

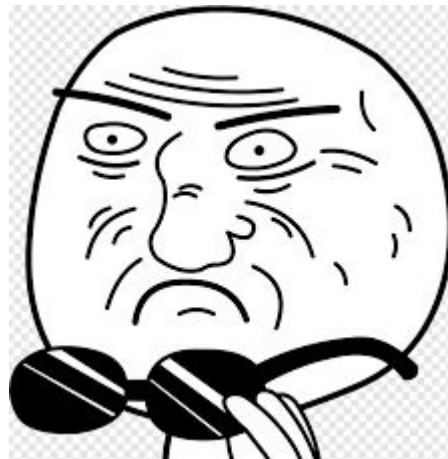


Advance **GIT** Commands



What are we going to talk about?

1. Reseting head changes using `reset`.
2. Editing recent commit using `--ammend`.
3. Saving changes without committing using `stash` and `pop`.
4. Integrating remote changes using `--rebase`.
5. Renaming local and remote branch.
6. Applying commit onto other branch using `cherry-pick`.
7. Recovering deleted local branch using `git reflog`.
8. Creating downloadable package using `git tag`.



TODO:

1. Setting up ssh keys.
2. Force pushing using `git push --force`.
3. Fetching only single branch from remote repository using `git fetch`.



1. Resetting head changes using *'git reset'*

Git reset is a powerful command that is used to undo local changes to the state of a Git repo.



See also:

- `git reset [--soft, --mixed] HEAD.`
- `git checkout [commit-hash]`
- `git revert HEAD.`

Editing most recent commit using `git --amend`

Git amend is to make modifications to the most recent commit.



How do the pros do it?

Integrating remote changes using `--rebase`

Git rebase

- Git rebase is one of two Git utilities that specializes in integrating changes from one branch onto another.





Renaming local and remote branch

In renaming local branch,

```
$ git branch -m new-name
```

When you are on a different branch:

```
$ Git branch -m old-name new-name
```



Renaming local and remote branch (cont'd)

Delete the old-name remote branch and push the new-name local branch

```
$ git push origin :old-name new-name
```

Reset the upstream branch for the new-name local branch.

- `git push origin -u new-name`



Deleting a local and remote branch

Delete a Local GIT branch

```
$ git branch -d branch_name
```

```
$ git branch -D branch_name
```

Delete a remote GIT branch

```
$ git push <remote_name> --delete <branch_name>
```

Applying commit onto another branch using cherry-pick

Cherry picking in Git means to choose a commit from one branch and apply it onto another.

- This is in contrast with other ways such as `merge` and `rebase` which normally apply many commits onto another branch.



Recovering deleted local branch using git reflog

- Git reflog tracks any updates of branches, it serves as reference a storage





Creating a downloadable package using `git tag`

Git tagging is generally used to capture a point in history that is used for a marked version release (i.e. v1.0.1). *A tag is like a branch that doesn't change.*