CASE STUDY- RESOLVING MERGE CONFLICTS

Problem Statement:

You work for Zendrix Software & Co. You have been assigned the task of updating the Master branch of their Git repository with all the features from the feature branches.

Following is the GitHub account, https://github.com/devops-intellipaat/merge-conflict.git

Consider,

- Feature1 branch to be a public branch
- Feature2 branch to be a private branch

The company relies on a monolithic architecture, and for now all the code resides in one file "main.c". The respective features have been added in the feature branches for main.c. Meanwhile, a security patch was made to the master branch, and now feature1 and feature2 branches are behind from master by 1 commit.

Following tasks have to be done:

1. Update Feature1 and Feature2 branch with the Security Patch
2. Apply changes of Feature1 and Feature2 branch on master
3. Finally push all the branches to GitHub

For Solving this, please fork the repository to your Github account and then work.

As a solution, please submit your GitHub's repository link.

1. Update Feature1 and Feature2 branch with the Security Patch

Switch to feature1 and feature2 branch, then merge the master branch security patch code.
Git the merge conflict issue, need to fix it by opening and editing the correct code.

```
ubuntu@ip-172-31-24-62:~/merge-conflict$ git status
On branch feature1
Your branch is ahead of 'origin/feature1' by 3 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
ubuntu@ip-172-31-24-62:~/merge-conflict$
```

```
ubuntu@ip-172-31-24-62:~/merge-conflict$ git checkout feature2
Branch 'feature2' set up to track remote branch 'feature2' from 'origin'.
Switched to a new branch 'feature2'
ubuntu@ip-172-31-24-62:~/merge-conflict$ git merge master
Auto-merging main.c
CONFLICT (content): Merge conflict in main.c
Automatic merge failed; fix conflicts and then commit the result.
ubuntu@ip-172-31-24-62:~/merge-conflict$ git status
On branch feature2
Your branch is up to date with 'origin/feature2'.

You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
        both modified:   main.c

no changes added to commit (use "git add" and/or "git commit -a")
ubuntu@ip-172-31-24-62:~/merge-conflict$
```

```
ubuntu@ip-172-31-24-62: ~/merge-conflict                                    —  □  ✕
ubuntu@ip-172-31-24-62:~/merge-conflict$ git add .
ubuntu@ip-172-31-24-62:~/merge-conflict$ git commit -m "Resolving the conflict"
[feature2 e962af1] Resolving the conflict
 Committer: Ubuntu <ubuntu@ip-172-31-24-62.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

ubuntu@ip-172-31-24-62:~/merge-conflict$ git sta
stage     stash     status
ubuntu@ip-172-31-24-62:~/merge-conflict$ git status
On branch feature2
Your branch is ahead of 'origin/feature2' by 3 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
ubuntu@ip-172-31-24-62:~/merge-conflict$ █
```
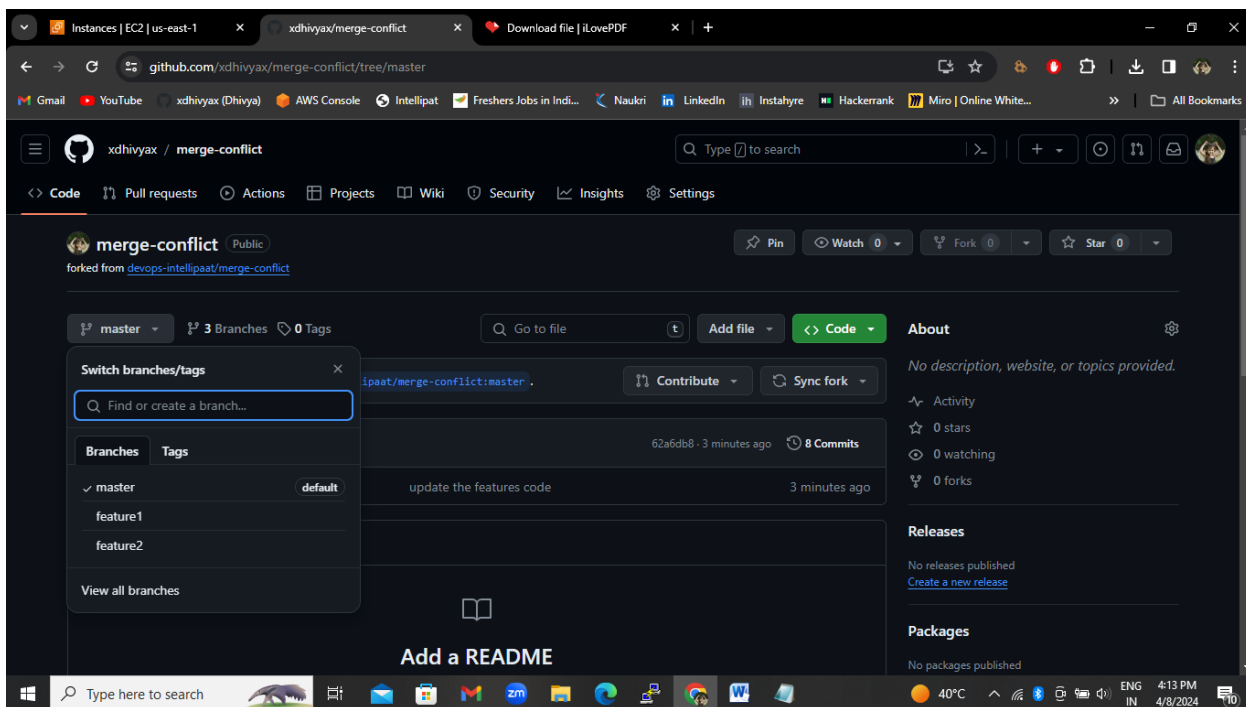
2. Apply changes of Feature1 and Feature2 branch on master
   Checkout to master branch and merge the feature1 and feature2 branch code.

3. Finally push all the branches to GitHub

GITHUB REPO LINK : https://github.com/xdhivyax/merge-conflict/tree/master