Module 3: Git And Jenkins

Assignment - 1 Solution

Problem Statement

Hooli Co. has put Harry and Kane to implement a new set of features in their feature software. Both started working separately, making their own copies of the same source code. Now, it has become difficult for them to track the changes they've made in the original code, and they are finding it difficult to merge their code together.

Action Items

- Create two separate branches from master
- Make changes in the same function of the source code in both the branches
- Merge branch1 into the master
- Try and merge branch2 into the master (merge conflict should arise)
- Install a merge tool of your choice and resolve the merge conflict using git mergetool command
- Link to download the code base

Solution

1. Initialize git repository in your source code directory

   Syntax: git init

```
edureka@master:~/Documents/projWeb$ git init
Initialized empty Git repository in /home/edureka/Documents/projWeb/.git/
```

2. Add and commit all the source files to the repository

   Syntax: git add .
   git commit -m 'message'

```
edureka@master:~/Documents/projWeb$ git add .
edureka@master:~/Documents/projWeb$ git commit -m "initial Commit"
[master (root-commit) 882b200] initial Commit
 9 files changed, 198 insertions(+)
 create mode 100644 css/site.css
 create mode 100644 fonts/segoeuil.ttf
 create mode 100644 img/cloneWhite.svg
 create mode 100644 img/deployWhite.svg
 create mode 100644 img/lightbulbWhite.svg
 create mode 100644 img/stackWhite.svg
 create mode 100644 img/successCloudNew.svg
 create mode 100644 img/tweetThis.svg
 create mode 100644 index.html
```

3. Now, Create a feature branch from the master

   Syntax: git checkout -b <branchName>

```
edureka@master:~/Documents/projWeb$ git checkout -b feature1
Switched to a new branch 'feature1'
```

4. Edit or Add some code/line inside one of the files

   Syntax: vi <fileName>
   Note: Here the file edited is index.html

```
</div>
    <div class="content-body">
        <div class="success-text">Success!</div>
        <div class="description line-1"> Azure DevOps Project has been successfully setup. Adding feature-1.</div>
        <div class="description line-2"> Your HTML app is up and running on Azure</div>
        <div class="next-steps-container">
            <div class="next-steps-header">Next up</div>
            <div class="next-steps-body">
```

5. Stage and commit changes inside the feature branch

   Syntax: git commit -a -m ‘message’

```
edureka@master:~/Documents/projWeb$ git commit -a -m 'Feature 1 added'
[feature1 6876a51] Feature 1 added
 1 file changed, 1 insertion(+), 1 deletion(-)
```

6. Move back to the master branch and create a new Feature branch

   Syntax: git checkout master
   git checkout -b feaure2

```
edureka@master:~/Documents/projWeb$ git checkout master
Switched to branch 'master'
edureka@master:~/Documents/projWeb$ git checkout -b feature2
Switched to a new branch 'feature2'
```

7. Now, Edit the same line(area) or code block you edited in the first branch

   Syntax: vi <fileName>

```
</div>
    <div class="content-body">
        <div class="success-text">Success!</div>
        <div class="description line-1"> Azure DevOps Project has been successfully setup. Adding feature 2</div>
        <div class="description line-2"> Your HTML app is up and running on Azure. Feature 2 continued</div>
        <div class="next-steps-container">
            <div class="next-steps-header">Next up</div>
            <div class="next-steps-body">
```

8. Stage and commit the changes to the new branch

   Syntax: git commit -a -m ‘message’

```
edureka@master:~/Documents/projWeb$ git commit -a -m 'Feature 2 added'
[feature2 a39a978] Feature 2 added
 1 file changed, 2 insertions(+), 2 deletions(-)
```

9. Now, checkout the master branch and merge it with any one of the new branches that were created

   Syntax: git merge <branchName>

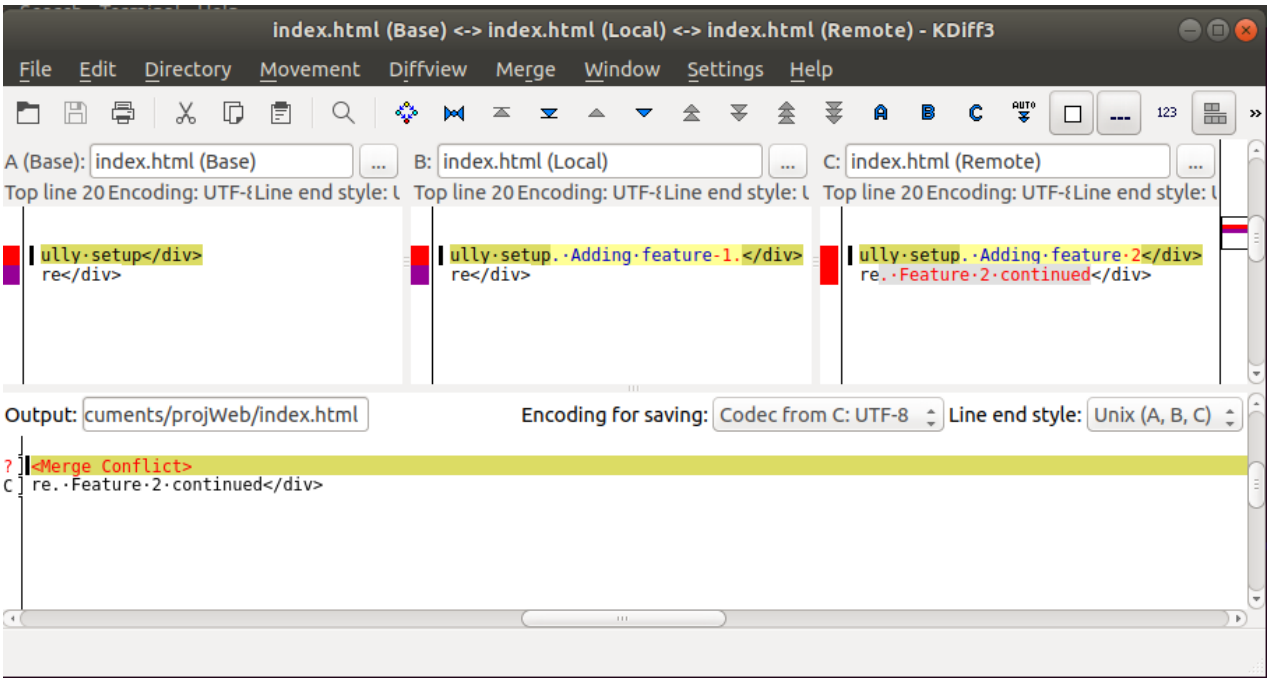
```
edureka@master:~/Documents/projWeb$ git checkout master
Switched to branch 'master'
edureka@master:~/Documents/projWeb$ git merge feature1
Updating 882b200..6876a51
Fast-forward
 index.html | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
edureka@master:~/Documents/projWeb$
```

10. On trying and merging the second branch into master, a merge conflict will arise

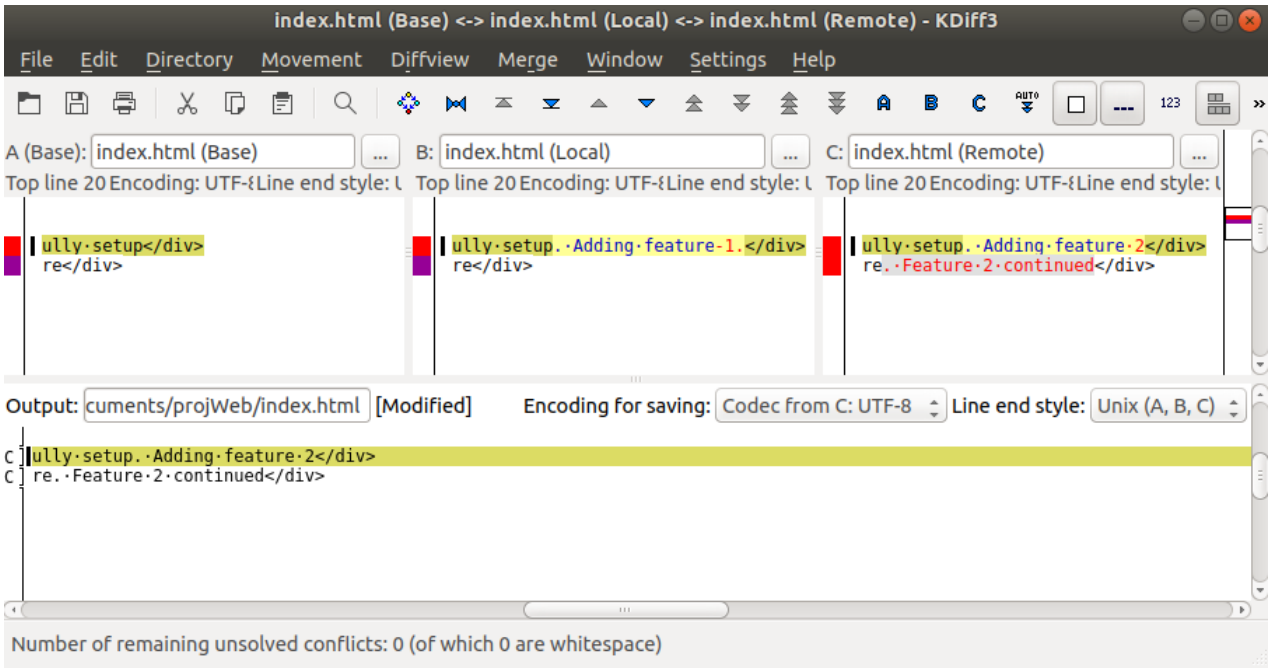
```
edureka@master:~/Documents/projWeb$ git merge feature2
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.
edureka@master:~/Documents/projWeb$
```

11. To solve the merge conflict check set a merge tool for git (kdiff3 is used here. If not installed, please install it using apt-get install kdiff3)

    Syntax: git mergetool



Choose and save one of the files



12. The merge conflict has now been solved