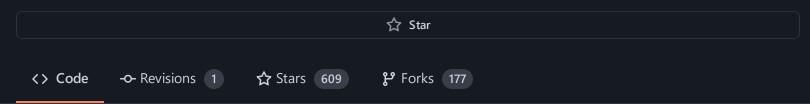
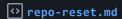
heiswayi / repo-reset.md

Created 5 years ago • Report abuse



GitHub - Delete commits history with git commands



First Method

Deleting the .git folder may cause problems in our git repository. If we want to delete all of our commits history, but keep the code in its current state, try this:

```
# Check out to a temporary branch:
git checkout --orphan TEMP_BRANCH

# Add all the files:
git add -A

# Commit the changes:
git commit -am "Initial commit"

# Delete the old branch:
git branch -D master

# Rename the temporary branch to master:
git branch -m master

# Finally, force update to our repository:
git push -f origin master
```

This will not keep our old commits history around. **But if this doesn't work, try the next method below.**

Second Method

```
# Clone the project, e.g. `myproject` is my project repository:
git clone https://github/heiswayi/myproject.git

# Since all of the commits history are in the `.git` folder, we have to remove it:
cd myproject

# And delete the `.git` folder:
```

```
git rm -rf .git

# Now, re-initialize the repository:
git init
git remote add origin https://github.com/heiswayi/myproject.git
git remote -v

# Add all the files and commit the changes:
git add --all
git commit -am "Initial commit"

# Force push update to the master branch of our project repository:
git push -f origin master
```

NOTE: You might need to provide the credentials for your GitHub account.



Thank you!

prageet commented on Sep 23, 2017

Thank you.

BrianKSFDO commented on Sep 29, 2017

First method worked, thanks for sharing!

🧰 jayeshdshah commented on Oct 12, 2017

while executing second method, it shown below error message pathspec '.git' did not match any files

xiongmai commented on Oct 18, 2017

yay thanks! I used the first method.

Rob-van-B commented on Oct 24, 2017

great!
TomHaii commented on Nov 8, 2017
thanks
he-dev commented on Nov 19, 2017
Thank you! The first method worked for me ;-)
iasonmeurer commented on Dec 4, 2017
For those using the second method, remove 'git' from the beginning of the rm command.
And delete the .git folder:
rm -rf .git
kirkvicent commented on Dec 14, 2017
多谢多谢 from china
Anihilakos commented on Dec 15, 2017
thank you!
maxdonetsk commented on Dec 27, 2017
Works like a charm. Thanks!
Zibri commented on Dec 28, 2017
error: git rm -rf .git >>> rm -rf .git



mohammadmahdikarimi commented on Feb 14, 2018

I am completely new and it worked for me. Thanks

Gabboxl commented on Apr 24, 2018

Omg, Thanks bro!

FadeMind commented on Apr 28, 2018

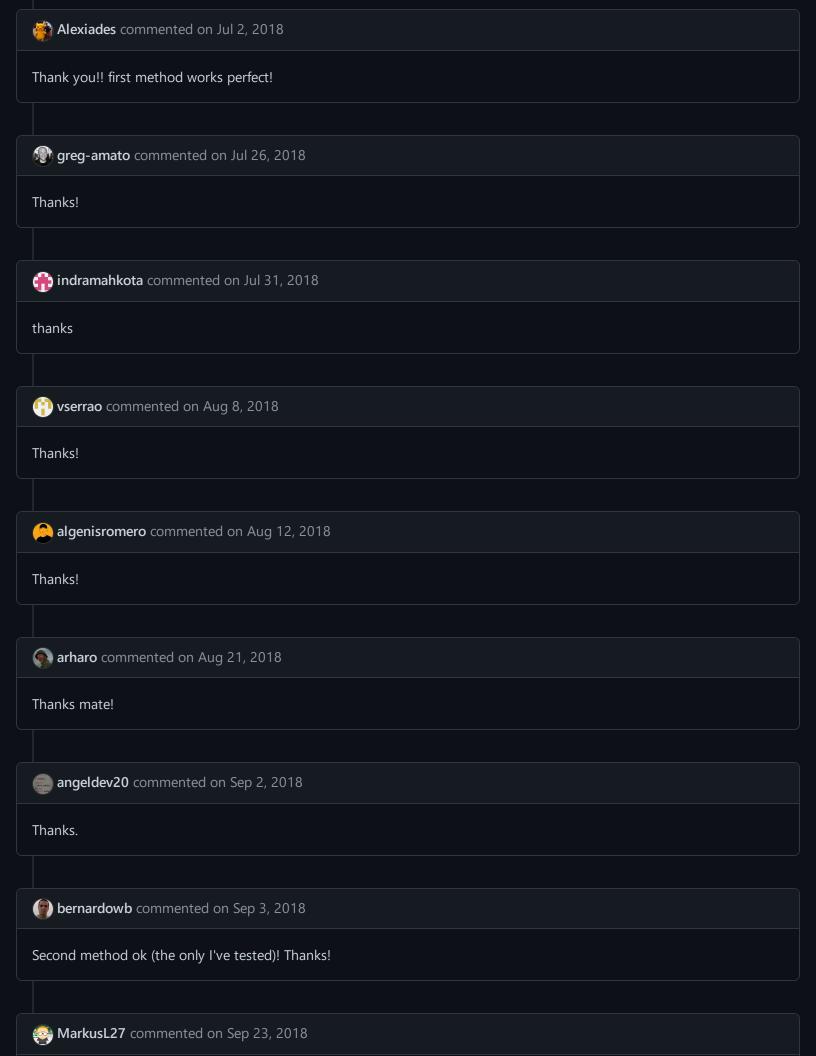
```
rm -rf .git
git init
git remote add origin git@github.com:JohnDoe/foobar.git
git remote -v
git add --all
git commit -am "Initial commit"
git push -f origin master
```

maxlefou commented on Jun 15, 2018 • edited 🔻

First method didn't worked on my private git server. remote: error: denying non-fast-forward refs/heads/master (you should pull first)

lucasbuetto commented on Jun 19, 2018

Thanks Bro!



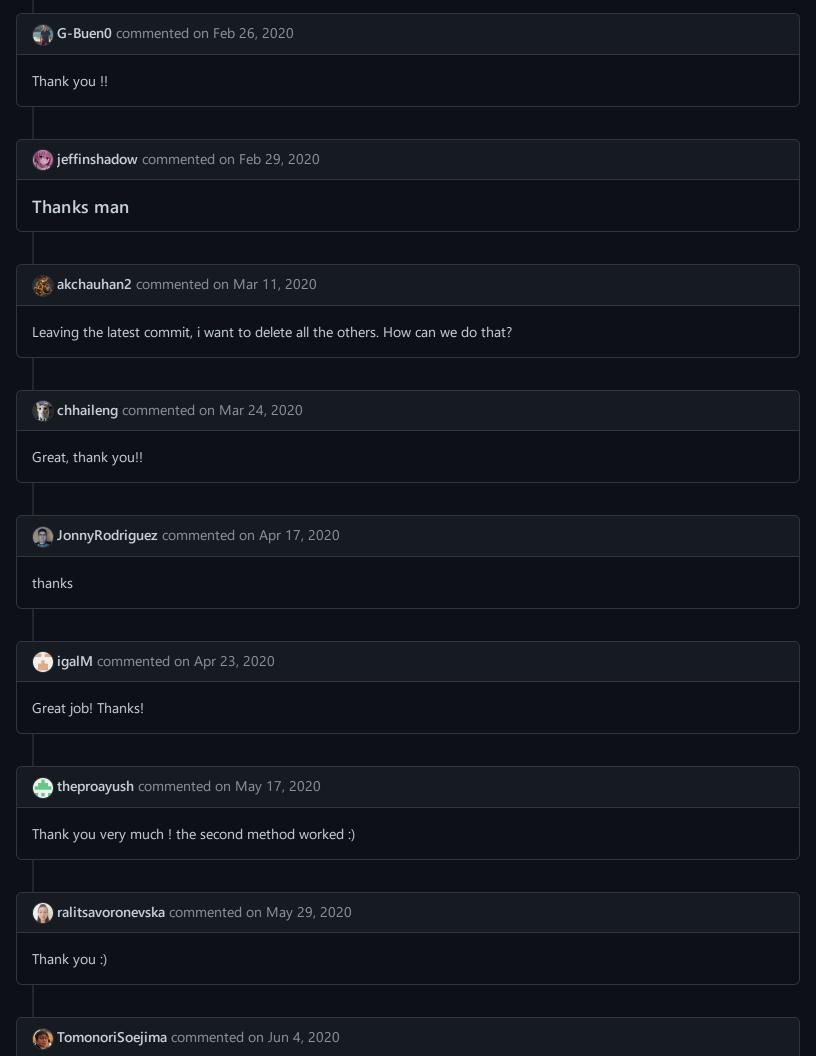
Thanks! Works perfect!
Paltonic commented on Sep 23, 2018
Thanks Man
ShinekaBarnett commented on Oct 17, 2018
Thanks!
DianaLauraSaavedra commented on Oct 17, 2018
donde se hace todo ese metodo
deezhizyu commented on Oct 21, 2018
Thanks you!
WiliTest commented on Oct 21, 2018 • edited →
thanks you but please correct the second method:
And delete the .git folder:
git rm -rf .git
rm -rf .git
(for windows: at the begining create a new folder, open a git bash inside: through windows context menu)
EricWu23 commented on Nov 1, 2018
Brilliant! Saved my life!

igwerner commented on Nov 28, 2018

Awesome thanks dude
phoenixproject commented on Nov 30, 2018
thanks. Very thanks!
srinath654321 commented on Dec 17, 2018
Thank you so much the first method worked out well.
danchamorro commented on Jan 3, 2019
Awesome!
crosslife commented on Jan 13, 2019
Thank you!
numoonchld commented on Jan 16, 2019 • edited ▼
thanks! - the first method worked - didn't need to try the second
renatodysouza commented on Jan 17, 2019
thank you, so much
Sofia190 commented on Jan 26, 2019
Thank you! It's brilliant!
DanielDimanov commented on Jan 27, 2019
Thank you!

EPiC-APOC commented on Feb 15, 2019
this works for my head repository (origin) how can I pull (or whatever) this to my base repository (upstream)?
begueradj commented on Mar 12, 2019
None of these methods worked for me.
deppeler commented on Apr 15, 2019
Just what I was looking for. Thank you for posting and sharing.
MPGAJEWSKI commented on Jul 21, 2019
The first method worked. Thanks!
guyueshui commented on Aug 28, 2019
I would say: mark! And thank you very much, I've wrote down this method to my blog, thank you!
widoz commented on Nov 9, 2019
holitics commented on Nov 18, 2019
First method worked for me!
seiasmribeiro commented on Nov 23, 2019
thanks

Wujin1992 commented on Dec 7, 2019
Awesome!
shasmdh commented on Dec 10, 2019
The first method worked for me, tnx.
actionanand commented on Jan 17, 2020
awesome. Thanks for the update
Everyone remember your all commits still accessible by checking the Environment.
Of course for you only.
skrantzman commented on Jan 23, 2020
I really appreciate you posting this. An immense help in clearing up the unneeded clutter. The first method worked perfectly.
ameyaKetkar commented on Jan 28, 2020
Awesome! Thanks a lot.
proberto-pg commented on Jan 28, 2020
This is very useful, thanks.
ambrosektal commented on Feb 19, 2020
Thank you



Thank you!
corestackdev commented on Jun 11, 2020
The old commits are still accessible if you have github actions enabled, the action page displays all previous commits and workflow executions. Anyone can still gain access to those previous commits clicking the commit id.
Dionid commented on Jun 27, 2020
Thanks!
JayeshThamke commented on Jun 30, 2020
Hi i also have branches on my repo. and want to ensure that those will remain intact and only commits will be deleted.
🔊 halmogbl commented on Jul 16, 2020
thank you
sergiocarp10 commented on Jul 19, 2020
Thank you so much!
Thank you so much!
Thank you so much! TheSpanishBread commented on Aug 8, 2020
TheSpanishBread commented on Aug 8, 2020
TheSpanishBread commented on Aug 8, 2020
TheSpanishBread commented on Aug 8, 2020 Thank you !!! :D
TheSpanishBread commented on Aug 8, 2020 Thank you !!! :D ightharpoonup in the state of the st

If using 2FA for authentication with Github, the HTTPs remote won't be possible. You need SSH instead. Replace git remote add origin https://github.com/heiswayi/myproject.git With git remote add origin git@github.com:heiswayi/myproject.git sidharrth2002 commented on Sep 7, 2020 Method 1 worked like a charm suyash0103 commented on Sep 15, 2020 Method 1 worked! Thanks a lot! alfonmga commented on Sep 25, 2020 An engineer is a man who can do for a dime what any fool can do for a dollar meta-n4vn33t commented on Sep 27, 2020 Method 1 works fine on my Gitea test instance thanks! SORC3r3r commented on Nov 4, 2020 凸 🌒 itsGreghere commented on Dec 3, 2020 First Method = beautiful! Exactly what I wanted to do. Thanks! rommzestz commented on Dec 5, 2020

all steps ######### git pull origin master git checkout --orphan TEMP_BRANCH git add -A git config --global user.email "you@example.com" git config --global user.name "Your Name" git commit -am "Initial commit" git branch -D master git branch -m master git push -f origin master **peloncano** commented on Jan 12 This helped - thanks. acmd3BOT commented on Feb 11 Thanks brother 👍 Dr-McFish commented on Feb 21 helfull +1 arpanEnbake commented on Feb 26 thanks for giving commands SincentYeh-dev commented on Feb 27 Thanks, first method work perfectly. 📦 ileuskov commented on Feb 28 Thanks a lot!

First is easy, just need enter your email and name

morethanbooks commented on Mar 5 Awesome! Thanks! 🐲 calag4n commented on Mar 10 Thanks:) nassandraga commented on Mar 20 First method worked, Thank you for sharing this! **iii** trinib commented on Mar 29 THANK YOU VERY very MUCHcould not understand on stackflow 😥 julian128 commented on Mar 31 Hi I have a problem related with commit histories. I worked on branch - br1. and created pull request. before merging it, I created new branch - br2 from master. after that, the PR related br1 has been merged. by the way, now, the PR related br2 includes all commit histories from the PR related br1. can I remove only these commit histories? current pr has new commit histories related br2. I don't want to delete new commit histories. can you help me? 🀞 trinib commented on Mar 31

Hi I have a problem related with commit histories.

I worked on branch - br1. and created pull request.

before merging it, I created new branch - br2 from master.

after that, the PR related br1 has been merged.

by the way, now, the PR related br2 includes all commit histories from the PR related br1.

can I remove only these commit histories? current pr has new commit histories related br2.

I don't want to delete new commit histories.

can you help me?

TarthikKN007 commented on Apr 23
thanks alot worked with the first one
iginijhawan commented on May 2
Thanks A lot
leejongyoung commented on May 3
Awesome! 😩
amanarora9848 commented 17 days ago
Worked very well, thank you!
GaborVxxx commented 9 days ago
Thank you. Helped me a lot.
flcoder commented 6 days ago
Thank you!