1. Configure your user name and email.

```
C:\Users\Lenovo> git config --global user.name"deeksha-rapidops"

C:\Users\Lenovo>git config --global user.email"deeksha.rapidops@rapidops.com"
```

2. Clone repo of your name from GitHub to the local system.

   cmd-> git clone “address”

```
C:\Users\Lenovo\Desktop>git clone https://github.com/Training-Rapidops/Deeksha-Varshney.git
Cloning into 'Deeksha-Varshney'...
warning: You appear to have cloned an empty repository.
```

3. Create a file inside the repo, and make your first commit "My First Commit".

   cmd-> echo “hello”>>file name

   git add .

   Git commit -m “My first commit”

```
C:\Users\Lenovo\Desktop\Deeksha-Varshney>echo "Hello" >>Hello.txt

C:\Users\Lenovo\Desktop\Deeksha-Varshney>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        Hello.txt

nothing added to commit but untracked files present (use "git add" to track)

C:\Users\Lenovo\Desktop\Deeksha-Varshney>git add .

C:\Users\Lenovo\Desktop\Deeksha-Varshney>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   Hello.txt
```

```
C:\Users\Lenovo\Desktop\Deeksha-Varshney>git commit -m "My First Commit"
[master (root-commit) 435bf3f] My First Commit
 1 file changed, 1 insertion(+)
 create mode 100644 Hello.txt
```

4. Create and switch to the branch 'test/development' (create from the master branch and it should be from origin).

   cmd-> git branch test , git checkout test

```
C:\Users\Lenovo\Desktop\Deeksha-Varshney>git branch test

C:\Users\Lenovo\Desktop\Deeksha-Varshney>git checkout test
Switched to branch 'test'
```

5. Add a file in this branch and commit your changes with the message "Practice started in test branch".

   cmd-> echo”Hii” >>Hii.txt

   git add .

   git commit -m ”Practice Started in Test Branch”

```
C:\Users\Lenovo\Desktop\Deeksha-Varshney>echo "Hii" >>Hii.txt

C:\Users\Lenovo\Desktop\Deeksha-Varshney>git status
On branch test
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        Hii.txt

nothing added to commit but untracked files present (use "git add" to track)

C:\Users\Lenovo\Desktop\Deeksha-Varshney>git add .

C:\Users\Lenovo\Desktop\Deeksha-Varshney>git status
On branch test
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   Hii.txt


C:\Users\Lenovo\Desktop\Deeksha-Varshney>git commit -m "Practice Started in Test Branch"
[test cb37de4] Practice Started in Test Branch
 1 file changed, 1 insertion(+)
 create mode 100644 Hii.txt
```

6. Now push your changes and this branch to the remote.

   Cmd -> git push -u origin test

```
C:\Users\Lenovo\Desktop\Deeksha-Varshney>git push -u origin test
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (6/6), 497 bytes | 248.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Training-Rapidops/Deeksha-Varshney.git
 * [new branch]      test -> test
Branch 'test' set up to track remote branch 'test' from 'origin'.
```

7. Go to your GitHub repository and create a pull request to merge this branch in master. Also, add Ravindra & your mentor as reviewers.

   Created pull request and merge branch.

8. Fork a public repository **Test-Training**

   Forked in github.

```
C:\Users\Lenovo\Desktop>git clone https://github.com/deeksha-rapidops/Test-Training.git
Cloning into 'Test-Training'...
remote: Enumerating objects: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (28/28), done.
remote: Total 38 (delta 15), reused 19 (delta 7), pack-reused 0
Receiving objects: 100% (38/38), 6.80 KiB | 773.00 KiB/s, done.
Resolving deltas: 100% (15/15), done.
```

9. Make a change in the template.html file by adding 'author: {your-name}' and add commit 'Author updated by {your-name}'.

   Cmd-> git add .

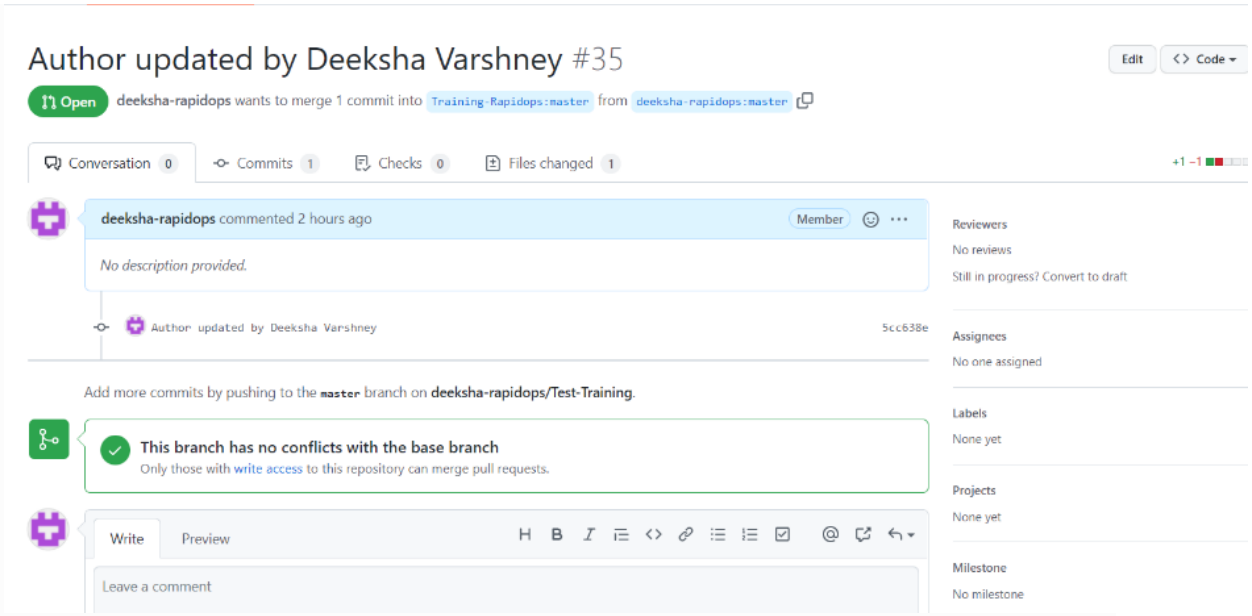
   git commit -m “Author updated by Deeksha Varshney”

```
C:\Users\Lenovo\Desktop>cd Test-Training

C:\Users\Lenovo\Desktop\Test-Training>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:   template.html

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

C:\Users\Lenovo\Desktop\Test-Training>git add .

C:\Users\Lenovo\Desktop\Test-Training>git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   template.html


C:\Users\Lenovo\Desktop\Test-Training>git commit -m "Author updated by Deeksha Varshney"
[master 5cc638e] Author updated by Deeksha Varshney
 1 file changed, 1 insertion(+), 1 deletion(-)
```

10. Give PR to merge your forked repo in the **Test-Training** repo

    Made a pull request on github.

```
C:\Users\Lenovo\Desktop\Test-Training>git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 323 bytes | 323.00 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.
To https://github.com/deeksha-rapidops/Test-Training.git
   65f8803..5cc638e  master -> master

C:\Users\Lenovo\Desktop\Test-Training>
```



11. Come back to your repo, where you create a branch 'practice' and add 3 commits then rebase it with the master.

Cmd-> git branch practice , git checkout practice

Echo for 3 file creations

Git commit for commit each file(Missed screenshots of these commands)

All 3 commits are rebased with master.

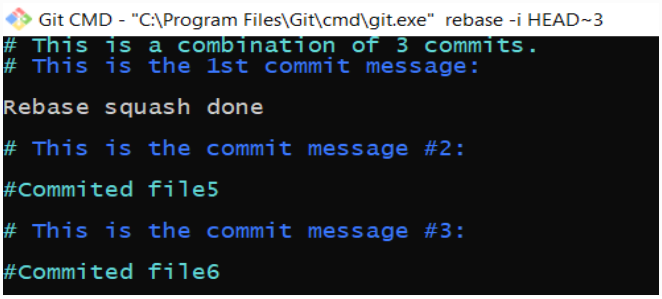
```
C:\Users\Lenovo\Desktop\Test-Training>git rebase master
Current branch practice is up to date.

C:\Users\Lenovo\Desktop\Test-Training>git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

C:\Users\Lenovo\Desktop\Test-Training>git rebase practice
Successfully rebased and updated refs/heads/master.

C:\Users\Lenovo\Desktop\Test-Training>git log
commit 5382a312cbf1b8ae97533956deb3c80b4d1ad919 (HEAD -> master, practice)
Author: deeksha varshney <deeksha.varshney@rapidops.com>
Date:   Mon Jan 17 17:12:21 2022 +0530

commit 5382a312cbf1b8ae97533956deb3c80b4d1ad919 (HEAD -> master, practice)
Author: deeksha varshney <deeksha.varshney@rapidops.com>
Date:   Mon Jan 17 17:12:21 2022 +0530

    Commited file3
```

12. Push your changes to remote, then Add 3 commits again and squash them into the first commit by keeping the message "Rebase squash done".

Cmd-> git push origin master(Missed Screeshots)

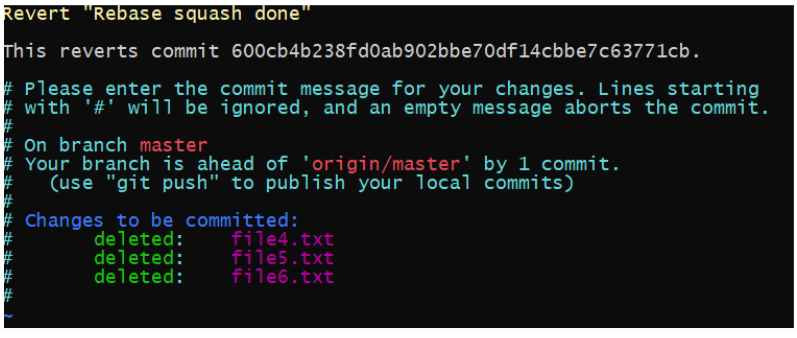
Cmd-> git rebase -I HEAD~3

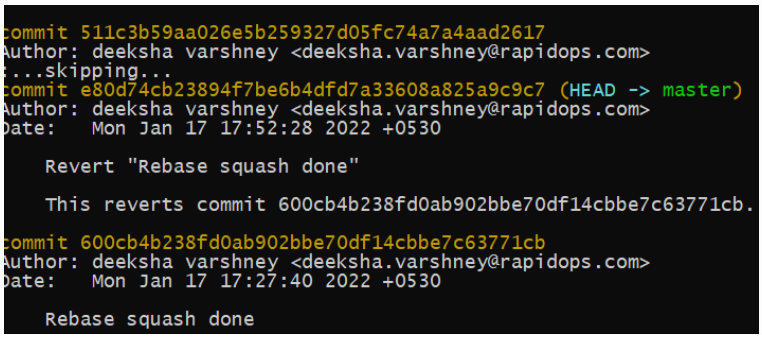
Rebase squash done



13. Now revert these changes but note that the changes must be retained in the commit history. (use default revert commit message)

Cmd-> git revert “commit id”





14. Reset your last commit without losing the changes and then commit with the message "Finished revert with staging changes".

Cmd-> git reset –soft HEAD~1

```
C:\Users\Lenovo\Desktop\Test-Training>git reset --soft HEAD~1

C:\Users\Lenovo\Desktop\Test-Training>git log
commit 600cb4b238fd0ab902bbe70df14cbbe7c63771cb (HEAD -> master)
Author: deeksha varshney <deeksha.varshney@rapidops.com>
Date:   Mon Jan 17 17:27:40 2022 +0530

    Rebase squash done

commit 5382a312cbf1b8ae97533956deb3c80b4d1ad919 (origin/master, origin/HEAD, practice)
Author: deeksha varshney <deeksha.varshney@rapidops.com>
Date:   Mon Jan 17 17:12:21 2022 +0530

    Commited file3

commit 46e1bbc9a56b4ea1f74f9c885abc89f8dac2f8fe
Author: deeksha varshney <deeksha.varshney@rapidops.com>
Date:   Mon Jan 17 17:11:46 2022 +0530

    Commited file2

commit 511c3b59aa026e5b259327d05fc74a7a4aad2617
Author: deeksha varshney <deeksha.varshney@rapidops.com>
Date:   Mon Jan 17 17:09:57 2022 +0530

    Commited file1
```

15. Create one commit then perform a hard reset such that you're back to the commit with the message "Rebase squash done".

cmd-> git reset –hard HEAD~1

```
C:\Users\Lenovo\Desktop\Test-Training>git reset --hard HEAD~1
HEAD is now at 600cb4b Rebase squash done

C:\Users\Lenovo\Desktop\Test-Training>
```

16. Create a file index.html, and add it to the staging index then stash it.

Cmd-> git stash

```
C:\Users\Lenovo\Desktop\Test-Training>echo >>index.html

C:\Users\Lenovo\Desktop\Test-Training>git add index.html

C:\Users\Lenovo\Desktop\Test-Training>git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   index.html


C:\Users\Lenovo\Desktop\Test-Training>git stash
Saved working directory and index state WIP on master: 600cb4b Rebase squash done
```

17. Check the list of stash, what changes are there in the stash, then bring your changes from stash.

Cmd = git stash list , git stash pop 0

```
C:\Users\Lenovo\Desktop\Test-Training>git stash list
stash@{0}: WIP on master: 600cb4b Rebase squash done

C:\Users\Lenovo\Desktop\Test-Training>git stash pop 0
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   index.html

Dropped refs/stash@{0} (5321798353b5cb67af85a634ea617408331edfb2)
```

18. Commit with a message "Revert, Reset & Stash done".

```
C:\Users\Lenovo\Desktop\Test-Training>git commit -m "Revert,Reset& Stash done."
[master ba57b44] Revert,Reset& Stash done.
 1 file changed, 1 insertion(+)
 create mode 100644 index.html

C:\Users\Lenovo\Desktop\Test-Training>
```