About forks

In this article

A fork is a new repository that shares code and visibility settings with the original "upstream" repository.

About forks 🔗

Forks let you make changes to a project without affecting the original repository, also known as the "upstream" repository. After you fork a repository, you can fetch updates from the upstream repository to keep your fork up to date, and you can propose changes from your fork to the upstream repository with pull requests. A fork can be owned by either a personal account or an organization.

When you view a forked repository on GitHub Enterprise Server, the upstream repository is indicated below the name of the fork.
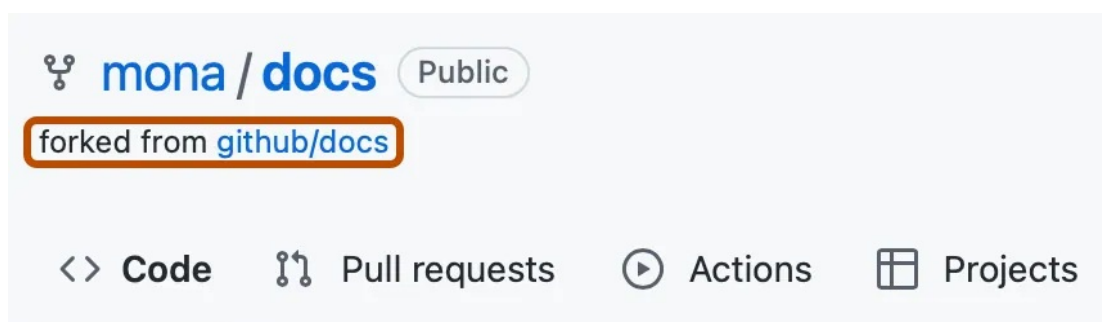
In open source projects, forks are often used to iterate on ideas or changes before incorporating the changes into the upstream repository. If you fork a public repository to your personal account, make changes, then open a pull request to propose your changes to the upstream repository, you can give anyone with push access to the upstream repository permission to push changes to your pull request branch (including deleting the branch). This speeds up collaboration by allowing repository maintainers to make commits or run tests locally to your pull request branch from a user-owned fork before merging. You cannot give push permissions to a fork owned by an organization. For more information, see "[Allowing changes to a pull request branch created from a fork](#)."

Deleting a fork will not delete the original upstream repository. You can make any changes you want to your fork, and there will be no effect on the upstream. For example, you can add collaborators, rename files, or generate GitHub Pages on the fork without affecting the upstream. If you delete a private repository, all forks of the repository are deleted.

You can view, sort, and filter the forks of a repository on the repository's forks page. For more information, see "[Understanding connections between repositories](#)."

About creating forks 🔗

You can fork a private or internal repository to your personal account or to an organization on your GitHub Enterprise Server instance where you have permission to create repositories, provided that the settings for the repository and your enterprise policies allow forking. Generally, you can fork any public repository to your personal account or to an organization where you have permission to create repositories.

For instructions for forking a repository, see "[Fork a repo](#)." For more information about when you can create forks, and the permission and visibility settings of forks, see "[About permissions and visibility of forks](#)."

> **Tip:** You can use GitHub Desktop to fork a repository. For more information, see "[Cloning and forking repositories from GitHub Desktop](#)."

Forking a repository versus duplicating a repository 🔗

If you want to create a new repository from the contents of an existing repository but don't want to merge your changes to the upstream in the future, you can duplicate the repository or, if the repository is a template, you can use the repository as a template. For more information, see "[Duplicating a repository](#)" and "[Creating a repository from a template](#)".

Forking a repository is similar to duplicating a repository, with the following differences.

- You can use a pull request to suggest changes from your fork to the upstream repository.
- You can bring changes from the upstream repository to your fork by synchronizing your fork with the upstream repository.
- Forks have their own members, branches, tags, labels, policies, issues, pull requests, discussions, actions, projects, and wikis.
- Forks inherit the restrictions of their upstream repositories. For example, branch protection rules cannot be passed down if the upstream repository belongs to an organization on a GitHub Free plan.

Further reading 🔗

- "[About collaborative development models](#)"
- "[Creating a pull request from a fork](#)"
- [Open Source Guides](#)