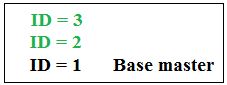
How to revert the second commit

Purpose: Delete the second commit in the list.

Use case: There are 3 commits – ID #1 is base commit.



If you want to delete commit #3, simply use: git reset --hard HEAD

But you want to keep commit #3, and only delete commit #2. In that case, use below commands.

How?

|  |
| --- |
| $ git rebase –i 1 (the ID of base commit)  *# delete commit #2 in editor*  *# after resolve conflicts*  $ git rebase --continue  $ git push origin … |

This will open an editor and show a list of all commits since the base commit (#2, #3).

Simply remove the line with the offending commit (#2) (vi: delete current line = dd). Then save and close the editor (:wq!)

Resolve any conflicts if there are any. Then git rebase --continue.

Now git log to check, commit #2 has been removed. Push your commit #3 to remote.

Source:

<http://christoph.ruegg.name/blog/git-howto-revert-a-commit-already-pushed-to-a-remote-reposit.html>