

Assignment 3: Feature Branches and Hotfixes

Create a 'hotfix' branch to fix an issue in the main code. Merge the 'hotfix' branch into 'main' ensuring that the issue is resolved.

A)

Here's how you can create a 'hotfix' branch to fix an issue in the main code, then merge the 'hotfix' branch into 'main' to resolve the issue:

1.Create a new 'hotfix' branch:

git checkout -b hotfix

This command creates a new branch named 'hotfix' and switches to it. Now you can make your fixes in this branch without affecting the main code.

2.Make the necessary changes to fix the issue in the 'hotfix' branch:

Edit the files in your project directory to fix the issue.

3.Add the changes to the staging area:

git add .

4.Commit the changes:

git commit -m "Fixed the issue in the hotfix branch"

Replace "Fixed the issue in the hotfix branch" with a descriptive commit message.

5.Switch back to the 'main' branch:

git checkout main

6.Merge the 'hotfix' branch into 'main':

git merge hotfix

This command merges the changes from the 'hotfix' branch into the 'main' branch. If there are any conflicts, you'll need to resolve them before completing the merge.

7.Verify that the issue is resolved:

Test your code to ensure that the issue has been fixed in the 'main' branch.

8.Commit the merge (if necessary):

If the merge resulted in any conflicts that you resolved manually, you'll need to commit the merge:

git commit -m "Merged hotfix into main"

Replace "Merged hotfix into main" with an appropriate commit message.

Now, the issue should be resolved in the 'main' branch after merging the 'hotfix' branch.