**Version control system – Git**

* Branching and Rebasing

1] Create a branch ‘bug’, checkout to bug branch to solve the bug in file1.

Bug is – Put a “ ”(space) between numerical and letters

e.g.:- Change1 🡪 Change 1

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ 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: File1.txt

no changes added to commit (use "git add" and/or "git commit -a")

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git branch bug

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git checkout bug

Switched to branch 'bug'

M File1.txt

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug)

$

1. Commit this change with message as “Bug Resolved – Space added between numerical and letters”

=>rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug)

$ git status

On branch bug

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: File1.txt

no changes added to commit (use "git add" and/or "git commit -a")

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug)

$ git commit -am "Bug Resolved – Space added between numerical and letters"

[bug 7ee3111] Bug Resolved – Space added between numerical and letters

1 file changed, 1 insertion(+), 1 deletion(-)

1. Merge the changes to master assuming bug is resolved.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug)

$ git checkout master

Switched to branch 'master'

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git merge bug

Updating a088341..7ee3111

Fast-forward

File1.txt | 2 +-

1 file changed, 1 insertion(+), 1 deletion(-)

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git log --oneline

7ee3111 (HEAD -> master, bug) Bug Resolved – Space added between numerical and letters

a088341 Revert "WI:File1.txt renamed as TestFile.txt."

9e64a69 WI:File1.txt renamed as TestFile.txt.

466e850 WI:File1.txt updated for Change2 in File1.

c992c77 WI:File1.txt updated for Change1 in File1.

17c4f0c WI1:File1.txt, FileToDelete.txt and .gitignore were created.

1. Delete branch ‘bug’

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git branch -d bug

Deleted branch bug (was 7ee3111).

1. Recover the deleted branch ‘bug’ and rename it to ‘bug123’

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git checkout -b bug a088341

Switched to a new branch 'bug'

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug)

$ git branch

\* bug

master

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug)

$ git log --oneline --graph --decorate

\* a088341 (HEAD -> bug) Revert "WI:File1.txt renamed as TestFile.txt."

\* 9e64a69 WI:File1.txt renamed as TestFile.txt.

\* 466e850 WI:File1.txt updated for Change2 in File1.

\* c992c77 WI:File1.txt updated for Change1 in File1.

\* 17c4f0c WI1:File1.txt, FileToDelete.txt and .gitignore were created.

6] Consider the following scenario and achieve the same to Rebase:-

- Create new branch ‘idea’ and make a new commit on ‘idea’ branch

- Now checkout to master and make a new commit on master

Refer the below image:-

master

A

idea

B

D

C

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git branch idea

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git checkout idea

Switched to branch 'idea'

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (idea)

$ git checkout master

Switched to branch 'master'

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git commit -am "commit performed to show rebase"

On branch master

nothing to commit, working tree clean

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git log --oneline --graph --decorate

\* 7ee3111 (HEAD -> master, idea) Bug Resolved – Space added between numerical and letters

\* a088341 (bug) Revert "WI:File1.txt renamed as TestFile.txt."

\* 9e64a69 WI:File1.txt renamed as TestFile.txt.

\* 466e850 WI:File1.txt updated for Change2 in File1.

\* c992c77 WI:File1.txt updated for Change1 in File1.

\* 17c4f0c WI1:File1.txt, FileToDelete.txt and .gitignore were created.

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git commit -am "commit performed to show rebase"

[master 9c402d3] commit performed to show rebase

1 file changed, 2 insertions(+), 2 deletions(-)

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git log --oneline --graph --decorate

\* 9c402d3 (HEAD -> master) commit performed to show rebase

\* 7ee3111 (idea) Bug Resolved – Space added between numerical and letters

\* a088341 (bug) Revert "WI:File1.txt renamed as TestFile.txt."

\* 9e64a69 WI:File1.txt renamed as TestFile.txt.

\* 466e850 WI:File1.txt updated for Change2 in File1.

\* c992c77 WI:File1.txt updated for Change1 in File1.

\* 17c4f0c WI1:File1.txt, FileToDelete.txt and .gitignore were created.

E

***Now use Rebase to get the linear story line as follows:-***

master

A

E

D

B

C

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git checkout idea

Switched to branch 'idea'

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (idea)

$ git rebase master

First, rewinding head to replay your work on top of it...

Fast-forwarded idea to master.

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (idea)

$ git checkout idea

Already on 'idea'

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (idea)

$ git checkout master

Switched to branch 'master'

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git log --oneline --graph --decorate

\* 9c402d3 (HEAD -> master, idea) commit performed to show rebase

\* 7ee3111 Bug Resolved – Space added between numerical and letters

\* a088341 (bug) Revert "WI:File1.txt renamed as TestFile.txt."

\* 9e64a69 WI:File1.txt renamed as TestFile.txt.

\* 466e850 WI:File1.txt updated for Change2 in File1.

\* c992c77 WI:File1.txt updated for Change1 in File1.

\* 17c4f0c WI1:File1.txt, FileToDelete.txt and .gitignore were created.

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git branch -d idea

Deleted branch idea (was 9c402d3).

8] Now say bug is resolved and you are good to release this stable version, tag this release as ‘v1.0’ with message as “Stable version 1.0 released”.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git tag V1.0 9c402d3 -m "stable git version 1 released"

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git log --oneline --graph --decorate

\* 9c402d3 (HEAD -> master, tag: V1.0) commit performed to show rebase

\* 7ee3111 Bug Resolved – Space added between numerical and letters

\* a088341 (bug) Revert "WI:File1.txt renamed as TestFile.txt."

\* 9e64a69 WI:File1.txt renamed as TestFile.txt.

\* 466e850 WI:File1.txt updated for Change2 in File1.

\* c992c77 WI:File1.txt updated for Change1 in File1.

\* 17c4f0c WI1:File1.txt, FileToDelete.txt and .gitignore were created.

9] Consider the following scenario to work with Stashing:-

Checkout to ‘bug123’ branch first and add text as “Change 3” in File1.txt, stage this change and don’t commit the same so that you can get a dirty state of working directory.

Now assume that you have a priority task to be made on master branch, checkout to master and store this dirty state (on ‘bug123’) to stash so that you get back to this state later.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git branch bug123

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git checkout bug123

Switched to branch 'bug123'

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug123)

$ git checkout master

Switched to branch 'master'

M File1.txt

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git stash list

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git stash save Change3bug123

Saved working directory and index state On master: Change3bug123

1. Display the list of all stash.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git stash list

stash@{0}: On master: Change3bug123

11] Checkout to ‘bug123’ branch, reapply the stash which is stored recently and finally commit this change with the message as “Change 3 in File1”

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug123)

$ git stash pop

On branch bug123

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: File1.txt

no changes added to commit (use "git add" and/or "git commit -a")

Dropped refs/stash@{0} (01767456f52daaf8132d5e062c326f3b8111ad0c)

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug123)

$ git commit -am "change 3:file1.txt updated for change 3"

[bug123 3d050b5] change 3:file1.txt updated for change 3

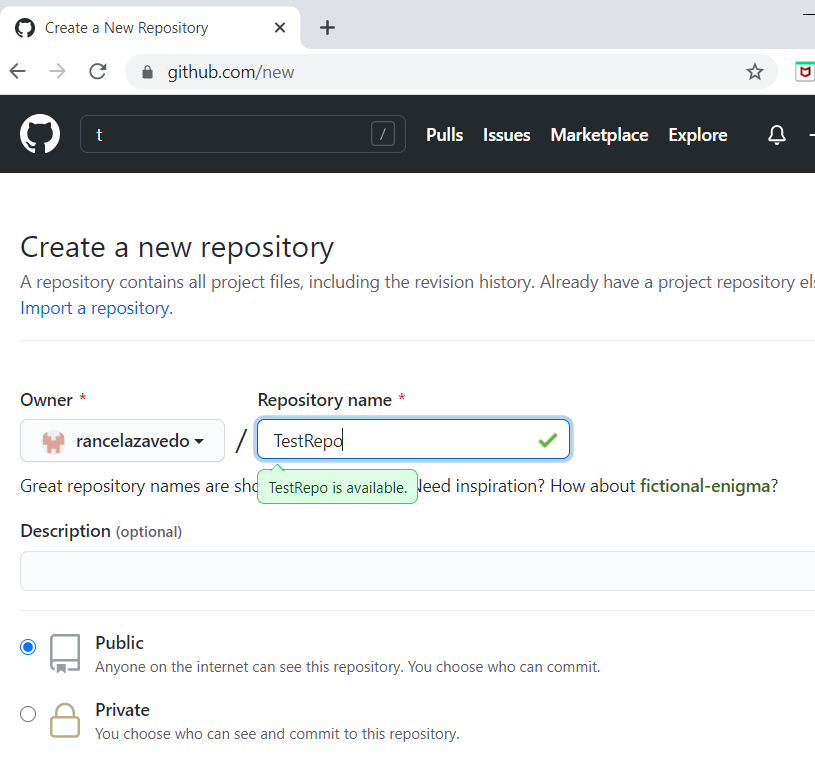
1 file changed, 2 insertions(+), 1 deletion(-)

* **Basic remote operations:-**

1] Create (Signup) GitHub account by visiting <https://github.com/>

2] Create new repository as “TestRepo”.

=>



1. Add remote on to your local repository and name it as ‘Origin’.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug123)

$ git remote add origin https://github.com/rancelazavedo/TestRepo.git

1. Display all remote names with their details.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug123)

$ git remote

origin

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (bug123)

$ git push -u origin master

Enumerating objects: 19, done.

Counting objects: 100% (19/19), done.

Delta compression using up to 8 threads

Compressing objects: 100% (13/13), done.

Writing objects: 100% (19/19), 1.92 KiB | 654.00 KiB/s, done.

Total 19 (delta 2), reused 0 (delta 0)

remote: Resolving deltas: 100% (2/2), done.

To https://github.com/rancelazavedo/TestRepo.git

\* [new branch] master -> master

Branch 'master' set up to track remote branch 'master' from 'origin'.

5] Now create a new file as ‘RemoteFile.txt’ on ***remote*** repository, make 1 commit by making following change in the RemoteFile.txt

Add changes to file as:-

Change1 – commit with message as “Change1 in RemoteFile”

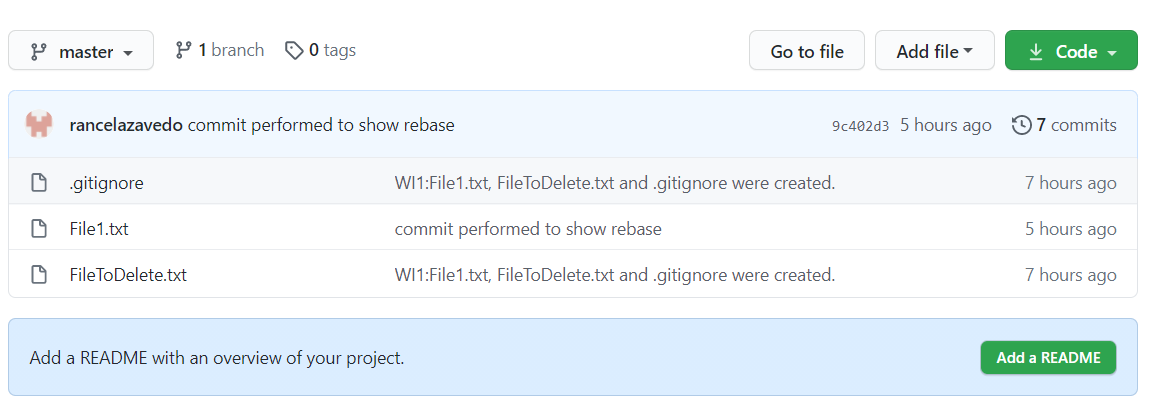
And if you check the log on ***remote*** it should display in following manner:-

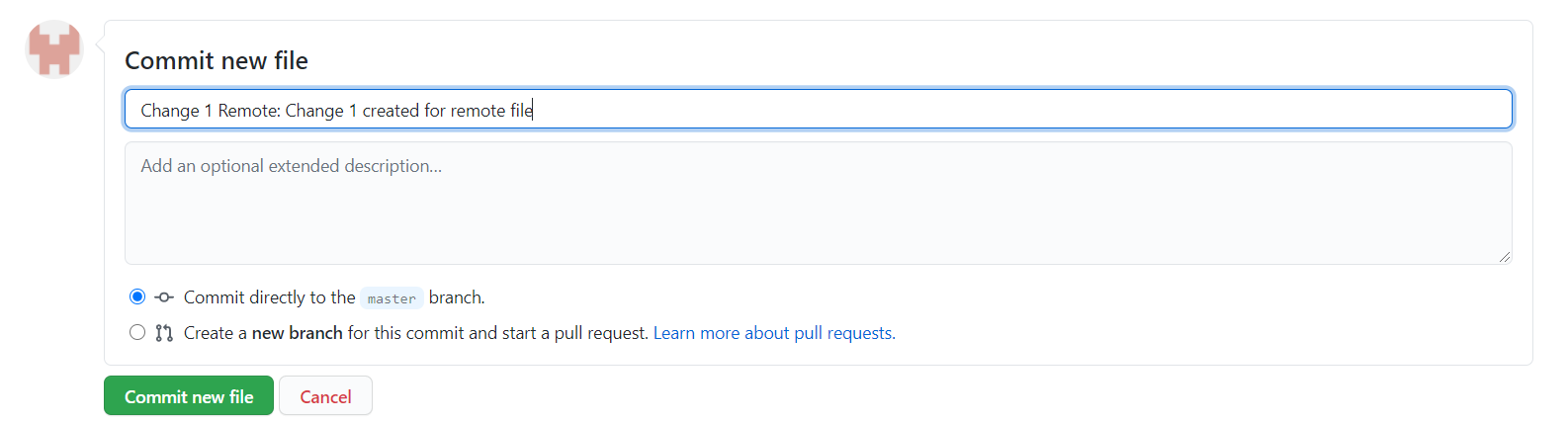
Change1 in RemoteFile

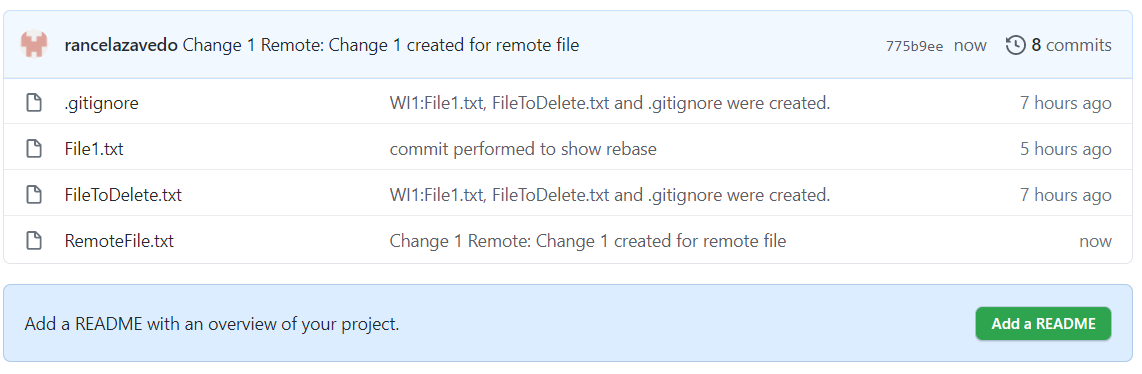
Note: - If you check log on local there will be no commits which are made on remote.

=>

Git Hub:









Git Bash:

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git log --oneline --graph --decorate

\* 9c402d3 (HEAD -> master, tag: V1.0, origin/master) commit performed to show rebase

\* 7ee3111 Bug Resolved – Space added between numerical and letters

\* a088341 (bug) Revert "WI:File1.txt renamed as TestFile.txt."

\* 9e64a69 WI:File1.txt renamed as TestFile.txt.

\* 466e850 WI:File1.txt updated for Change2 in File1.

\* c992c77 WI:File1.txt updated for Change1 in File1.

\* 17c4f0c WI1:File1.txt, FileToDelete.txt and .gitignore were created.

1. Fetch remote ‘Origin’ to synch up with remote.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git fetch

remote: Enumerating objects: 4, done.

remote: Counting objects: 100% (4/4), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0

Unpacking objects: 100% (3/3), 700 bytes | 70.00 KiB/s, done.

From https://github.com/rancelazavedo/TestRepo

9c402d3..775b9ee master -> origin/master

1. Finally merge the remote changes to local master.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git merge

Updating 9c402d3..775b9ee

Fast-forward

RemoteFile.txt | 1 +

1 file changed, 1 insertion(+)

create mode 100644 RemoteFile.txt

8] Now, make 1 more commit on ***remote (GitHub)*** by making following change in the RemoteFile.txt

Add changes to file as:-

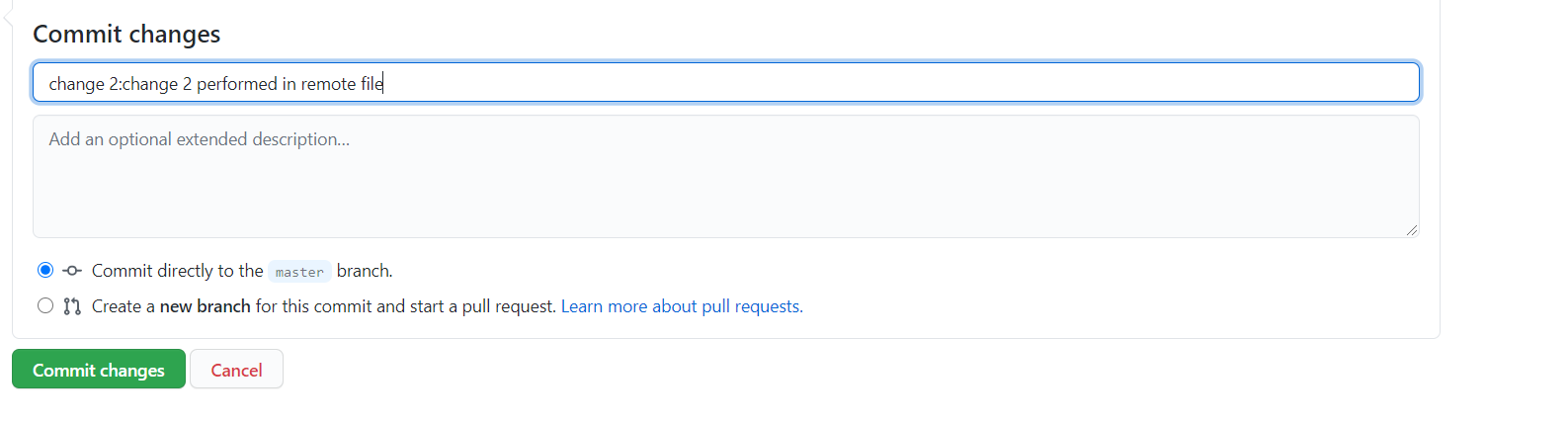
Change2 – commit with message as “Change2 in RemoteFile”

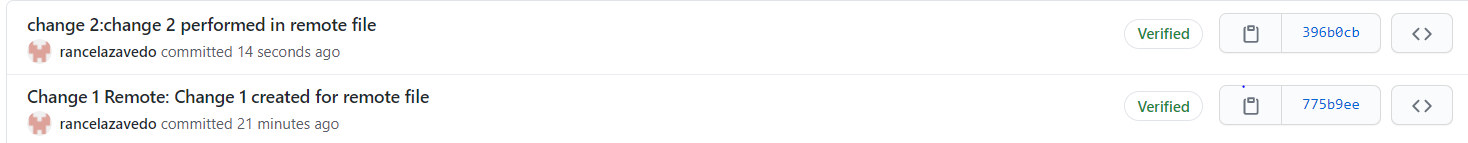
And if you check the log on ***remote*** it should display in following manner:-

Change2 in RemoteFile

Change1 in RemoteFile

=>





1. Use Pull to take these recent changes to local.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git log --oneline --graph --decorate

\* 775b9ee (HEAD -> master, origin/master) Change 1 Remote: Change 1 created for remote file

\* 9c402d3 (tag: V1.0) commit performed to show rebase

\* 7ee3111 Bug Resolved – Space added between numerical and letters

\* a088341 (bug) Revert "WI:File1.txt renamed as TestFile.txt."

\* 9e64a69 WI:File1.txt renamed as TestFile.txt.

\* 466e850 WI:File1.txt updated for Change2 in File1.

\* c992c77 WI:File1.txt updated for Change1 in File1.

\* 17c4f0c WI1:File1.txt, FileToDelete.txt and .gitignore were created.

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git pull

remote: Enumerating objects: 5, done.

remote: Counting objects: 100% (5/5), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0

Unpacking objects: 100% (3/3), 686 bytes | 68.00 KiB/s, done.

From https://github.com/rancelazavedo/TestRepo

775b9ee..396b0cb master -> origin/master

Updating 775b9ee..396b0cb

Fast-forward

RemoteFile.txt | 1 +

1 file changed, 1 insertion(+)

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git log --oneline --graph --decorate

\* 396b0cb (HEAD -> master, origin/master) change 2:change 2 performed in remote file

\* 775b9ee Change 1 Remote: Change 1 created for remote file

\* 9c402d3 (tag: V1.0) commit performed to show rebase

\* 7ee3111 Bug Resolved – Space added between numerical and letters

\* a088341 (bug) Revert "WI:File1.txt renamed as TestFile.txt."

\* 9e64a69 WI:File1.txt renamed as TestFile.txt.

\* 466e850 WI:File1.txt updated for Change2 in File1.

\* c992c77 WI:File1.txt updated for Change1 in File1.

\* 17c4f0c WI1:File1.txt, FileToDelete.txt and .gitignore were created.

10] Now, make 2 new commits on ***local*** by making following change in the RemoteFile.txt

Add changes to file as:-

Change3 – commit with message as “Change3 in RemoteFile”

Change4 – commit with message as “Change4 in RemoteFile”

And if you check the log on ***local*** it should display in following manner:-

Change4 in RemoteFile

Change3 in RemoteFile

Change2 in RemoteFile

Change1 in RemoteFile

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git status

On branch master

Your branch is up to date with 'origin/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: RemoteFile.txt

no changes added to commit (use "git add" and/or "git commit -a")

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git commit -am "Change3 in RemoteFile"

[master 0acea07] Change3 in RemoteFile

1 file changed, 1 insertion(+)

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git status

On branch master

Your branch is ahead of 'origin/master' by 1 commit.

(use "git push" to publish your local commits)

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: RemoteFile.txt

no changes added to commit (use "git add" and/or "git commit -a")

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git commit -am "Change4 in RemoteFile"

[master 873baee] Change4 in RemoteFile

1 file changed, 1 insertion(+)

1. Push these recent changes to remote.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git push origin

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% (6/6), 580 bytes | 290.00 KiB/s, done.

Total 6 (delta 3), reused 0 (delta 0)

remote: Resolving deltas: 100% (3/3), completed with 1 local object.

To https://github.com/rancelazavedo/TestRepo.git

396b0cb..873baee master -> master

1. Take a copy (Clone) of remote repository to your local machine on different drive or folder.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/Dummy (master)

$ git clone https://github.com/rancelazavedo/TestRepo.git

Cloning into 'TestRepo'...

remote: Enumerating objects: 31, done.

remote: Counting objects: 100% (31/31), done.

remote: Compressing objects: 100% (18/18), done.

remote: Total 31 (delta 7), reused 25 (delta 5), pack-reused 0

Unpacking objects: 100% (31/31), 3.68 KiB | 37.00 KiB/s, done.

1. Rename remote branch ‘Origin’ to ‘RemoteBranch’.

=>

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git remote rename origin RemoteBranch

rance@LAPTOP-9CQDJ3S1 MINGW64 ~/Desktop/PSL\_GIT/Assignments/GitAssignment1 (master)

$ git remote

RemoteBranch