

Introduction to GitHub

Welcome! This document serves as a general introduction to making an account on GitHub, joining the Poulin Lab organization and adding code.

Table of Contents

Creating a GitHub Account	2
GitHub Education	2
Joining the Poulin Lab Organization	4
Storing & Sharing Code.....	6
Creating Repositories	6
Adding Code	8
Sharing Code with Members Outside the Lab	9
Modifying Code and Development.....	12
Creating Branches Within Repositories	12
Committing Changes	15
Pull Requests & Merging Branches	18

While this document focuses primarily on uploading and sharing code, the second half does touch on the basics of development. For further reading on using GitHub as a workspace, resources are available below:

<https://docs.github.com/fr/get-started/start-your-journey/hello-world>

<https://skills.github.com/>

<https://docs.github.com/en/get-started/start-your-journey/about-github-and-git>

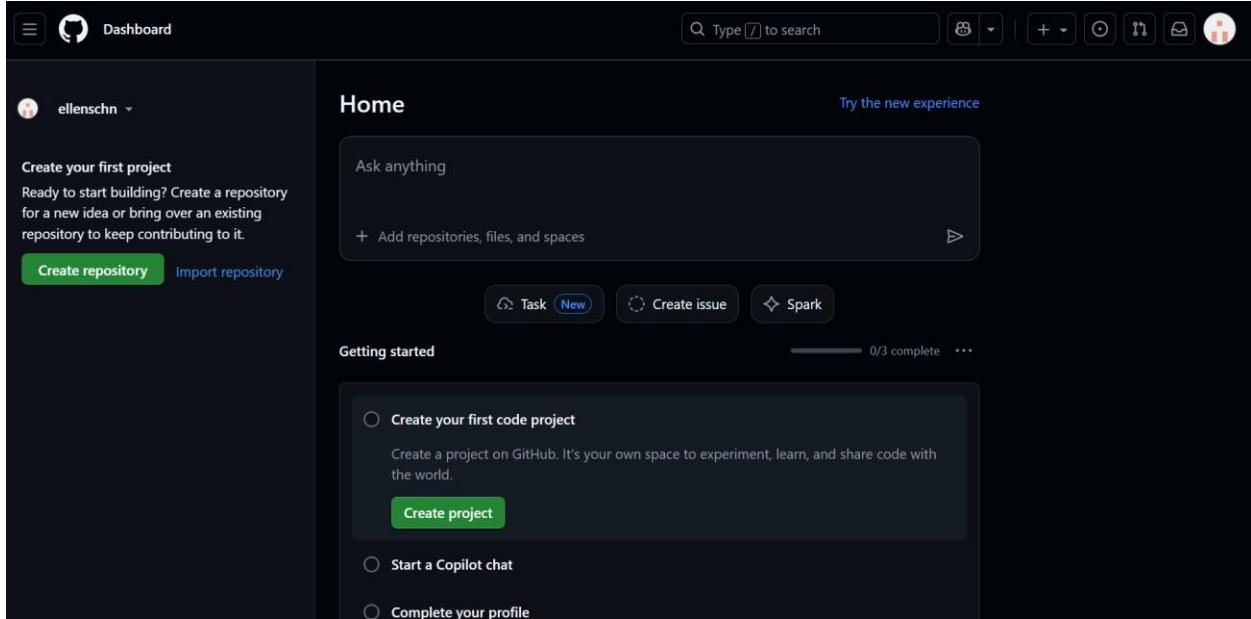
Creating a GitHub Account

To add code and join the lab organization, you should create a GitHub account. Ideally, each member of the lab should have their own GitHub account.

To create a GitHub personal account, navigate to the main GitHub page:

<https://github.com/>. Enter your email (ideally McGill email) and create a password.

Once made, log into GitHub. The main page of the website should look like this:



GitHub Education

Note: this part isn't necessary, just a nice feature of GitHub.

As a McGill student, you can sign up for GitHub Education for free! You can read on the benefits here: <https://github.com/education>

To sign up for GitHub Education, navigate to your settings. Click Billing and licensing -> Education benefits.

The image consists of two side-by-side screenshots of the GitHub mobile application interface.

Screenshot 1 (Left): Shows the user profile screen for 'ellenschn'. At the top right, there is a circular icon with a bar chart and a gear symbol, which is highlighted by a red arrow labeled '1'. Below the profile picture, there is a 'Set status' button. A red arrow labeled '2' points to the 'Settings' option in the menu below.

Screenshot 2 (Right): Shows the 'Billing and licensing' section of the user settings. This section includes links for 'Overview', 'Usage', 'Budgets and alerts', 'Licensing', 'Payment information', 'Payment history', and 'Additional billing details'. At the bottom, there is a dark grey bar containing the 'Education benefits' link, which is highlighted by a red arrow labeled '4'.

Start an application and follow the prompts. You will need to share your location and upload a proof of enrolment (can be in the form of Student ID, transcripts, enrolment letter, etc.)

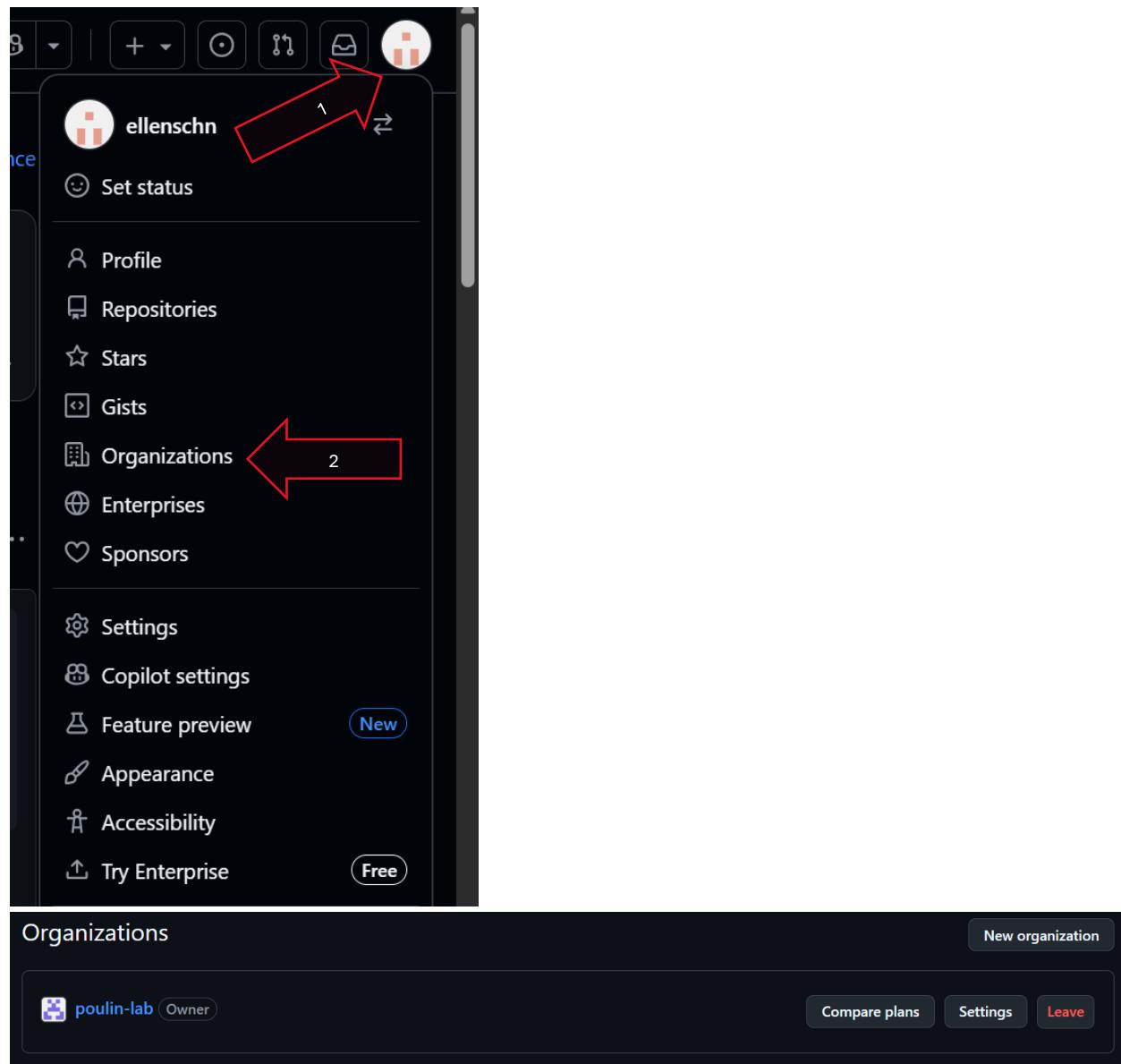
The image shows the 'GitHub Education' application page. At the top, it says 'GitHub Education'. Below that, there is a section titled 'Education Benefits' with a graduation cap icon. It contains the text: 'Complete a teacher or student application to unlock tools and resources for your educational journey.' To the right of this text is a large green arrow pointing right, and next to it is a green button with the text 'Start an application'.

Once your application is approved, you can access all the Education benefits!

Joining the Poulin Lab Organization

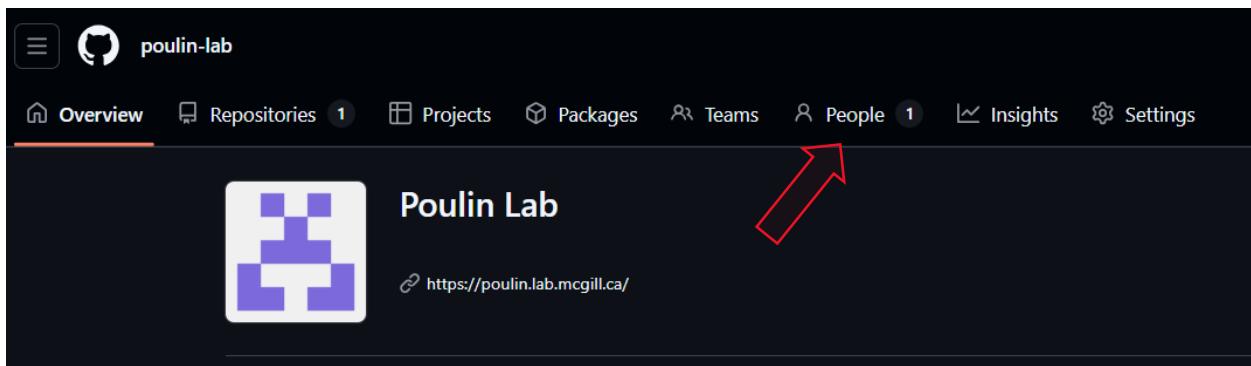
An organization is a shared space between all lab members to add, modify and share their code. To join the Poulin Lab Organization, you must be invited by an organization admin. Admins include JF Poulin and Duncan Spencer. Ensure that you are receiving an invitation to the email you signed up to GitHub with.

To access the organization in GitHub, navigate to the Organizations section, and then click Poulin Lab.



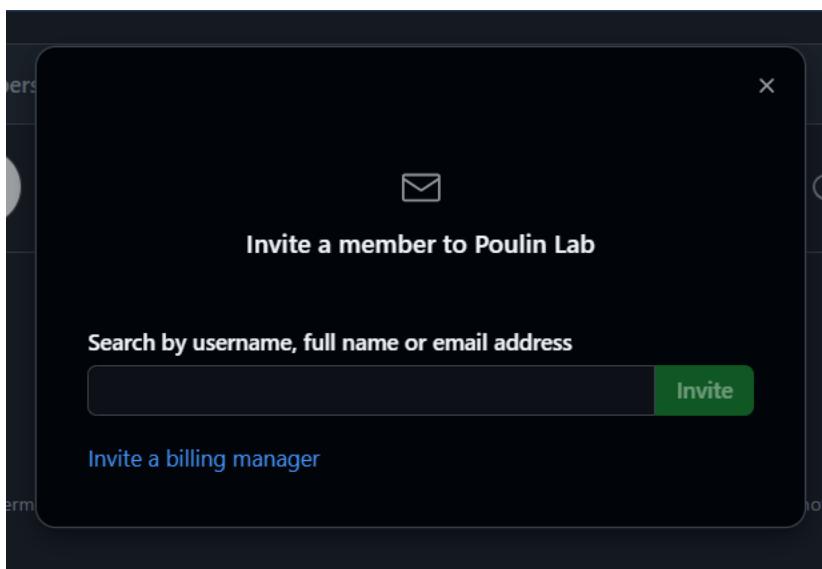
For Admins: How to Invite Members

Start on the main page of the Poulin Lab organization and navigate to the “People” tab.



Members	1
ellenschn	2FA Private Owner 0 teams 0 roles ...

Simply type in their email address and invite!



Note that it will ask you to give the person Member or Admin benefits. The lab organization is configured so that Members can add code, create and delete repos but **cannot invite other collaborators**. An administrator on the other hand, has full access to everything. You can additionally give “admin access” to a person for a particular repository.

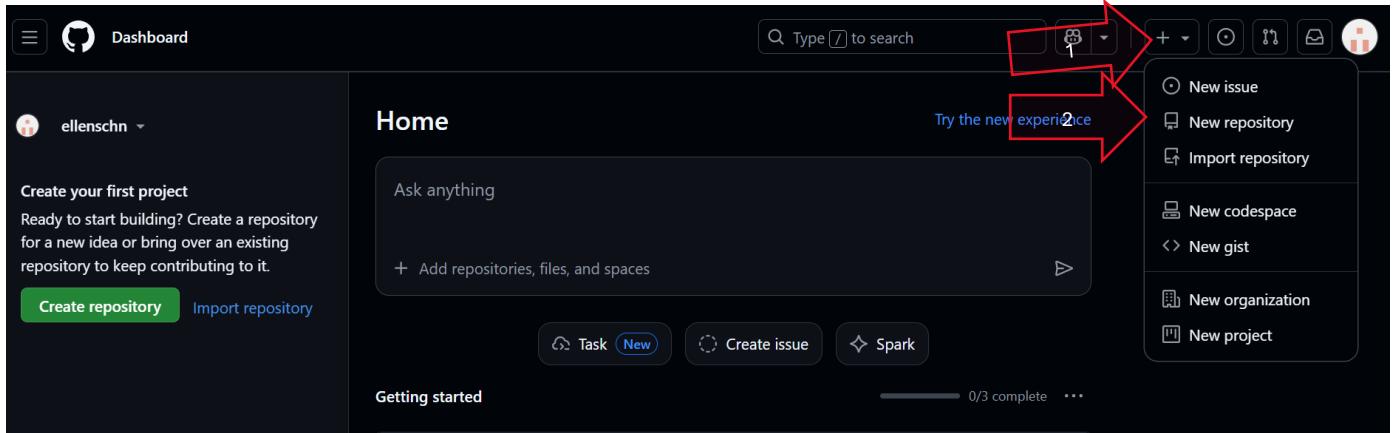
Storing & Sharing Code

All code created and shared should be stored under the lab organization for ease of access and trackability.

Creating Repositories

A repository (or repo) can be thought of like a folder. It will contain related items to a particular experiment or project. Each repository should have a README file to detail information about that particular experiment.

To create a repository, navigate to the “+” button on the right hand side of the screen.



Name the repository accordingly. Ensure that the owner falls under “Poulin-Lab” and not your account so that everyone has access to the repository. Add a description if you like. Visibility should ideally be public, and README should be checked.

Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).
Required fields are marked with an asterisk (*).

1 General

Owner * poulin-lab **Repository name *** /

Great repository names are short and memorable. How about [automatic-octo-spork](#)?

Description

0 / 350 characters

2 Configuration

Choose visibility * Public

Choose who can see and commit to this repository

Add README On

READMEs can be used as longer descriptions. [About READMEs](#)

Add .gitignore No .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

Add license No license

Licenses explain how others can use your code. [About licenses](#)

Create repository

poulin-lab / .github

Type / to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

.github Public

main 1 Branch 0 Tags Go to file Add file Code

ellensch Create README.md 1b68075 · 1 hour ago 2 Commits

profile Create README.md 1 hour ago

README.md Initial commit 1 hour ago

README

.github

All information needed to get started w/ GitHub, in regards to making repositories, adding code, pulling & changing it, along with sharing.

poulin-lab/.github is a special repository. The profile/README.md will appear on the organization's profile.

Edit README Visit profile

About

All information needed to get started w/ GitHub, in regards to making repositories, adding code, pulling & changing it, along with sharing.

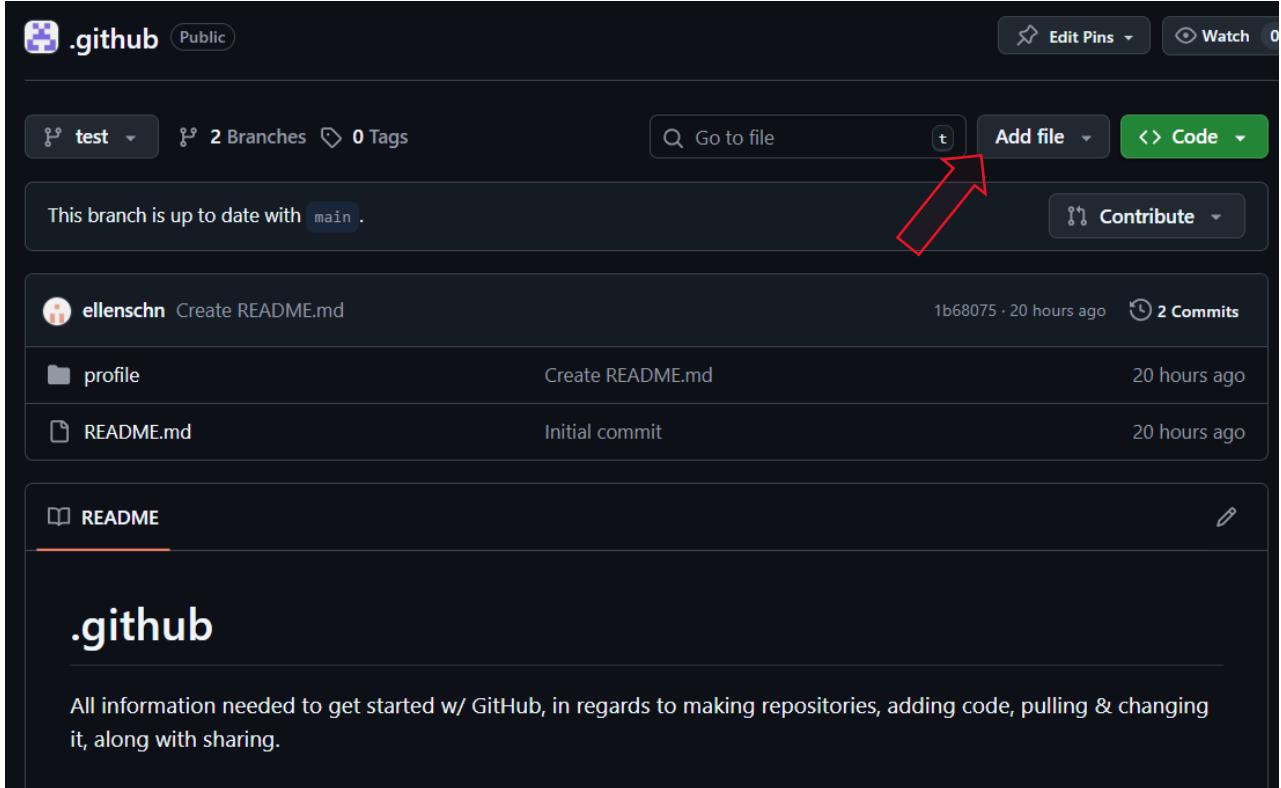
Readme Activity Custom properties

0 stars 0 watching 0 forks

Report repository

Adding Code

You can add code to any branch in a repository. For most cases in this lab, we'll keep the final or official version in the main branch.



A screenshot of a GitHub repository page for '.github'. The page shows two branches: 'test' and 'main'. The 'test' branch is selected. A red arrow points to the 'Add file' button in the top right corner of the header bar. The repository has 2 commits from user 'ellensch'.

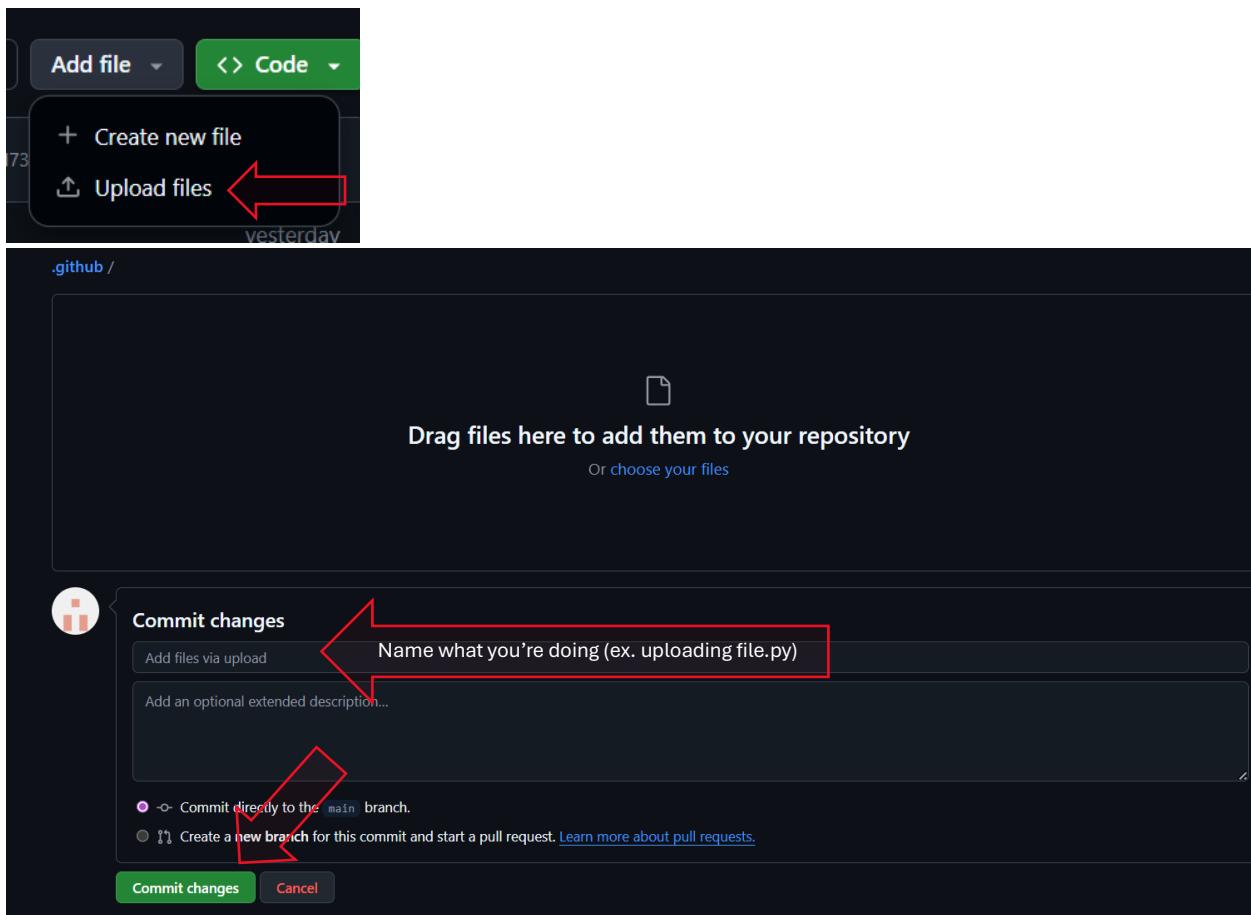
This branch is up to date with `main`.

Commit	File	Date	Author
Create README.md	README.md	1b68075 · 20 hours ago	ellensch
Create README.md	profile	20 hours ago	
Initial commit	README.md	20 hours ago	

README

.github

All information needed to get started w/ GitHub, in regards to making repositories, adding code, pulling & changing it, along with sharing.



Note: you can upload just about anything (doesn't have to be code)!

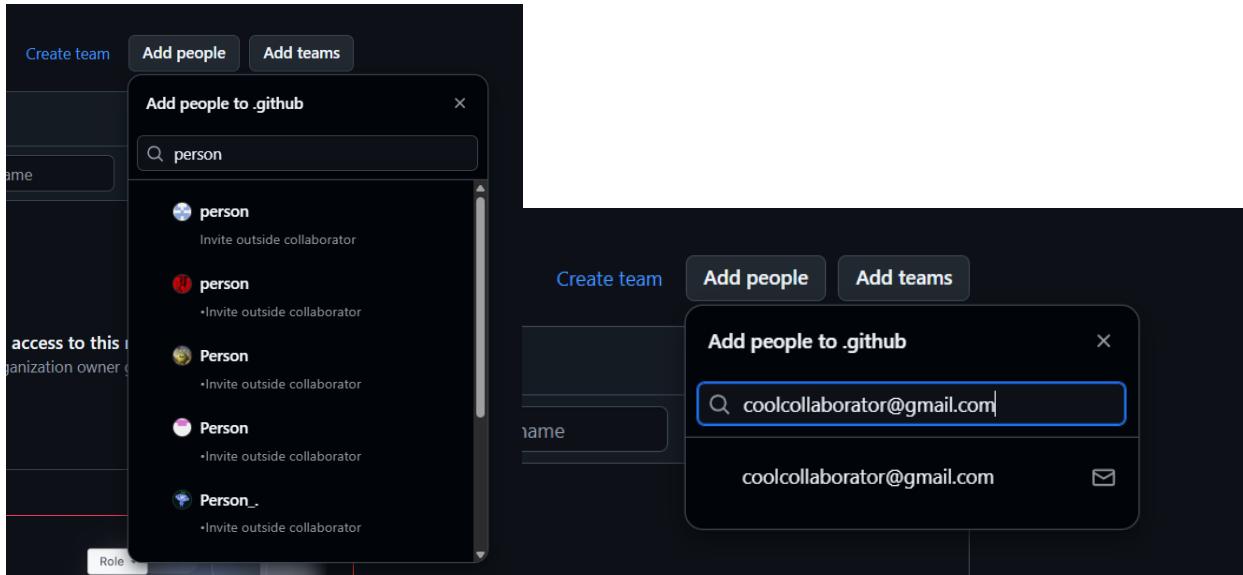
Sharing Code with Members Outside the Lab

Note: this is sharing for a particular repository, if you want someone to have full access to the organization, an administrator should invite them as a member of the lab.

Sharing code can be done to someone's GitHub account or to an email address. To share a repository directly to a person, go to the repository's Settings → Manage Access → Invite collaborators.

A screenshot of a GitHub repository page for the organization 'poulin-lab'. The repository name is '.github' and it is public. The page shows a list of commits, including one from 'ellenschn' to 'Create README.md'. The main content area displays the 'README' file with the heading '.github' and a description: 'All information needed to get started w/ GitHub, in regards to making repositories, adding code, pulling & changing it, along with sharing.' On the right side, there is an 'About' section with repository statistics: 1 branch, 0 tags, 2 commits, 0 forks, 0 stars, 0 watching, and 0 forks. A red arrow points from the top of the image towards the 'Edit Pins' button in the header.

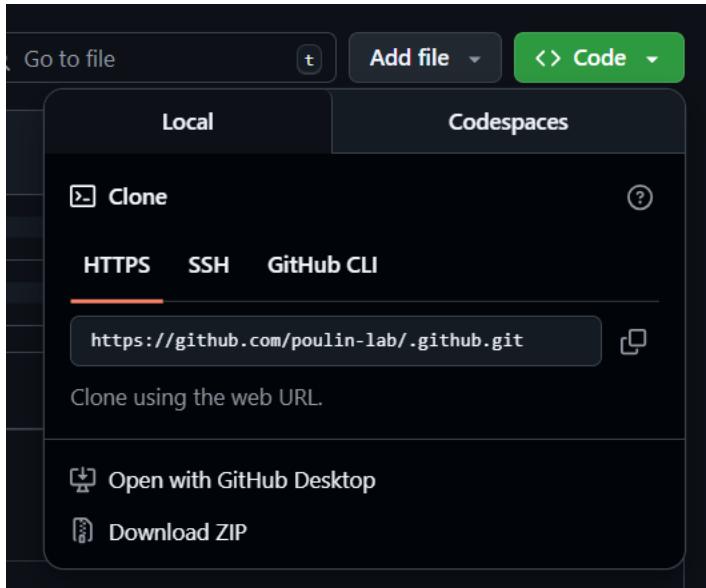
A screenshot of the 'General' settings page for the '.github' repository. The left sidebar contains sections for Access, Collaborators and teams (which has a red arrow pointing to it), Moderation options, Code and automation (with branches, tags, rules, actions, models, webhooks, copilot, environments, pages, and custom properties), and Security (with advanced security, code quality, and deploy keys). The main content area shows the 'Collaborators and teams' section, which is a 'Public repository'. It details access roles: 'Base role' (Read, 1 member), 'Direct access' (0 teams or members, only Owners can contribute), and 'Organization access' (0 users and 0 teams). Below this is the 'Manage access' section, which is currently set to 'Organization access'. It shows 0 actors, a search bar, and filters for Type: All and Role: All. A red arrow points from the bottom of the image towards the 'Organization access' tab in the 'Manage access' section.



You can search to find people who already have a GitHub account or invite people directly through email. You can additionally modify someone's access under the "Manage Access" box.

If the repository is public, an alternative is through a URL link. Note that this only allows others to see the code, they can not modify or add anything.

A screenshot of a GitHub repository page for '.github'. The repository is public. At the top, there are buttons for 'Edit Pins', 'Watch', and 'Code'. A red arrow points to the 'Contribute' button, which is located next to the 'Add file' button. The repository has 2 branches and 0 tags. It shows a commit from 'ellenschn' creating a README.md file 20 hours ago. The repository also contains a 'profile' folder and an 'Initial commit' for the README.md file. A section titled '.github' contains the text: 'All information needed to get started w/ GitHub, in regards to making repositories, adding code, pulling & changing it, along with sharing.'



Simply copy the URL and anyone will be able to view the repository!

Modifying Code and Development

Code uploaded to GitHub can be modified and changed as needed. The best way to approach this is by creating branches off of main repositories, committing new changes, and sending pull requests to other lab members.

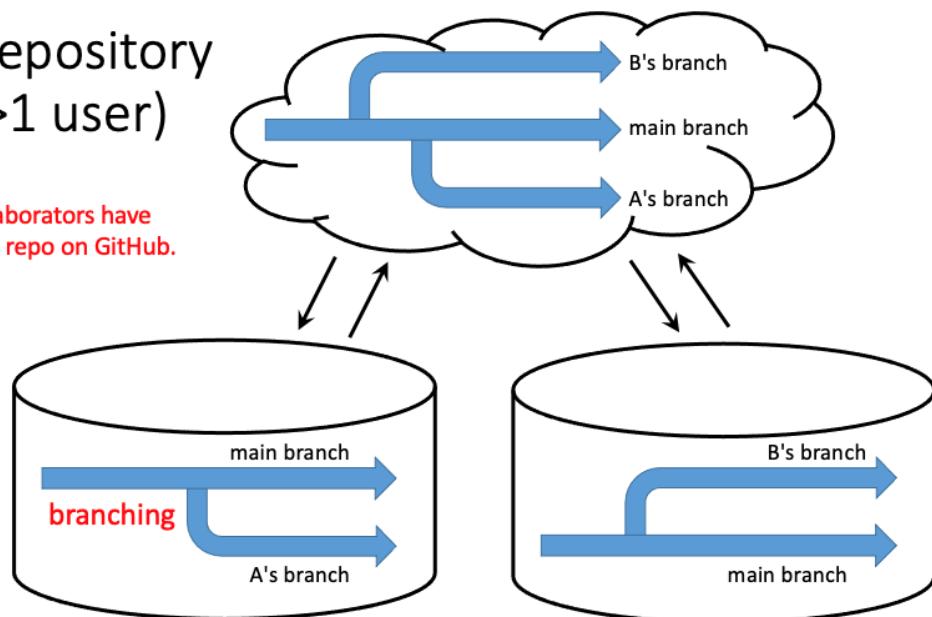
Creating Branches Within Repositories

A branch is a separate version of the code within a repository. It acts like a workspace where a user can make changes without affecting the main version. Each branch can have its own commits and updates, letting you experiment safely.

By default, all repositories have a main branch where the source code should be stored. There is no limit on how many branches a repository can have.

Shared repository model (>1 user)

Used when all collaborators have write access to the repo on GitHub.



Version conflicts avoided when working on separate branches

To create a new branch, go to the dropdown menu that says main. Create a name for your branch and click create.

A screenshot of a GitHub repository page for a repository named '.github'. The repository is public. At the top, there are buttons for 'Edit Pins' and 'Watch'. Below the header, the main branch is set to 'main' (indicated by a red arrow), with 1 Branch and 0 Tags. A search bar says 'Go to file' and there are buttons for 'Add file' and 'Code'. The repository has 2 commits from user 'ellenschn' (Create README.md) made 20 hours ago. There are also two other commits: 'profile' (Create README.md) and 'README.md' (Initial commit), both made 20 hours ago. The README file is open, showing the text: '.github All information needed to get started w/ GitHub, in regards to making repositories, adding code, pulling & changing it, along with sharing.'

A screenshot of the GitHub interface showing the creation of a new branch. A modal window titled 'Switch branches/tags' is open. It contains a search bar with 'test' and a text input field 'Name it as you'd like'. Below the search bar are tabs for 'Branches' and 'Tags'. A button at the bottom says 'Create branch test from main'. Red arrows point to each of these elements: the search bar, the text input, the 'Branches' tab, and the 'Create branch' button. The background shows the same '.github' repository page as the first screenshot.

.github Public

2 Branches 0 Tags

Says *test* instead of *main*

This branch is up to date with *main*

ellenschn Create README.md 1b68075 · 20 hours ago 2 Commits

profile Create README.md 20 hours ago

README.md Initial commit 20 hours ago

README

.github

All information needed to get started w/ GitHub, in regards to making repositories, adding code, pulling & changing it, along with sharing.

A new branch starts as an exact copy of the main branch, but any changes you make in the new branch won't affect the main branch. If the main branch gets updated after your branch is created, you can pull those updates into your branch as needed. If you want your branch's changes to become part of the main branch, you'll need to merge the branches.

To read more about branches: <https://heardlibrary.github.io/digital-scholarship/manage/control/github/branch/>

Committing Changes

In GitHub, saving any changes is called "committing". Each commit has a title and an optional description (which is what you likely did just now by uploading code!) This allows for everyone to be aware of what changes are happening in the organization.

To modify any file, simply click on it and change it as you'd like. Once you're done, you must click on "commit changes" and write a commit message that describes what you have done.

.github Public

test 2 Branches 0 Tags

Go to file Add file Code

This branch is up to date with main.

Contribute

ellenschn	Create README.md	1b68075 · 20 hours ago	2 Commits
profile	Create README.md	20 hours ago	
README.md	Virtual commit	20 hours ago	
README			edit

Click on whatever document you want to modify

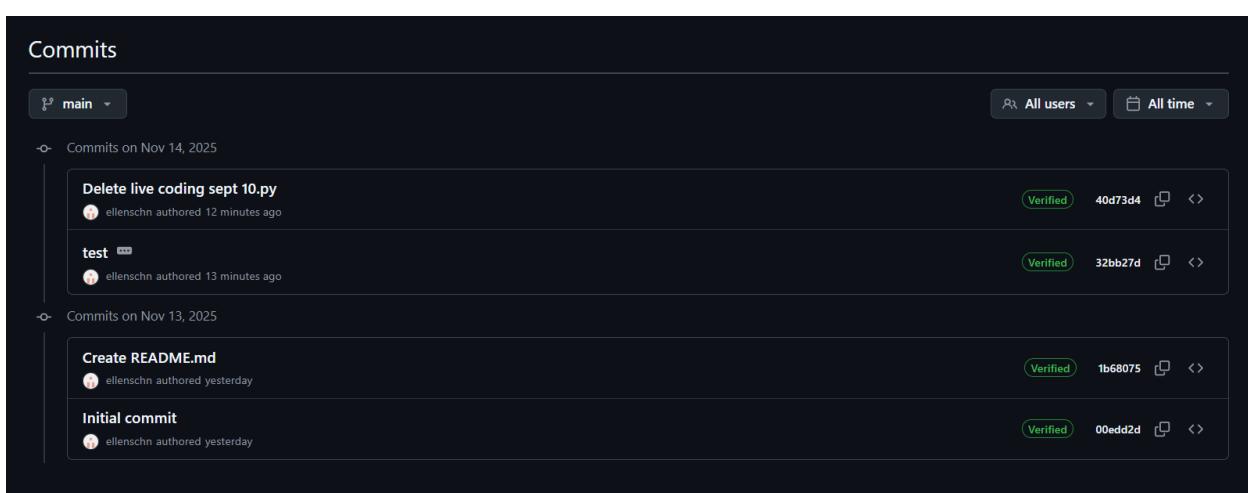
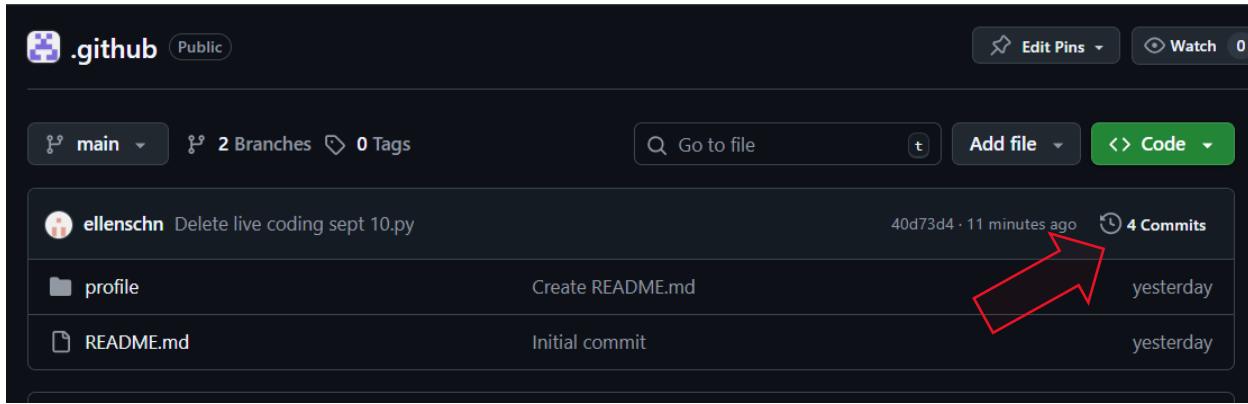
.github

All information needed to get started w/ GitHub, in regards to making repositories, adding code, pulling & changing it, along with sharing.

The image consists of three vertically stacked screenshots of the GitHub README editor interface, showing different stages of the commit process.

- Screenshot 1: Preview Stage**
 - Shows the file content: ".github" and a descriptive text block.
 - Shows the top right toolbar with options like Raw, Copy, Paste, and Edit.
 - A red arrow points from the "Raw" button to the "Edit" button in the second screenshot.
- Screenshot 2: Edit Stage**
 - Shows the code editor with the file content.
 - Shows the top right toolbar with "Cancel changes" and "Commit changes..." buttons.
 - A red arrow points from the "Commit changes..." button to the "Commit changes" dialog in the third screenshot.
 - A red box labeled "Modify as needed" surrounds the code editor area.
- Screenshot 3: Commit Changes Dialog**
 - Shows the "Commit changes" dialog with fields for "Commit message" (containing "Update README.md") and "Name your change".
 - Shows the "Extended description" field with placeholder text.
 - Shows the branch selection radio buttons:
 - Commit directly to the `main` branch
 - Create a **new branch** for this commit and start a **pull request** [Learn more about pull requests](#)
 - A red arrow points from the "Create a new branch" radio button to the "Commit changes" button.
 - At the bottom are "Cancel" and "Commit changes" buttons.

All recent commits are listed at the top of all files. Click on the number of commits to see a full list of changes.



Commits

main

All users All time

Commits on Nov 14, 2025

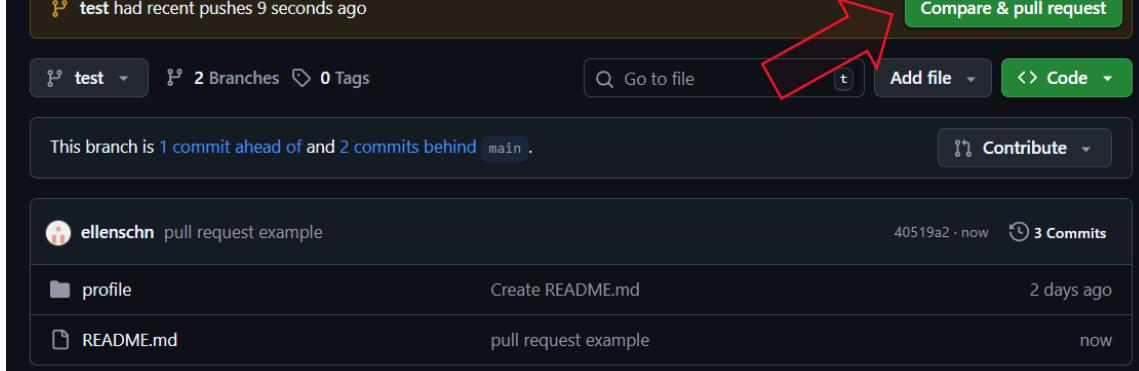
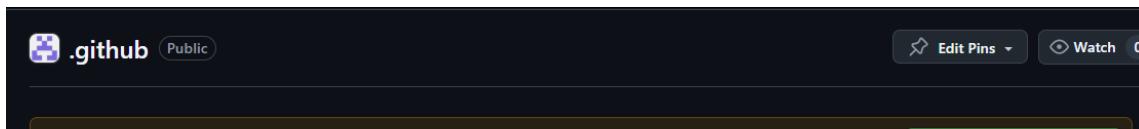
- Delete live coding sept 10.py
ellenschn authored 12 minutes ago Verified 40d73d4
- test
ellenschn authored 13 minutes ago Verified 32bb27d

Commits on Nov 13, 2025

- Create README.md
ellenschn authored yesterday Verified 1b68075
- Initial commit
ellenschn authored yesterday Verified 00edd2d

Pull Requests & Merging Branches

Pull requests are for when you want to merge your branch with the main branch. Creating a pull request will signal to other collaborators for their input and display all the commits you are proposing. To create a pull request, first go into the branch that you want to “pull” into the main branch. There should be a button at the top saying “Compare & pull request”.



test

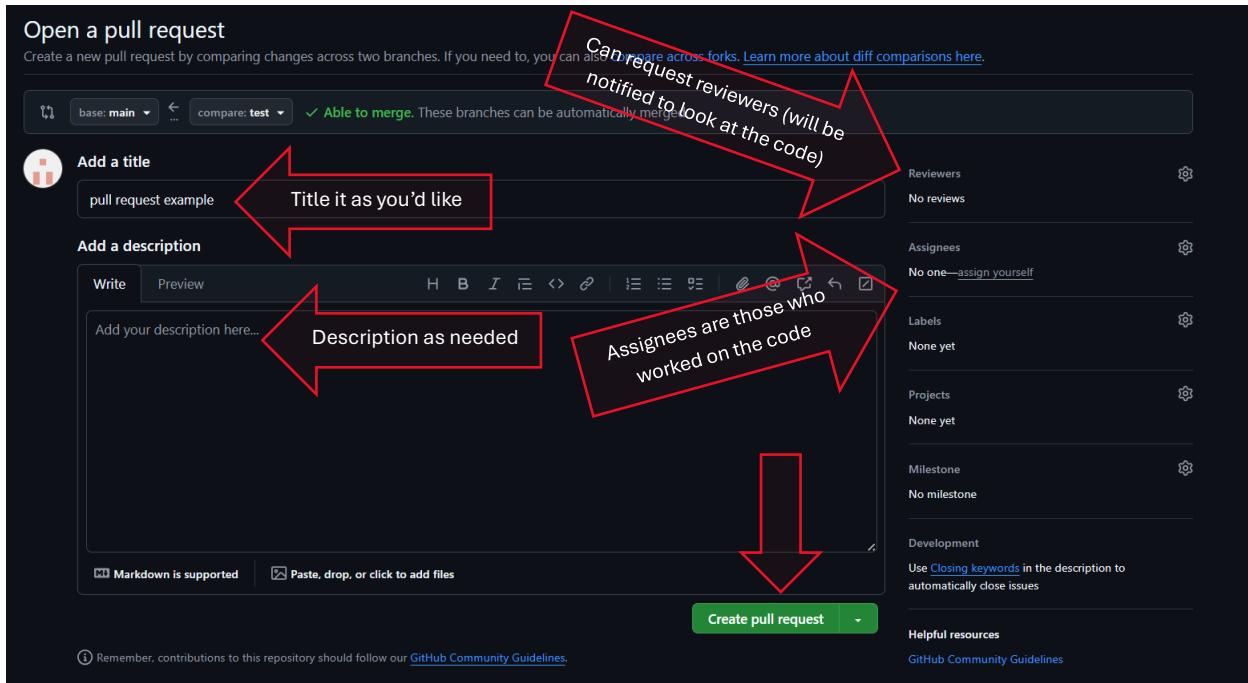
2 Branches 0 Tags

Go to file Add file Code

This branch is 1 commit ahead of and 2 commits behind main.

ellenschn pull request example
profile Create README.md 2 days ago

README.md pull request example now



pull request example #1

Open ellenschn wants to merge 1 commit into `main` from `test`

The screenshot shows a GitHub pull request with the title 'pull request example #1'. It indicates 1 commit, 0 checks, and 1 file changed. A red arrow points from the 'See exactly what gets added' annotation to the 'Files changed' section. A callout box with a red border contains the text 'Look at the commits proposed'. Below the commit list, it says 'No description provided.' and shows a smiley face emoji. The user 'ellenschn' has commented 'pull request example' and self-assigned the PR. A red arrow points from the 'ellenschn commented now' annotation to the comment. At the bottom, a green box shows 'No conflicts with base branch' and 'Merging can be performed automatically.' A red arrow points from the 'Merge pull request' button to the 'Add a comment' section below. The 'Add a comment' section includes a 'Write' tab, a preview area, and a placeholder 'Add your comment here...'. A red arrow points from the 'Add to the conversation about the code' annotation to the 'Add your comment here...' placeholder. There is also a 'Still in progress? Convert to draft' link at the bottom right.

pull request example #1

A screenshot of a GitHub pull request page. At the top, it says "ellenschin wants to merge 1 commit into `main` from `test`". Below this, there are tabs for "Conversation" (0), "Commits" (1), "Checks" (0), and "Files changed" (1). A red arrow points to the "Files changed" tab with the text "Say anything you want about the changes". The main area shows a diff of `README.md`. The diff shows three additions:

```
diff --git a/README.md b/README.md
index 1234567..8901234 100644
--- a/README.md
+++ b/README.md
@@ -1,2 +1,3 @@
 1   1  # .github
 2   2  All information needed to get started w/ GitHub, in regards to making repositories, adding code, pulling & changing it, along with sharing.
 3 + editing! will enter main branch!
```

A screenshot of the GitHub review interface. It shows a modal window titled "Finish your review" with a "Write" tab selected. The text area contains "Leave a comment". Below it, it says "Markdown is supported" and "Paste, drop, or click to add files". To the right, there's a sidebar with a "Review changes" button and a "Viewed" section. At the bottom left of the modal, there are three radio buttons: "Comment" (selected), "Approve", and "Request changes". At the bottom right is a "Submit review" button.

If everything looks good, return to the conversation tab and click on “Merge pull request”.

A screenshot of a pull request merge dialog box. At the top, a green icon with a circular arrow and a checkmark is followed by the text "No conflicts with base branch" and "Merging can be performed automatically." Below this is a green button labeled "Merge pull request". A red arrow points from the text above the button down to the button itself. To the right of the button, a dropdown arrow is visible. Below the button, a message says "You can also merge this with the command line. [View command line instructions](#)". At the bottom right of the dialog, there is a link "Still in progress? [Convert to draft](#)".

Commit message
Merge pull request #1 from poulin-lab/test

Extended description
pull request example

This commit will be authored by ellen.s.chen@mail.mcgill.ca.

Confirm merge Cancel

A screenshot of a pull request merge history. The title is "pull request example #1". A purple button labeled "Merged" is followed by the text "ellenschn merged 1 commit into main from test now". Below this are navigation links: "Conversation 0", "Commits 1", "Checks 0", and "Files changed 1".

ellenschn commented 5 minutes ago
No description provided.

pull request example

Verified 40519a2

ellenschn self-assigned this 5 minutes ago

ellenschn merged commit 42a7c8e into main now

Revert

Pull request successfully merged and closed
You're all set — the test branch can be safely deleted.

Delete branch

Now the two branches have successfully merged! Note that the “pulled” branch doesn’t get deleted. If you no longer have use for the branch, you will have to manually delete it.

Forking is another way to contribute to workflows. This is a more advanced technique, feel free to read about it here: <https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/working-with-forks/fork-a-repo>