**Q1. What is cherry-pick? Explain the use case of it also write steps to do it.**

**Steps:**

1: commit changes in a branch A that will genrate a specific id <commit\_hash>

2: create new branch

git checkout -b <branch\_name>

3: to pull specific commit use cherry-pick

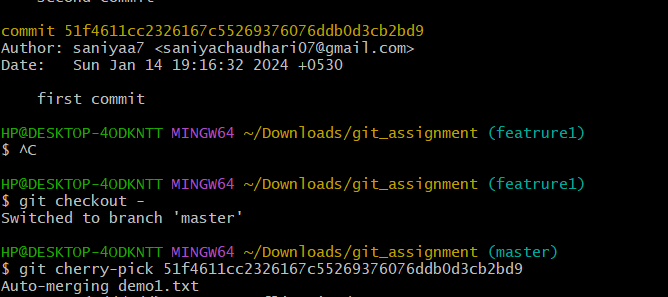
git cherry-pick <commit\_hash>

4:Now that commit is pull in local sysytem

5:to push in remote repository

git add .

git push origin <branch\_name>



**Q2. How to set multiple remote repositories for the same project? Explain the use case. you should have a working demo of it.**

**Steps:**

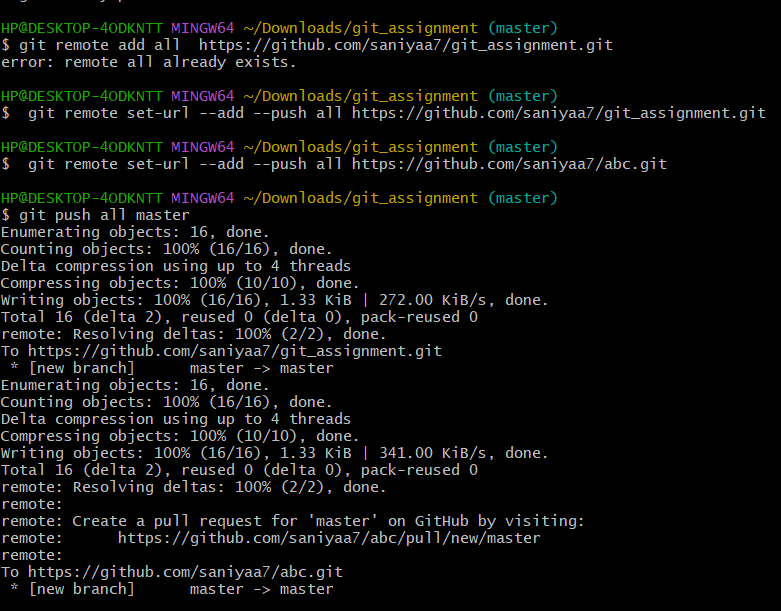
1.git remote add all REMOTE\_URL\_A.

2.git remote set-url --add --push all REMOTE\_URL\_A.

3. git remote set-url --add --push all REMOTE\_URL\_B.

Push a branch to all the remotes with

4. git push all BRANCH



**Q3. How to combine two commits into one commit? For Example, you have 4 different commits C1, C2, C3, and C4 and you to combine C2 and C3 into one single commit.**

**Steps:**

1.Run git rebase -i HEAD~4 in the terminal.

2.Change pick to edit for both C2 and C3.

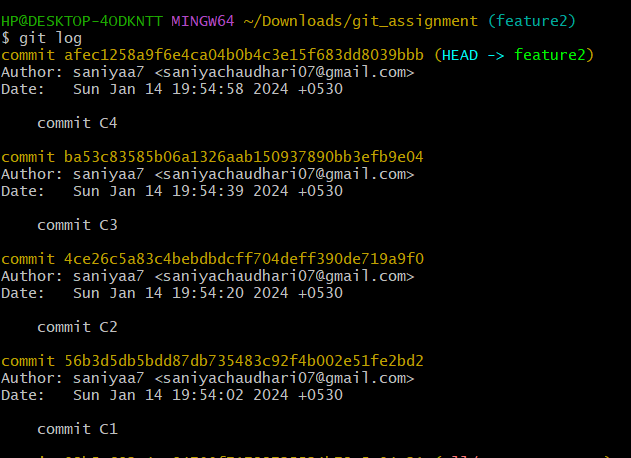
For each commit, run:

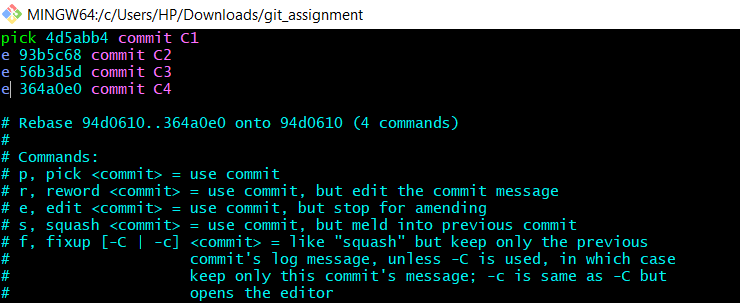
3. git reset HEAD~1

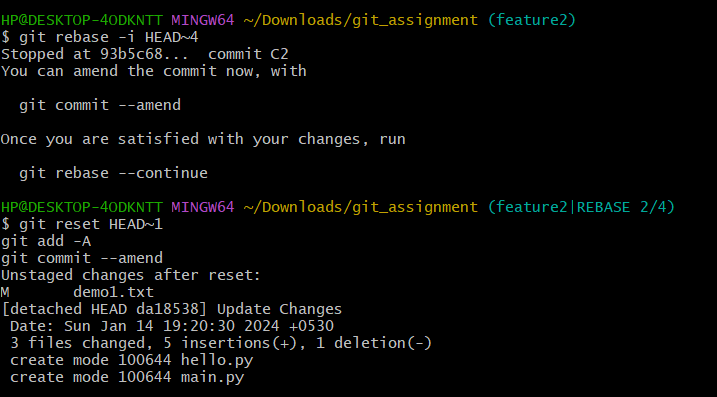
git add -A

git commit --amend

Complete the rebase with git rebase --continue.

This process combines C2 and C3 into one commit.



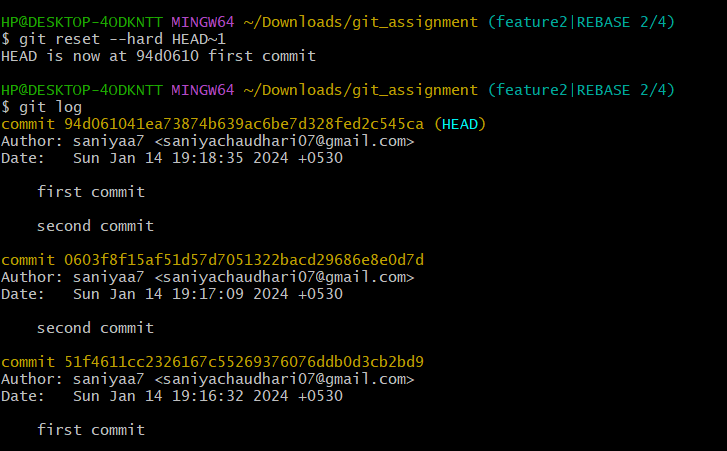


**Q4.How to delete one commit or multiple commits permanently Write command for it.**

**Steps:**

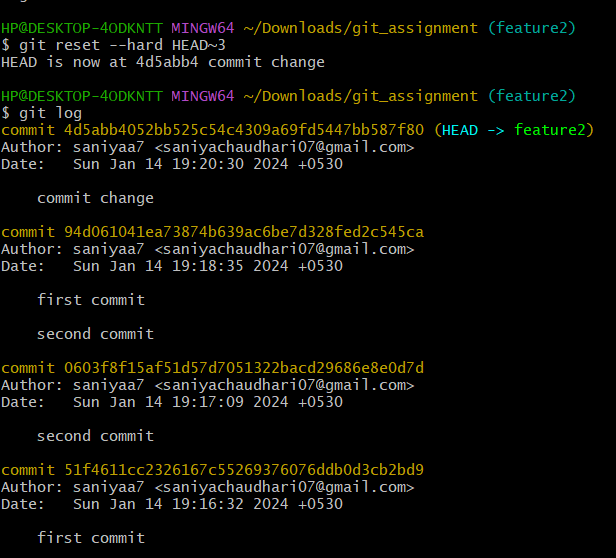
1.to remove the last commit from your Git history

git reset --hard HEAD~1



2. Delete Multiple Commits:

git rebase --hard HEAD~3

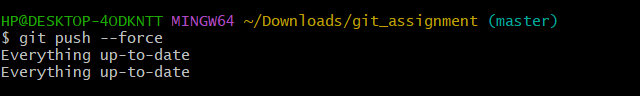


**Q.Difference between --force push and --force-with-lease, You have to show the use of both, also you have to show in what scenario --force-with-lease is used?**

**Steps:**

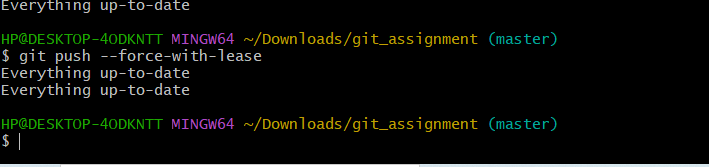
1)git push --force:

This command will still push if there is extra commit on remote branch



2)git push --force-with-lease

This command will not let you push if there is extra commit on remote branch



**Q5.How to un stage changes from the staged area. write command for it.**

git reset --soft HEAD~1

