Lab-Pull Request & Merge

Pre-requisites:

1.) Lab-Clone a Repository locally should be executed.

Steps to Follow

1. To create a new branch, Type the command: $ git branch <yourname>
2. To switch to the newly created branch, Type the command: $ git checkout <yourname>
3. To modify the file with some content, Type the command: $ echo “Hello World, Updated” >> test.txt
4. To add the modified file to staging, Type the command: $ git add test.txt
5. To commit the changes, Type the command: $ git commit -m ‘<any message>’

This will commit changes locally.

6. To push the changes in GitHub repo in your branch, Type the command: $ git push origin <your branch name>

Note: In order to push changes directly to master, replace <your branch name> with master

7. Refresh the browser where GitHub repo is opened.
8. View the repo content.

It should create a new branch with your name in GitHub under the repo and branch should have the test.txt file with changes pushed recently.

9. Select your branch from Branch drop down
10. Click on the test.txt file

It should display the modified changes i.e. “Hello World, updated”.

11. Click on the Pull Request tab
12. Click on New pull request button
13. Select Base as master and Compare as <your branch>
14. Click on Create pull request button
15. Enter any comment
16. Scroll down to see the changed file with comparison.
17. Click Create pull request.

Pull request tab will start showing 1 in count

18. Click on the Pull Request tab agin
19. View the list of open pull request

20. Select the pull request just created
21. Click Merge pull request button, if no conflicts exist.

Refresh the Master branch and view the changes reflected in the changed file.

User1 implements a change in his code and after testing, he commits and pushes his changes to the new branch.

Note that user1 is pushing these changes to the new branch, which is why he used the branch name **test** instead of **master** branch.

Now, if user2 wants the same functionality in the master branch instead of re-implementing,one can achieve this by merging user1's branch with the master branch.

$ git checkout test

Add some test to test.txt as shown below

$ echo "New line" >> test.txt

$ git add *

$ git commit -m "Merge Commit"

$ git push origin test

$ git checkout master

$ ls

Here you will not see the same test.txt file. Now to merge the test_branch with the master branch do.

$ git merge origin/test

$ git push origin master (to push the changes to the master branch)

To verify the changes do

$ git log -1