



Placement Empowerment Program
Cloud Computing and DevOps Centre

100 Days of DevOps Challenge

Day 32 : Git Rebase

Name: Monisha J R

Department: CSE



Git Rebase

Objective :

The goal is to rebase a developer's **feature** branch onto the latest changes from the **master** branch in a Git repository, ensuring no new merge commits are created. You'll then push the updated branch.

Steps Performed :


- **Clone the repository:** The repository is located at /opt/news.git and needs to be cloned to /usr/src/kodekloudrepos on the storage server in Stratos DC.
 - `git clone /opt/news.git /usr/src/kodekloudrepos`
- **Change to the repository directory:** Navigate into the newly cloned repository directory.
 - `cd /usr/src/kodekloudrepos`
- **Checkout the feature branch:** Switch to the feature branch where the developer's work is located.
 - `git checkout feature`
- **Rebase the feature branch:** Rebase the current feature branch onto the master branch. This command will apply the feature branch's commits on top of the master branch's latest commit, ensuring a clean history without a merge commit.
`git rebase master`
- **Push the changes:** After the rebase is complete, push the updated feature branch to the remote repository. Since you've rewritten the history with the rebase, you'll need to use the --force flag (or --force-with-lease for a safer push).
 - `git push -force`
- **Verify:** The task is now complete. You can confirm the branch has been updated by checking the git log on the remote server.

End

00:13

Terminal 1

Stop Lab



You're done!

CONGRATULATIONS!!!!

You have successfully completed the challenge. Results have been saved. Ref ID: 6807867b399a2462b6cc6683

That might have been an easy task for you. But don't worry. Depending on the tasks you complete, you will get more difficult and advanced tasks assigned in the future.

You can also view your results in your dashboard under the "Active Practice" page.


```
e3e1b33 initial commit
[root@ststor01 news]# git push
fatal: The current branch feature has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin feature

To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetupRemote' in 'git help config'.


[root@ststor01 news]# git push -u origin feature
To /opt/news.git
+ a66ab73...d8381f5 feature -> feature (forced update)
branch 'feature' set up to track 'origin/feature'.
[root@ststor01 news]#
```

Feedback





Congrats!


You have completed the task successfully.




100 Days of DevOps

Day 32: Git Rebase 

80 

800 

 Share your success card

