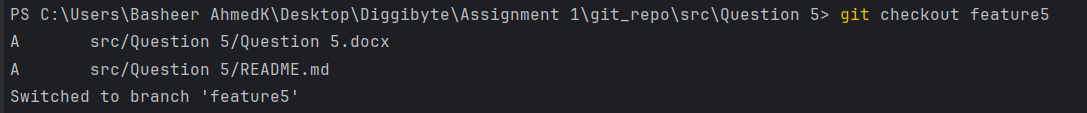
**Question 5**

Step 1: Step 1: Create a feature branch.



Step 2: Switch to the new branch.



open the file and make some changes to it.

A screen shot of a computer

Description automatically generated

Add and commit the changes to the new branch.

A screen shot of a computer

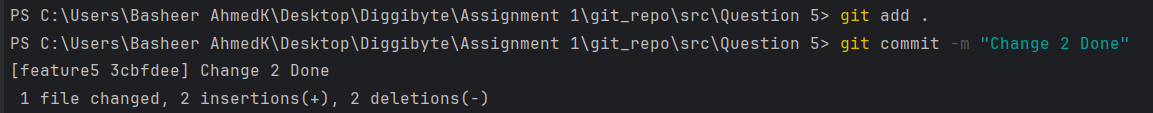
Description automatically generated

open the same file and make some changes to it.

A screen shot of a computer

Description automatically generated

Add and commit the changes to the new branch.

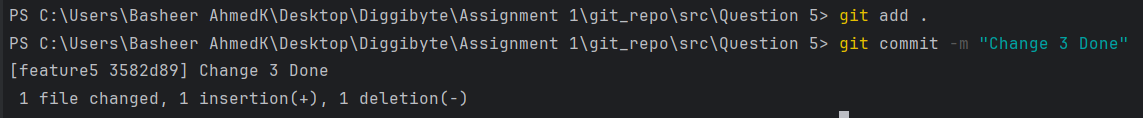


open the same file and make some changes to it.

A screen shot of a computer

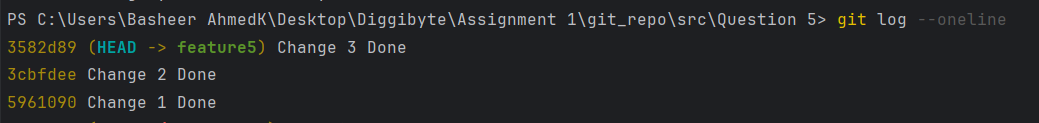
Description automatically generated

Add and commit the changes to the new branch.

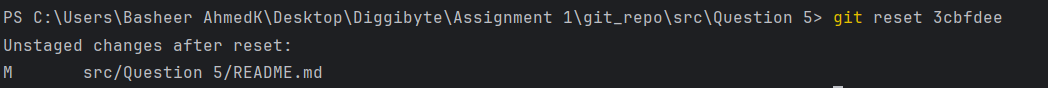


Step 3: Use the "git log" command to view the commit history and identify the commit to which you want to reset.

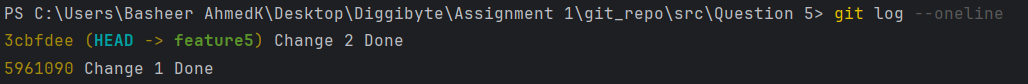
Here I want “Change 2 Done” that I want to reset and its hash is “3cbfdee”



Step 4: Use the "git reset" command followed by the desired reset type and the commit hash

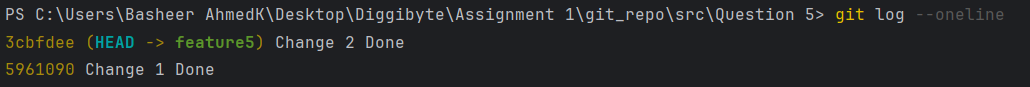


Step 5: Verify that the reset was successful by using the "git log" command again.



Step 6: Use the "git log" command to view the commit history and identify the commit that you want to reverse.

Here I want to revert “Change 1 Done” and its hash is “5961090”

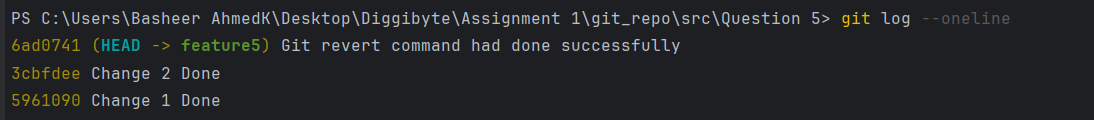


Step 7: Use the "git revert" command followed by the commit hash or reference to which you want to revert. (Hint: git revert <commit hash>)

A computer screen shot of a black screen

Description automatically generated

Step 8: Verify that the revert was successful by using the "git log" command again.



Note: Identify the difference between git log after git reset and git revert.