Lesson 04 Demo 06

Merging Branches in Git

Objective: To merge branches in Git for maintaining a coherent and functional codebase

Tools required: Git

Prerequisites: None

Steps to be followed:

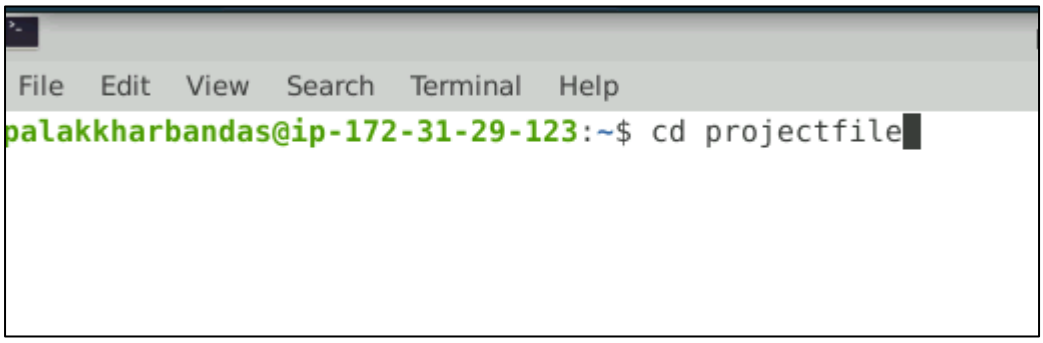
1. Create a new branch
2. Create a new file in the new branch
3. Switch back to the main branch and merge the branches
4. Push the changes to the remote repository

Note: A **projectfile** is already created in Lesson 4 Demo 5. If you do not have the repository, first perform the demo first, then execute the steps from this demo.

Step 1: Create a new branch

1.1 Open the terminal in your lab and navigate to the sample-repository folder by using the following command:

cd projectfile



1.2 Execute the following command on the terminal to create a new branch and check the existing branch:

git branch sample_branch

git branch

```
[redacted]@ip-172-31-71-23:~/projectfile$ git branch sample_branch
[redacted]@ip-172-31-71-23:~/projectfile$  git branch
* new_branch1
  sample_branch
  text_branch
[redacted]@ip-172-31-71-23:~/projectfile$
```

Step 2: Create a new file in the new branch

2.1 Use the following command to switch to the sample_branch:

git checkout sample_branch

```
[redacted]@ip-172-31-71-23:~/projectfile$ git checkout sample_branch
Switched to branch 'sample_branch'
```

2.2 Use the following commands to create an empty HTML file **index1.html** and add it to the test_branch:

touch index1.html

git add index1.html

git commit -a -m "index1.html added to the sample_branch"

```
[redacted]@ip-172-31-71-23:~/projectfile$ git checkout sample_branch
Switched to branch 'sample_branch'
[redacted]@ip-172-31-71-23:~/projectfile$ touch index1.html
[redacted]@ip-172-31-71-23:~/projectfile$ git add index1.html
[redacted]@ip-172-31-71-23:~/projectfile$ git commit -a -m "index1.html added to the sample_branch"
[sample_branch 67ded56] index1.html added to the sample_branch
 Committer: manikumarsimpli <manikumarsimpli@ip-172-31-71-23.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index1.html
[redacted]@ip-172-31-71-23:~/projectfile$
```

Step 3: Switch back to the main branch and merge the branches

3.1 Use the following command to switch back to the main branch:

git checkout new_branch1

```
[redacted]@ip-172-31-71-23:~/projectfile$ git checkout new_branch1
Switched to branch 'new_branch1'
Your branch is up to date with 'origin/main'.
[redacted]@ip-172-31-71-23:~/projectfile$
```

3.2 Use the following command to merge the sample_branch to the main branch:

git merge sample_branch

```
[redacted]@ip-172-31-71-23:~/projectfile$ git merge sample_branch
Updating d77a2b5..67ded56
Fast-forward
 index1.html | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index1.html
[redacted]@ip-172-31-71-23:~/projectfile$
```

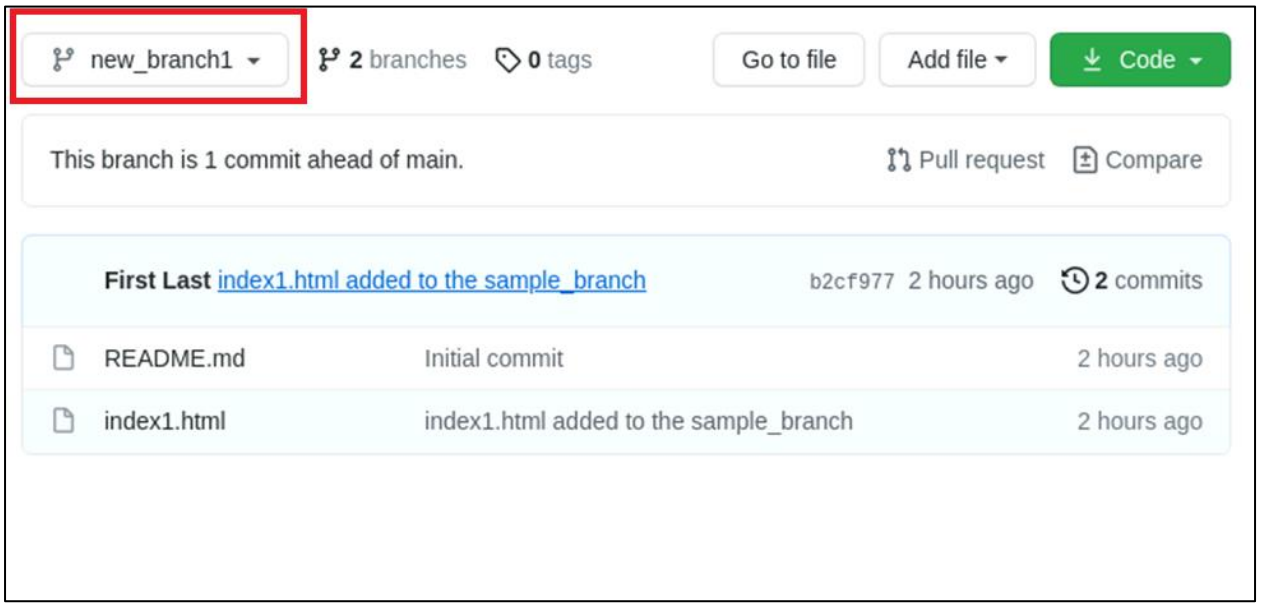
Step 4: Push the changes to the remote repository

4.1 Use the following command to push the new_branch1 to the remote repository:

git push -u origin new_branch1

```
[redacted]@ip-172-31-71-23:~/projectfile$ git push -u origin new_branch1
Username for 'https://github.com': Simplilearn-Edu
Password for 'https://Simplilearn-Edu@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 310 bytes | 310.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'new_branch1' on GitHub by visiting:
remote:      https://github.com/Simplilearn-Edu/projectfile/pull/new/new_branch1
remote:
To https://github.com/Simplilearn-Edu/projectfile.git
 * [new branch]      new_branch1 -> new_branch1
Branch 'new_branch1' set up to track remote branch 'new_branch1' from 'origin'.
[redacted]@ip-172-31-71-23:~/projectfile$
```

4.2 Go to the remote repository to check whether the branch is updated



By following these steps, you have successfully merged branches in Git, combining the changes from one branch into another, facilitating the integration of new features or bug fixes into the main codebase.