

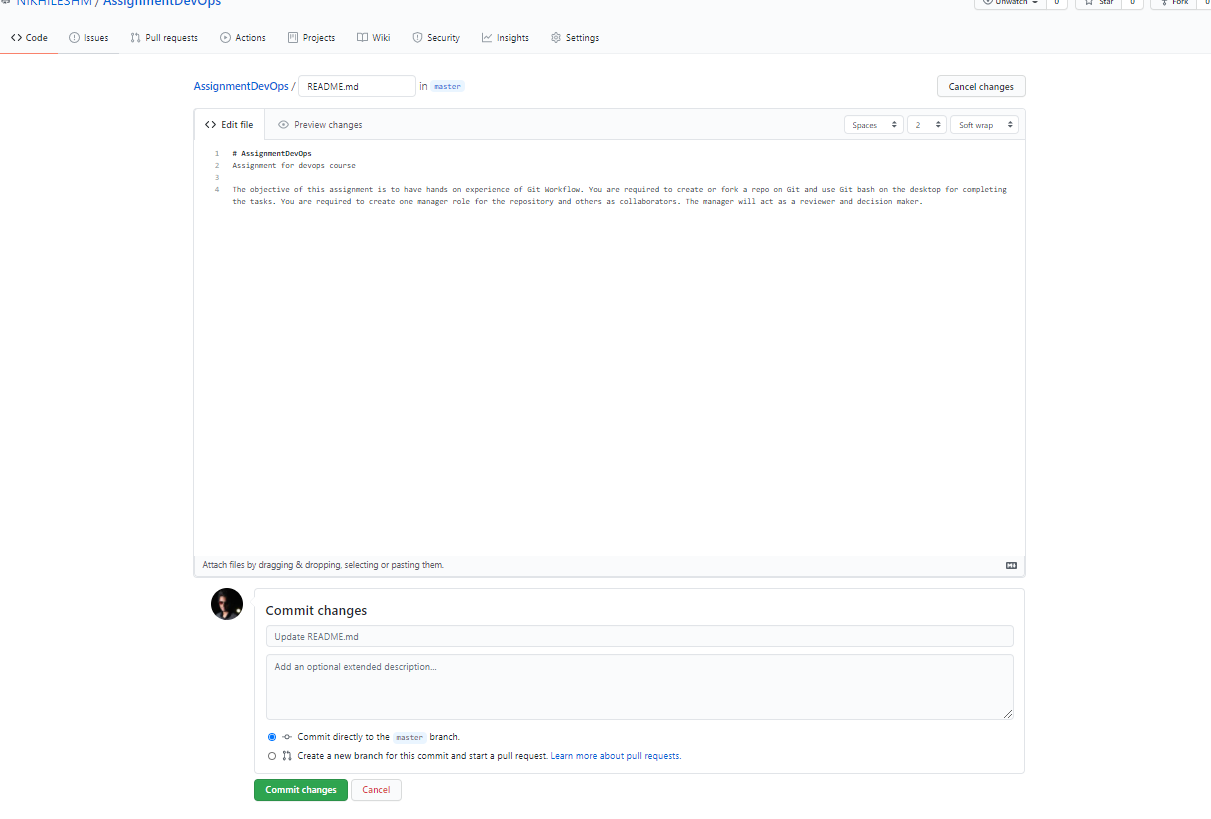
|  |
| --- |
| ASSIGNMENT-1  DEVOPS  2018 |
|  |
| GIT WORK FLOW  BITS Pilani  Authored by: NIKHILESH M(2020MT93513) |



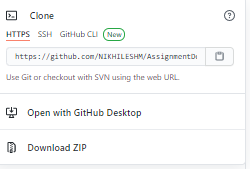
# GIT Work Flow

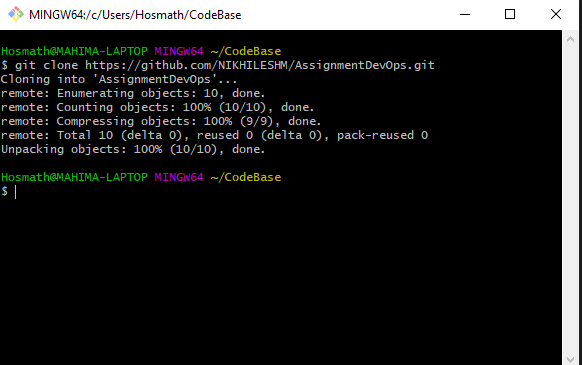
|  |
| --- |
| Step 1 - Create Repository Login to Managers GitHub Account to create Git Repo    Create Repository    Set up PR Notifications |
| Step 2 – Invite Collaborator    Step 3 – Create Branch and set Branch permissions     Setup Branch Permissions – |
|  |

## Step 3 – Edit File Readme with Commit



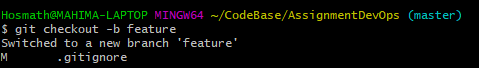
## Step 4 – Clone the Repo





## Step 5 – Setup Some Code

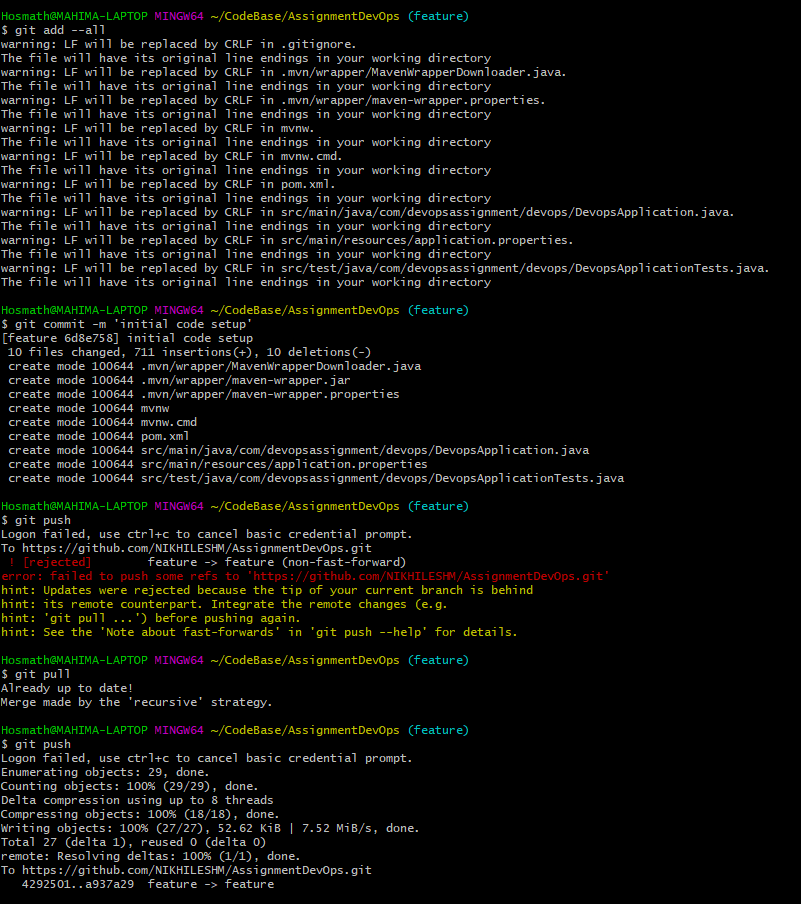
Create local Branch



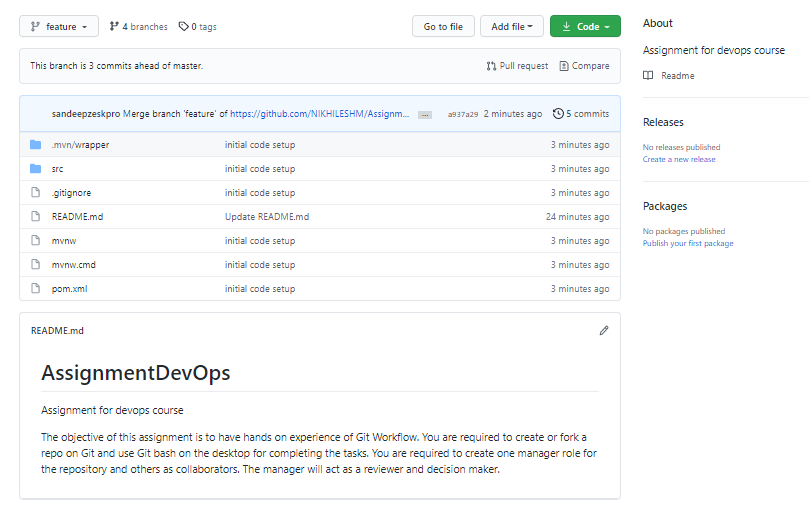
Connect it to origin



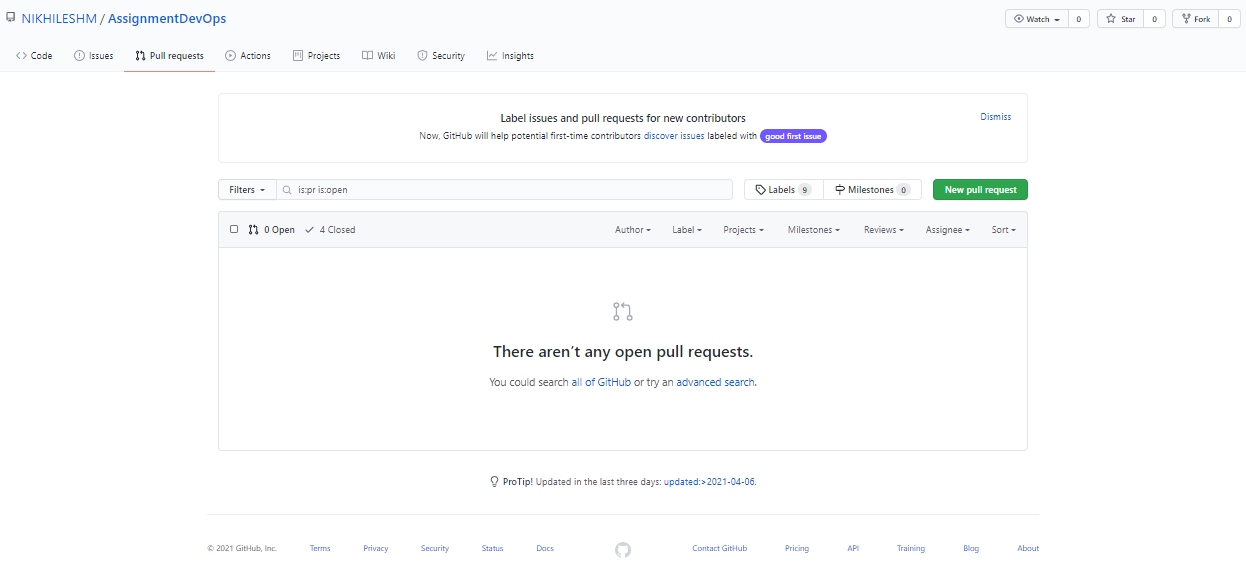
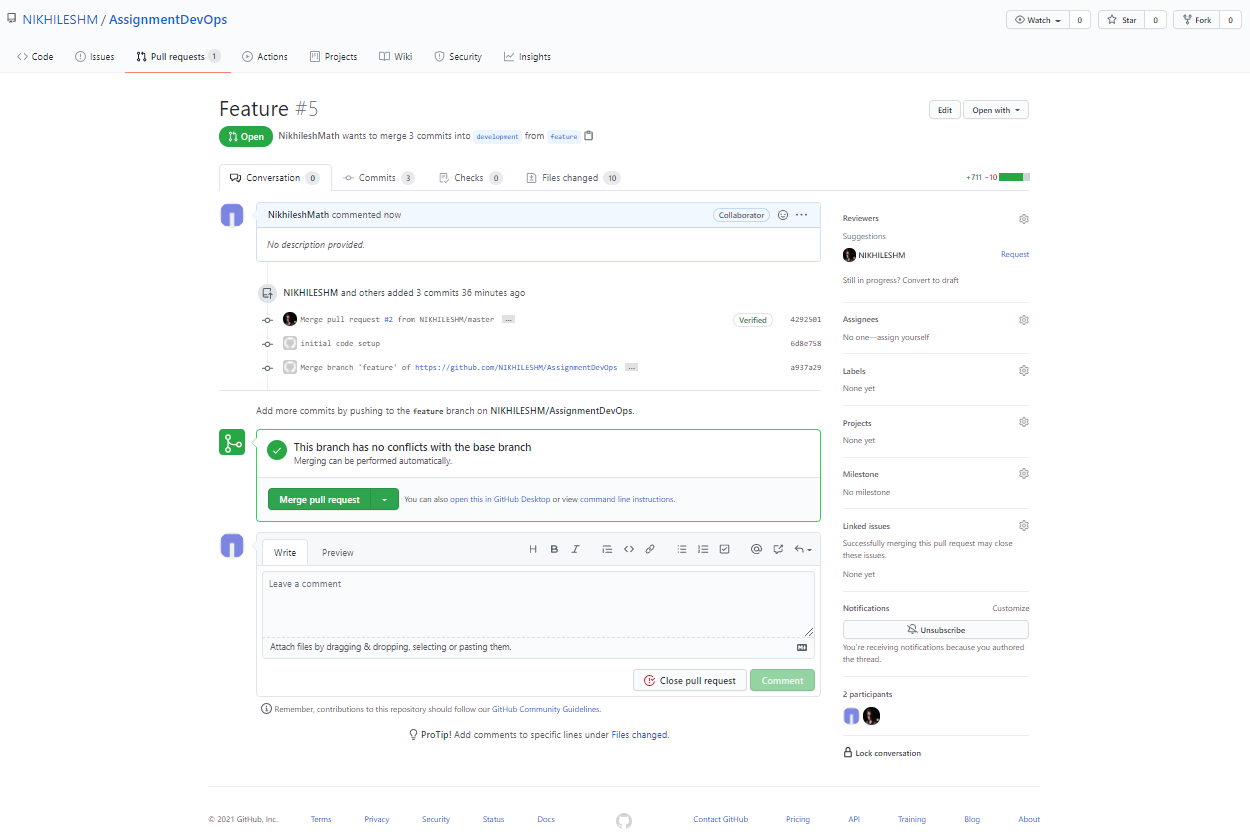
Add and push new code



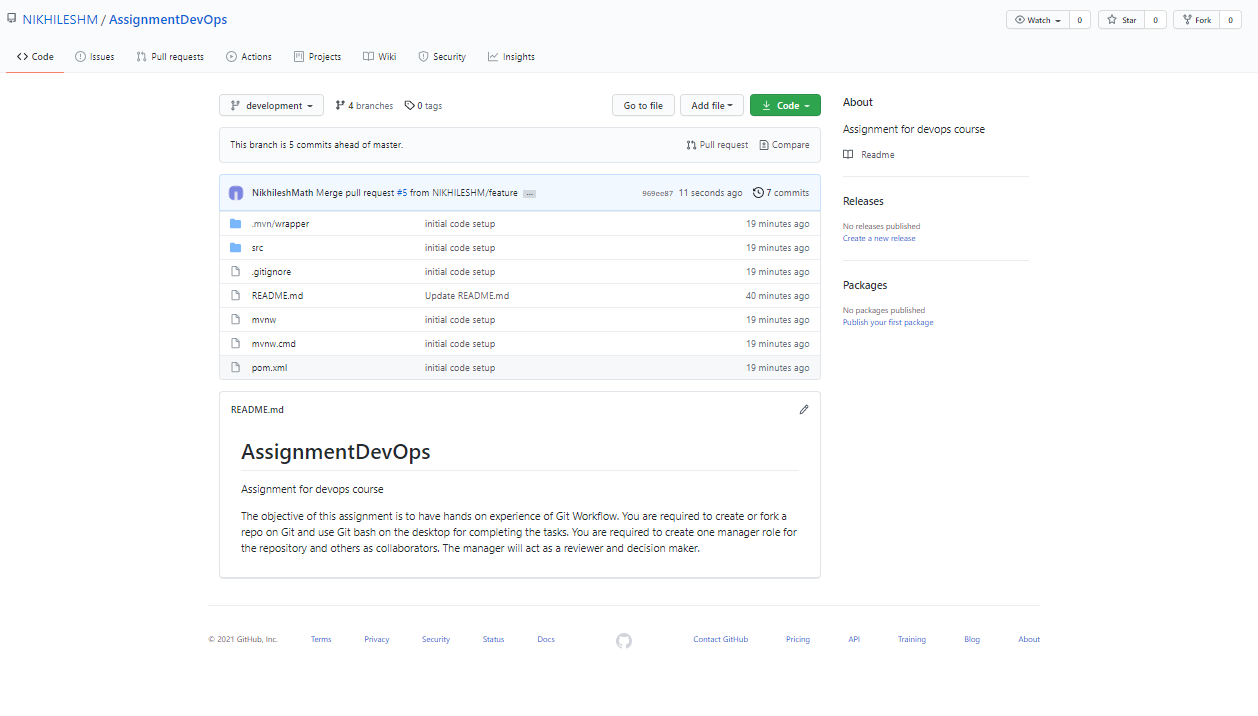
Code in remote branch –



## Step 6 – Create Pull Request

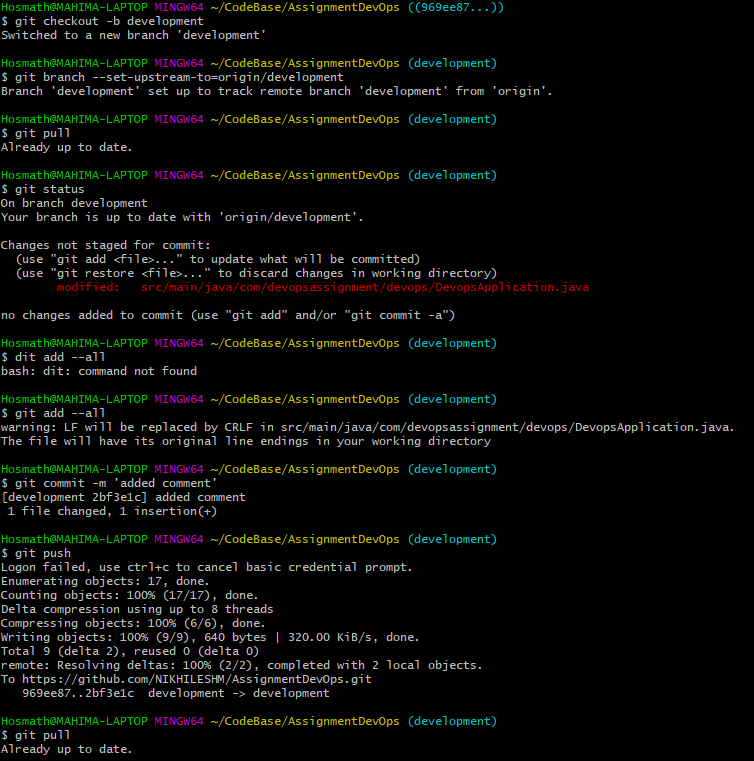
Merged into development –

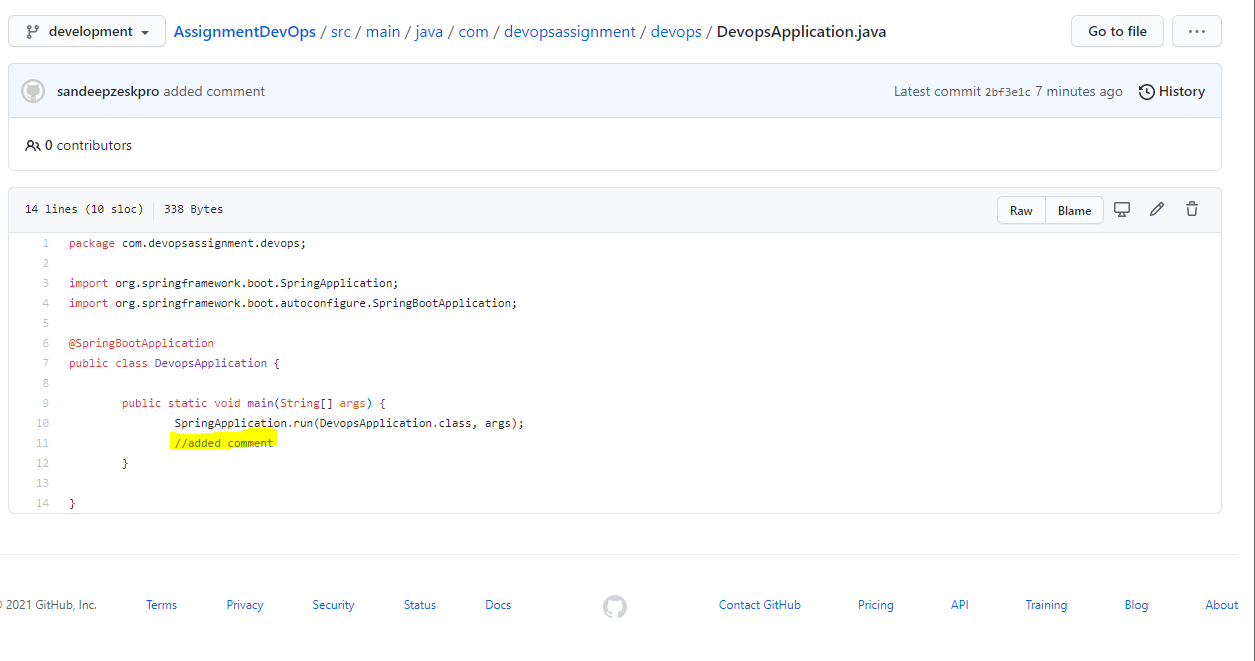


## Step 7 – Resolving collaborator conflict

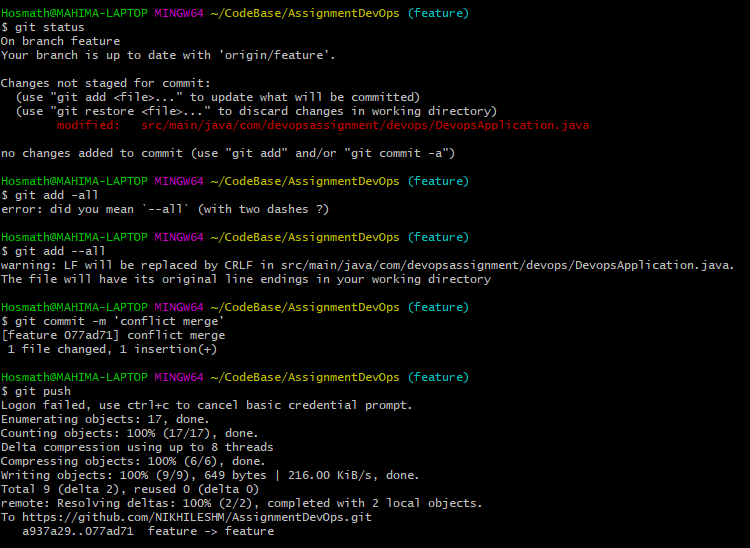
Added change in development branch from manager and added different change in same file in feature branch from collaborator. Will create PR and resolve conflicts

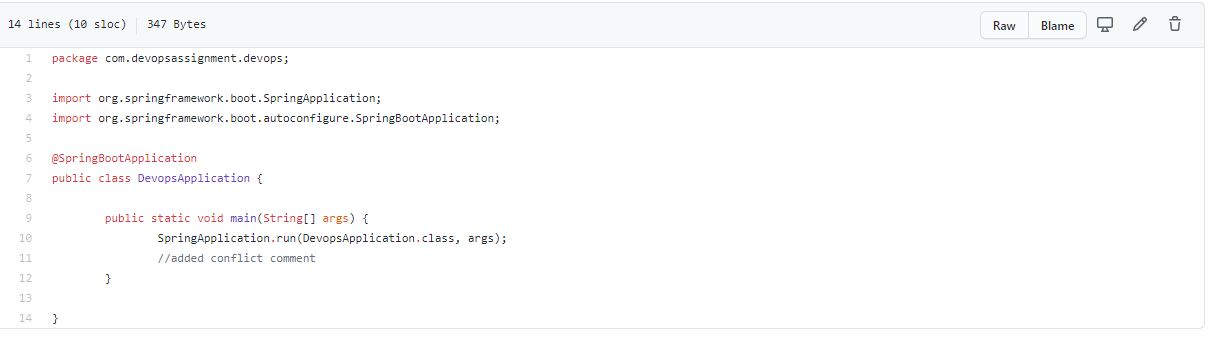
Added development branch change –





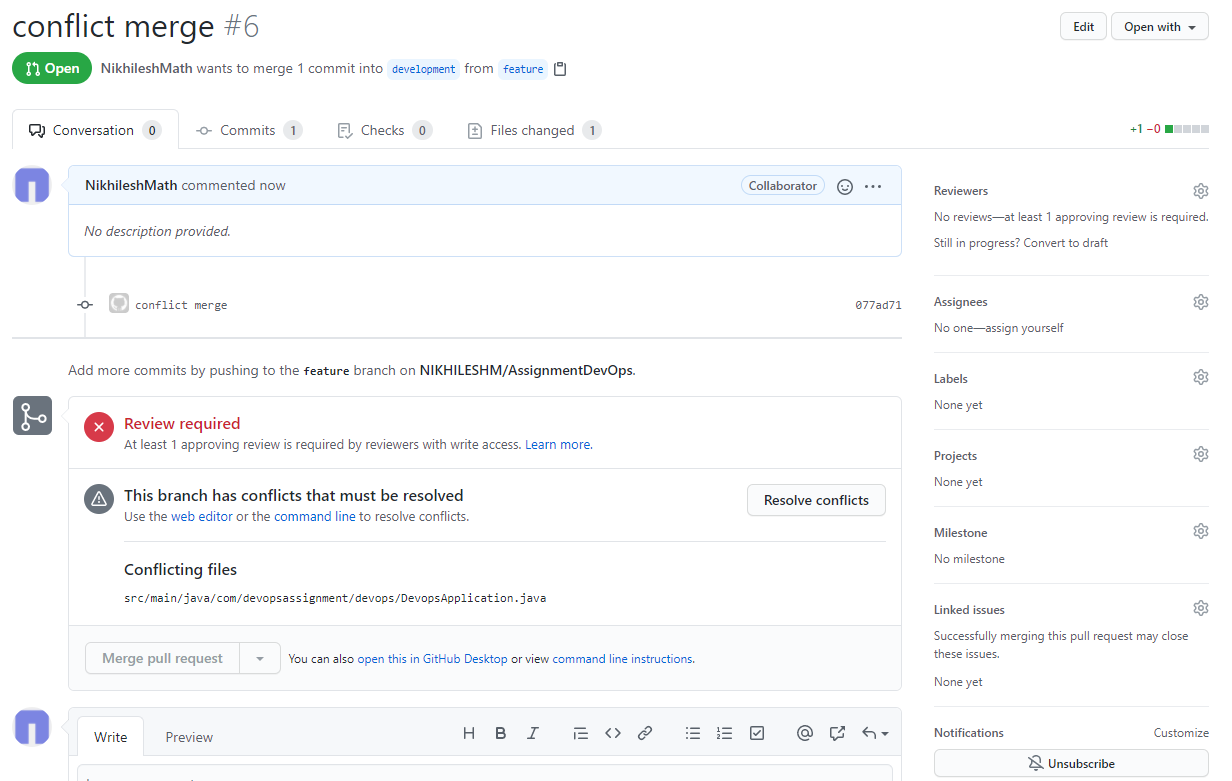
Added Feature Branch change from another collaborator



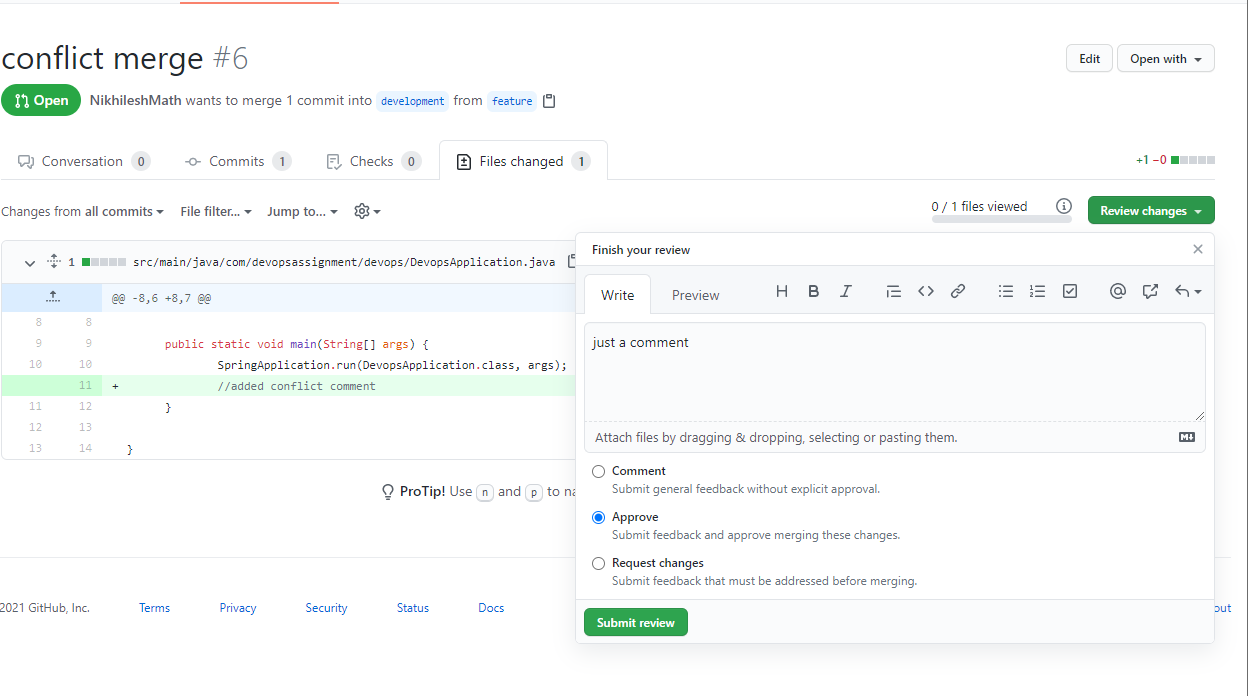


Create PR from Feature to dev to resolve conflicts

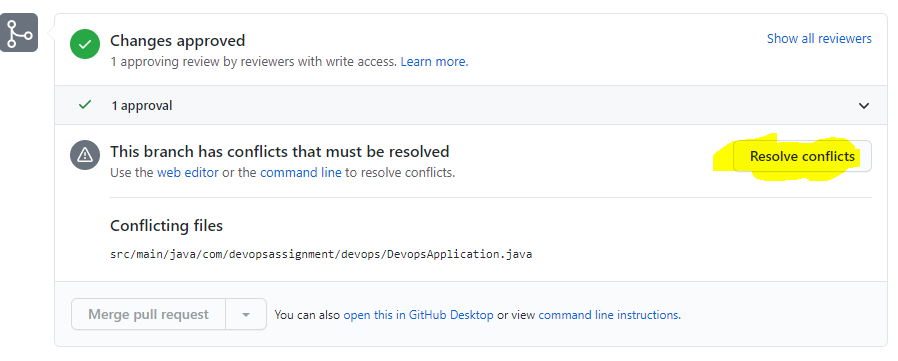
Collaborator created PR – Hence Review is required to merge into development –

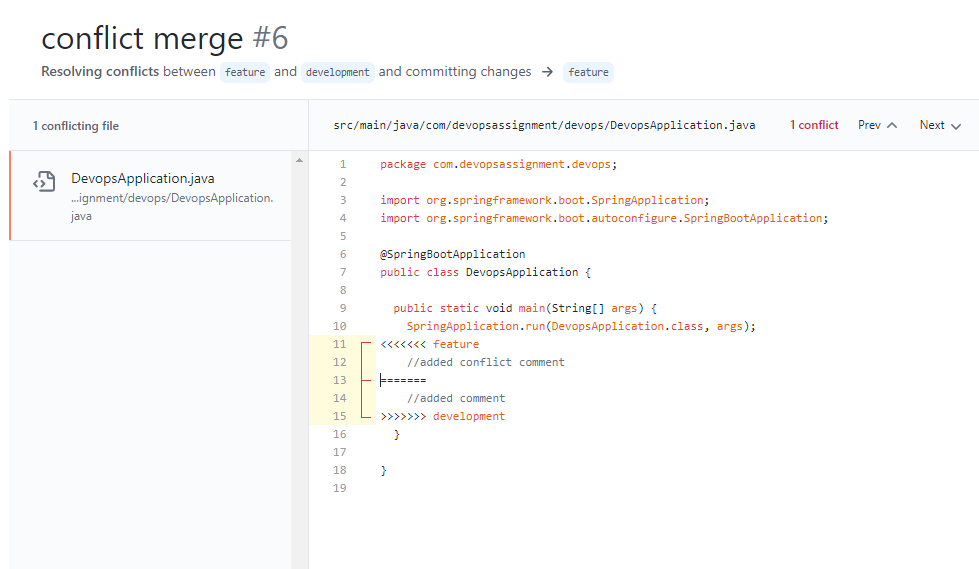


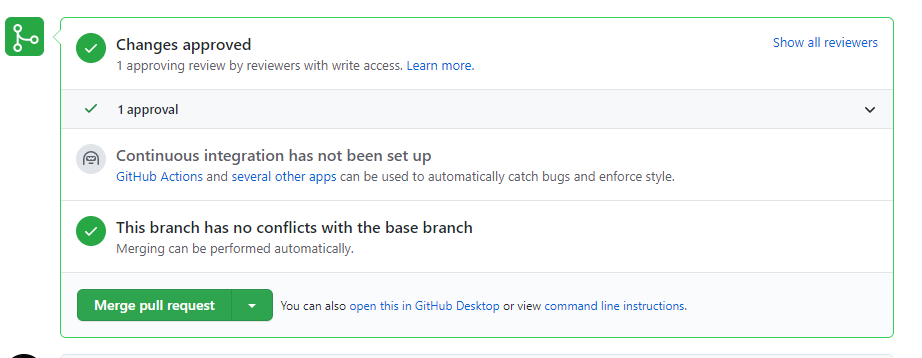
Adding review

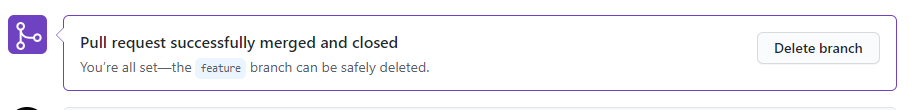


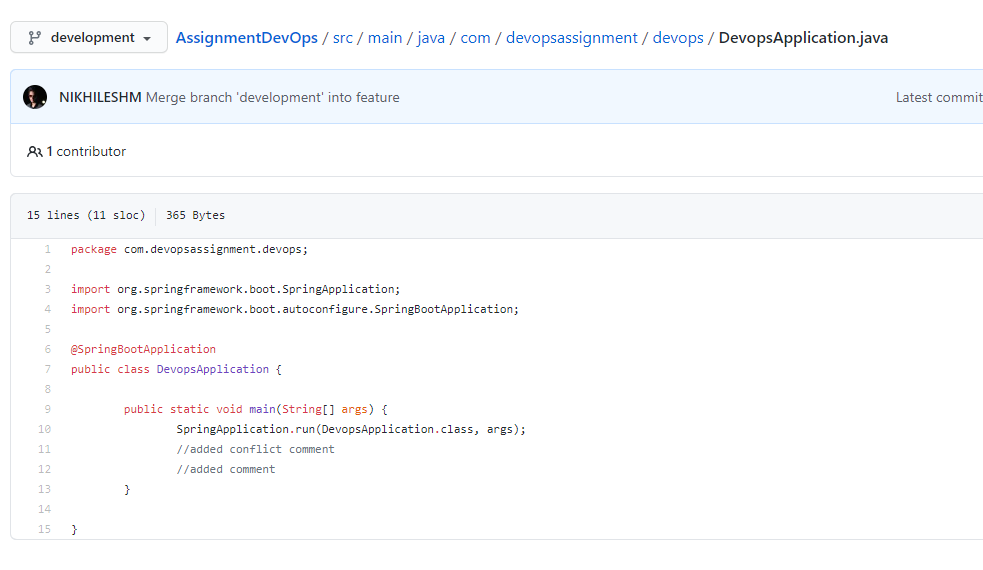
Resolve conflicts –



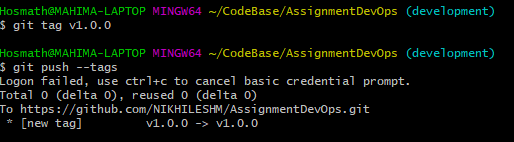


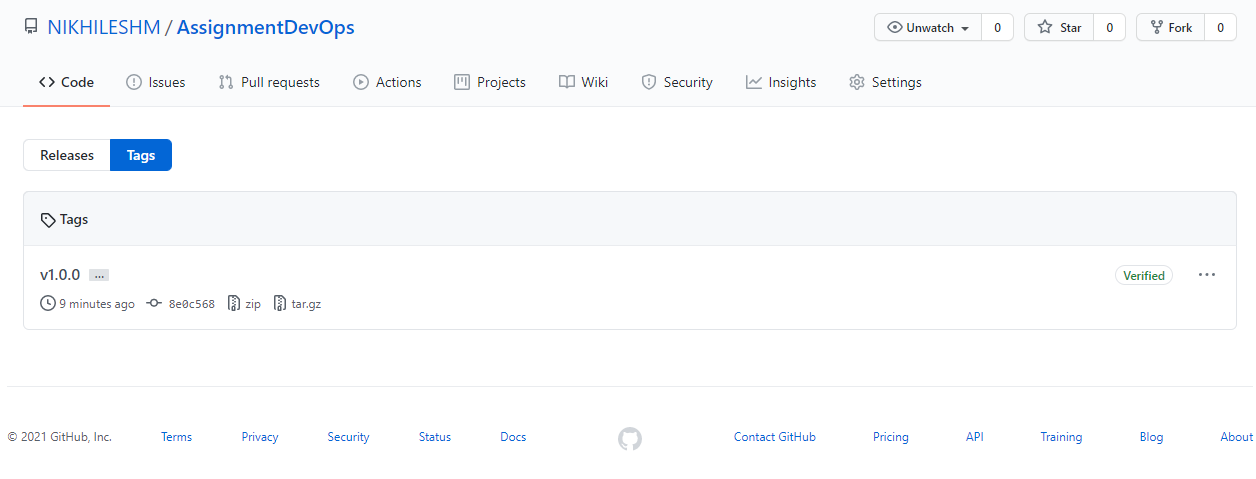




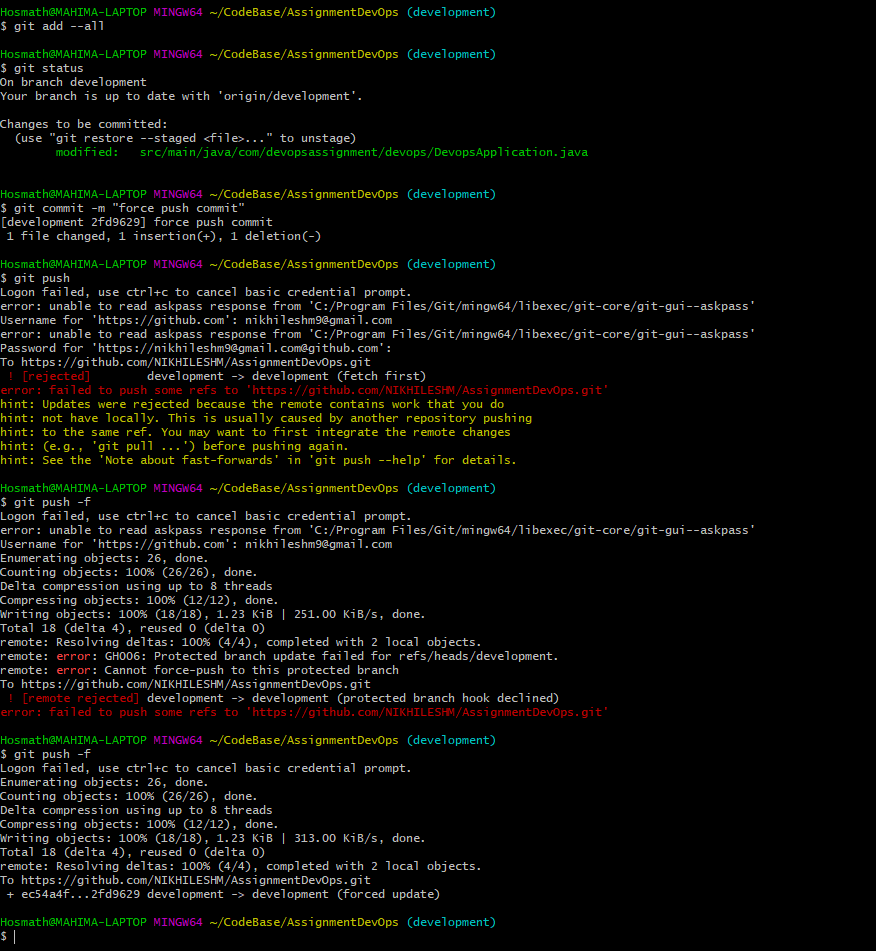


## Step 8 – Creating Tags

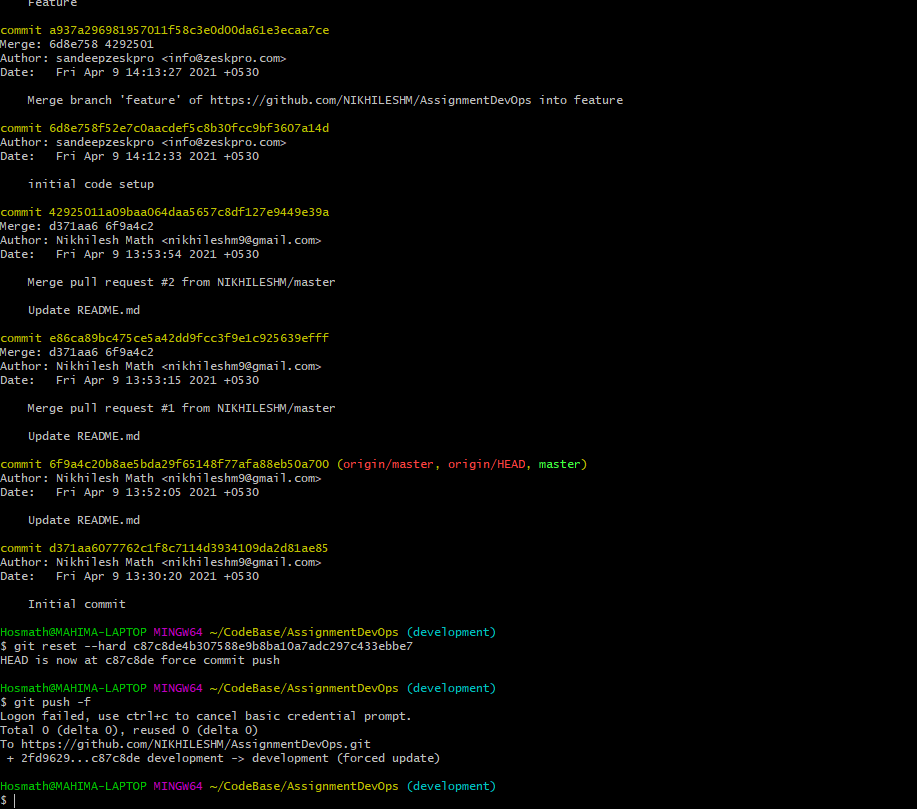




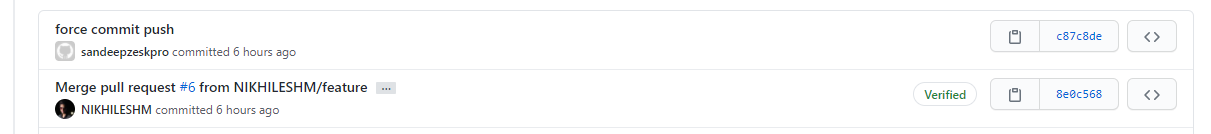
## Step 9 – Do a force push and then later reset the changes



I Reset to my previous commit and did another force push

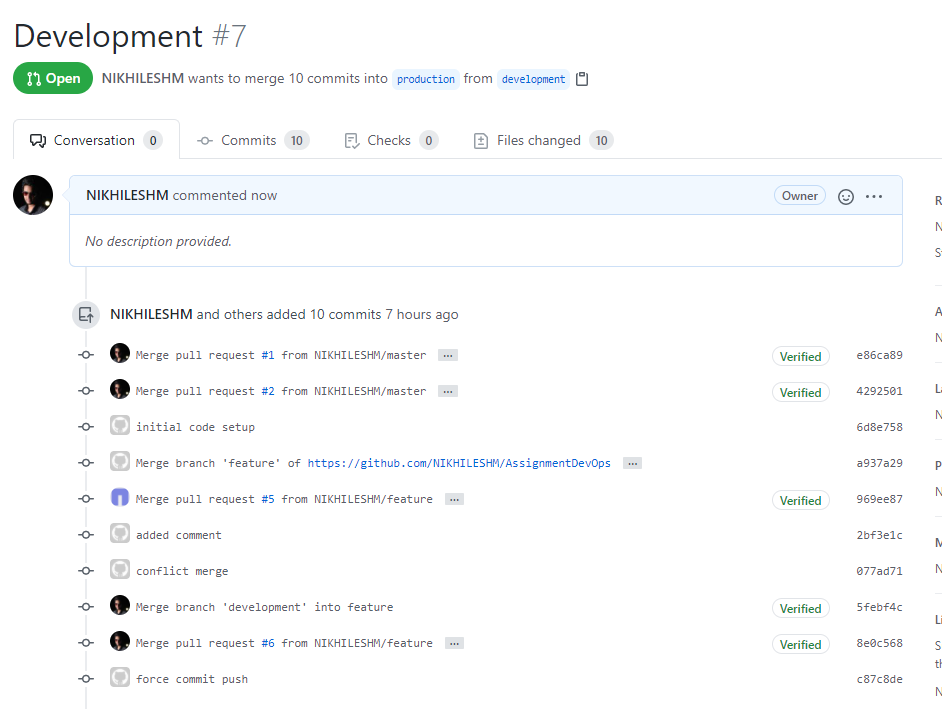


I reset the commit in remote

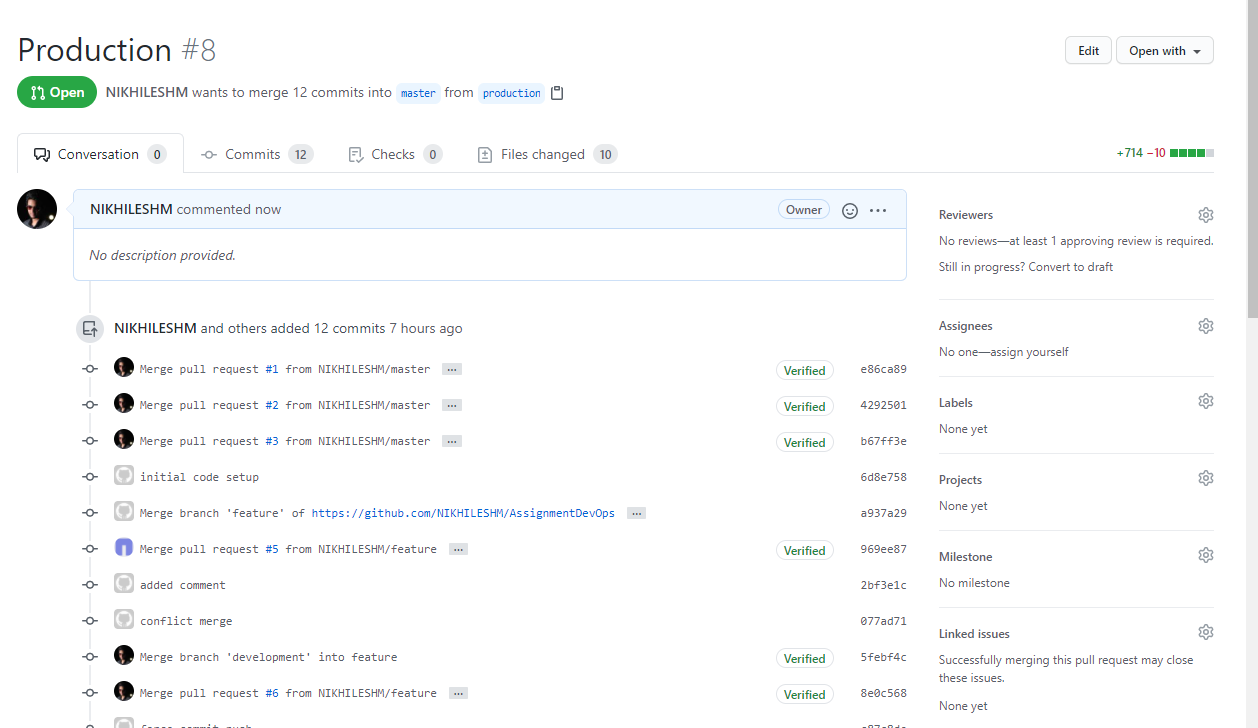


## Step 10 – Staging Development to Production

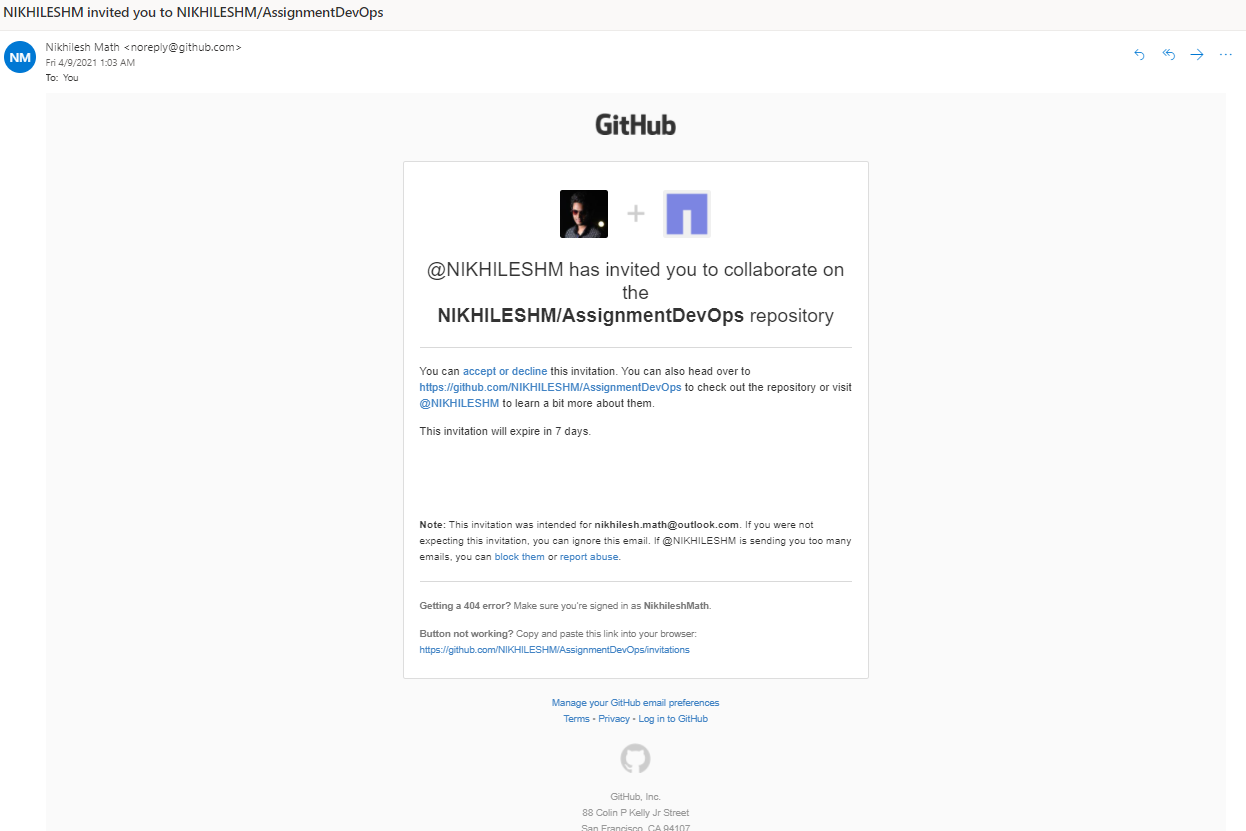
Here we do it using pull request from development to production



## Step 11 – released merging production to master branch



## Step 12 – Bonus Point Email notification to manager



## Importance of readme and git Ignore

Readme provides the user with the overview of the Git Repository and the code it contains. It is really important to provide all the details required to kick start and using this repo for a starter.

.gitignore files play a really important role where unwanted or intermediate/locally generated/setup code such as .mvn or nodemodules are ignored by the git so that only Important code is pushed into the remote