

Aim: Version control using Git.

Objectives: To understand working on github and solving conflicts.

Tools used: Github, Virtual box, ubuntu, linux terminal, Kdiff3

Concept: To understand how git works.

Problem statement:

2.1:

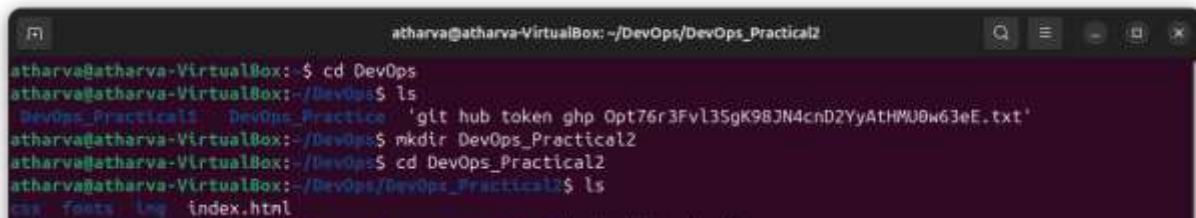
- Create a local git repository and move the entire code base to it
- Create a new branch for a new feature you want to add to the application
- Merge back the created branch with the master branch
- Create a remote repository
- Push the local repository to company's remote repository

2.2:

- Create two separate branches from master
- Make changes in the same function of the source code in both the branches
- Merge branch1 into the master
- Try and merge branch2 into the master (merge conflict should arise)
- Install a merge tool of your choice and resolve the merge conflict using git
- mergetool command.

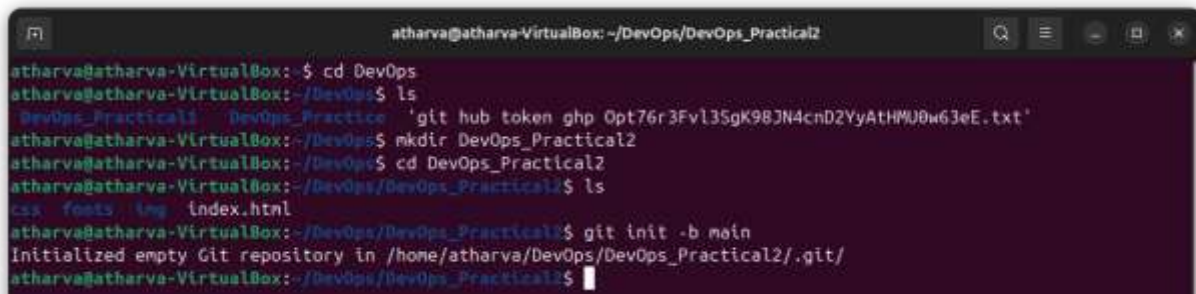
2.1:

1. Create a Sample Project and open the terminal and navigate to the sample project folder



```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2
atharva@atharva-VirtualBox:~$ cd DevOps
atharva@atharva-VirtualBox:~/DevOps$ ls
DevOps_Practical1  DevOps_Practice  'git hub token ghp_Opt76r3Fv13SgK98JN4cnD2YyAtHMU0w63eE.txt'
atharva@atharva-VirtualBox:~/DevOps$ mkdir DevOps_Practical2
atharva@atharva-VirtualBox:~/DevOps$ cd DevOps_Practical2
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ ls
css  fonts  img  index.html
```

2. Initialize this folder as local git repository by git init command



```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2
atharva@atharva-VirtualBox:~$ cd DevOps
atharva@atharva-VirtualBox:~/DevOps$ ls
DevOps_Practical1  DevOps_Practice  'git hub token ghp_Opt76r3Fv13SgK98JN4cnD2YyAtHMU0w63eE.txt'
atharva@atharva-VirtualBox:~/DevOps$ mkdir DevOps_Practical2
atharva@atharva-VirtualBox:~/DevOps$ cd DevOps_Practical2
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ ls
css  fonts  img  index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git init -b main
Initialized empty Git repository in /home/atharva/DevOps/DevOps_Practical2/.git/
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

3. Add all the files to staging area and use status command

```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2$ git add .
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   css/site.css
    new file:   fonts/segoeui.ttf
    new file:   img/cloneWhite.svg
    new file:   img/deployWhite.svg
    new file:   img/lightbulbWhite.svg
    new file:   img/stackWhite.svg
    new file:   img/successCloudNew.svg
    new file:   img/tweetThis.svg
    new file:   index.html

atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

4. Make initial Commit

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git commit -m "initial commit"
[main (root-commit) 564bcd7] initial commit
 9 files changed, 198 insertions(+)
 create mode 100644 css/site.css
 create mode 100644 fonts/segoeui.ttf
 create mode 100644 img/cloneWhite.svg
 create mode 100644 img/deployWhite.svg
 create mode 100644 img/lightbulbWhite.svg
 create mode 100644 img/stackWhite.svg
 create mode 100644 img/successCloudNew.svg
 create mode 100644 img/tweetThis.svg
 create mode 100644 index.html

atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

5. Now create a new branch named branch1 and switch to branch1 by using checkout command

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git branch
* main
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git branch branch1
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git checkout branch1
Switched to branch 'branch1'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

6. Change the title in index.html by using nano command making changes save the changes

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ nano index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

```
<head>
  <title>HTML Application branch1</title>
  <link href="css/site.css" rel="stylesheet">
</head>
<body>
  <div class="main-container">
    <div class="cloud-image">
      
    </div>
    <div class="content">
```

8. Now check the status by using git status

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ nano index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git status
On branch branch1
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:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

9. Now add all the files in the staging area and then commit

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git add .
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git commit -m "changes in branch1"
[branch1 fbf0fef] changes in branch1
1 file changed, 1 insertion(+), 1 deletion(-)
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

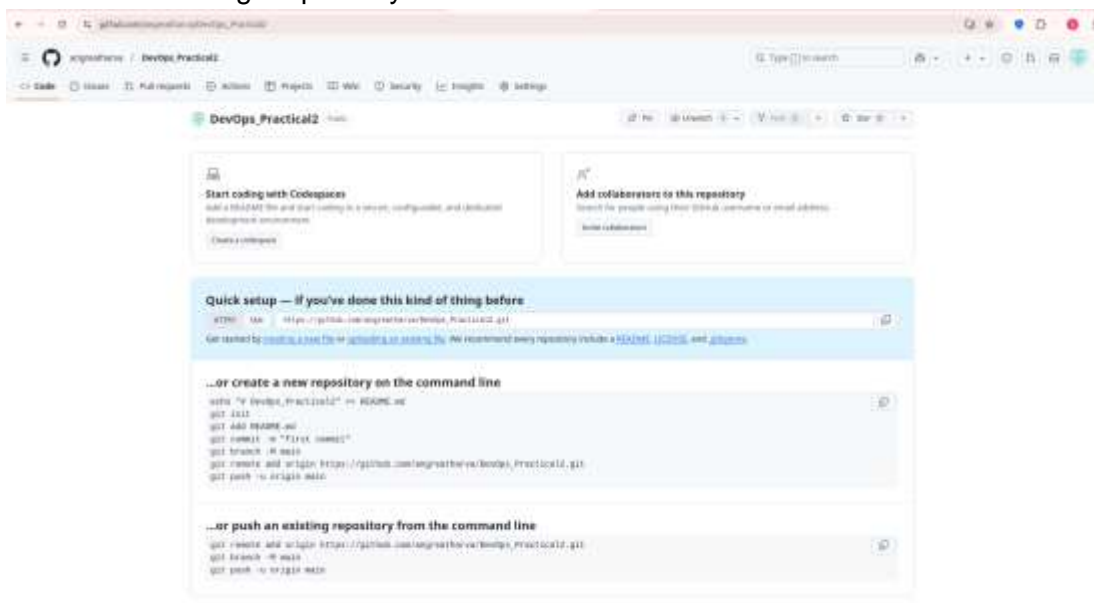
10. Now switch to main branch

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git checkout main
Switched to branch 'main'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

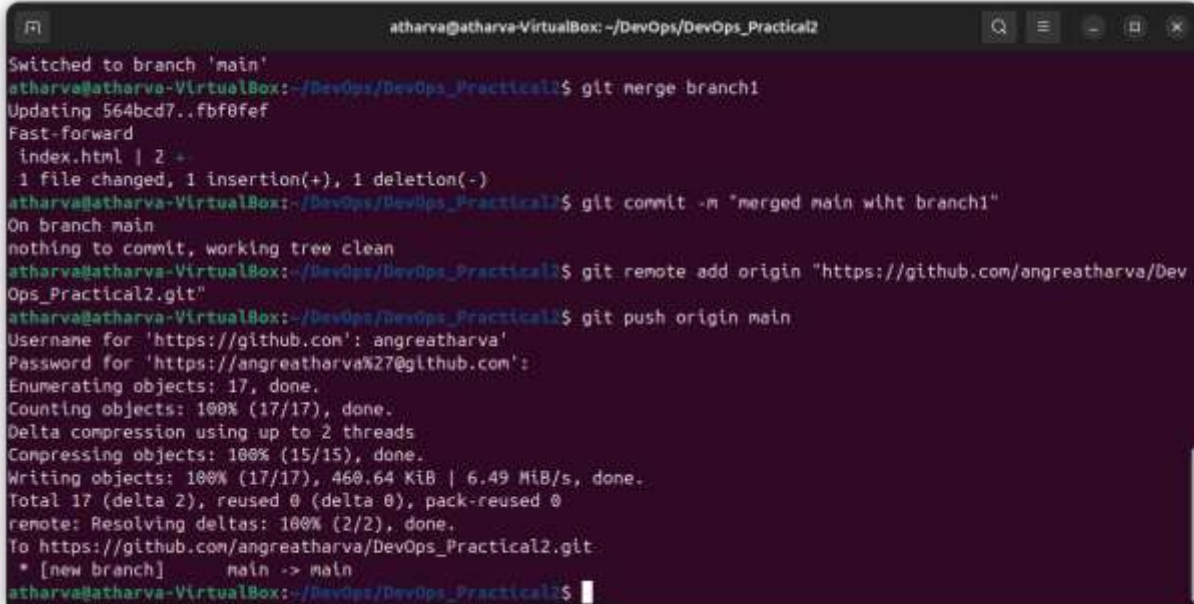
11. Now the merge the changes and make the commit

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git merge branch1
Updating 564bcd7..fbf0fef
Fast-forward
 index.html | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git commit -m "merged main wiht branch1"
On branch main
nothing to commit, working tree clean
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

12. Now create a git repository



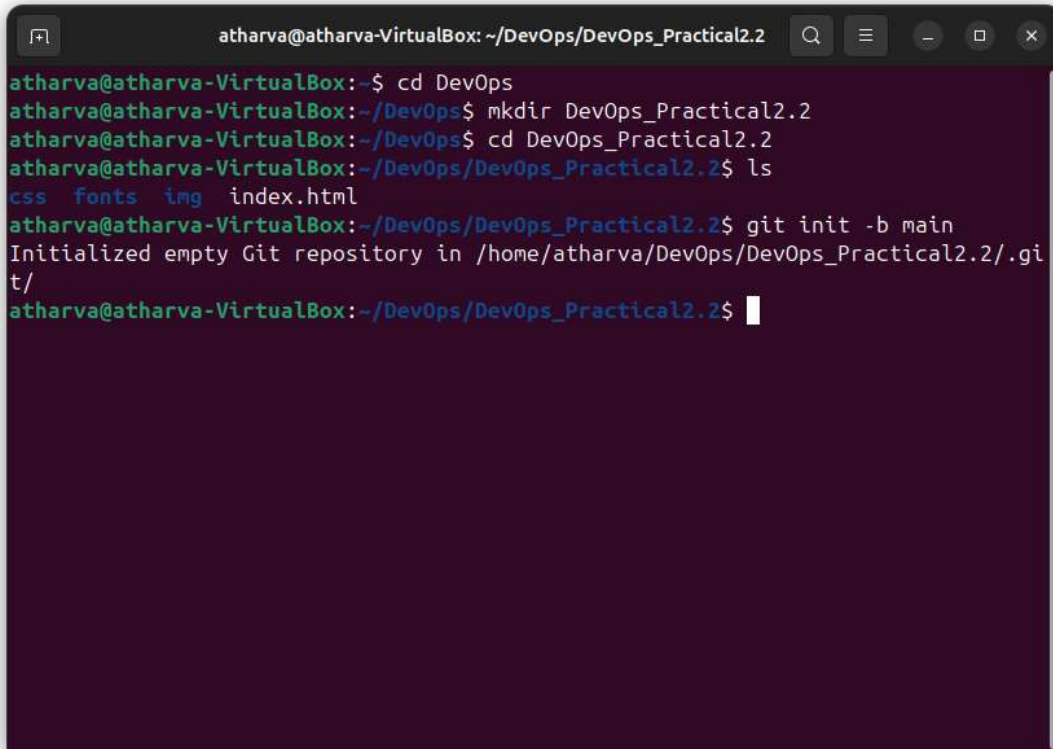
13. Add remote and Push it.

A terminal window titled 'atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2'. The terminal shows the following sequence of commands and output:

```
Switched to branch 'main'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git merge branch1
Updating 564bcd7..fbf8fef
Fast-forward
 index.html | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git commit -m "merged main w/ht branch1"
On branch main
nothing to commit, working tree clean
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git remote add origin "https://github.com/angreatharva/DevOps_Practical2.git"
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$ git push origin main
Username for 'https://github.com': angreatharva
Password for 'https://angreatharva%27@github.com':
Enumerating objects: 17, done.
Counting objects: 100% (17/17), done.
Delta compression using up to 2 threads
Compressing objects: 100% (15/15), done.
Writing objects: 100% (17/17), 460.64 KiB | 6.49 MiB/s, done.
Total 17 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/angreatharva/DevOps_Practical2.git:
 * [new branch]      main -> main
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2$
```

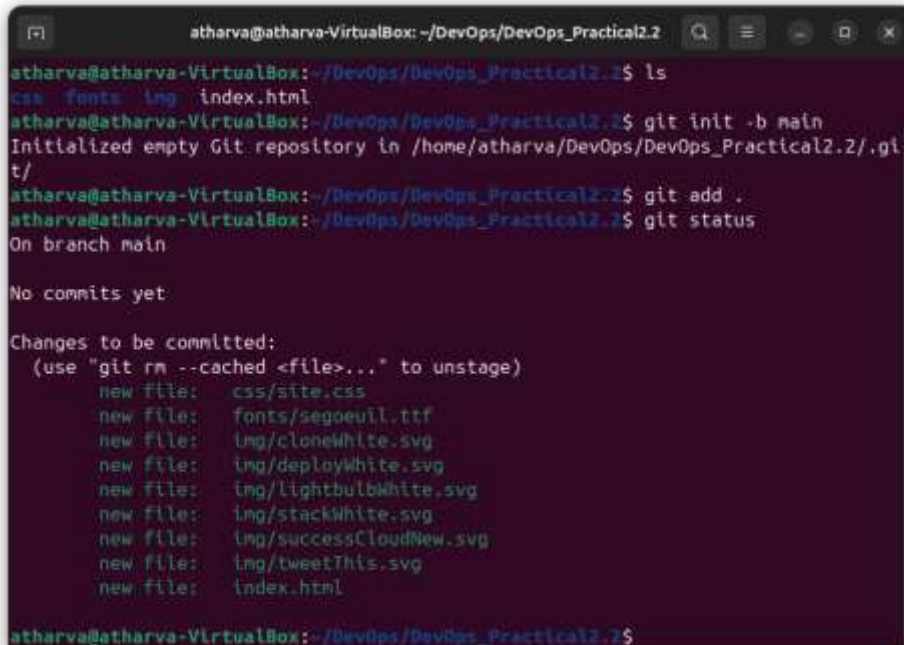
2.2

Step 1: Create a local repo and initialize a git



```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2
atharva@atharva-VirtualBox:~$ cd DevOps
atharva@atharva-VirtualBox:~/DevOps$ mkdir DevOps_Practical2.2
atharva@atharva-VirtualBox:~/DevOps$ cd DevOps_Practical2.2
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ ls
css  fonts  img  index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git init -b main
Initialized empty Git repository in /home/atharva/DevOps/DevOps_Practical2.2/.git/
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$
```

Step 2: Add all the files to staging area and use status command



```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ ls
css  fonts  img  index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git init -b main
Initialized empty Git repository in /home/atharva/DevOps/DevOps_Practical2.2/.git/
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git add .
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   css/site.css
    new file:   fonts/segoeui1.ttf
    new file:   img/cloneWhite.svg
    new file:   img/deployWhite.svg
    new file:   img/lightbulbWhite.svg
    new file:   img/stackWhite.svg
    new file:   img/successCloudNew.svg
    new file:   img/tweetThis.svg
    new file:   index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$
```

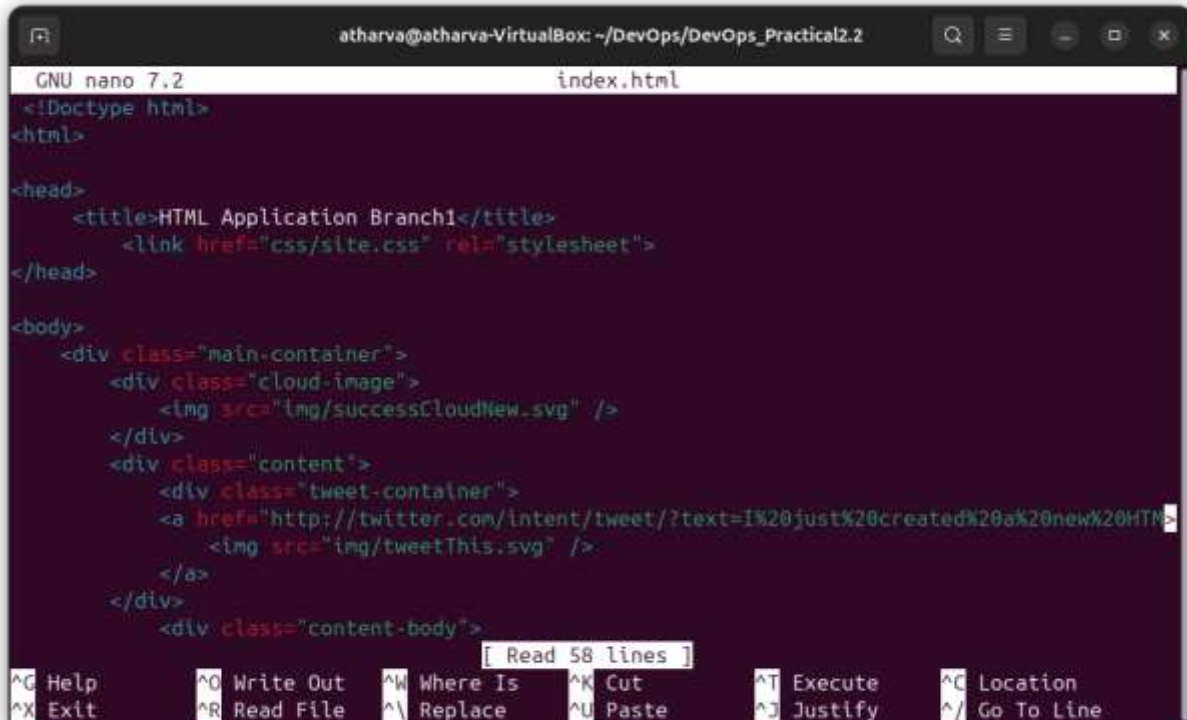
Step 3: Make initial Commit


```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   css/site.css
    new file:   fonts/segoeui.ttf
    new file:   img/cloneWhite.svg
    new file:   img/deployWhite.svg
    new file:   img/lightbulbWhite.svg
    new file:   img/stackWhite.svg
    new file:   img/successCloudNew.svg
    new file:   img/tweetThis.svg
    new file:   index.html
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2$ git commit -m "initial commit"
[main (root-commit) 5393fdd] initial commit
 9 files changed, 198 insertions(+)
 create mode 100644 css/site.css
 create mode 100644 fonts/segoeui.ttf
 create mode 100644 img/cloneWhite.svg
 create mode 100644 img/deployWhite.svg
 create mode 100644 img/lightbulbWhite.svg
 create mode 100644 img/stackWhite.svg
 create mode 100644 img/successCloudNew.svg
 create mode 100644 img/tweetThis.svg
 create mode 100644 index.html
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2$
```

Step 4: create a new branch named branch1 and move to that branch

```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2
    new file:   img/deployWhite.svg
    new file:   img/lightbulbWhite.svg
    new file:   img/stackWhite.svg
    new file:   img/successCloudNew.svg
    new file:   img/tweetThis.svg
    new file:   index.html
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2$ git commit -m "initial commit"
[main (root-commit) 5393fdd] initial commit
 9 files changed, 198 insertions(+)
 create mode 100644 css/site.css
 create mode 100644 fonts/segoeui.ttf
 create mode 100644 img/cloneWhite.svg
 create mode 100644 img/deployWhite.svg
 create mode 100644 img/lightbulbWhite.svg
 create mode 100644 img/stackWhite.svg
 create mode 100644 img/successCloudNew.svg
 create mode 100644 img/tweetThis.svg
 create mode 100644 index.html
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2$ git branch
* main
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2$ git branch branch1
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2$ git checkout branch1
Switched to branch 'branch1'
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2$
```

Step 5: make changes to the index.html file in branch1

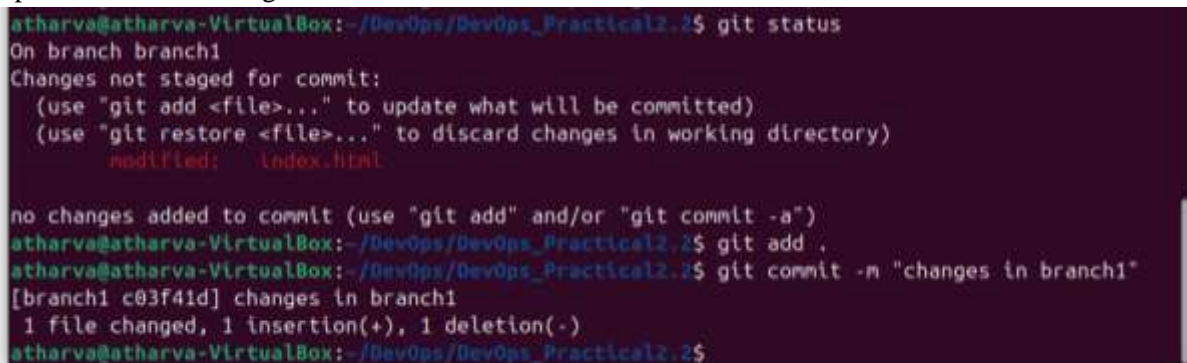


```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2
GNU nano 7.2 index.html
<!doctype html>
<html>

<head>
  <title>HTML Application Branch1</title>
  <link href="css/site.css" rel="stylesheet">
</head>

<body>
  <div class="main-container">
    <div class="cloud-image">
      
    </div>
    <div class="content">
      <div class="tweet-container">
        <a href="http://twitter.com/intent/tweet/?text=I%20just%20created%20a%20new%20HTML">
          
        </a>
      </div>
      <div class="content-body">
[ Read 58 lines ]
^G Help      ^O Write Out  ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File  ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

Step 6: commit the changes in branch1



```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git status
On branch branch1
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:   index.html

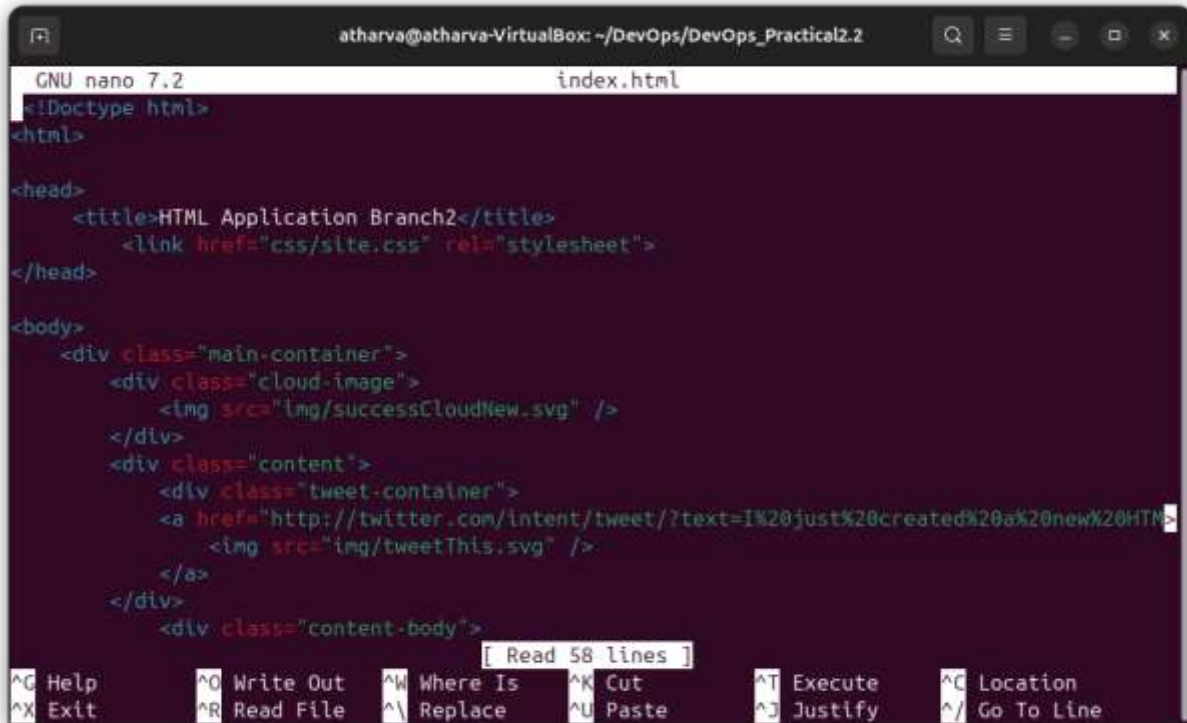
no changes added to commit (use "git add" and/or "git commit -a")
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git add .
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git commit -m "changes in branch1"
[branch1 c03f41d] changes in branch1
1 file changed, 1 insertion(+), 1 deletion(-)
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$
```

Step 7: create a new branch named branch2 and move to that branch



```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git branch
* branch1
  main
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git branch branch2
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git checkout branch2
Switched to branch 'branch2'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git branch
  branch1
* branch2
  main
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$
```

Step 8: make changes to the index.html file in branch2



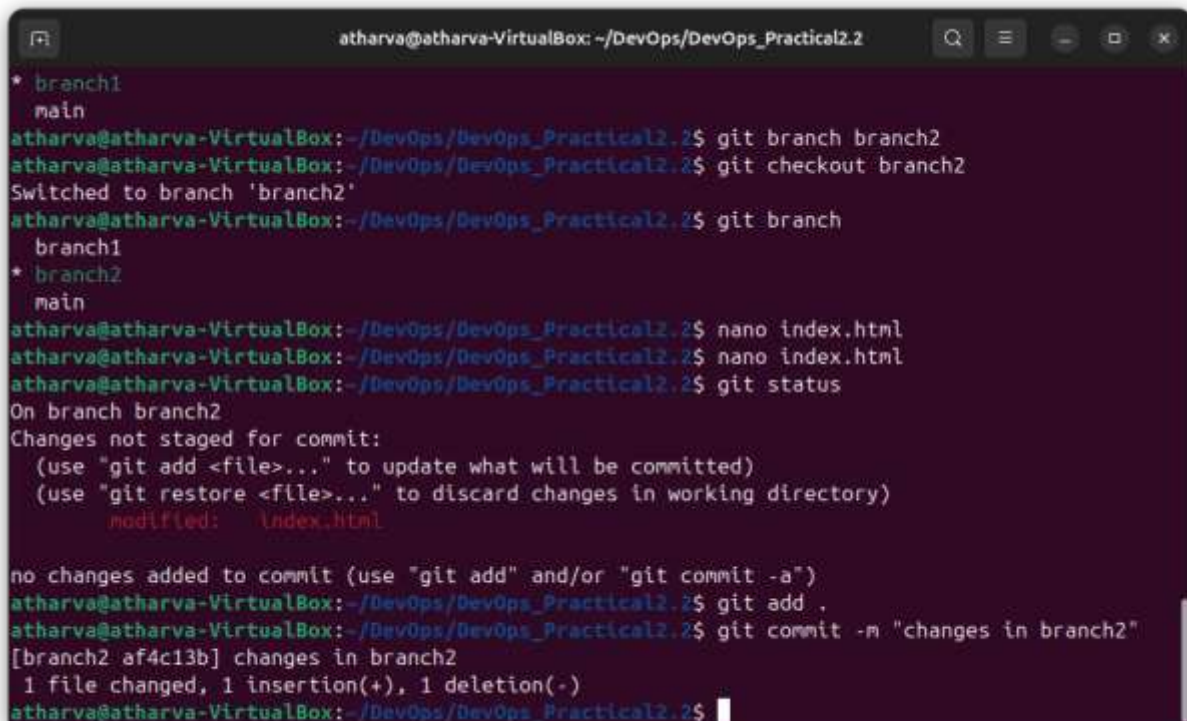
```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2
GNU nano 7.2 index.html
<!doctype html>
<html>

<head>
  <title>HTML Application Branch2</title>
  <link href="css/site.css" rel="stylesheet">
</head>

<body>
  <div class="main-container">
    <div class="cloud-image">
      
    </div>
    <div class="content">
      <div class="tweet-container">
        <a href="http://twitter.com/intent/tweet/?text=I%20just%20created%20a%20new%20HTML">
          
        </a>
      </div>
      <div class="content-body">
        [ Read 58 lines ]
      </div>
    </div>
  </div>
</body>
</html>

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

Step 9: commit the changes in branch2



```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2
* branch1
  main
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git branch branch2
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git checkout branch2
Switched to branch 'branch2'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git branch
  branch1
* branch2
  main
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ nano index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ nano index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git status
On branch branch2
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:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git add .
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git commit -m "changes in branch2"
[branch2 af4c13b] changes in branch2
1 file changed, 1 insertion(+), 1 deletion(-)
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$
```

Step 10:

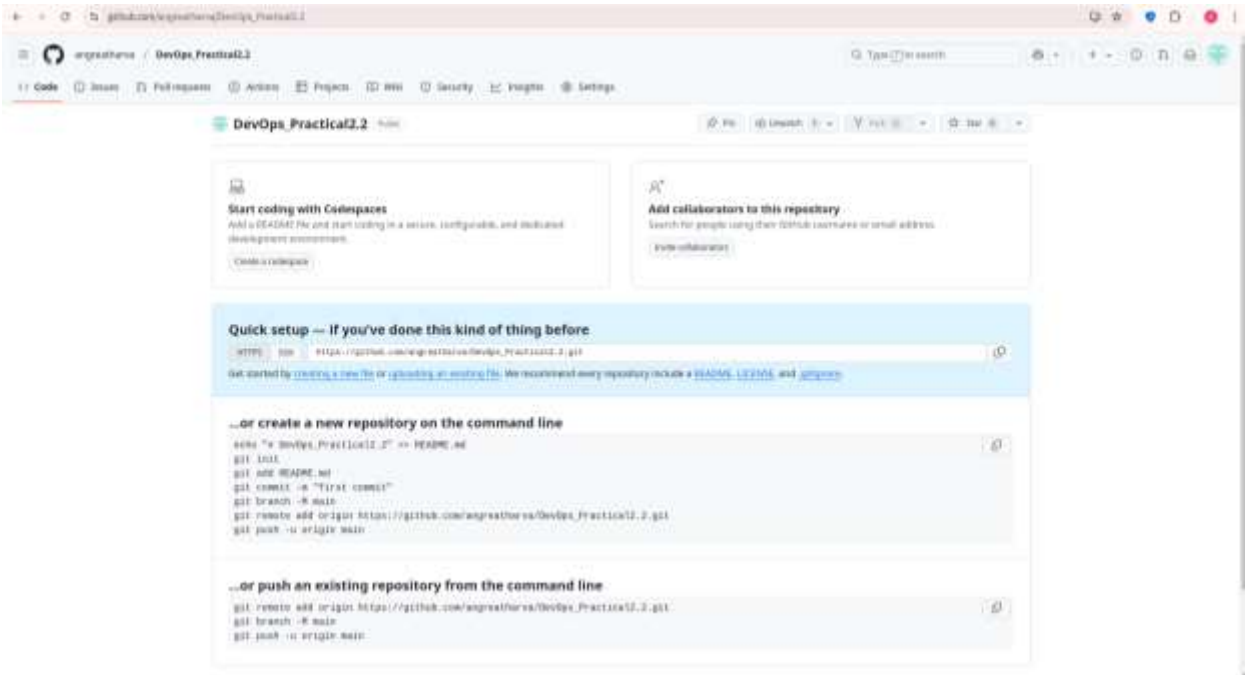
So now we have 3 branches -> main, branch1, branch2 each having different title in the index.html file.

Switch to main branch and add the remote of the git repository

Push main branch

Switch to branch1 and push it

Switch to branch2 and push it



```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git checkout main
Switched to branch 'main'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git remote add origin "https://github
.com/angreatharva/DevOps_Practical2.2.git"
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git push origin main
Username for 'https://github.com': angreatharva
Password for 'https://angreatharva@github.com':
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 2 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (14/14), 460.43 KiB | 12.79 MiB/s, done.
Total 14 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/angreatharva/DevOps_Practical2.2.git
 * [new branch]    main -> main
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$
```

```
atharva@atharva-VirtualBox: ~/DevOps/DevOps_Practical2.2
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git checkout branch1
Switched to branch 'branch1'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git push origin branch1
Username for 'https://github.com': angreatharva
Password for 'https://angreatharva@github.com':
remote: Invalid username or password.
fatal: Authentication failed for 'https://github.com/angreatharva/DevOps_Practical2.2.git/'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git push origin branch1
Username for 'https://github.com': angreatharva
Password for 'https://angreatharva@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 293 bytes | 293.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'branch1' on GitHub by visiting:
remote:   https://github.com/angreatharva/DevOps_Practical2.2/pull/new/branch1
remote:
To https://github.com/angreatharva/DevOps_Practical2.2.git
 * [new branch]      branch1 -> branch1
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$
```

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git checkout branch2
Switched to branch 'branch2'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git push origin branch2
Username for 'https://github.com': angreatharva
Password for 'https://angreatharva@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 284 bytes | 284.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'branch2' on GitHub by visiting:
remote:   https://github.com/angreatharva/DevOps_Practical2.2/pull/new/branch2
remote:
To https://github.com/angreatharva/DevOps_Practical2.2.git
 * [new branch]      branch2 -> branch2
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$
```

Step 11: Now to move to main branch and then merge branch1 into it

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git checkout main
Switched to branch 'main'
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git merge branch1
Updating 5393fdd..c03f41d
Fast-forward
 index.html | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```

Also push it to the main branch

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ nano index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git push origin main
Username for 'https://github.com': angreatharva
Password for 'https://angreatharva@github.com':
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/angreatharva/DevOps_Practical2.2.git
 5393fdd..c03f41d main -> main
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$
```

Step 12: Now try to merge branch1

```
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git merge branch1
Switched to branch 'main'
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git status
On branch main
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

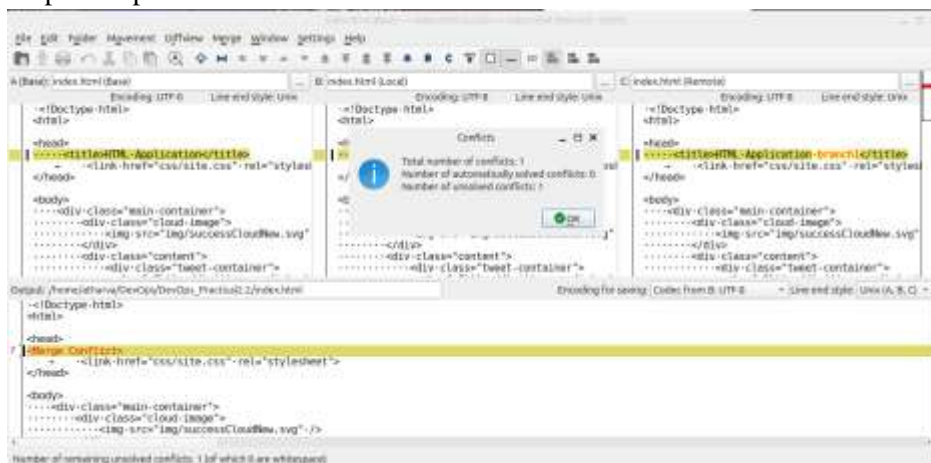
Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ nano index.html
atharva@atharva-VirtualBox:~/DevOps/DevOps_Practical2.2$ git mergetool

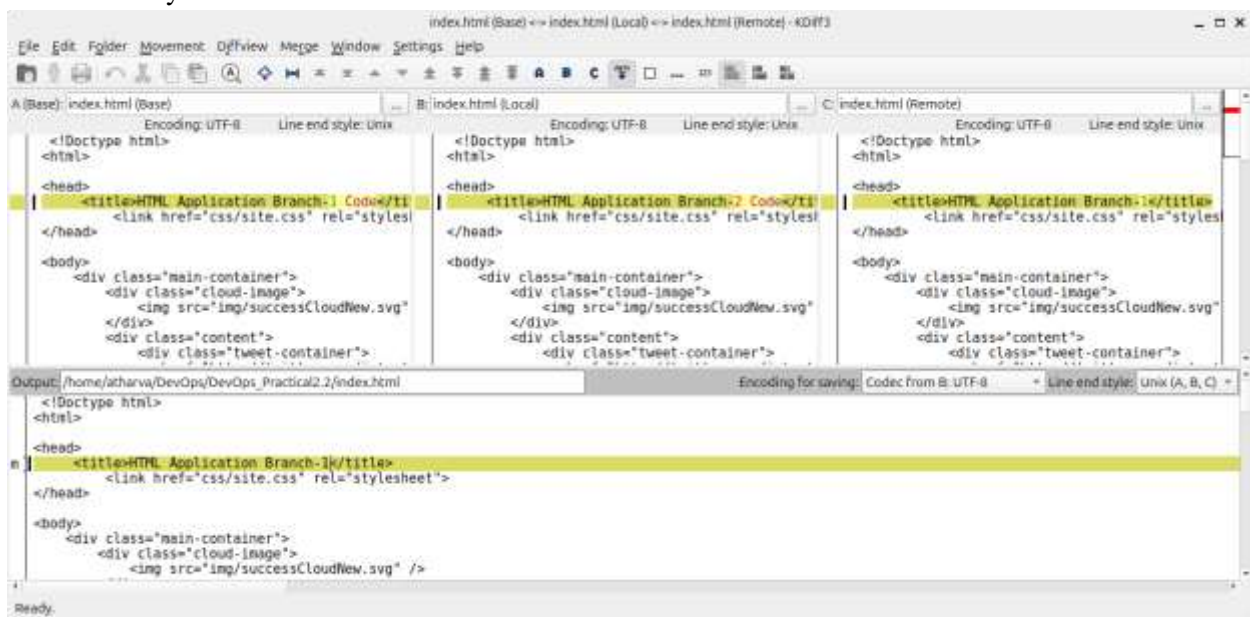
This message is displayed because 'merge.tool' is not configured.
See 'git mergetool --tool-help' or 'git help config' for more details.
'git mergetool' will now attempt to use one of the following tools:
meld opendiff kdiff3 tkdiff xxdiff tortoisemerge gvimdiff diffuze diffmerge ecmerge p4merge araxis bc codecompare emerge emerge vindiff nvindiff
Merging:
index.html

Normal merge conflict for 'index.html':
  [local]: modified file
  [remote]: modified file
Hit return to start merge resolution tool (kdiff3):
```

Step 16: Open the kdiff tool



Step 17: Select which line u want to involve from with file and resolve the conflict this process has to be done manually



Step 19: Push the changes to github.

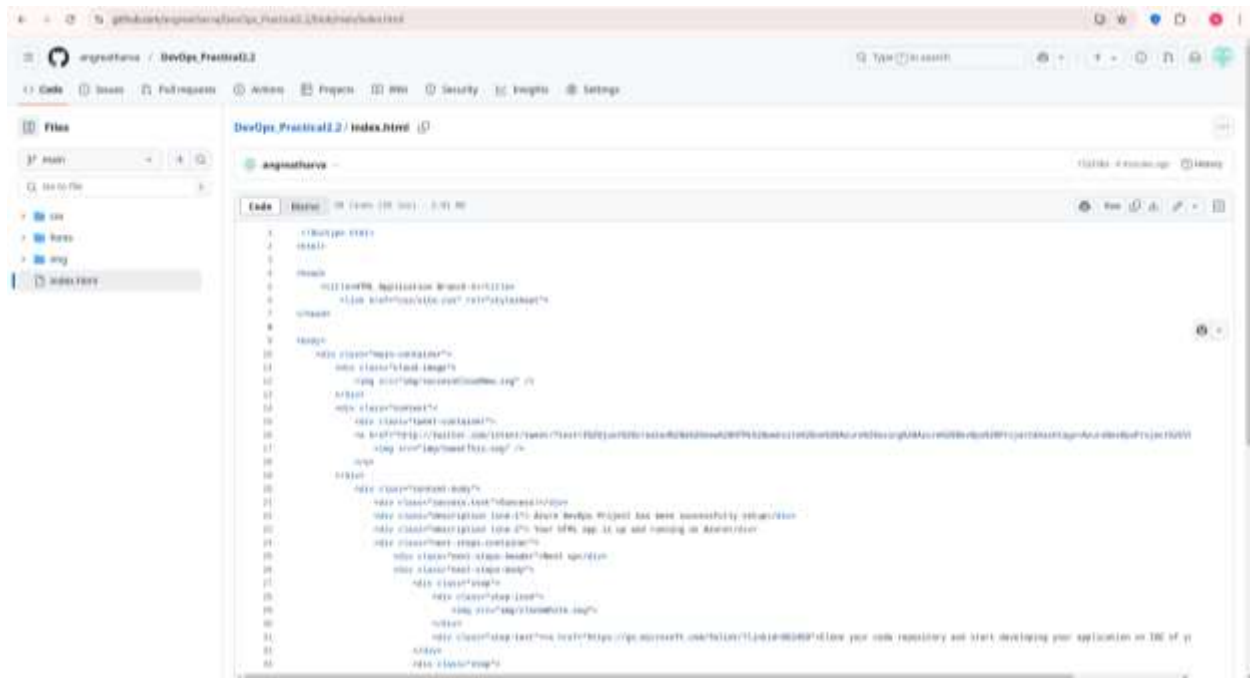
```
atharv@atharva-VirtualBox: /home/atharva/DevOps_Practical2 $ git status
On branch main
All conflicts fixed but you are still merging.
(use 'git commit' to conclude merge)

Changes to be committed:
  modified:   index.html

Untracked files:
  (use 'git add <file>...' to include in what will be committed)
  index.html

atharv@atharva-VirtualBox: /home/atharva/DevOps_Practical2 $ git commit -m "merged successfully"
[main 90ce6d0] merged successfully
atharv@atharva-VirtualBox: /home/atharva/DevOps_Practical2 $ git push origin main
Username for 'https://github.com': angreatharva
Password for 'https://angreatharva@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 2 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 336 bytes | 336.00 KiB/s, done.
Total 4 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To https://github.com/angreatharva/DevOps_Practical2.git
   5198d06..90ce6d0  main -> main
atharv@atharva-VirtualBox: /home/atharva/DevOps_Practical2 $
```

Step 20: Check if the changes are made in github

The screenshot shows a web browser window displaying a GitHub repository page for 'DevOps_Practical2.2'. The repository is owned by 'angreatharva'. The 'Files' tab is selected, showing the 'index.html' file. The file content is a shell script that sets up a development environment. It includes comments in Hindi and English, and commands to install Node.js, npm, and other tools. The script also sets up a local development environment with a 'dev' directory and a 'dev' branch. The script is as follows:

```
1. #!/bin/bash
2. # DevOps_Practical2.2
3.
4. # Install Node.js and npm
5. # Install Node.js and npm
6. # Install Node.js and npm
7. # Install Node.js and npm
8. # Install Node.js and npm
9. # Install Node.js and npm
10. # Install Node.js and npm
11. # Install Node.js and npm
12. # Install Node.js and npm
13. # Install Node.js and npm
14. # Install Node.js and npm
15. # Install Node.js and npm
16. # Install Node.js and npm
17. # Install Node.js and npm
18. # Install Node.js and npm
19. # Install Node.js and npm
20. # Install Node.js and npm
21. # Install Node.js and npm
22. # Install Node.js and npm
23. # Install Node.js and npm
24. # Install Node.js and npm
25. # Install Node.js and npm
26. # Install Node.js and npm
27. # Install Node.js and npm
28. # Install Node.js and npm
29. # Install Node.js and npm
30. # Install Node.js and npm
31. # Install Node.js and npm
32. # Install Node.js and npm
33. # Install Node.js and npm
34. # Install Node.js and npm
35. # Install Node.js and npm
36. # Install Node.js and npm
37. # Install Node.js and npm
38. # Install Node.js and npm
39. # Install Node.js and npm
40. # Install Node.js and npm
41. # Install Node.js and npm
42. # Install Node.js and npm
43. # Install Node.js and npm
44. # Install Node.js and npm
45. # Install Node.js and npm
46. # Install Node.js and npm
47. # Install Node.js and npm
48. # Install Node.js and npm
49. # Install Node.js and npm
50. # Install Node.js and npm
51. # Install Node.js and npm
52. # Install Node.js and npm
53. # Install Node.js and npm
54. # Install Node.js and npm
55. # Install Node.js and npm
56. # Install Node.js and npm
57. # Install Node.js and npm
58. # Install Node.js and npm
59. # Install Node.js and npm
60. # Install Node.js and npm
61. # Install Node.js and npm
62. # Install Node.js and npm
63. # Install Node.js and npm
64. # Install Node.js and npm
65. # Install Node.js and npm
66. # Install Node.js and npm
67. # Install Node.js and npm
68. # Install Node.js and npm
69. # Install Node.js and npm
70. # Install Node.js and npm
71. # Install Node.js and npm
72. # Install Node.js and npm
73. # Install Node.js and npm
74. # Install Node.js and npm
75. # Install Node.js and npm
76. # Install Node.js and npm
77. # Install Node.js and npm
78. # Install Node.js and npm
79. # Install Node.js and npm
80. # Install Node.js and npm
81. # Install Node.js and npm
82. # Install Node.js and npm
83. # Install Node.js and npm
84. # Install Node.js and npm
85. # Install Node.js and npm
86. # Install Node.js and npm
87. # Install Node.js and npm
88. # Install Node.js and npm
89. # Install Node.js and npm
90. # Install Node.js and npm
91. # Install Node.js and npm
92. # Install Node.js and npm
93. # Install Node.js and npm
94. # Install Node.js and npm
95. # Install Node.js and npm
96. # Install Node.js and npm
97. # Install Node.js and npm
98. # Install Node.js and npm
99. # Install Node.js and npm
100. # Install Node.js and npm
```

Conclusion: In this practical Lab, we gained hands-on experience in creating different project versions, managing branches, and establishing both local and remote repositories using terminal commands. We also learned the essential workflow of committing local changes to the global repository. Additionally, we acquired skills in efficiently resolving conflicts using tools like kdiff. Overall, this session equipped us with essential techniques for structured version control and collaborative development.