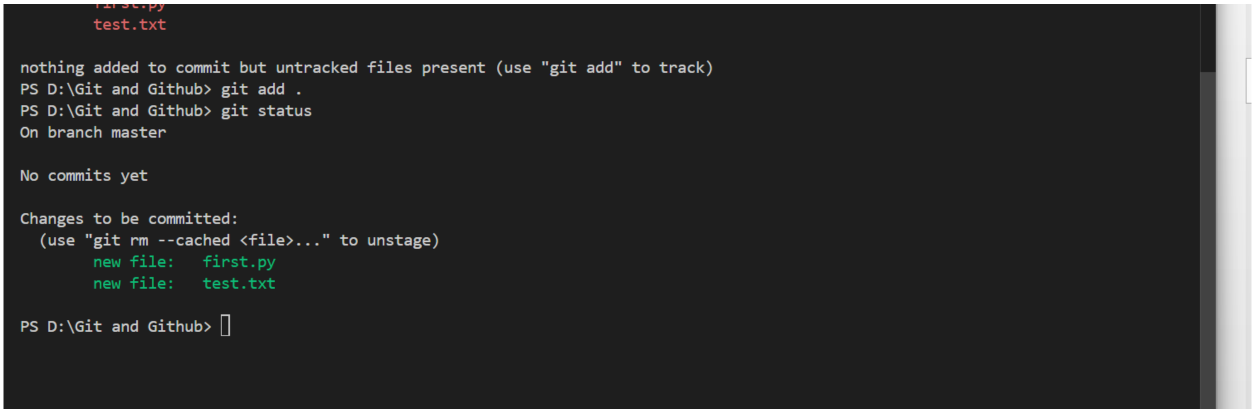
Now let us create a folder and inside it files as per the user desire so that we can identify the changes inside the file using the version control (Git).

On creating the new files, initially the files are in the untracked stage so sent the untracked files to the staging stage. To do so first initialize the directory and staged the files.

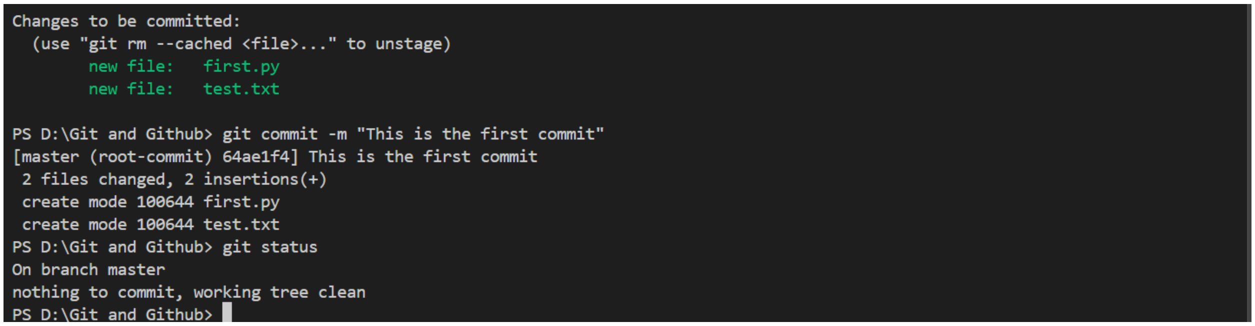


```
PS D:\Git and Github> git init
Initialized empty Git repository in D:/Git and Github/.git/
PS D:\Git and Github> git status
On branch master
On branch master
On branch master
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
PS D:\Git and Github> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        first.py
        test.txt

nothing added to commit but untracked files present (use "git add" to track)
PS D:\Git and Github> git add .
PS D:\Git and Github> git status
On branch master

No commits yet
```

Now commit the files such that the files are stored in the local repository.

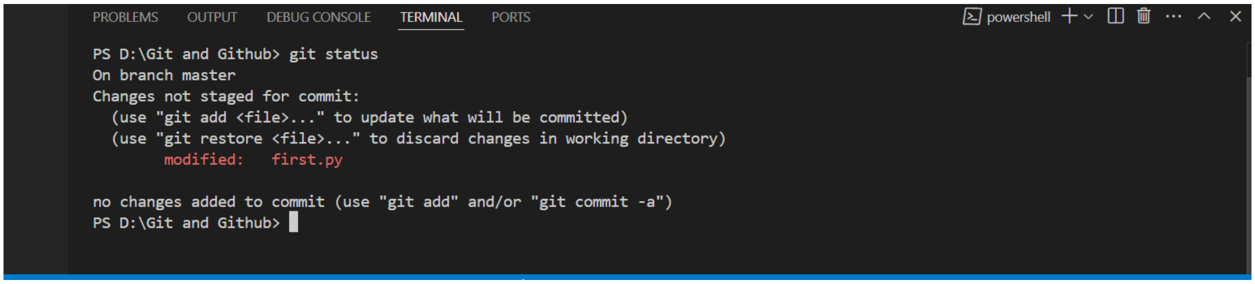


```
        first.py
        test.txt

nothing added to commit but untracked files present (use "git add" to track)
PS D:\Git and Github> git add .
PS D:\Git and Github> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   first.py
        new file:   test.txt

PS D:\Git and Github> 
```



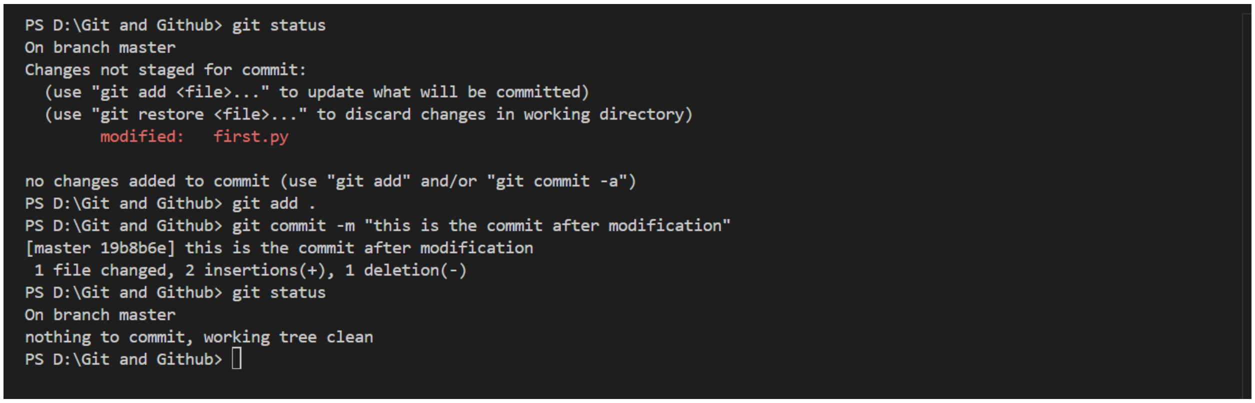
```
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   first.py
        new file:   test.txt

PS D:\Git and Github> git commit -m "This is the first commit"
[master (root-commit) 64ae1f4] This is the first commit
 2 files changed, 2 insertions(+)
 create mode 100644 first.py
 create mode 100644 test.txt
PS D:\Git and Github> git status
On branch master
nothing to commit, working tree clean
PS D:\Git and Github> 
```

Make certain changes inside the file to see the changes in the file status.



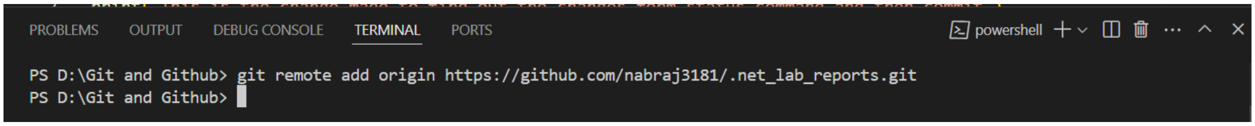
```
PS D:\Git and Github> git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   first.py

no changes added to commit (use "git add" and/or "git commit -a")
PS D:\Git and Github>
```

After changing the contents in the file **“first.py”** add the file and commit it.



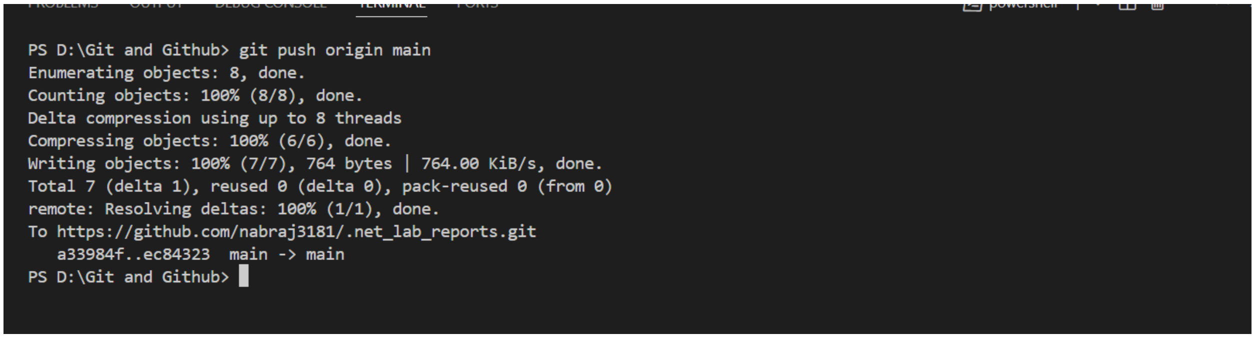
```
PS D:\Git and Github> git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   first.py

no changes added to commit (use "git add" and/or "git commit -a")
PS D:\Git and Github> git add .
PS D:\Git and Github> git commit -m "this is the commit after modification"
[master 19b8b6e] this is the commit after modification
 1 file changed, 2 insertions(+), 1 deletion(-)
PS D:\Git and Github> git status
On branch master
nothing to commit, working tree clean
PS D:\Git and Github>
```

All of these files are saved in the local repository. Now to add these files in the remote repository create the repository in the GitHub and copy the url of the repo and use the following code.



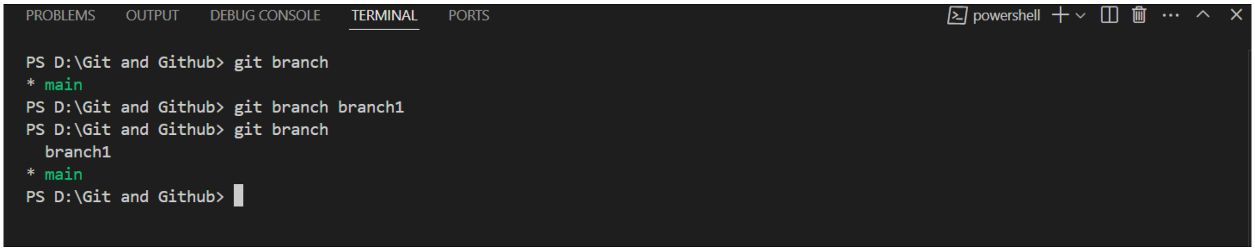
```
PS D:\Git and Github> git remote add origin https://github.com/nabraj3181/.net_lab_reports.git
PS D:\Git and Github>
```

Now push the files in the repository created.



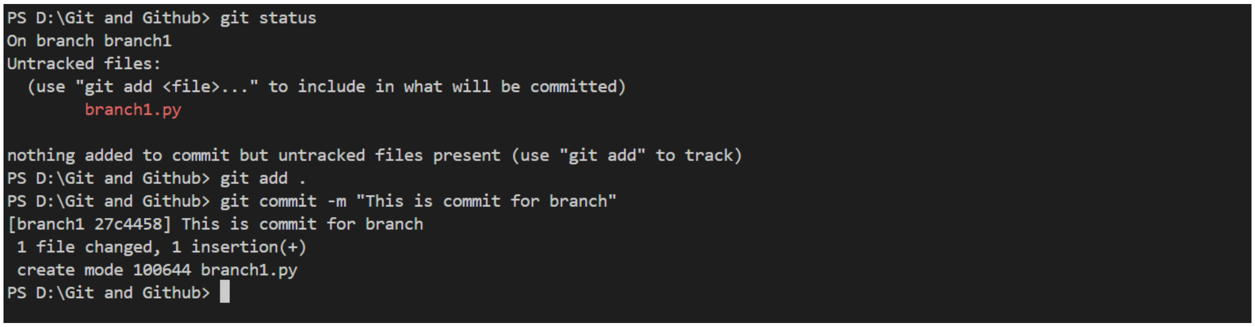
```
PS D:\Git and Github> git push origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 764 bytes | 764.00 KiB/s, done.
Total 7 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/nabraj3181/.net_lab_reports.git
   a33984f..ec84323  main -> main
PS D:\Git and Github>
```

Now creating branches, allowing the work on different version of a project without affecting the main codebase.



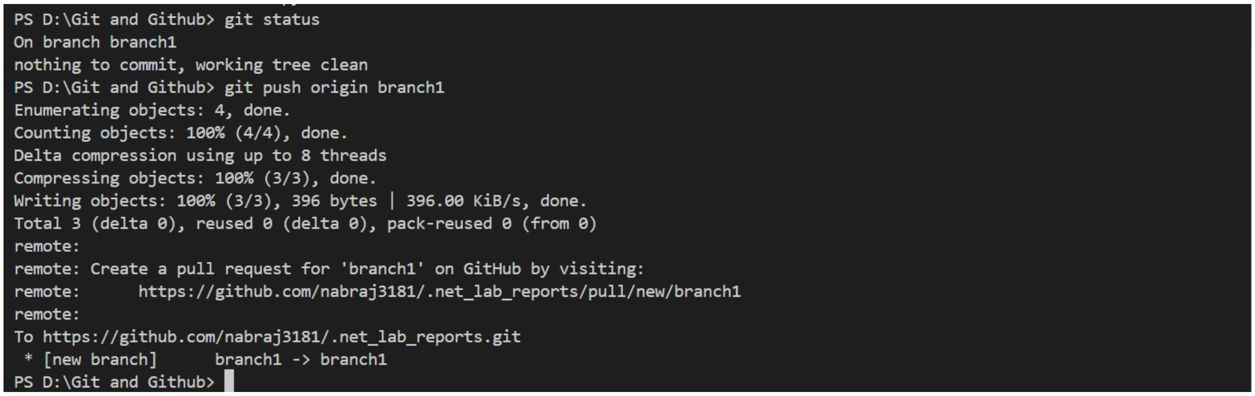
```
PS D:\Git and Github> git branch
* main
PS D:\Git and Github> git branch branch1
PS D:\Git and Github> git branch
  branch1
* main
PS D:\Git and Github>
```

Moving on to the recently created branch to modify the contents in the file without affecting the main codebase.



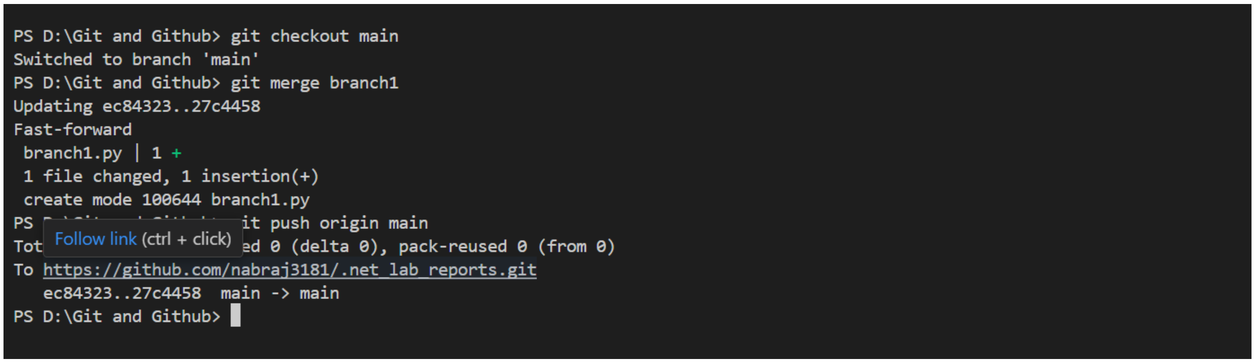
```
PS D:\Git and Github> git status
On branch branch1
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        branch1.py

nothing added to commit but untracked files present (use "git add" to track)
PS D:\Git and Github> git add .
PS D:\Git and Github> git commit -m "This is commit for branch"
[branch1 27c4458] This is commit for branch
 1 file changed, 1 insertion(+)
 create mode 100644 branch1.py
PS D:\Git and Github> 
```

To change the branch, we can use the command *"git switch main"*. To make sure the branch is visible to other users of the repository push the branch in the GitHub.



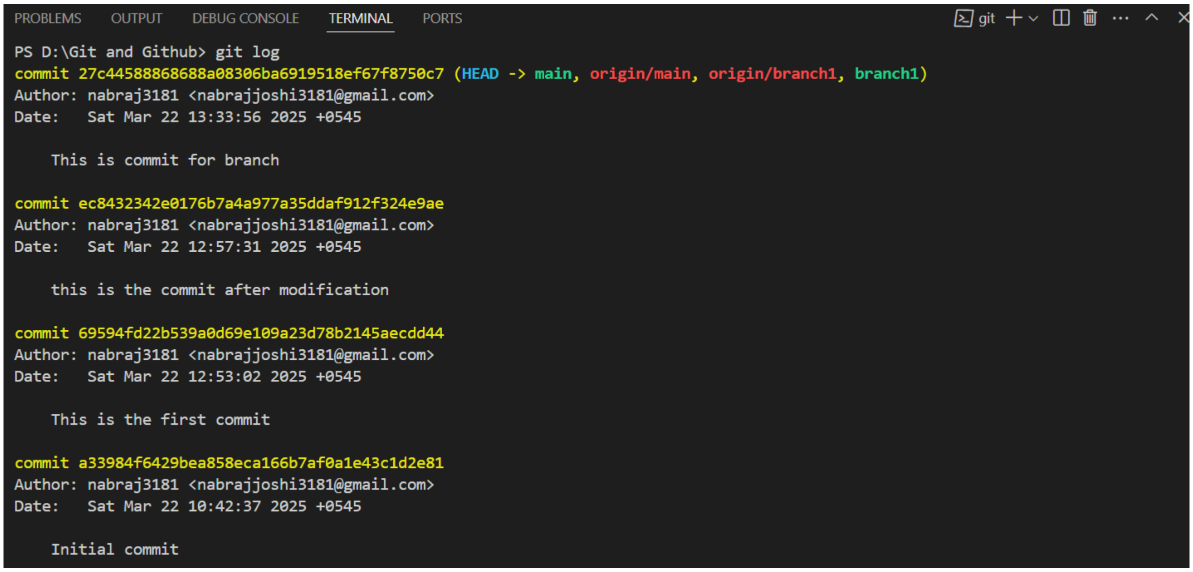
```
PS D:\Git and Github> git status
On branch branch1
nothing to commit, working tree clean
PS D:\Git and Github> git push origin branch1
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 396 bytes | 396.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'branch1' on GitHub by visiting:
remote:      https://github.com/nabraj3181/.net_lab_reports/pull/new/branch1
remote:
To https://github.com/nabraj3181/.net_lab_reports.git
 * [new branch]      branch1 -> branch1
PS D:\Git and Github> 
```

Merging the branches such that the changes in the new branch or new features added in the new branch is added to the main code base.



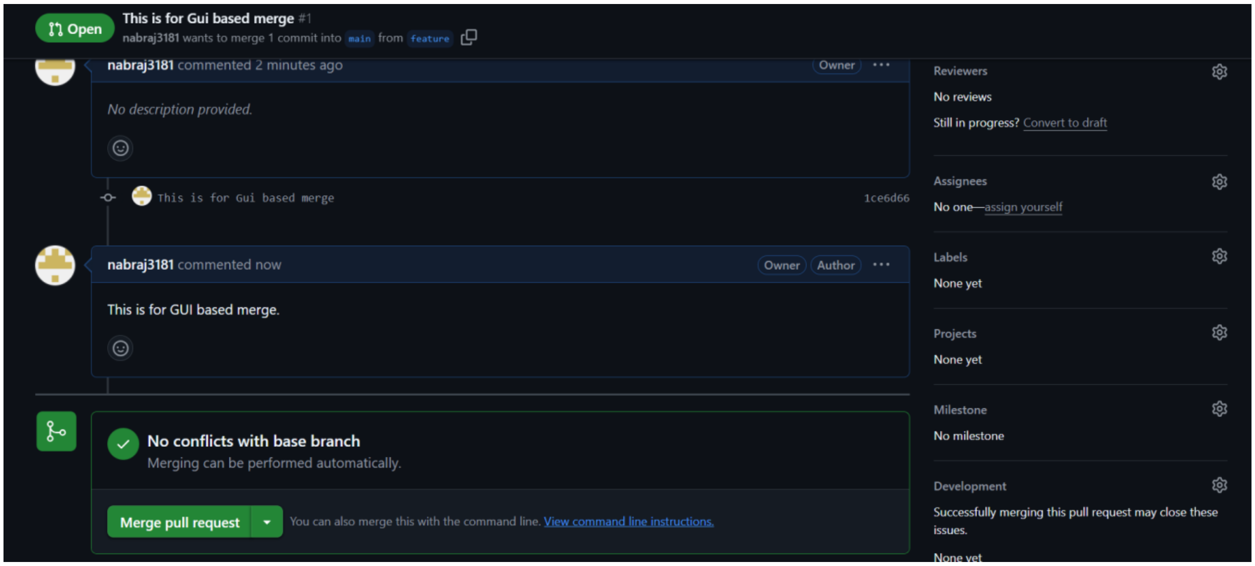
```
PS D:\Git and Github> git checkout main
Switched to branch 'main'
PS D:\Git and Github> git merge branch1
Updating ec84323..27c4458
Fast-forward
 branch1.py | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 branch1.py
PS                       it push origin main
Tot                      ed 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/nabraj3181/.net_lab_reports.git
   ec84323..27c4458  main -> main
PS D:\Git and Github> 
```

To check the commits performed in the past



```
PS D:\Git and Github> git log
commit 27c44588868688a08306ba6919518ef67f8750c7 (HEAD -> main, origin/main, origin/branch1, branch1)
Author: nabraj3181 <nabrajjoshi3181@gmail.com>
Date:   Sat Mar 22 13:33:56 2025 +0545

    This is commit for branch

commit ec8432342e0176b7a4a977a35ddaf912f324e9ae
Author: nabraj3181 <nabrajjoshi3181@gmail.com>
Date:   Sat Mar 22 12:57:31 2025 +0545

    this is the commit after modification

commit 69594fd22b539a0d69e109a23d78b2145aecdd44
Author: nabraj3181 <nabrajjoshi3181@gmail.com>
Date:   Sat Mar 22 12:53:02 2025 +0545

    This is the first commit

commit a33984f6429bea858eca166b7af0a1e43c1d2e81
Author: nabraj3181 <nabrajjoshi3181@gmail.com>
Date:   Sat Mar 22 10:42:37 2025 +0545

    Initial commit
```

Merging the branch in the GUI GitHub (Web)



This is for Gui based merge #1

nabraj3181 wants to merge 1 commit into main from feature

nabraj3181 commented 2 minutes ago

No description provided.

This is for Gui based merge 1ce6d66

nabraj3181 commented now

This is for GUI based merge.

No conflicts with base branch

Merging can be performed automatically.

You can also merge this with the command line. View command line instructions.

No reviews

Still in progress? Convert to draft

No one—assign yourself

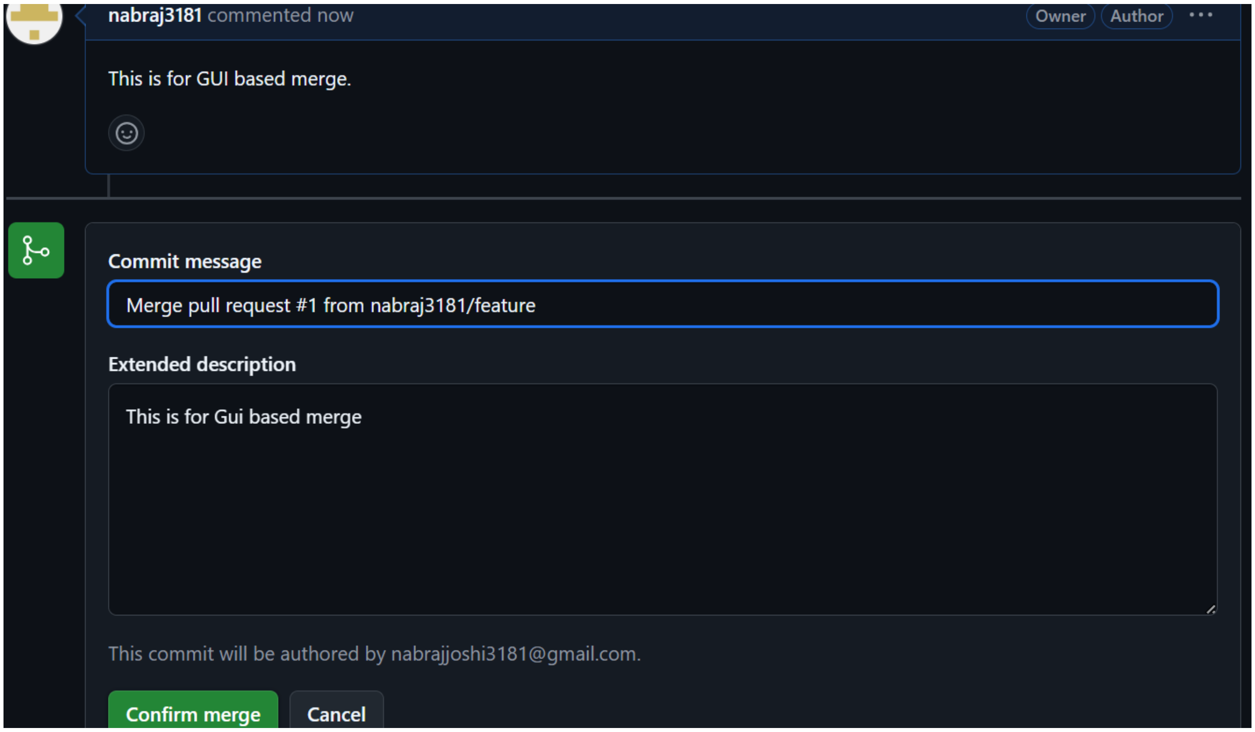
None yet

None yet

No milestone

Successfully merging this pull request may close these issues.

None yet



This is for GUI based merge.

Commit message

Merge pull request #1 from nabraj3181/feature

Extended description

This is for Gui based merge

This commit will be authored by nabrajjoshi3181@gmail.com.

Conclusion:

In this lab, we learn about the basics of the Git and GitHub. We perform initialization, branching, merging, pushing and commit.