Step 1: Clone the repository you have created in GitHub.
Step 2: Create a new branch using the command.

```
PS C:\Users\arjun\Desktop\Git_Practice\git_assignment> git branch branch1
```

Step 3: Switch to the new branch.

```
PS C:\Users\arjun\Desktop\Git_Practice\git_assignment> git switch branch1
Switched to branch 'branch1'
```

Step 4: Make some changes to the code in your local copy of the repository.
Step 5: Commit changes to the new branch.

```
PS C:\Users\arjun\Desktop\Git_Practice\git_assignment> git status
On branch branch1
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        udfs.py

nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\arjun\Desktop\Git_Practice\git_assignment> git add .
PS C:\Users\arjun\Desktop\Git_Practice\git_assignment> git commit -m 'adding udfs.py file'
[branch1 c50077b] adding udfs.py file
 1 file changed, 7 insertions(+)
 create mode 100644 udfs.py
```

Step 6: Switch back to the original branch

```
PS C:\Users\arjun\Desktop\Git_Practice\git_assignment> git switch main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```

Step 7: Merge the new branch

```
PS C:\Users\arjun\Desktop\Git_Practice\git_assignment> git merge branch1
Updating adb0061..c50077b
Fast-forward
 udfs.py | 7 +++++++
 1 file changed, 7 insertions(+)
 create mode 100644 udfs.py
PS C:\Users\arjun\Desktop\Git_Practice\git_assignment>
```

Step 8: Push changes to the original branch

```
PS C:\Users\arjun\Desktop\Git_Practice\git_assignment> git push
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), 364 bytes | 121.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Mallikharjuna71/git_assignment.git
   adb0061..c50077b  main -> main
```