Squashing commits in GitHub Desktop

In this article

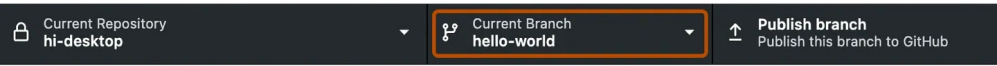You can use GitHub Desktop to squash commits in your branch's history.

Mac   Windows

About squashing a commit 🔗

Squashing allows you to combine multiple commits in your branch's history into a single commit. This can help keep your repository's history more readable and understandable.
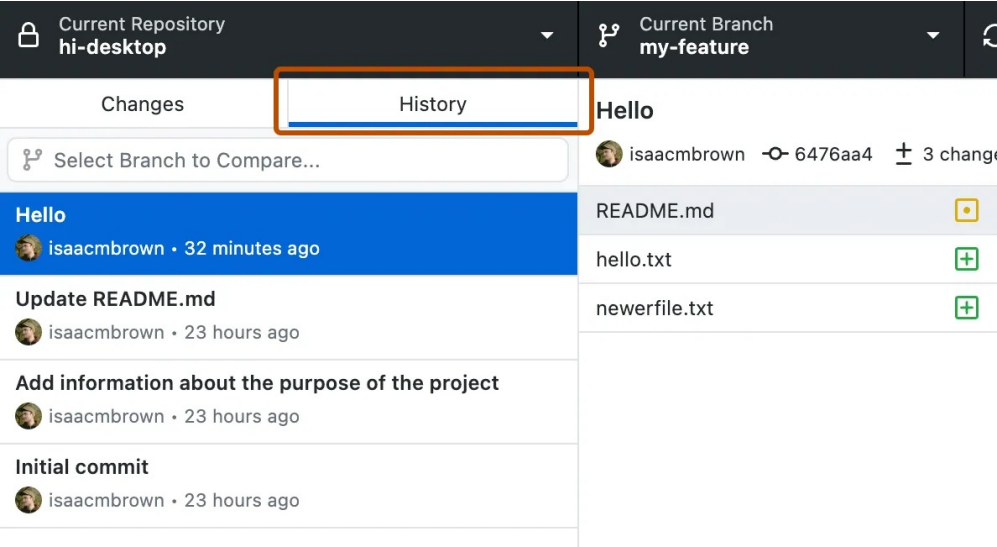
Squashing a commit 🔗

① In GitHub Desktop, click ⑂ **Current Branch**.

② In the list of branches, select the branch that has the commits that you want to squash.
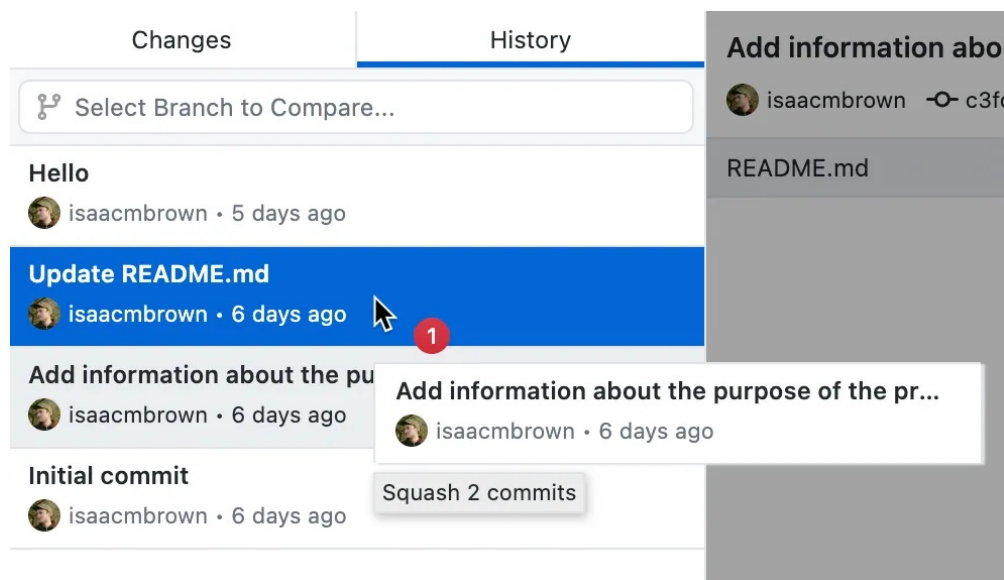
③ In the left sidebar, click **History**.

④ Select the commits to squash and drop them on the commit you want to combine

them with.

You can select one commit or select multiple commits using `Command` or `Shift` .

You can select one commit or select multiple commits using `Ctrl` or `Shift` .

5 Modify the commit message of your new commit. The commit messages of the selected commits you want to squash are pre-filled into the **Summary** and **Description** fields.

6 Click **Squash Commits**.

Error messages when squashing commits 🔗

When you squash commits, you may see one of the following notifications or error messages.

- A notification states that the requested change to the branch will require a force push to update the remote branch. Force pushing alters the commit history of the branch and will affect other collaborators who are working in that branch. Select **Begin Squash** to start the squash, and then click **Force push origin** to push your changes.
- An error states that the squash failed because there is a merge commit among the squashed commits.
- A notification is shown indicating that there are uncommitted changes present on your current branch. Select **Stash Changes and Continue** to store the changes and proceed, or select **Close** to dismiss the message and commit the changes. When there are no longer any uncommitted changes you can squash your commits.