

Lesson

Tuesday

Intermediate JavaScript (/intermediate-javascript)

/ Team Week (/intermediate-javascript/team-week)

/ Pull Requests with Forks

Text

Many of the JavaScript libraries and tools we use at Epicodus are **open source** including webpack, React, and Jest, to name a few. **Open source** means the code is available publicly and that anyone can contribute to it.

In order to do that, maintainers of open source projects welcome pull requests from the community. However, it wouldn't be a good idea for maintainers to make everyone in the community collaborators. That would be difficult to manage and also lead to potential security risks.

Instead, it's possible to make a PR with only read access to a repository. Instead of cloning the repository and making a branch, contributors **fork** the project, make changes to a branch on the fork, and then submit the PR.

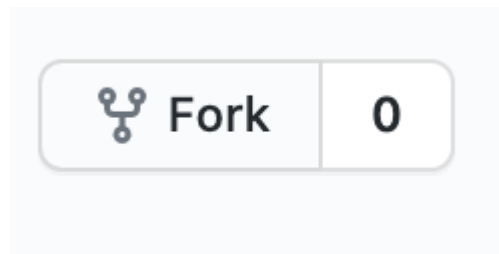
In this lesson, we'll walk through the necessary steps to make a PR via a fork. The steps for approving a PR are the same regardless of whether the PR comes from a branch or a branch on a fork so we won't cover that part.

Once you know how to submit a PR via a fork, you can start contributing to open source projects! There are many, many ways to do so — and you don't need to be an expert coder to contribute. We'll talk about contributing to open source more at the end of this lesson.

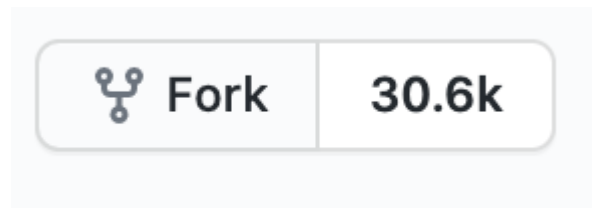
Making a PR Request Via a Fork

We covered the basics of forking in Practice: GitHub Remote Repositories (<https://www.learnhowtoprogram.com/introduction-to-programming/git-html-and-css/practice-github-remote-repositories>).

To fork a repository, we just need to click the *Fork* button in the upper right corner of the repository we want to fork.



There's even a little number by the button that shows how many times the repository has been forked. The above repository doesn't have any forks — but check out the *Fork* button for React:



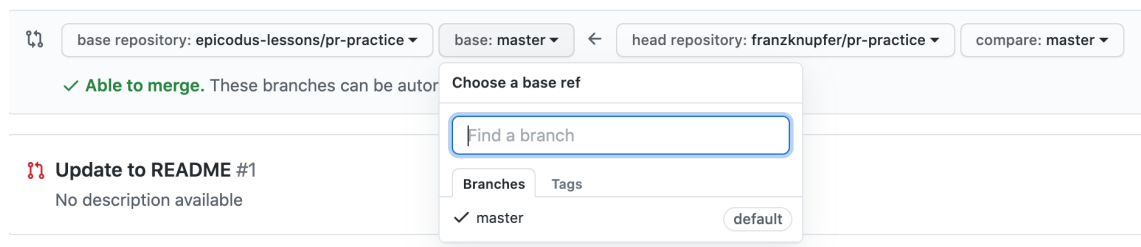
This one has nearly 31,000 forks at the time this screenshot was taken.

When you click the *Fork* button, you'll be prompted to choose a GitHub account to fork the project to. Once you choose an account (you likely only have one at this point), GitHub will automatically create the fork.

Once the project is forked, we can clone that project to our desktop, make a branch, and then make any updates to the branch as needed. *Do not make changes directly to the main branch.* We are really following almost the exact same process that we follow when making a PR without a fork.

Once the branch is updated, committed, and pushed back to our fork of the project, we can make a PR.

To do that, we can open a PR *in our fork of the project*. We don't even need to navigate to the original repository.



The image above shows the process of making a PR from a forked repository. GitHub automatically points at the original repository for the base repository while the compare repository is automatically the fork. Note that we should always verify that we are pointing at the correct repositories; we also need to make sure we are pointing at the right branches, too. In the example above, we can see that the compare branch is still on main — not what we want if we've just made changes to a branch on the fork! So even though GitHub should automatically point to the correct repositories, we still need to verify they are correct and update the branches as needed.

After that point, every other part of the PR process is the same as when we make a PR via a branch on the original repository itself.

Contributing to Open Source Projects

Open source projects are a huge part of the developer ecosystem. As we mentioned at the beginning of this lesson, many widely-used libraries are open source. By contributing to open source projects, we can get more involved in the developer community, improve our favorite libraries, and add great experience to our resume.

At this point, you might be thinking that you don't have enough experience to make an open source contribution. However, that is not the case. You don't need to be an expert coder to make a contribution. In fact, many open source projects have a pool of good first issues for people new to making contributions. Open source maintainers may need help on everything from updating READMEs to writing tests to updating the code itself. Updating a README for a library can be a great way to get to know the library better and get involved in an open source project.

We recommend taking the time to make a contribution. For more information on making open source contributions, see [How to Contribute to Open Source](http://opensource.guide/how-to-contribute/) (<http://opensource.guide/how-to-contribute/>), an excellent tutorial from Open Source Guides on making contributions.

[Previous \(/intermediate-javascript/team-week/pull-requests-with-branches\)](#)

[Next \(/intermediate-javascript/team-week/team-week-presentations-and-independent-project\)](#)

Lesson 12 of 13

Last updated more than 3 months ago.

[disable dark mode](#)



© 2023 Epicodus (<http://www.epicodus.com/>), Inc.