Professional Skills

Agile Fundamentals

Jira

Git

- Introduction to Source Control
- Basics
- Cloning
- Forking
- Branching
- Merging
- Reverting
- GitHub Pull Requests
- GitHub Reviews
- GitHub Actions

Databases Introduction

Java Beginner

Maven

Testing (Foundation)

Java Intermediate

HTML

CSS

Javascript

Spring Boot

Selenium

Sonarqube

Advanced Testing (Theory)

Cucumber

MongoDB

Express

NodeJS

React

GitHub Reviews

Contents

Overview

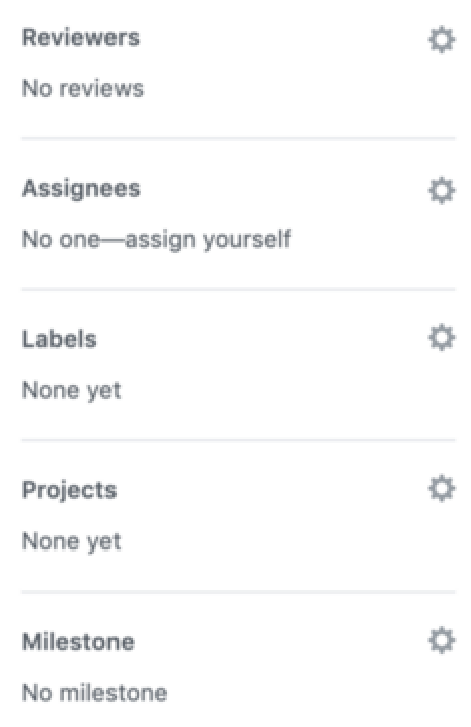
When submitting a Pull Request (PR) on GitHub (or most Git Providers), you can request a review to take place.

In the review, other developers and/or members of your team can propose/request changes, ask for clarity in certain areas, discuss certain files, and eventually (hopefully!) approve the PR.

Once the PR has been approved, it is then able to be merged to another branch, usually `develop` or `main`.

Requesting a Review

You request a review when submitting a PR. All you need to do is use the `Reviewers` section on the right hand side:



Reviewers
No reviews

Assignees
No one—assign yourself

Labels
None yet

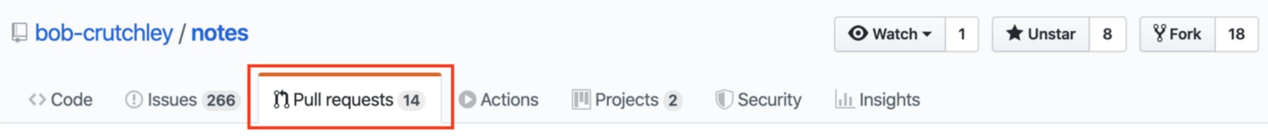
Projects
None yet

Milestone
No milestone

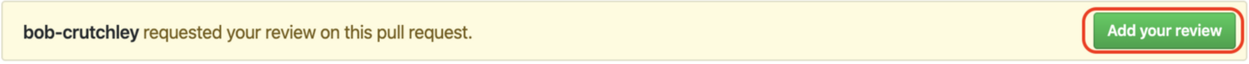
This will request a review from the person you choose from the dropdown. Please note that the person you would like to review your PR must be a collaborator on the GitHub project.

Reviewing a PR

If someone has requested a review from you, this will show up in the `Pull Requests` section, under the repository on GitHub:



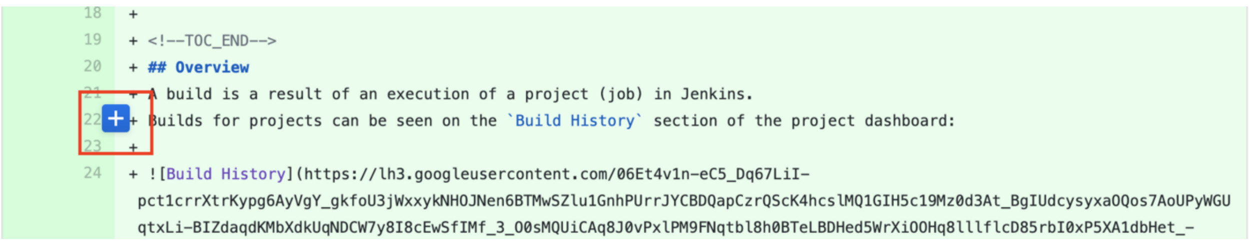
Once you have clicked on the correct PR, you will see that your review has been requested. You can click the `Add your review` button to start:



bob-crutchley requested your review on this pull request.

Making Comments

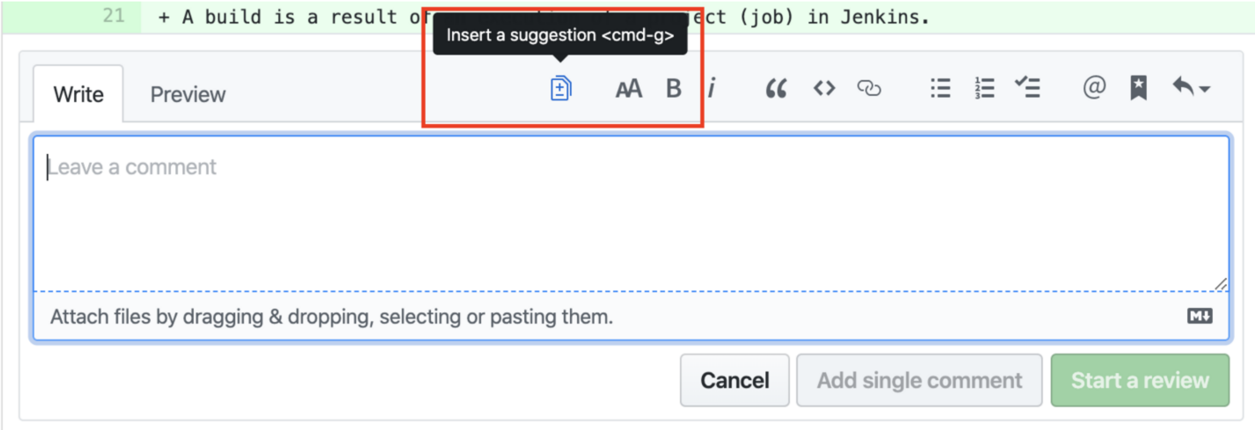
If you want to make a comment on a line in a file, you just use the blue `+` sign on the left hand side (it will appear as you hover over the `+` sign):



```
+
+ <!--TOC_END-->
+ ## Overview
+ A build is a result of an execution of a project (job) in Jenkins.
+ Builds for projects can be seen on the `Build History` section of the project dashboard:
+
+ ![Build History](https://lh3.googleusercontent.com/06Et4v1n-eC5_Dq67LiI-
  pct1crrXtrKypg6AyVgY_gkfoU3jWxxykNHOJNen6BTMwSZlu1GnhPUrrJYCBDQapCzrQScK4hcslMQ1GIH5c19Mz0d3At_BgIUdcysyxa0Qos7AoUPyWGU
  qtxLi-BIZdaqdKMbXdkUqNDCW7y8I8cEwSfIMf_3_O0sMQUiCAq8J0vPxlPM9FNqtbl8h0BTeLBDHed5WrXiOOHq8lllflcD85rbI0xP5XA1dbHet_-
```

Making a Suggestion in Your Comment

In the popup box, you can make a general comment if you wish or you can use the highlighted button to insert a suggestion:



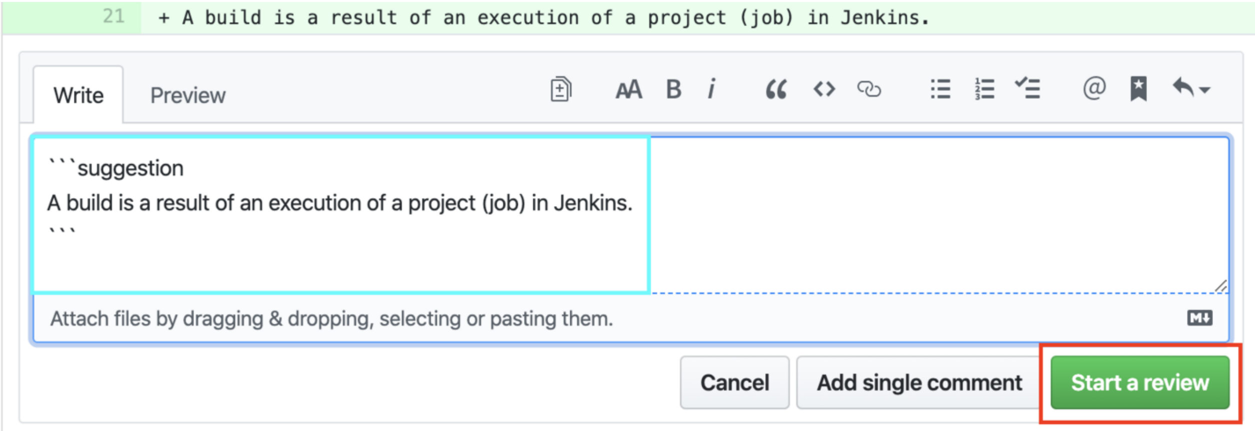
```
+ A build is a result of an execution of a project (job) in Jenkins.
```

Insert a suggestion <cmd-g>
Write
Preview
Leave a comment
Attach files by dragging & dropping, selecting or pasting them.
Cancel
Add single comment
Start a review

This will paste the line you are commenting on into the comment box, so you can explicitly state any changes you think would be beneficial.

The first time you do this, you will have the option to click on `Start review`, which will begin tracking any changes you make (including the one you just made).

From then on, when you make a comment/suggestion, the `Start review` button will be replaced with a button saying `Add Review Comment`:

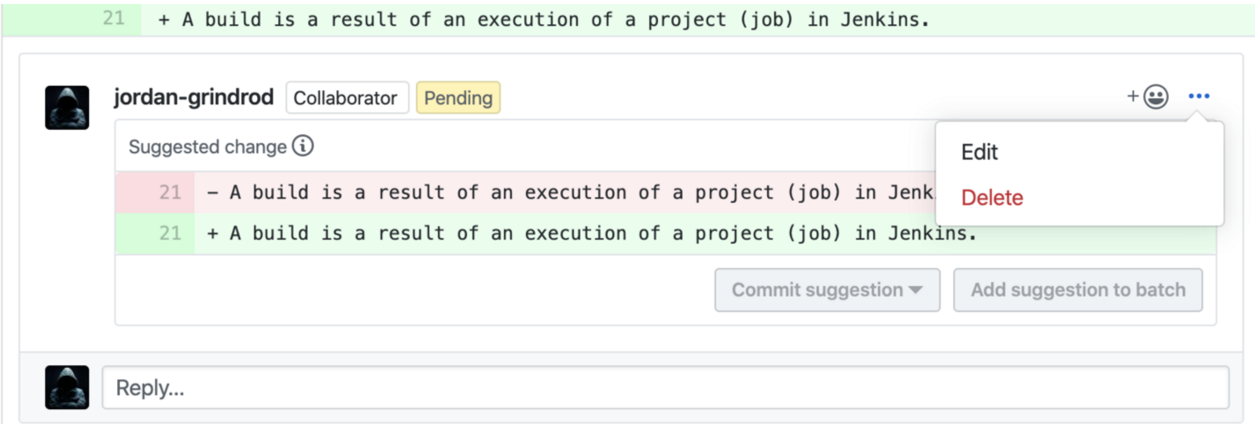


```
+ A build is a result of an execution of a project (job) in Jenkins.
```

~~~
```suggestion
A build is a result of an execution of a project (job) in Jenkins.
```
~~~

Editing and Deleting Comments

You can edit or delete any comments or suggestions you make:



```
+ A build is a result of an execution of a project (job) in Jenkins.
```

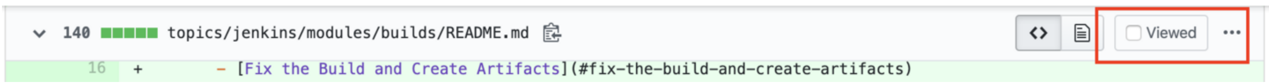
jordan-grindrod Collaborator Pending

Suggested change

```
- A build is a result of an execution of a project (job) in Jenk
+ A build is a result of an execution of a project (job) in Jenkins.
```

Keeping Track of Reviewed Files

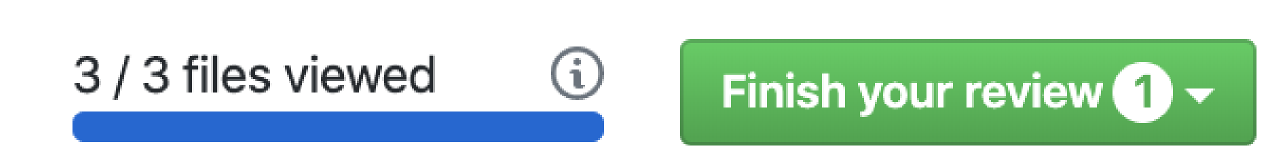
And, following good practice, you should check `viewed` when you are done reviewing a file:



```
16  +        - [Fix the Build and Create Artifacts](#fix-the-build-and-create-artifacts)
```

Finishing the Review

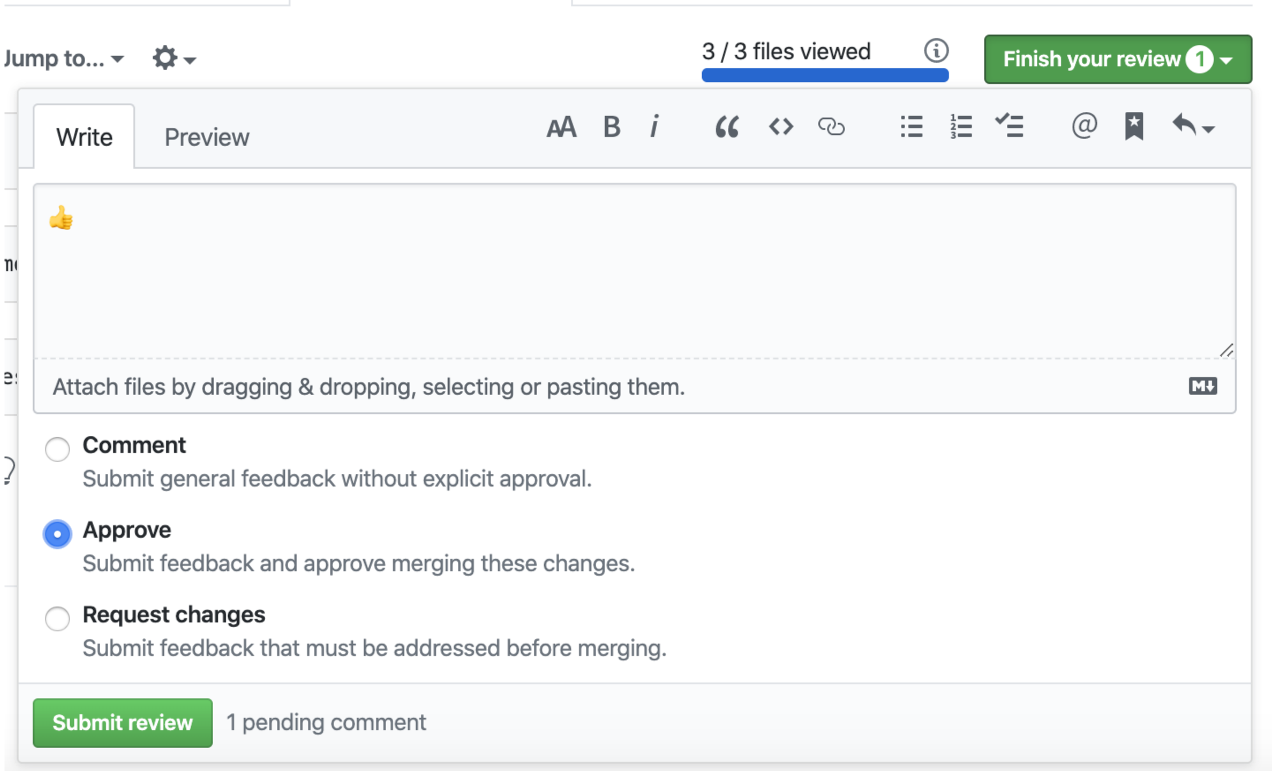
When you have viewed all files, you can click on `Finish your review`:



This will bring up a comment box where you can choose to:

- Make a comment without approval - this is usually asking for clarity around something
- Approve the review - the changes can then be merged into `develop` or `main`
- Request changes - this stops a merge from happening before the changes you've requested have been made

Once you have made your choice, you can go ahead and click `Submit review`, which will finish the review process and notify the person who requested the review that the review has been completed:



👍

Attach files by dragging & dropping, selecting or pasting them.

○ **Comment**
Submit general feedback without explicit approval.

● **Approve**
Submit feedback and approve merging these changes.

○ **Request changes**
Submit feedback that must be addressed before merging.

1 pending comment

Good Practice

It is always good practice to not only look over any files in a review, but also to **test** any code in there.

If there is code in the file, try to run it on your machine and make sure that:

- It runs correctly
- It follows best practices
- It is efficient
- Any instructions given are clear.

Tutorial

NB. This task works much better with somebody else. If you really want to do it individually, you will need to make another GitHub account...

1. Create a Pull Request, similar to how you will have done in the `Pull Requests` module, to merge a feature change from a `feature` branch into the `main` branch (ideally, you will make something that requires changing - for example, some broken code)

2. Add a Reviewer to the PR
3. Have the other person/other account do the same thing on their GitHub (requesting a review from you)
4. Start the review, making sure to test the new feature(s)
5. Make some comments for the other person/account
6. Either approve the review, or request changes
7. Complete the review
8. Complete the merge once your own PR has ben reviewed and approved by the other person/account

Exercises

There are no exercises for this module.