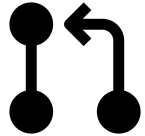
Pull request basics

Childhood Cancer Data Lab x Alex's Lemonade Stand

What is a **pull request**?



A request to merge new code changes in your branch into another – often, the default branch for a repository such as `main` or `development`.

Pull requests typically undergo **code review**, which is a quality assurance practice where someone who didn't write the code inspects it.

We'll talk much more about why (analytical) code review is a valuable practice in a later session, but we want to provide you with some context before we talk about tracking and planning work.

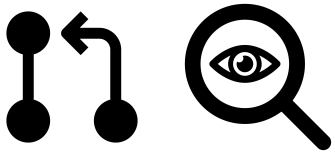
At this point, the most important thing to keep in mind about **pull requests** is that someone needs to look at them before they are merged

Would you rather have to read and vouch for 250 lines of code or 2500 lines of code?

Reasonably-sized “bites” are more likely to get (thoroughly) reviewed in a timely manner.

The longer code hangs out in a branch (“becomes stale”), the more likely it is that **merge conflicts** will pop up.

Pull request review primer



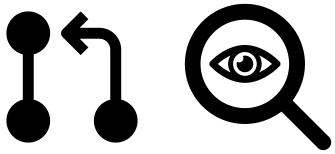
GitHub provides an interface that allows you to view the conversation around code, commits to the PR branch, and the files changed (the cumulative code changes between the target and PR branch).

The reviewer can comment on individual lines of code and individual files when viewing the diff.

The reviewer can submit a review approving the code, requesting changes, or commenting without approving.

- **Approval** means that the code can be merged.
- **Changes requested** mean that the reviewer that requested the changes must re-review and approve.
- **Commenting** means the code can be merged if someone else with the appropriate permissions approves.

Pull request review side note



GitHub also allows you to set up checks that need to *pass* before a pull request can be merged.

We'll talk a little bit more about this in an upcoming session!

To the demo pull request!