

# Managing branches in GitHub Desktop

## In this article

- About managing branches
- Creating a branch
- Creating a branch from a previous commit
- Publishing a branch
- Switching between branches
- Deleting a branch
- Further reading

You can use GitHub Desktop to create a new branch off of an existing branch in your repository so you can safely experiment with changes.

Mac Windows

## About managing branches

You can use branches to safely experiment with changes to your project. Branches isolate your development work from other branches in the repository. For example, you could use a branch to develop a new feature or fix a bug.

You always create a branch from an existing branch. Typically, you might create a branch from the default branch of your repository. You can then work on this new branch in isolation from changes that other people are making to the repository.

You can also create a branch starting from a previous commit in a branch's history. This can be helpful if you need to return to an earlier view of the repository to investigate a bug, or to create a hot fix on top of your latest release.

Once you're satisfied with your work, you can create a pull request to merge your changes in the current branch into another branch. For more information, see "[Creating an issue or pull request from GitHub Desktop](#)" and "[About pull requests](#)."


You can always create a branch in GitHub Desktop if you have read access to a repository, but you can only push the branch to GitHub if you have write access to the repository.

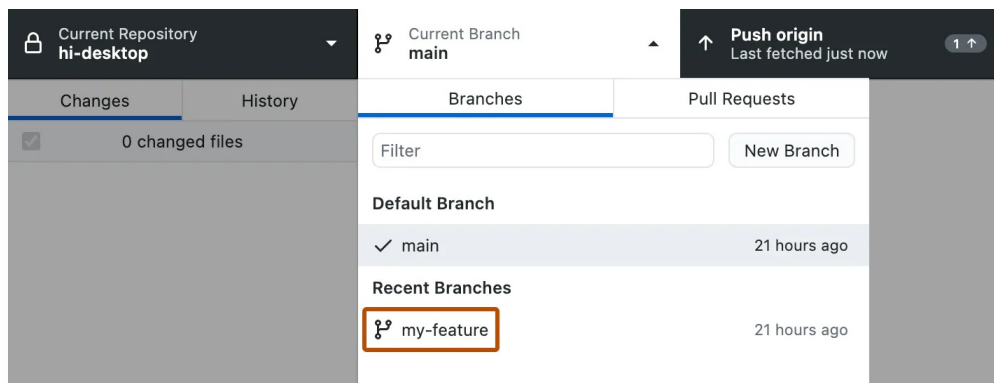
Repository administrators can enable protections on a branch. If you're working on a branch that's protected, you won't be able to delete or force push to the branch. Repository administrators can enable other protected branch settings to enforce specific workflows before a branch can be merged. For more information, see "[About protected branches](#)."

Repository administrators can also enable rulesets. Rulesets can be used to require specific branch names when creating a new branch, or to allow only users with bypass permissions to publish a new branch to the remote repository. GitHub Desktop will show a warning and prevent the branch from being created if the branch does not follow the rulesets. For more information, see "[About rulesets](#)."

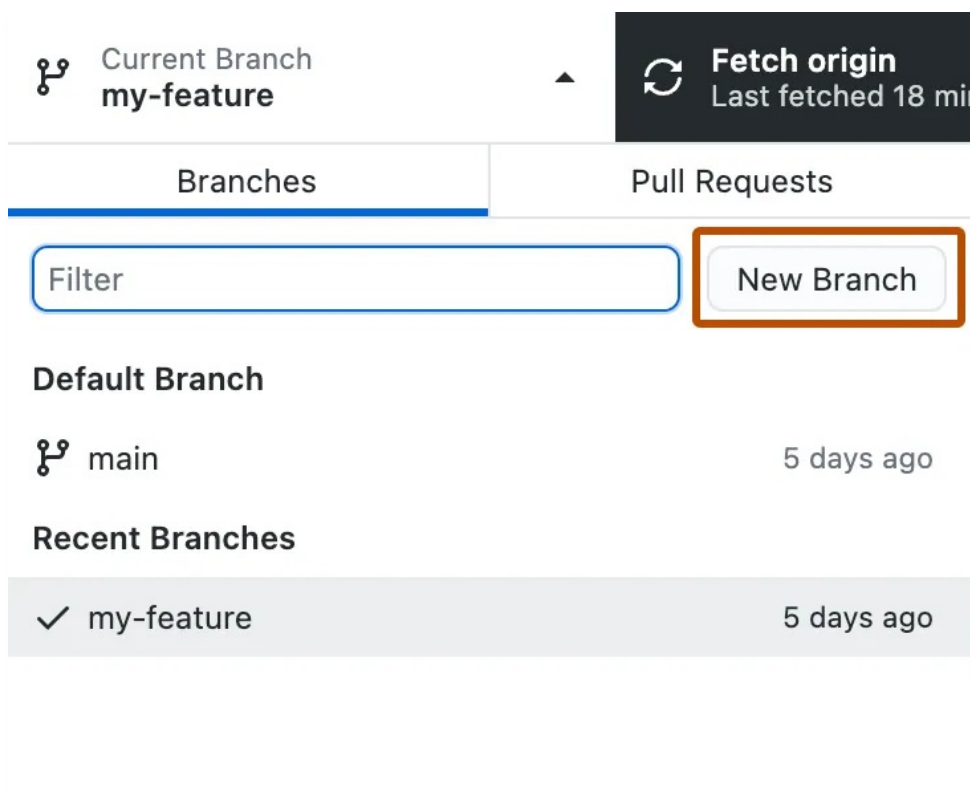
## Creating a branch [↗](#)

**Tip:** The first new branch you create will be based on the default branch. If you have more than one branch, you can choose to base the new branch on the currently checked out branch or the default branch.

- 1 At the top of the app, click  **Current Branch** and then in the list of branches, click the branch that you want to base your new branch on.



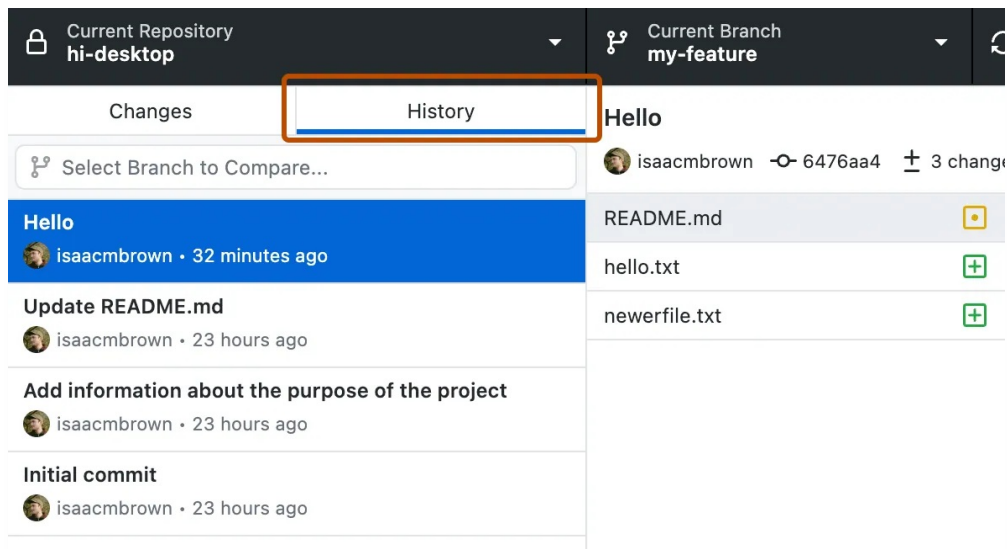
- 2 Click **New Branch**.



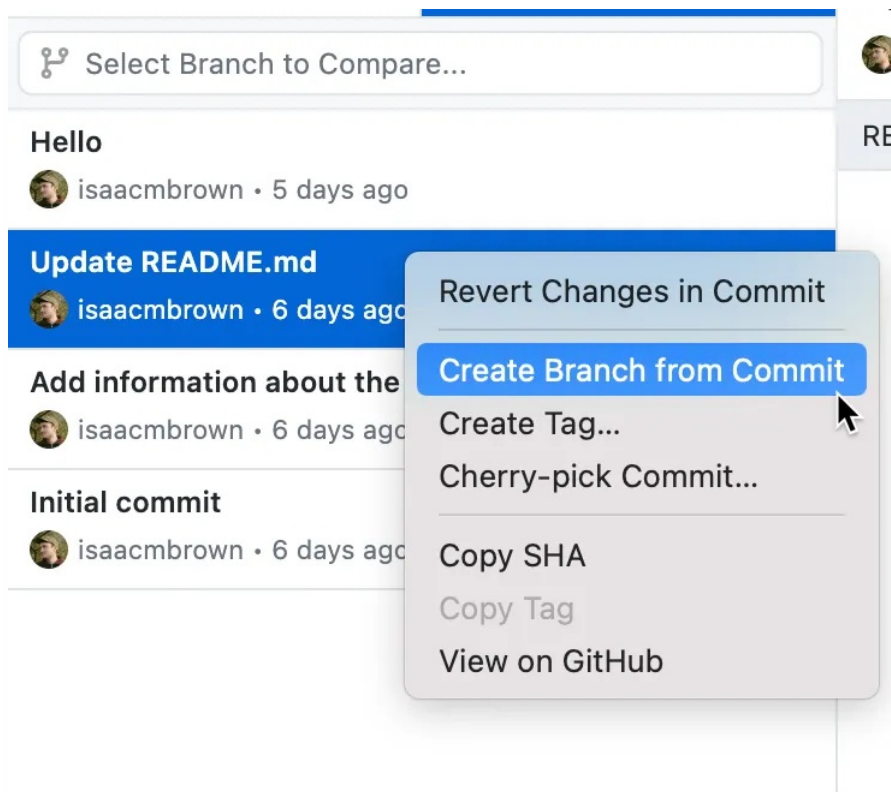
- 3 In the "Create a Branch" window, under "Name", type the name of the new branch.
- 4 Under "Create branch based on...", select a base branch for your new branch.
- 5 Click **Create Branch**.

## Creating a branch from a previous commit [↗](#)

- 1 In the left sidebar, click **History**.



- 2 Right-click on the commit you would like to create a new branch from and select **Create Branch from Commit**.

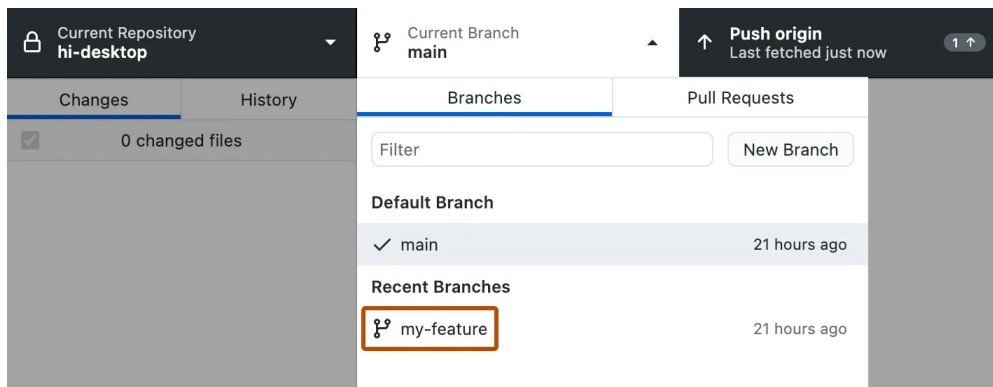


- 3 In the "Create a Branch" window, under "Name", type the name of the new branch.
- 4 Click **Create Branch**.

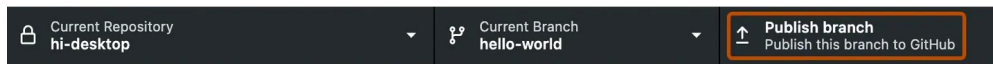
## Publishing a branch [🔗](#)

If you create a branch on GitHub, you'll need to publish the branch to make it available for collaboration on GitHub.

- 1 In the repository bar, click **Current Branch**, then click the branch that you want to publish.



- 2 Click **Publish branch**.

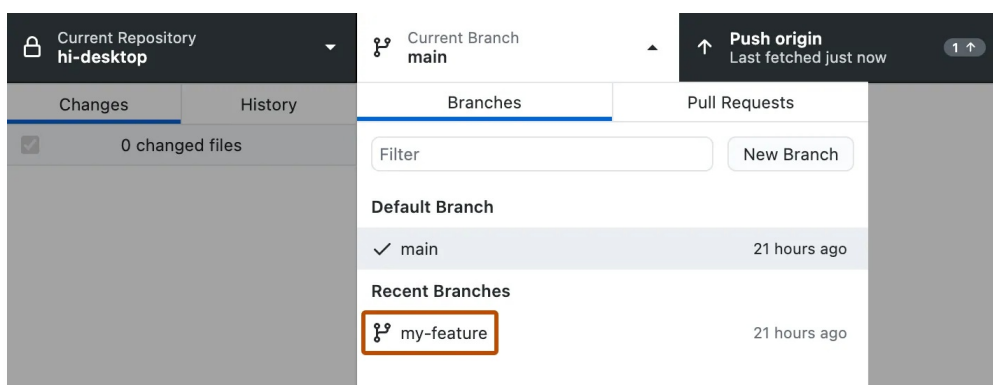


## Switching between branches [↗](#)

You can view and make commits to any of your repository's branches. If you have uncommitted, saved changes, you'll need to decide what to do with your changes before you can switch branches. You can commit your changes on the current branch, stash your changes to temporarily save them on the current branch, or bring the changes to your new branch. If you want to commit your changes before switching branches, see "[Committing and reviewing changes to your project in GitHub Desktop](#)."

**Tip:** You can set a default behavior for switching branches in the **Advanced** settings. For more information, see "[Configuring basic settings in GitHub Desktop](#)."

- 1 In the repository bar, click  **Current Branch**, then click the branch that you want to switch to.

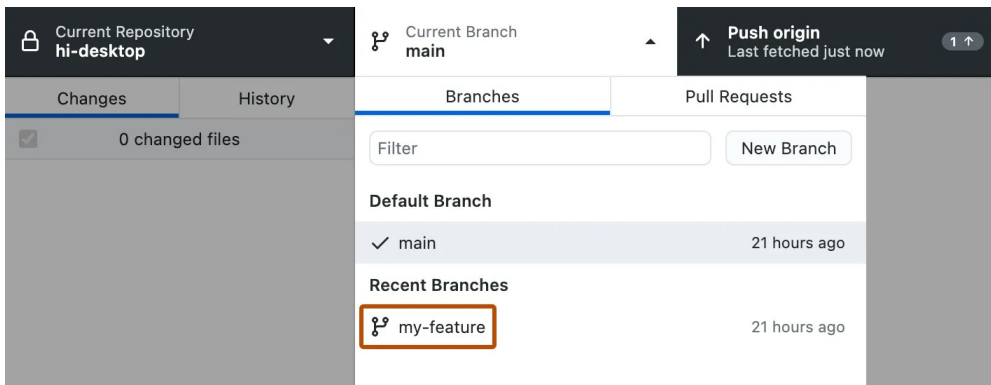


- 2 If you have saved, uncommitted changes, in the "Switch Branch" window, select **Leave my changes on CURRENT-BRANCH** or **Bring my changes to NEW-BRANCH**, then click **Switch Branch**.

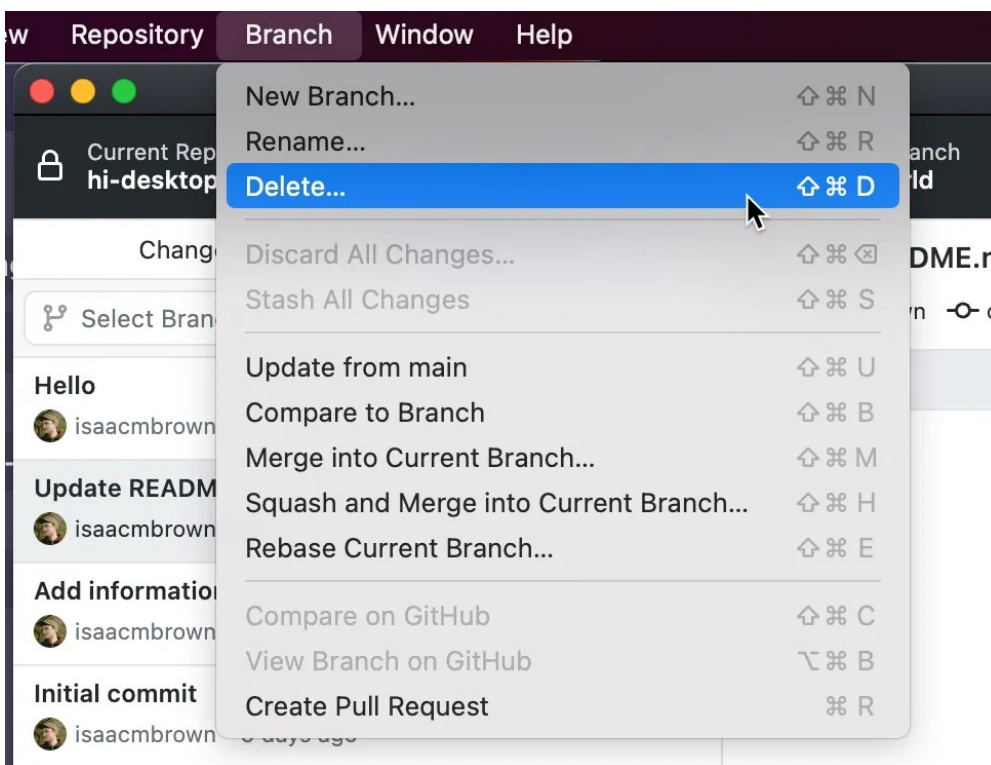
## Deleting a branch [↗](#)

You can't delete a branch if it's currently associated with an open pull request. You cannot undo deleting a branch.

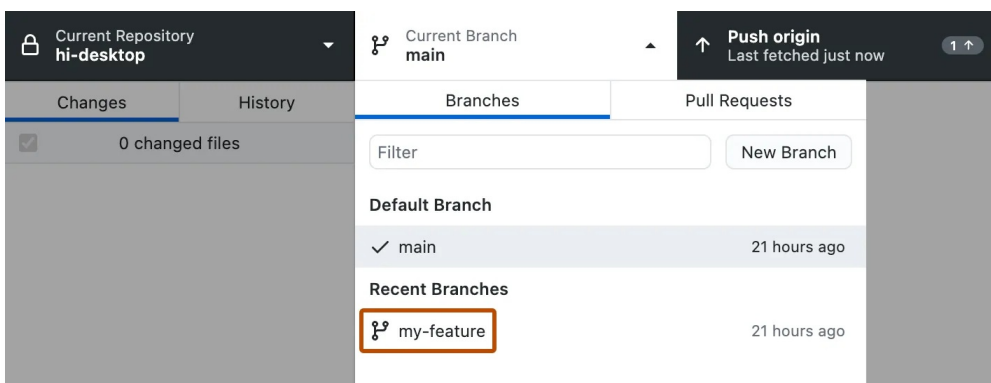
- 1 In the repository bar, click  **Current Branch**, then click the branch that you want to delete.



- 2 In your menu bar, click **Branch**, then click **Delete....** You can also press `Shift + Command + D`.

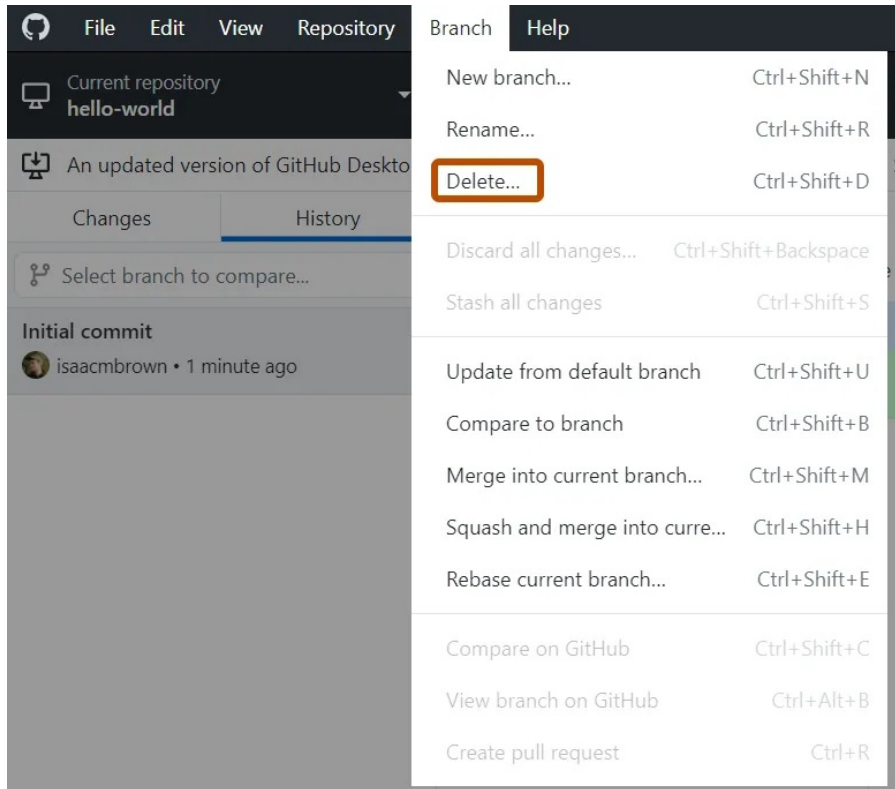


- 1 In the repository bar, click  **Current Branch**, then click the branch that you want to delete.



- 2 In your menu bar, click **Branch**, then click **Delete....** You can also press `Ctrl +`

Shift + D .



## Further reading [↗](#)

- "[Cloning a repository from GitHub to GitHub Desktop](#)"
- "[GitHub glossary](#)" in the GitHub glossary
- "[About branches](#)"
- "[Branches in a Nutshell](#)" in the Git documentation
- "[Stashing changes in GitHub Desktop](#)"

## Legal

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