

Checking out pull requests locally

In this article

- Modifying an active pull request locally
- Modifying an inactive pull request locally
- Error: Failed to push some refs

When someone sends you a pull request from a fork or branch of your repository, you can merge it locally to resolve a merge conflict or to test and verify the changes before merging on GitHub Enterprise Server.

Who can use this feature


Anyone with write access to a repository can pull a remote pull request down locally.

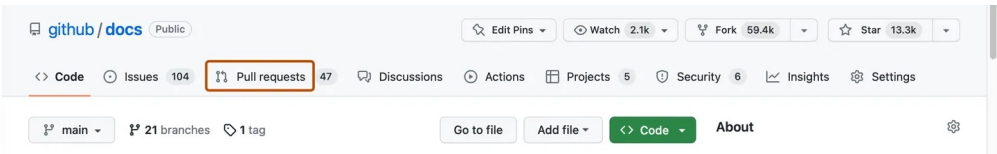
Mac Windows Linux

GitHub CLI Web browser

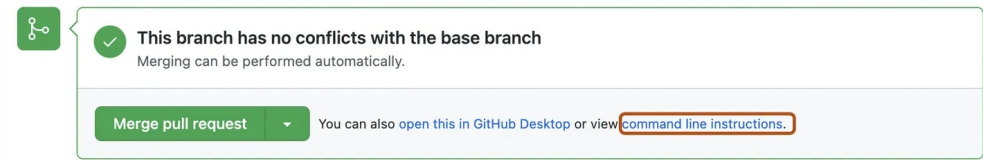
Note: Pull request authors can give upstream repository maintainers, or those with push access to the upstream repository, permission to make commits to their pull request's compare branch in a user-owned fork. For more information, see "[Allowing changes to a pull request branch created from a fork](#)."

Modifying an active pull request locally

- Under your repository name, click  **Pull requests**.



- In the list of pull requests, click the pull request you'd like to modify.
- In the merge box, click **command line instructions**. Follow the sequence of steps to bring down the proposed pull request.



- Optionally, to view proposed changes in GitHub Desktop, next to the **Merge pull**

request button, click **open this in GitHub Desktop**.

✓ **This branch has no conflicts with the base branch**
Merging can be performed automatically.

Merge pull request

You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

To learn more about GitHub CLI, see "[About GitHub CLI](#)."

To check out a pull request locally, use the `gh pr checkout` subcommand. Replace `pull-request` with the number, URL, or head branch of the pull request.

```
gh pr checkout PULL-REQUEST
```

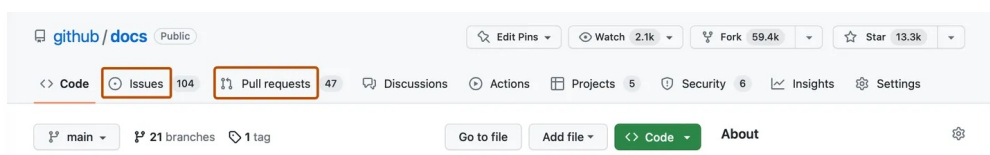
Modifying an inactive pull request locally [↗](#)

If a pull request's author is unresponsive to requests or has deleted their fork, the changes proposed in that pull request can still be merged via a new pull request. However, if you want to make changes to a pull request and the author is not responding, you'll need to perform some additional steps to update the pull request.

Once a pull request is opened, GitHub Enterprise Server stores all of the changes remotely. In other words, commits in a pull request are available in a repository even before the pull request is merged. You can fetch an open pull request and recreate it as your own.


Anyone can work with a previously opened pull request to continue working on it, test it out, or even open a new pull request with additional changes. However, only collaborators with push access can merge pull requests.

- 1 Under your repository name, click  **Issues** or  **Pull requests**.



- 2 In the "Pull Requests" list, click the pull request you'd like to merge.
- 3 Find the ID number of the inactive pull request. This is the sequence of digits right after the pull request's title.

Just adding stuff #24612

 Open

octocat wants to merge 1 commit into [github:main](#) from [octocat:branch](#) 

- 4 Open TerminalTerminalGit Bash.
- 5 Fetch the reference to the pull request based on its ID number, creating a new branch in the process.

```
git fetch origin pull/ID/head:BRANCH_NAME
```

- 6 Switch to the new branch that's based on this pull request:

```
[main] $ git switch BRANCH_NAME  
> Switched to a new branch 'BRANCH_NAME'
```

- 7 At this point, you can do anything you want with this branch. You can run some local tests, or merge other branches into the branch.

- 8 When you're ready, you can push the new branch up:

```
[pull-inactive-pull-request] $ git push origin BRANCH_NAME  
> Counting objects: 32, done.  
> Delta compression using up to 8 threads.  
> Compressing objects: 100% (26/26), done.  
> Writing objects: 100% (29/29), 74.94 KiB | 0 bytes/s, done.  
> Total 29 (delta 8), reused 0 (delta 0)  
> To https://HOSTNAME/USERNAME/REPOSITORY.git  
> * [new branch]      BRANCH_NAME -> BRANCH_NAME
```

- 9 [Create a new pull request](#) with your new branch.

Error: Failed to push some refs

The remote `refs/pull/` namespace is *read-only*. If you try to push any commits there, you'll see this error:

```
! [remote rejected] HEAD -> refs/pull/1/head (deny updating a hidden ref)  
error: failed to push some refs to 'git@github.local:USERNAME/REPOSITORY.git'
```

Tip: When you remove or rename a remote reference, your local `refs/pull/origin/` namespace will not be affected by calls to `git-remote`.

Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)