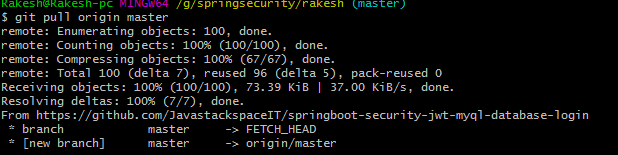
**Git Hub commands**

* **Step for git configuration**

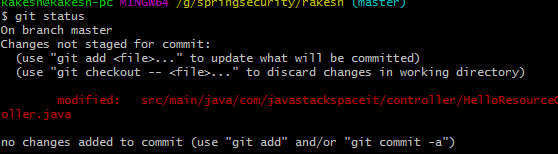
1. **Git init 🡺 to Initalize the git in our local working directory**
2. **git remote add origin https://github.com/JavastackspaceIT/springboot-security-jwt- myql-database-login.git**







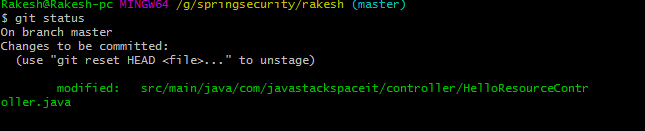
1. **git status**



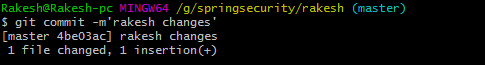
1. **git add .**



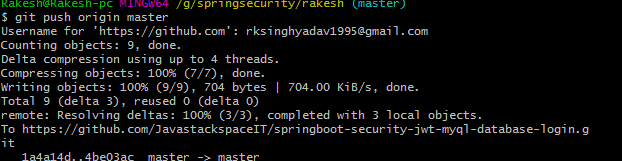
1. **git status**



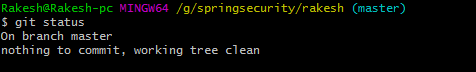
1. **git commt –m ‘message’**



1. **git push origin “origin name”**



1. **git status**

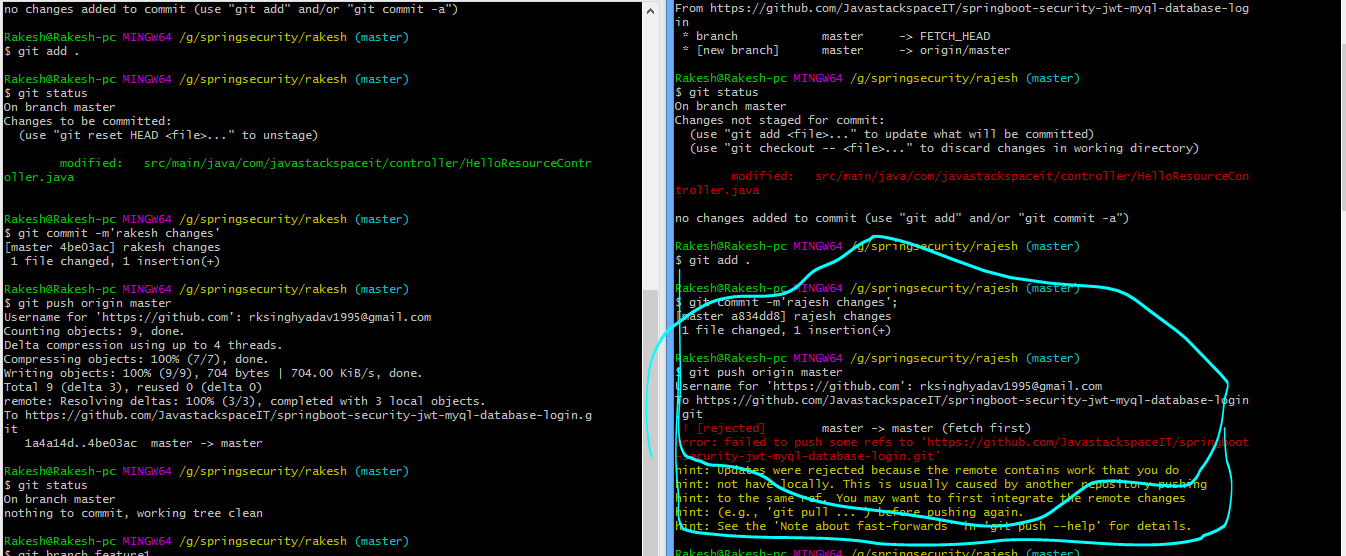


**How to resolve the conflict in the git**

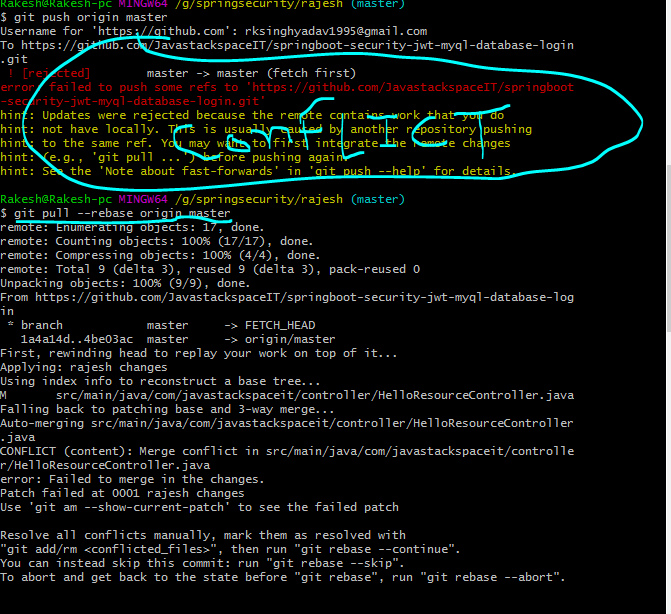
**If user one “rakesh “ changes in the “HelloResouceController” on the line number 10 and it add , commit and push the changes on the master branch.**

**Now the user “rajesh” won’t take any update from master branch and he wrote on the line number 10 so other code and he try to add, commit and push on master branch then he will get error**

**Bellow is screenshot .**



**To resolve the issue user “rajesh” has to use rebase command. Which are given bellow .**



**To see the current patch , that is giving conflict we have use following command.**

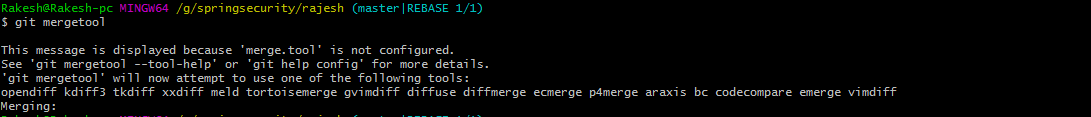


**To see the status of rebase we have to use following command.**



**To merge the changes which changes you want to use first or which is in last we have to use following command on git.**

Git mergetool



**It will open a new window with original code and rakesh and rajesh use which is changes all changes show.**

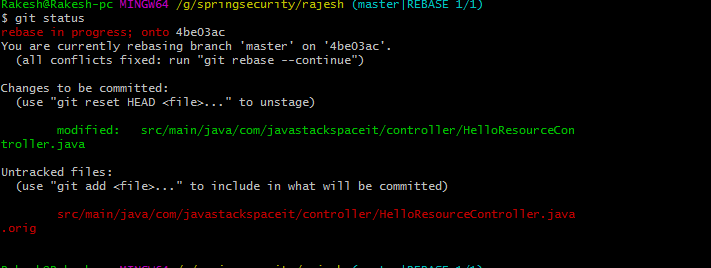
**So to write the change in file we have to use**

**First we have click on “ESC” button on keyboard the we have to use**

**:wq for write and quite.**

**After doing this we want to see the changes then we use**

**: git status**

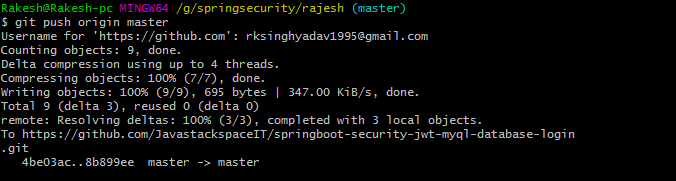


**Then we we have to use git command : git rebase –continue**



**Then to push the changes on repository we have to use**

**Following command.**



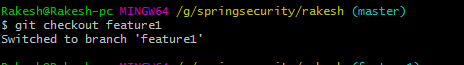
**Git basic commad.**

**How to create branch in git using command.**

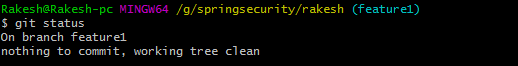
**: git branch <branch\_name> or git checkout –b <branch\_name>**



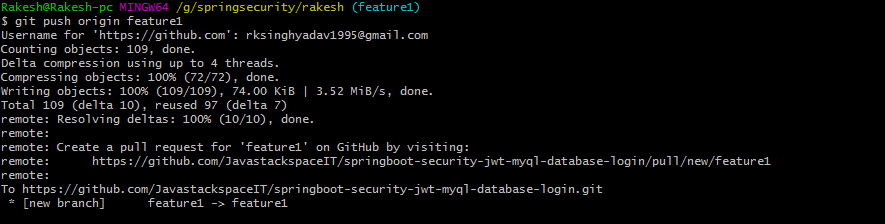
**To change master branch to feature1 branch use following command.**



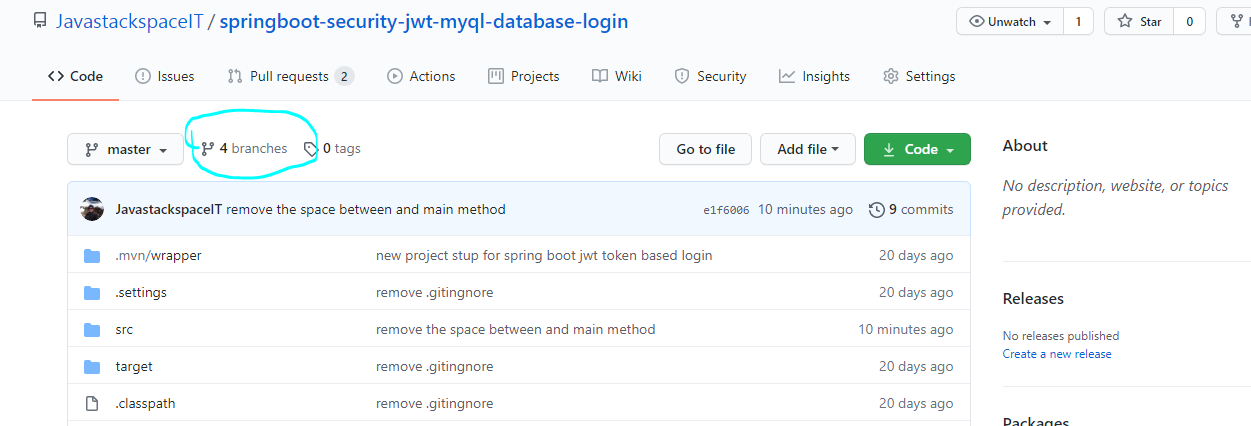
**To check changes use following command.**

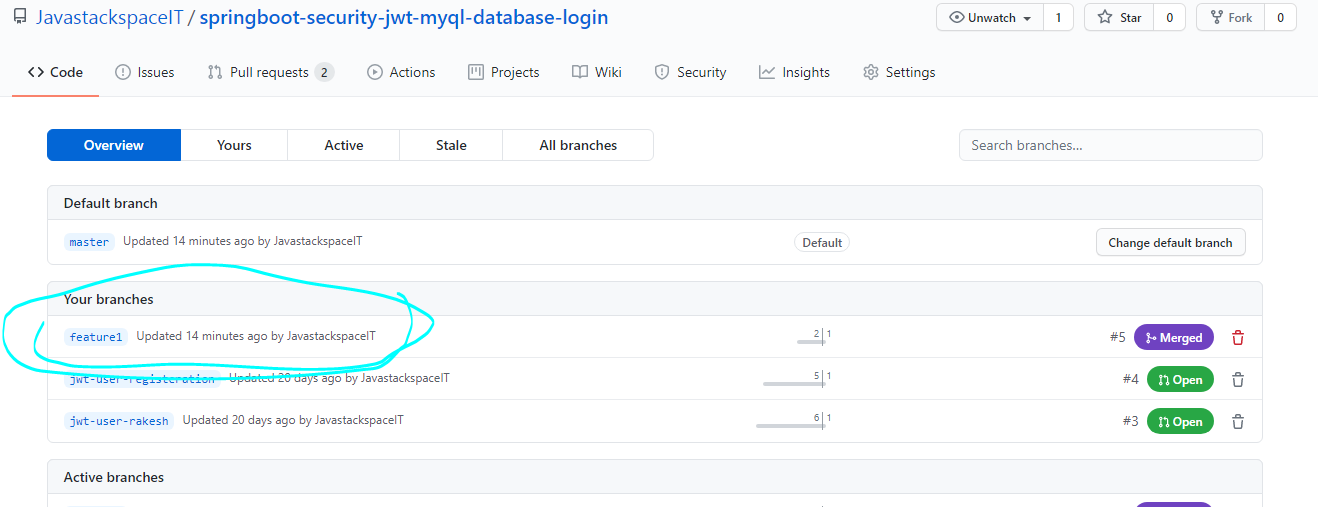


**To push the change form feature one to git repository we use following command.**

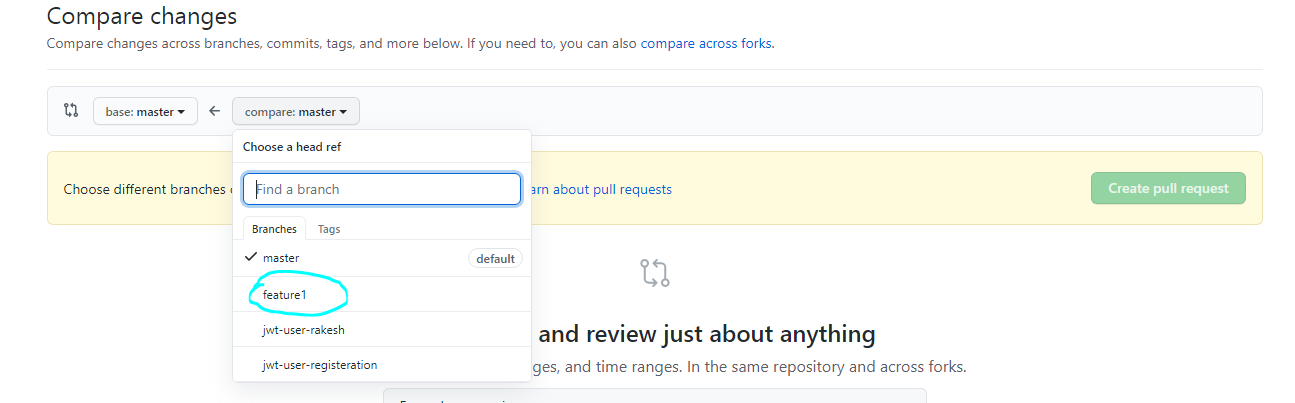


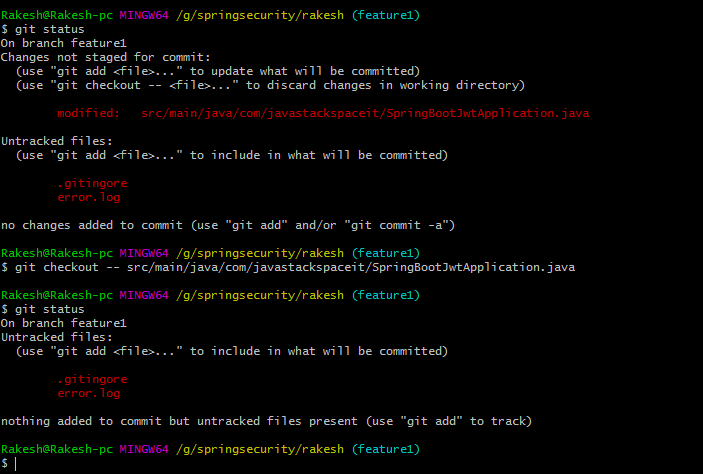
**When we push this changes new branch will be created on git repository**



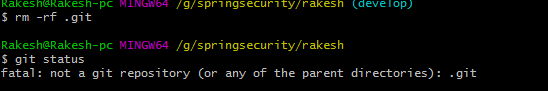


**To raise the pull request for merge the changes we have to use following .**



* **git restore filename : it will restore the modify file from staging to dev.**
* **Git checkout - - filename : it will discard the changes.**
* 
* **To Delete the Git repository we use following command**

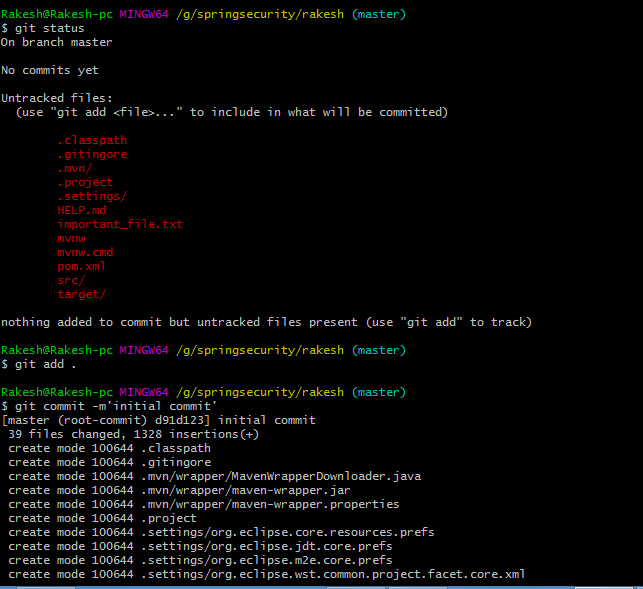
**: rm –rf .git**

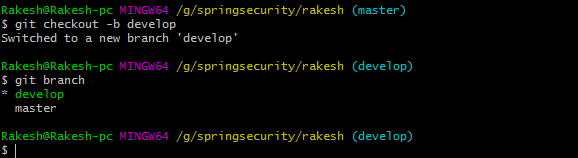


**To initialize the git again use bellow command.**



**To add in git use git add . command.**



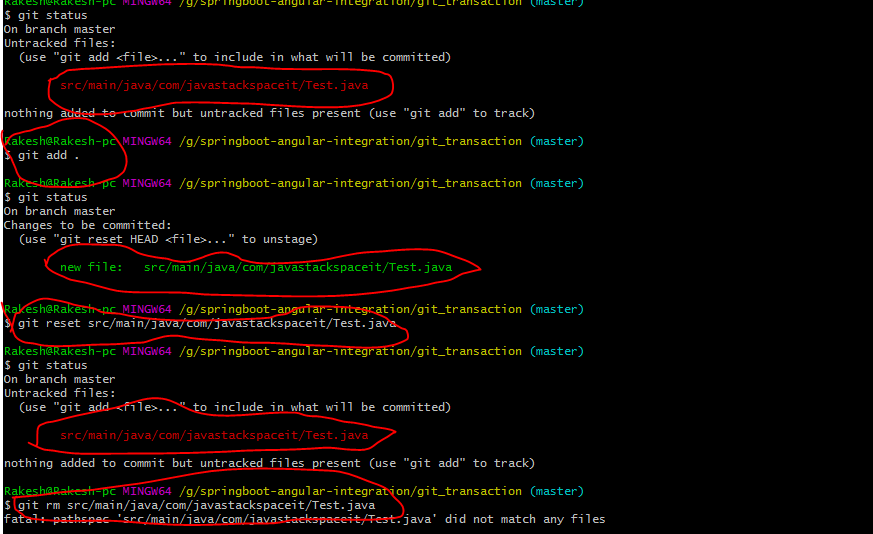
* Git branch : check how many branch is there
* 

To remove the file from git working directory use following command to delete.

: git rm <file\_name>



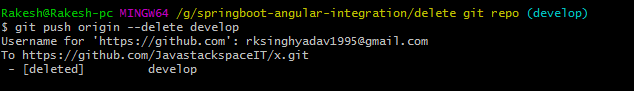
To remove the file from Git staging area.



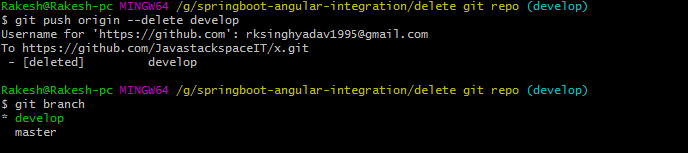
**How to delete the remote branch:**

We can delete the custom created branch example “ develop branch , feature1 branch” using following command.

**: git push origin - - delete <branch\_name>**

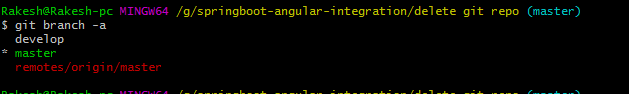


It will only delete remote branch. But local branch will exits. Bellow is screenshot.

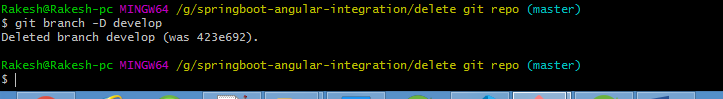


**To check local and remote branch using git command.**

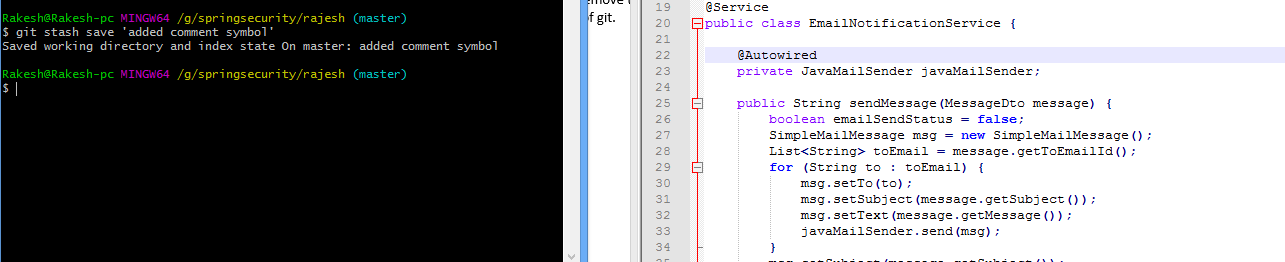
**: git branch –a**



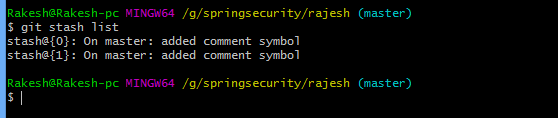
**How to delete local branch using git command.**



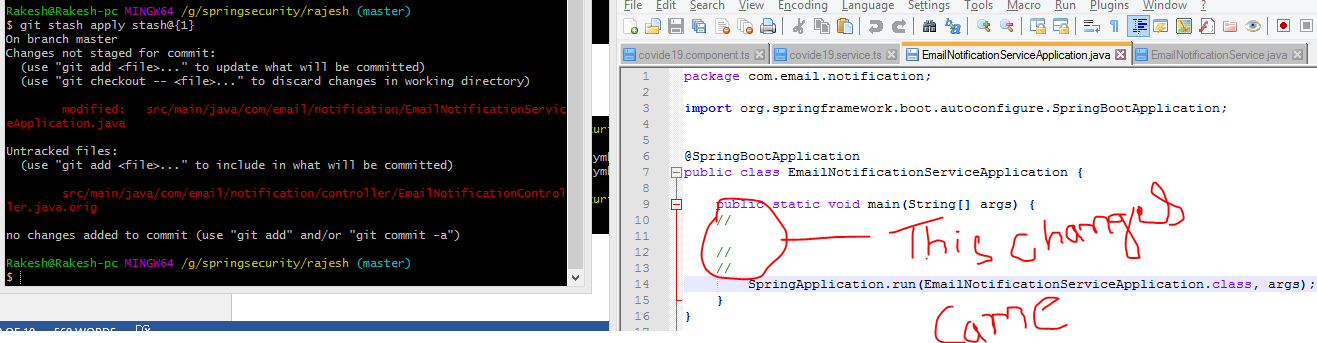
Git stash => Git stash command is used to remove the changes temporary from our file. It will create backup file and save in temporary memory of git.



Check git stash list:

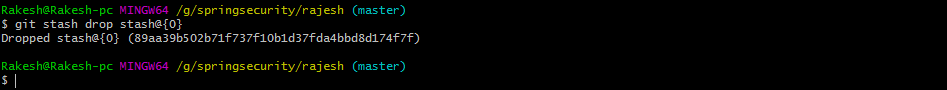


To retrieve the changes again in our local file use this command.



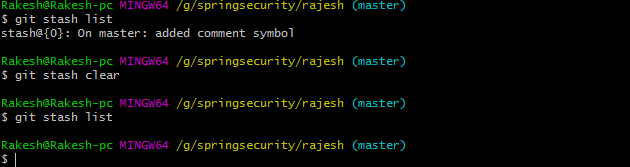
To delete the stash from local memory .

$ git stash drop <stash id> : it will delete only single stash



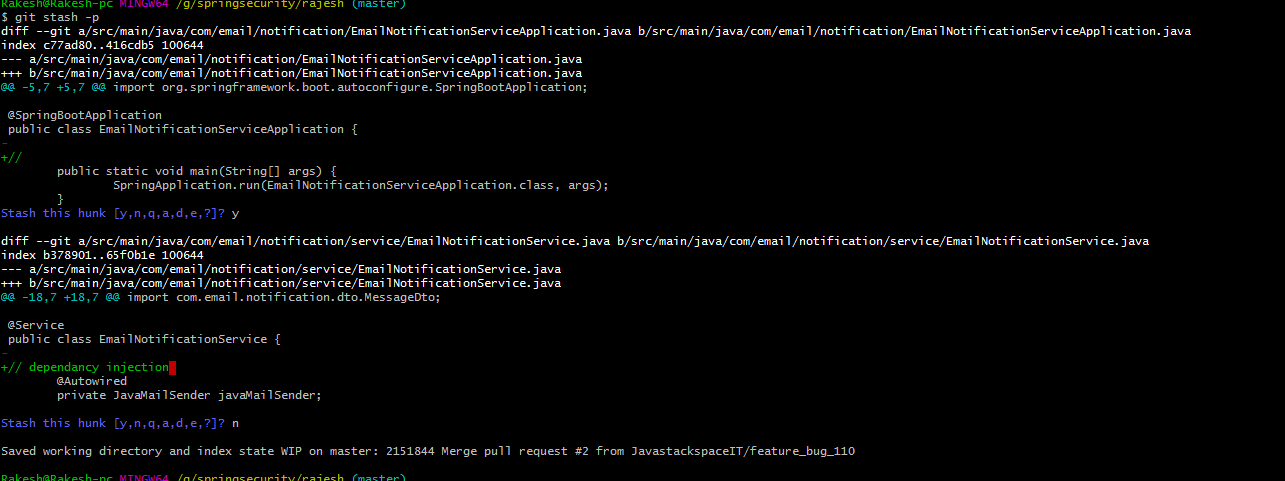
For delete all stash from local memory use following command.

$ git stash clear



Now we will see how to dash particular file from project. We will use

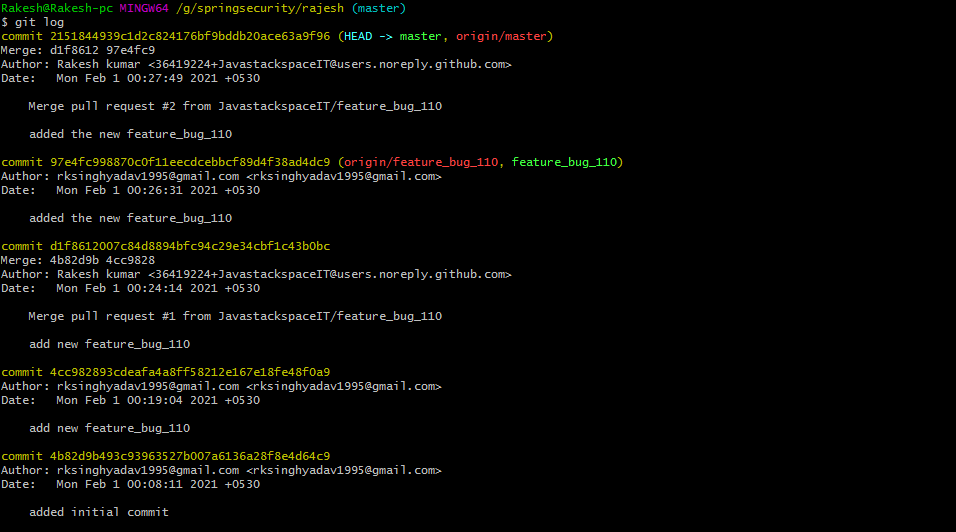
$ git stash –p , It will open interactive mode and it will ask which file you want to stash.



**GIT LOG Command**

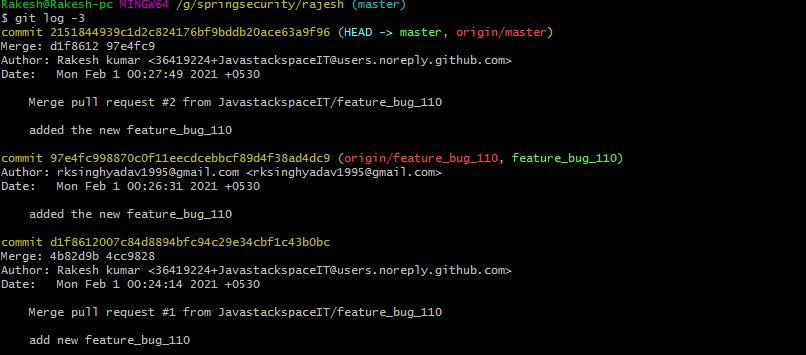
$ log log

This command is used to show the log history of the project.



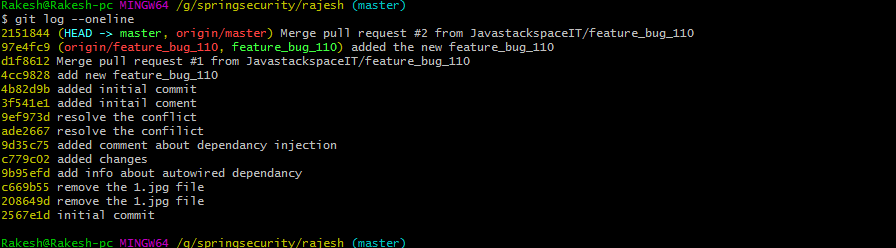
$ git log -3

It will show only last 3 commit.



$ git log –oneline

It will show commit message



$ git log - - grep=’message name’

It will search the commit file , message case sensitive

$ git log - i - - grep=’message name’ case in sensitive

