Code Review for Scientific Software

experiences building an online tutorial

Helen Kershaw
DAReS, NSF NCAR

Data
Assimilation
Research
Testbed

Terminology

BSSwF Better Scientific Software Fellowship

NSF National Science Foundation
NCAR National Center for Atmospheric Research

UCAR University Corporation for Atmospheric Research

SEA Software Engineering Assembly (UCAR/NCAR)

DART Data assimilation Research Testbed

DAReS Data Assimilation Research Section

AMS American Meteorological Society

Goals

- Tell you about my BSSwF project
- Share my experience building the tutorial
- Share practice and experience with code review from UCAR SEA

Goals

- Tell you about my BSSwF project
- Share my experience building the tutorial
- Share practice and experience with code review from UCAR SEA

- Get you to think about code review

Goals

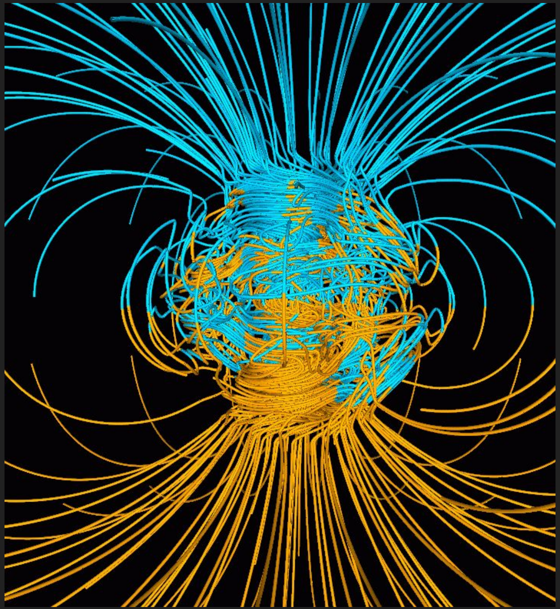
- Tell you about my BSSwF project
- Share my experience building the tutorial
- Share practice and experience with code review from UCAR SEA

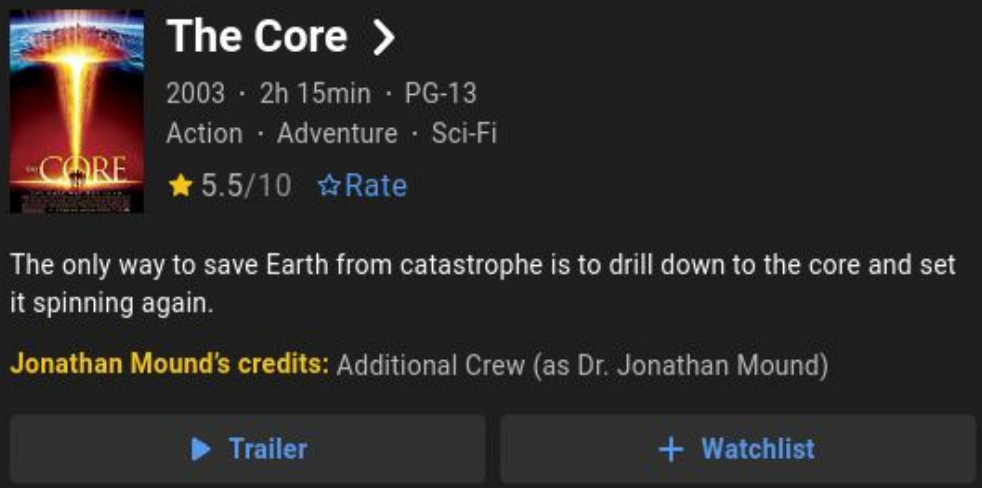
- Get you to think about code review
- And the joy of open source software

About me

hkershaw@ucar.edu

About me





The Core

2003 · 2h 15min · PG-13

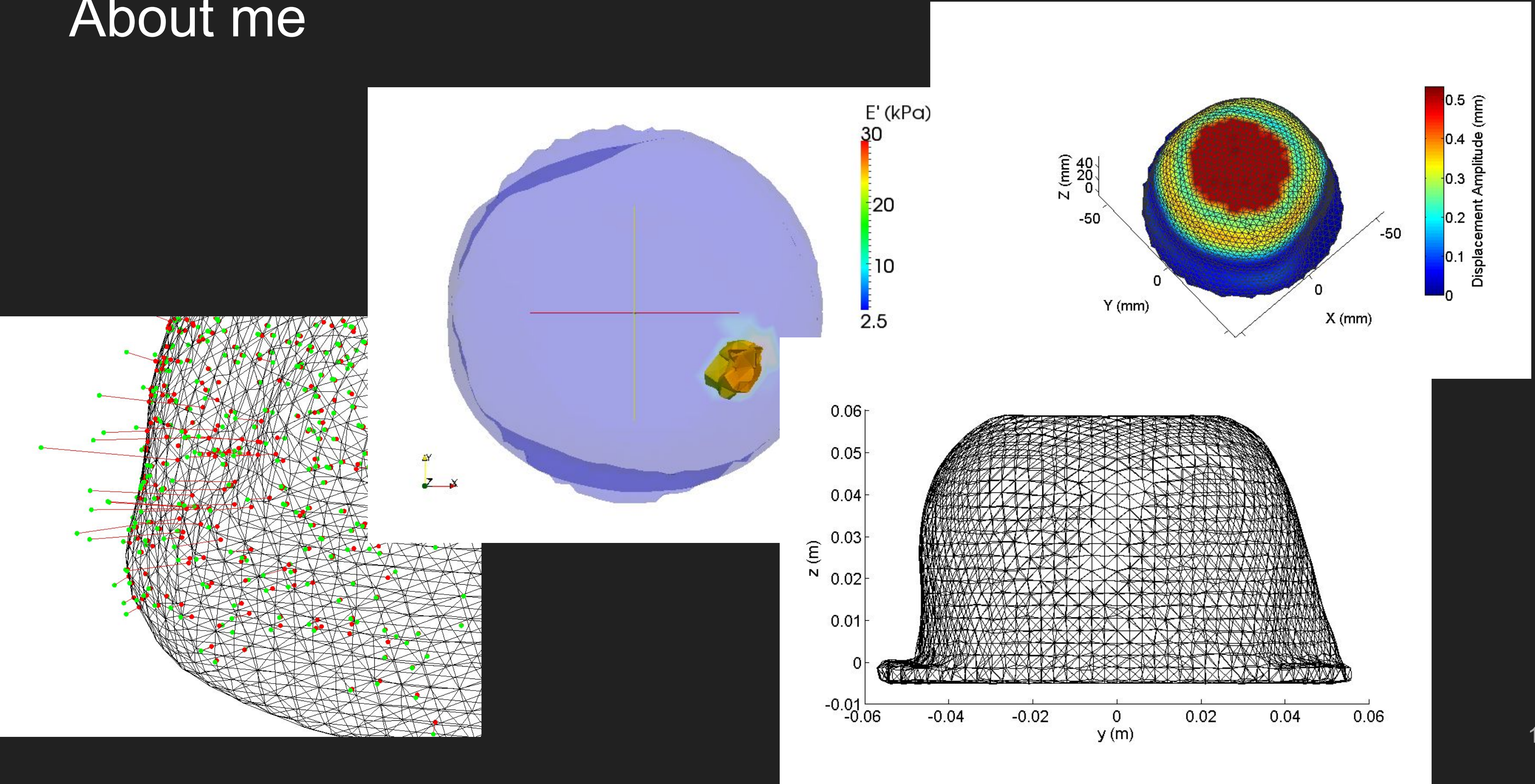
Action · Adventure · Sci-Fi

5.5/10

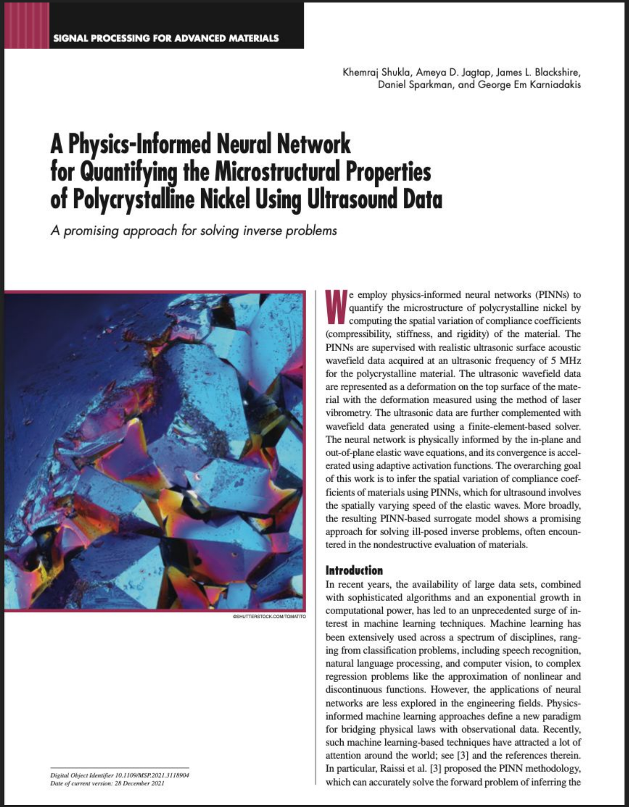
The only way to save Earth from catastrophe is to drill down to the core and set it spinning again.

Jonathan Mound's credits: Additional Crew (as Dr. Jonathan Mound)

About me


E' (kPa)
30
20
10
2.5
Z (mm)
40
20
0
-50
Y (mm)
0
X (mm)
0
-50
Displacement Amplitude (mm)
0.5
0.4
0.3
0.2
0.1
0
z (m)
0.06
0.05
0.04
0.03
0.02
0.01
0
-0.01
-0.06
-0.04
-0.02
0
0.02
0.04
0.06
y (m)

About me



SIGNAL PROCESSING FOR ADVANCED MATERIALS

Khemraj Shukla, Ameya D. Jagtap, James L. Blackshire,
Daniel Sparkman, and George Em Karniadakis

A Physics-Informed Neural Network for Quantifying the Microstructural Properties of Polycrystalline Nickel Using Ultrasound Data

A promising approach for solving inverse problems

©SHUTTERSTOCK.COM/TOMATITO

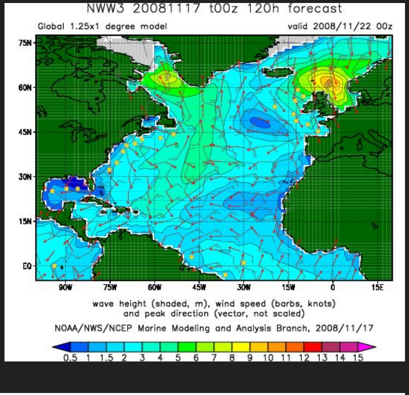
We employ physics-informed neural networks (PINNs) to quantify the microstructure of polycrystalline nickel by computing the spatial variation of compliance coefficients (compressibility, stiffness, and rigidity) of the material. The PINNs are supervised with realistic ultrasonic surface acoustic wavefield data acquired at an ultrasonic frequency of 5 MHz for the polycrystalline material. The ultrasonic wavefield data are represented as a deformation on the top surface of the material with the deformation measured using the method of laser vibrometry. The ultrasonic data are further complemented with wavefield data generated using a finite-element-based solver. The neural network is physically informed by the in-plane and out-of-plane elastic wave equations, and its convergence is accelerated using adaptive activation functions. The overarching goal of this work is to infer the spatial variation of compliance coefficients of materials using PINNs, which for ultrasound involves the spatially varying speed of the elastic waves. More broadly, the resulting PINN-based surrogate model shows a promising approach for solving ill-posed inverse problems, often encountered in the nondestructive evaluation of materials.

Introduction

In recent years, the availability of large data sets, combined with sophisticated algorithms and an exponential growth in computational power, has led to an unprecedented surge of interest in machine learning techniques. Machine learning has been extensively used across a spectrum of disciplines, ranging from classification problems, including speech recognition, natural language processing, and computer vision, to complex regression problems like the approximation of nonlinear and discontinuous functions. However, the applications of neural networks are less explored in the engineering fields. Physics-informed machine learning approaches define a new paradigm for bridging physical laws with observational data. Recently, such machine learning-based techniques have attracted a lot of attention around the world; see [3] and the references therein. In particular, Raissi et al. [3] proposed the PINN methodology, which can accurately solve the forward problem of inferring the

Digital Object Identifier 10.1109/MSP.2021.3118904
Date of current version: 28 December 2021

US
RSE



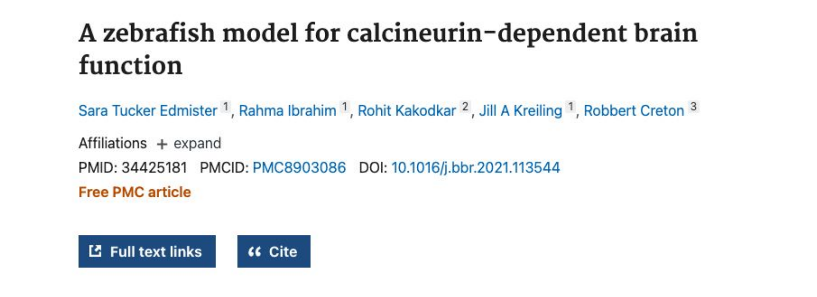
NWW3 20081117 t00z 120h forecast

Global 1.25x1 degree model

valid 2008/11/22 00z

wave height (shaded, m), wind speed (barbs, knots)
and peak direction (vector, not scaled)

NOAA/NWS/NCEP Marine Modeling and Analysis Branch, 2008/11/17



A zebrafish model for calcineurin-dependent brain function

Sara Tucker Edmister [1], Rahma Ibrahim [1], Rohit Kakodkar [2], Jill A Kreiling [1], Robbert Creton [3]

Affiliations + expand

PMID: 34425181 PMCID: PMC8903086 DOI: 10.1016/j.bbr.2021.113544

Free PMC article

DART

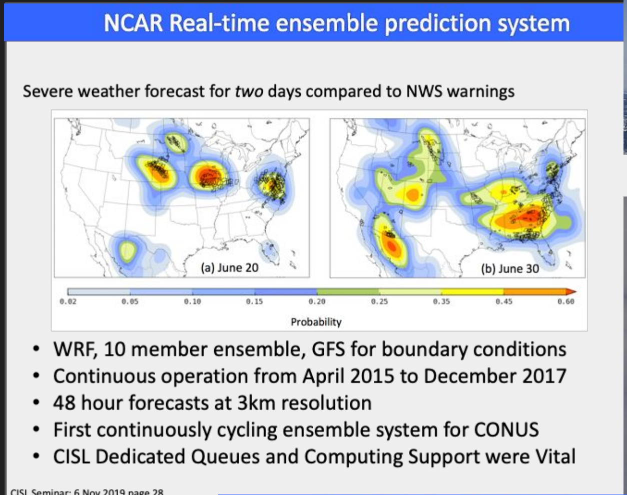
Data Assimilation Research Testbed

Cross-lab

Cross-institution

Cross-country

Cross-world

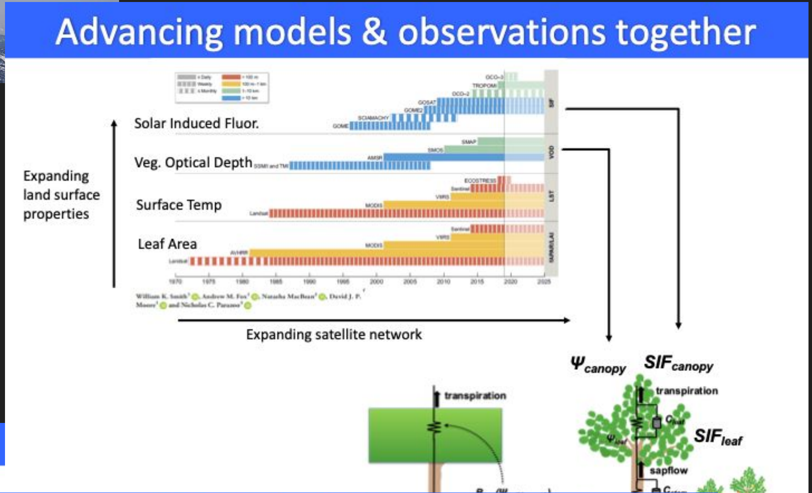


NCAR Real-time ensemble prediction system

Severe weather forecast for *two* days compared to NWS warnings

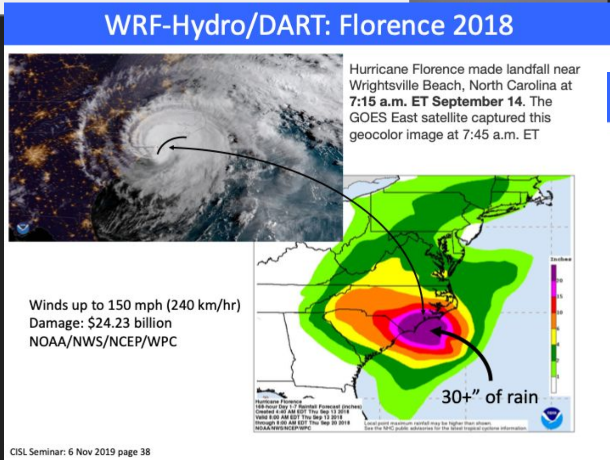
- WRF, 10 member ensemble, GFS for boundary conditions
- Continuous operation from April 2015 to December 2017
- 48 hour forecasts at 3km resolution
- First continuously cycling ensemble system for CONUS
- CISL Dedicated Queues and Computing Support were Vital

CISL Seminar: 6 Nov 2019 page 28



Advancing models & observations together

William K. Smith[1], Andrew M. Fox[1], Natasha MacBean[2], David J. P. Moore[1] and Nicholas C. Parazoo[3]

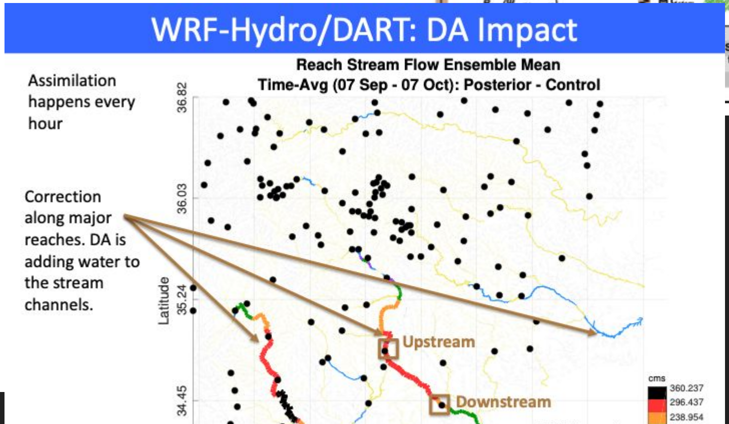


WRF-Hydro/DART: Florence 2018

Hurricane Florence made landfall near Wrightsville Beach, North Carolina at **7:15 a.m. ET September 14**. The GOES East satellite captured this geocolor image at 7:45 a.m. ET

Winds up to 150 mph (240 km/hr)
Damage: $24.23 billion
NOAA/NWS/NCEP/WPC

CISL Seminar: 6 Nov 2019 page 38



WRF-Hydro/DART: DA Impact

Assimilation happens every hour

Correction along major reaches. DA is adding water to the stream channels.

Reach Stream Flow Ensemble Mean
Time-Avg (07 Sep - 07 Oct): Posterior - Control

Featured project: Computational & Information Systems Lab & Research Applications Lab Collaboration

PREDICTING FLOODS AND PROTECTING LIVES

DATA ASSIMILATION FOR THE ENTIRE EARTH SYSTEM

Use ensemble DA techniques with geophysical models spanning the

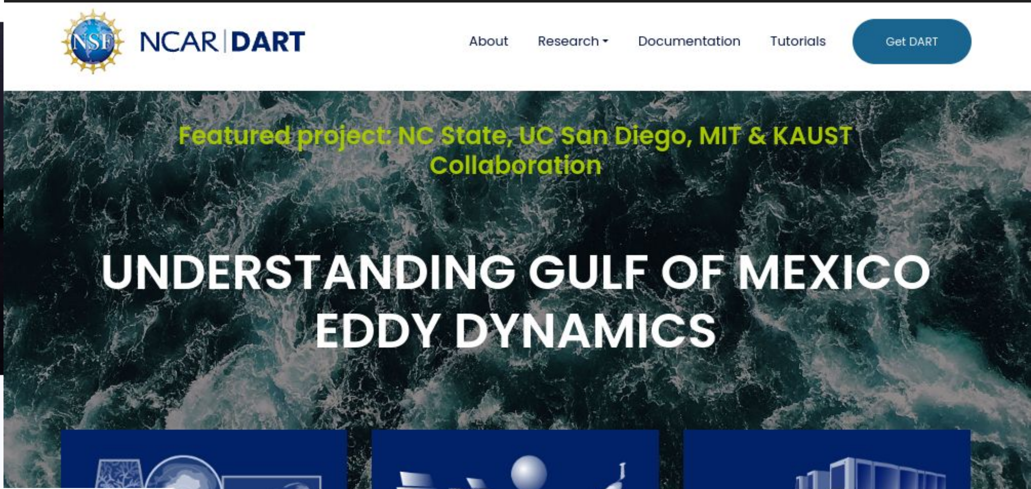
USE DATA FROM ANY SOURCE, TEST MANY ALGORITHMS

Assimilate any suitable observations. Swap out filter and

LEARN ON LAPTOPS, RUN ON SUPERCOMPUTERS

Compile without MPI for conceptual models or with MPI for GCMs on

#carouselIndicators



Featured project: NC State, UC San Diego, MIT & KAUST Collaboration

UNDERSTANDING GULF OF MEXICO EDDY DYNAMICS

IMILATION FOR THE EARTH SYSTEM

le DA techniques with models spanning the

USE DATA FROM ANY SOURCE, TEST MANY ALGORITHMS

Assimilate any suitable observations. Swap out filter and

LEARN ON LAPTOPS, RUN ON SUPERCOMPUTERS

Compile without MPI for conceptual models or with MPI for GCMs on

Featured project: University of Michigan, NCAR, NASA & NRL Collaboration

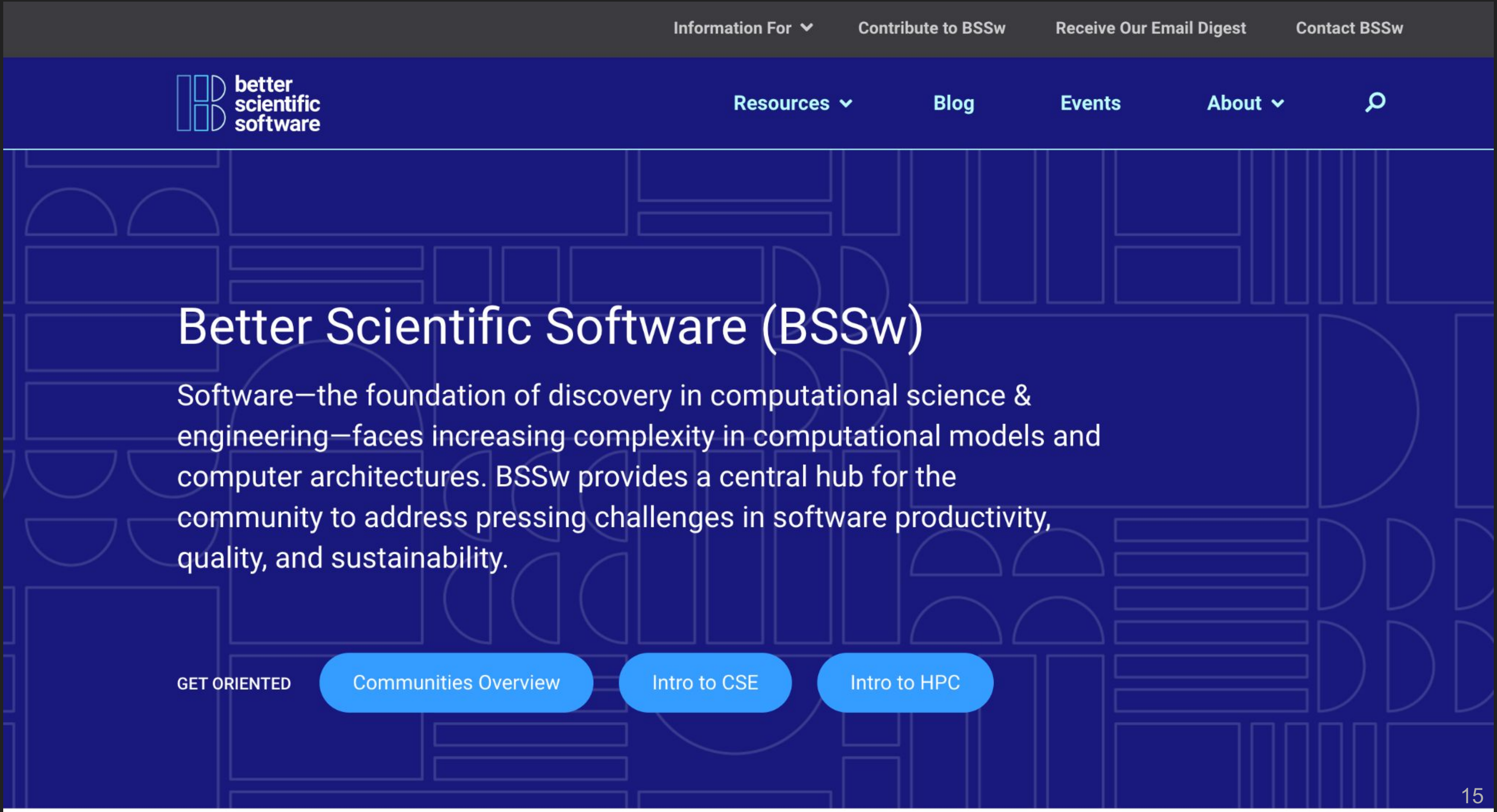
NEXT-GENERATION SPACE WEATHER PREDICTION

DATA ASSIMILATION FOR THE

USE DATA FROM ANY SOURCE,

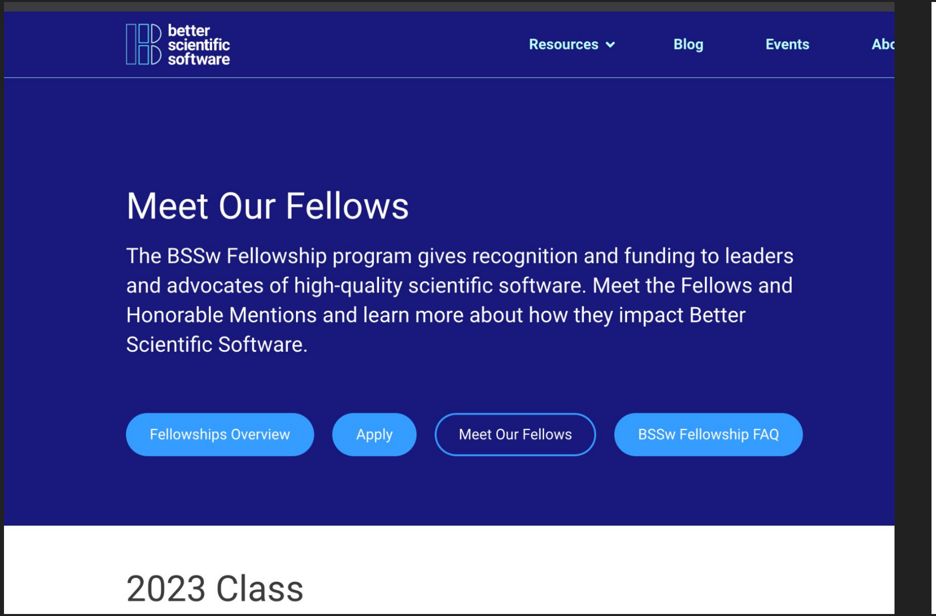
LEARN ON LAPTOPS, RUN ON

dart.ucar.edu



Better Scientific Software (BSSw)

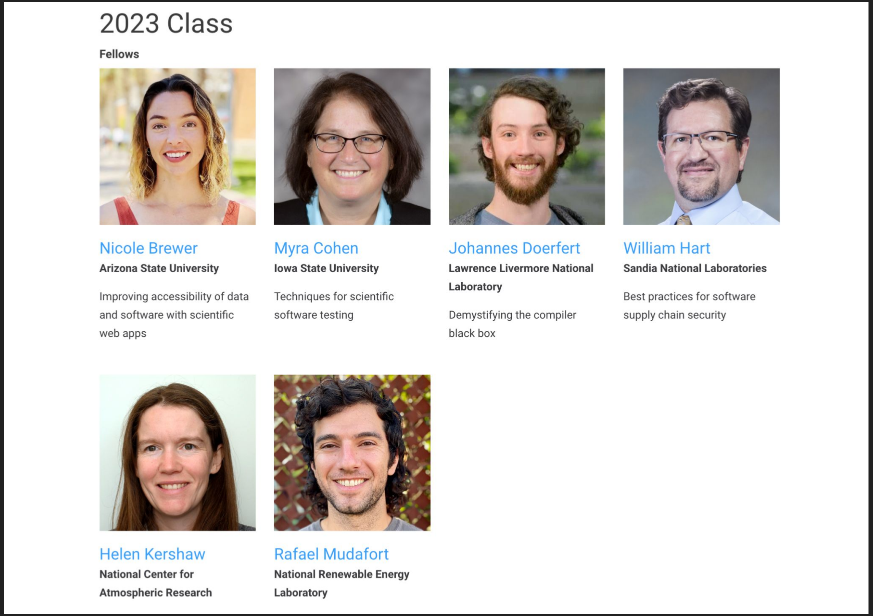
Software—the foundation of discovery in computational science & engineering—faces increasing complexity in computational models and computer architectures. BSSw provides a central hub for the community to address pressing challenges in software productivity, quality, and sustainability.



Meet Our Fellows

The BSSw Fellowship program gives recognition and funding to leaders and advocates of high-quality scientific software. Meet the Fellows and Honorable Mentions and learn more about how they impact Better Scientific Software.

2023 Class



2023 Class

Fellows

Nicole Brewer
Arizona State University
Improving accessibility of data and software with scientific web apps

Myra Cohen
Iowa State University
Techniques for scientific software testing

Johannes Doerfert
Lawrence Livermore National Laboratory
Demystifying the compiler black box

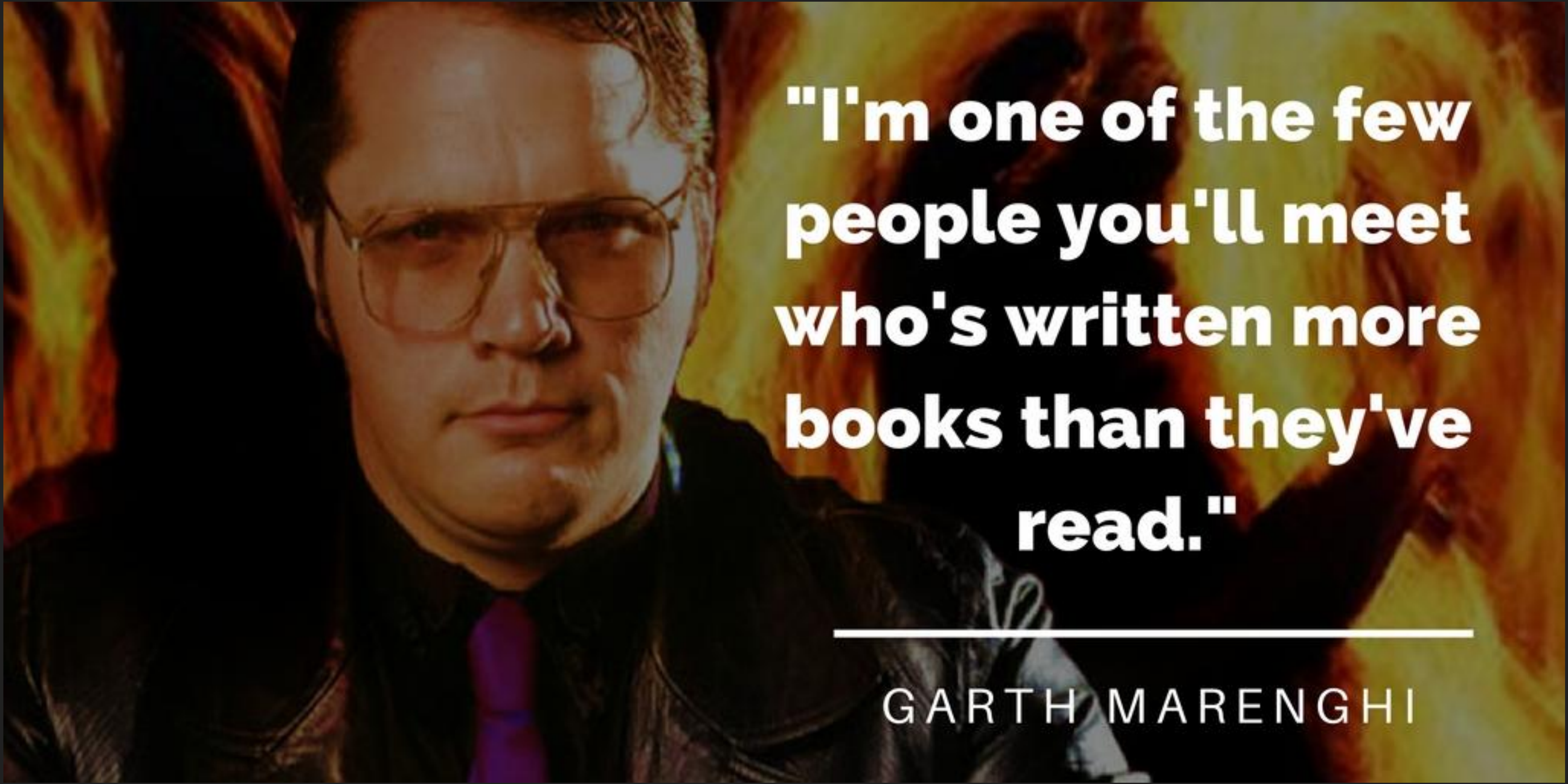
William Hart
Sandia National Laboratories
Best practices for software supply chain security

Helen Kershaw
National Center for Atmospheric Research

Rafael Mudafort
National Renewable Energy Laboratory

https://bssw.io/pages/meet-our-fellows

What problem am I trying to solve?



"I'm one of the few people you'll meet who's written more books than they've read."

GARTH MARENGHI

https://www.channel4.com/programmes/garth-marenghis-darkplace 2004

code-review.org

What outcomes would I like to see?

Outcomes

- People reviewing early and often
- People reviewing each others code
- Comfortable with napkin explanations of code
- Become a better reviewer
- Better code
- Take a look inside
- More open source contributors!

Outcomes

- People reviewing early and often
- People reviewing each others code
- Comfortable with napkin explanations of code
- Become a better reviewer
- Better code
- Take a look inside
- More open source contributors! *Ulterior Motive*

Onboard new contributors to DART

Onboard new contributors to DART

But not be specific to DART

Code review is a skill

Learning several things at once

- The mechanics of git and GitHub
- A new programming language
- New science
- Culture of new team

Learning several things at once

- The mechanics of git and GitHub
- A new programming language
- New science
- Culture of new team

Seasoned professional

Early career

Learning several things at once

- The mechanics of git and GitHub
- A new programming language
- New science
- Culture of new team
- And code review

Seasoned professional

Early career

The Tutorial

code-review.org

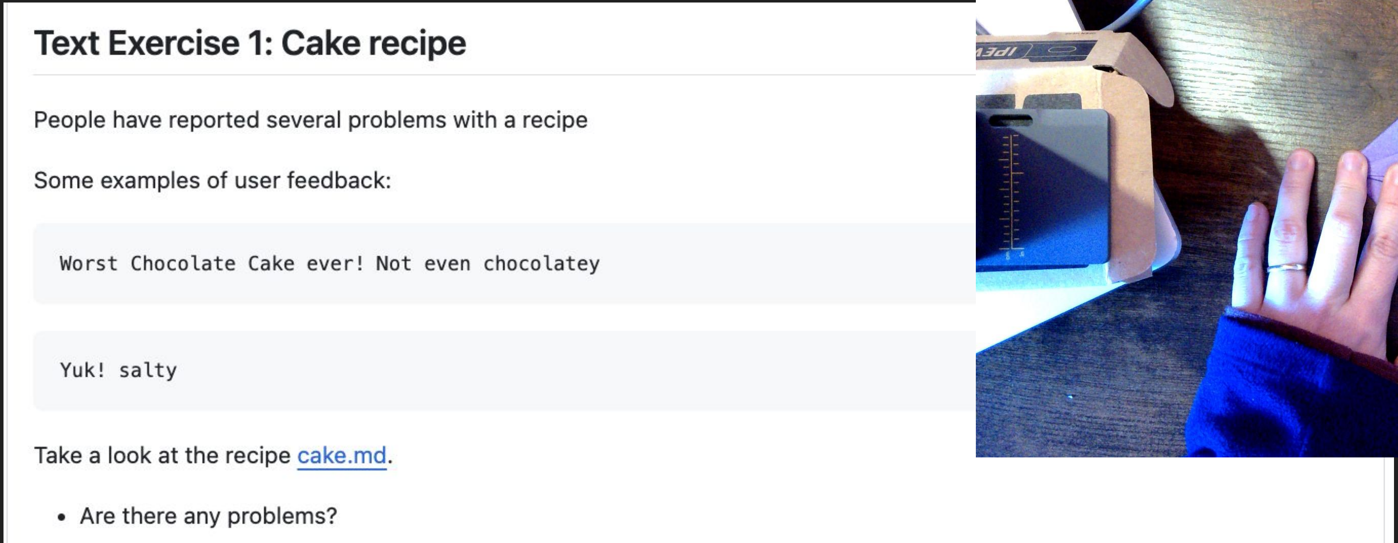
The Tutorial

Three sets of exercises

- No code
- Python
- Fortran

No code exercises

- Cake recipe
- Article on the women’s world cup
- Origami instructions to make a fish



Text Exercise 1: Cake recipe

People have reported several problems with a recipe

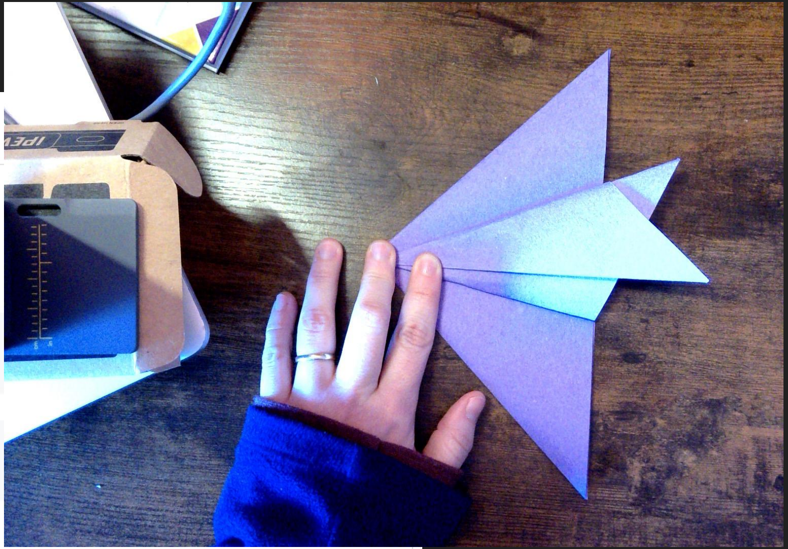
Some examples of user feedback:

```
Worst Chocolate Cake ever! Not even chocolatey
```

```
Yuk! salty
```

Take a look at the recipe cake.md.

- Are there any problems?



The Tutorial

Three sets of exercises

- No code
- Python
- Fortran

The Tutorial

Three sets of exercises

- No code
- Python
- Fortran

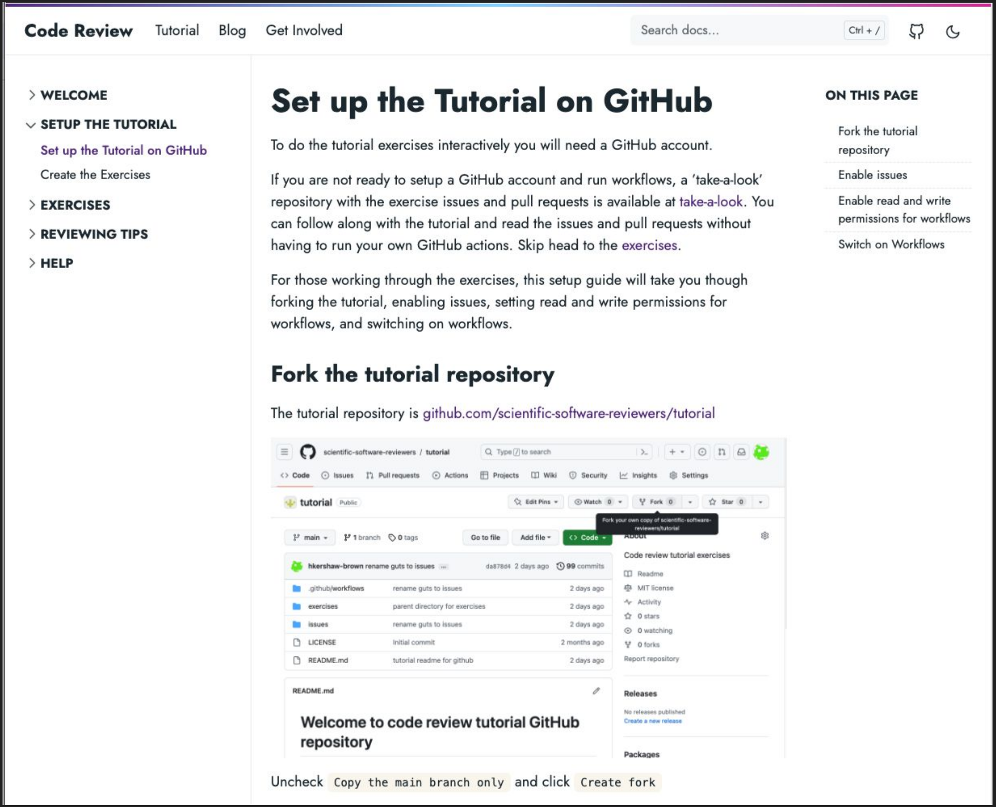
The Tutorial

Three sets of exercises

- No code
- Python
- Fortran

- Issue + prompts
- Pull request + prompts

Setting up the tutorial on GitHub



Set up the Tutorial on GitHub

To do the tutorial exercises interactively you will need a GitHub account.

If you are not ready to setup a GitHub account and run workflows, a 'take-a-look' repository with the exercise issues and pull requests is available at take-a-look. You can follow along with the tutorial and read the issues and pull requests without having to run your own GitHub actions. Skip head to the exercises.

For those working through the exercises, this setup guide will take you though forking the tutorial, enabling issues, setting read and write permissions for workflows, and switching on workflows.

Fork the tutorial repository

The tutorial repository is github.com/scientific-software-reviewers/tutorial

hkershaw-brown rename guts to issues da878d4 2 days ago 99 commits

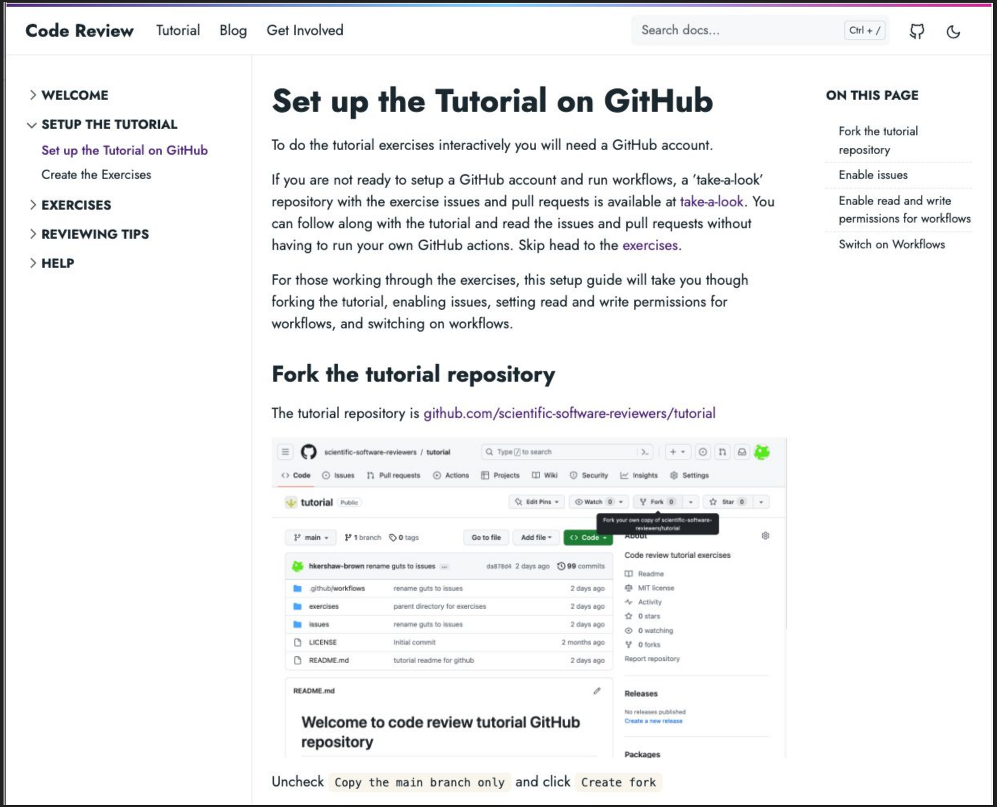
.github/workflows	rename guts to issues	2 days ago
exercises	parent directory for exercises	2 days ago
issues	rename guts to issues	2 days ago
LICENSE	Initial commit	2 months ago
README.md	tutorial readme for github	2 days ago

README.md

Welcome to code review tutorial GitHub repository

Uncheck `Copy the main branch only` and click `Create fork`

- Fork
- All branches
- Enable workflows
- …



Set up the Tutorial on GitHub

To do the tutorial exercises interactively you will need a GitHub account.

If you are not ready to setup a GitHub account and run workflows, a 'take-a-look' repository with the exercise issues and pull requests is available at take-a-look. You can follow along with the tutorial and read the issues and pull requests without having to run your own GitHub actions. Skip head to the exercises.

For those working through the exercises, this setup guide will take you though forking the tutorial, enabling issues, setting read and write permissions for workflows, and switching on workflows.

Fork the tutorial repository

The tutorial repository is github.com/scientific-software-reviewers/tutorial

hkershaw-brown rename guts to issues ... da878d4 2 days ago 99 commits

.github/workflows	rename guts to issues	2 days ago
exercises	parent directory for exercises	2 days ago
issues	rename guts to issues	2 days ago
LICENSE	Initial commit	2 months ago
README.md	tutorial readme for github	2 days ago

README.md

Welcome to code review tutorial GitHub repository

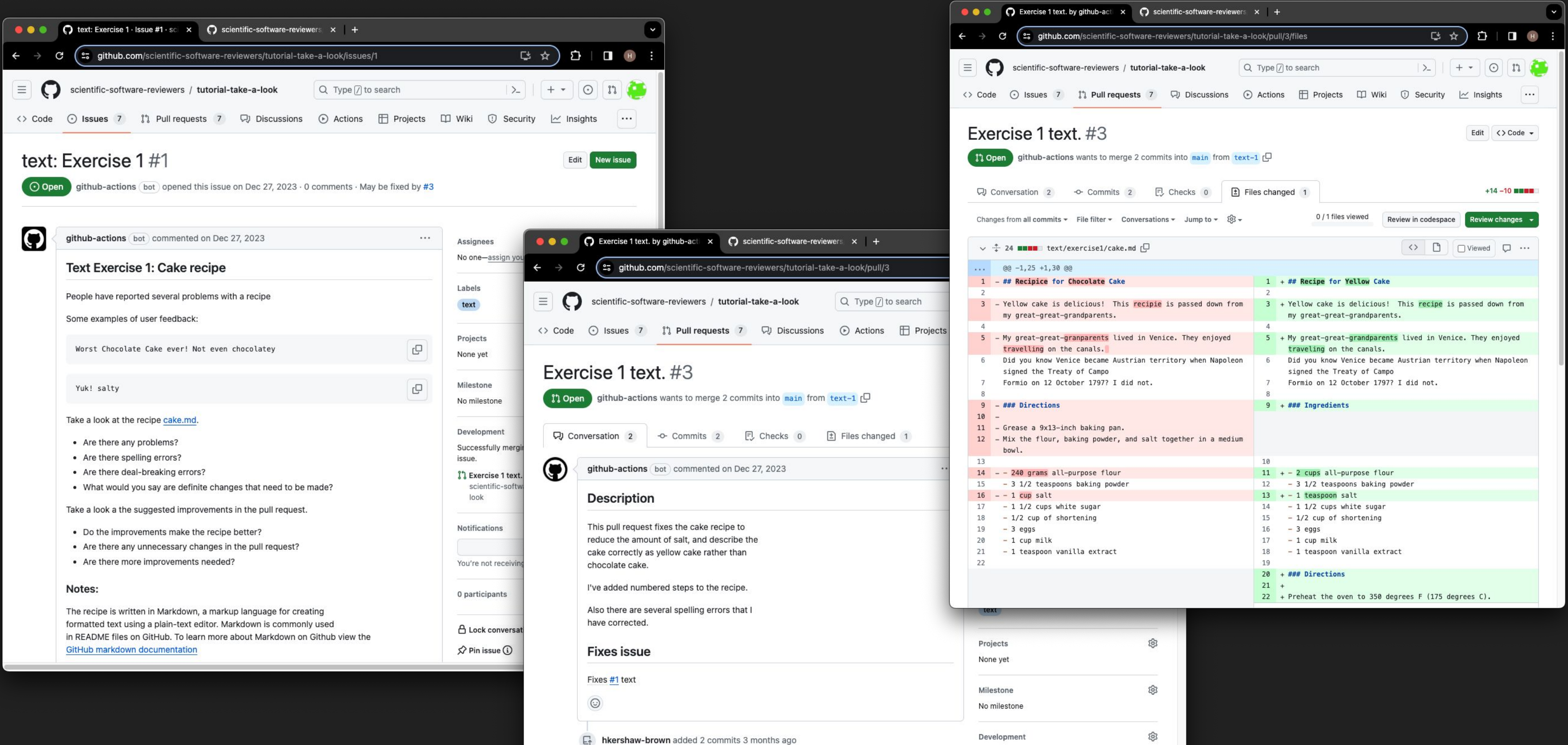
Code review tutorial exercises

Uncheck `Copy the main branch only` and click `Create fork`

- Fork
- All branches
- Enable workflows
- …

Barrier before I've started

take-a-look repository



text: Exercise 1 · Issue #1 · sc
scientific-software-reviewers
github.com/scientific-software-reviewers/tutorial-take-a-look/issues/1
scientific-software-reviewers / tutorial-take-a-look
Type / to search
Code
Issues 7
Pull requests 7
Discussions
Actions
Projects
Wiki
Security
Insights
Edit
New issue

text: Exercise 1 #1

Open **github-actions** bot opened this issue on Dec 27, 2023 · 0 comments · May be fixed by #3

github-actions bot commented on Dec 27, 2023

Text Exercise 1: Cake recipe

People have reported several problems with a recipe

Some examples of user feedback:

```
Worst Chocolate Cake ever! Not even chocolatey
```

```
Yuk! salty
```

Take a look at the recipe cake.md.

- Are there any problems?
- Are there spelling errors?
- Are there deal-breaking errors?
- What would you say are definite changes that need to be made?

Take a look a the suggested improvements in the pull request.

- Do the improvements make the recipe better?
- Are there any unnecessary changes in the pull request?
- Are there more improvements needed?

Notes:

The recipe is written in Markdown, a markup language for creating formatted text using a plain-text editor. Markdown is commonly used in README files on GitHub. To learn more about Markdown on Github view the GitHub markdown documentation

Assignees
No one—assign you

Labels
text

Projects
None yet

Milestone
No milestone

Development
Successfully mergi issue.
Exercise 1 text. scientific-softw look

Notifications
You're not receiving

0 participants

Lock conversat
Pin issue

Exercise 1 text. by github-act
scientific-software-reviewers
github.com/scientific-software-reviewers/tutorial-take-a-look/pull/3
scientific-software-reviewers / tutorial-take-a-look
Type / to search
Code
Issues 7
Pull requests 7
Discussions
Actions
Projects

Exercise 1 text. #3

Open **github-actions** wants to merge 2 commits into `main` from `text-1`

Conversation 2 · Commits 2 · Checks 0 · Files changed 1

github-actions bot commented on Dec 27, 2023

Description

This pull request fixes the cake recipe to reduce the amount of salt, and describe the cake correctly as yellow cake rather than chocolate cake.

I've added numbered steps to the recipe.

Also there are several spelling errors that I have corrected.

Fixes issue

Fixes #1 text

hkershaw-brown added 2 commits 3 months ago

text

Projects
None yet

Milestone
No milestone

Development

Exercise 1 text. by github-act
scientific-software-reviewers
github.com/scientific-software-reviewers/tutorial-take-a-look/pull/3/files
scientific-software-reviewers / tutorial-take-a-look
Type / to search
Code
Issues 7
Pull requests 7
Discussions
Actions
Projects
Wiki
Security
Insights
Edit
Code

Exercise 1 text. #3

Open **github-actions** wants to merge 2 commits into `main` from `text-1`

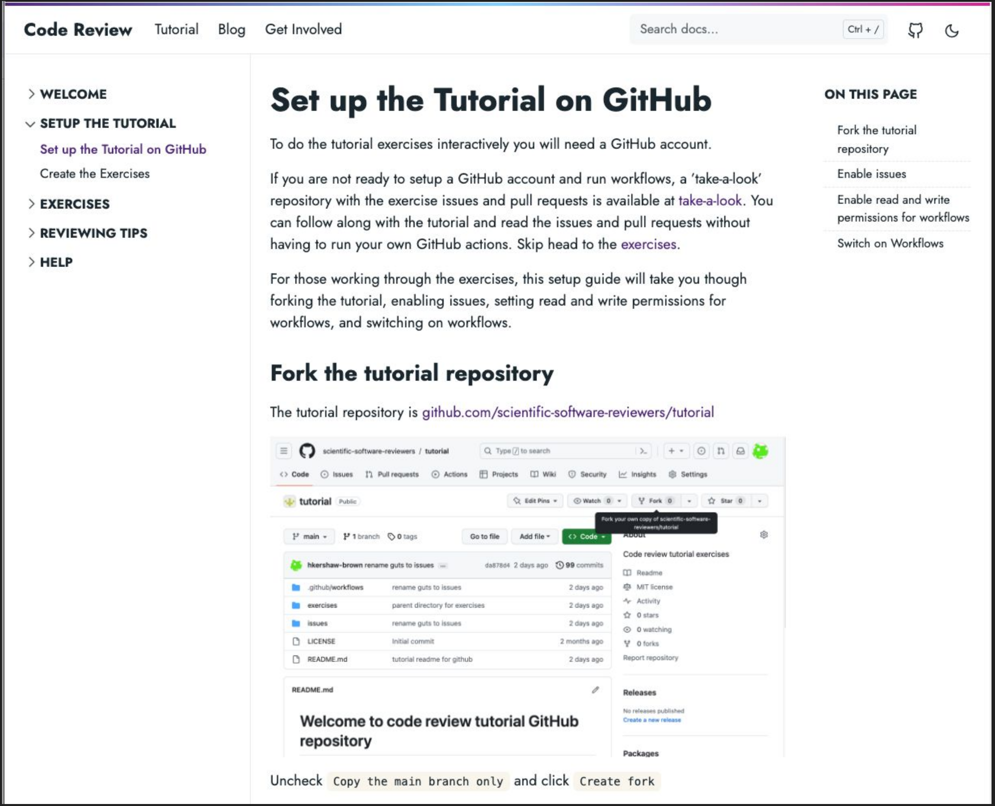
Conversation 2 · Commits 2 · Checks 0 · Files changed 1 · +14 −10

24 text/exercise1/cake.md

```
@@ -1,25 +1,30 @@
- ## Recipice for Chocolate Cake

- Yellow cake is delicious!  This recipie is passed down from
  my great-great-grandparents.

- My great-great-granparents lived in Venice. They enjoyed
  travelling on the canals.
  Did you know Venice became Austrian territory when Napoleon
  signed the Treaty of Campo
  Formio on 12 October 1797? I did not.

- ### Directions
-
- Grease a 9x13-inch baking pan.
- Mix the flour, baking powder, and salt together in a medium
  bowl.

- - 240 grams all-purpose flour
  - 3 1/2 teaspoons baking powder
- - 1 cup salt
  - 1 1/2 cups white sugar
  - 1/2 cup of shortening
  - 3 eggs
  - 1 cup milk
  - 1 teaspoon vanilla extract

```

```
+ ## Recipe for Yellow Cake

+ Yellow cake is delicious!  This recipe is passed down from
  my great-great-grandparents.

+ My great-great-grandparents lived in Venice. They enjoyed
  traveling on the canals.
  Did you know Venice became Austrian territory when Napoleon
  signed the Treaty of Campo
  Formio on 12 October 1797? I did not.

+ ### Ingredients

+ - 2 cups all-purpose flour
  - 3 1/2 teaspoons baking powder
+ - 1 teaspoon salt
  - 1 1/2 cups white sugar
  - 1/2 cup of shortening
  - 3 eggs
  - 1 cup milk
  - 1 teaspoon vanilla extract

+ ### Directions
+
+ Preheat the oven to 350 degrees F (175 degrees C).
```

Setting up the tutorial on GitHub



Set up the Tutorial on GitHub

To do the tutorial exercises interactively you will need a GitHub account.

If you are not ready to setup a GitHub account and run workflows, a 'take-a-look' repository with the exercise issues and pull requests is available at take-a-look. You can follow along with the tutorial and read the issues and pull requests without having to run your own GitHub actions. Skip head to the exercises.

For those working through the exercises, this setup guide will take you though forking the tutorial, enabling issues, setting read and write permissions for workflows, and switching on workflows.

Fork the tutorial repository

The tutorial repository is github.com/scientific-software-reviewers/tutorial

Fork your own copy of scientific-software-reviewers/tutorial

Code review tutorial exercises

hkershaw-brown rename guts to issues … da878d4 2 days ago 99 commits

.github/workflows	rename guts to issues	2 days ago
exercises	parent directory for exercises	2 days ago
issues	rename guts to issues	2 days ago
LICENSE	Initial commit	2 months ago
README.md	tutorial readme for github	2 days ago

No releases published
Create a new release

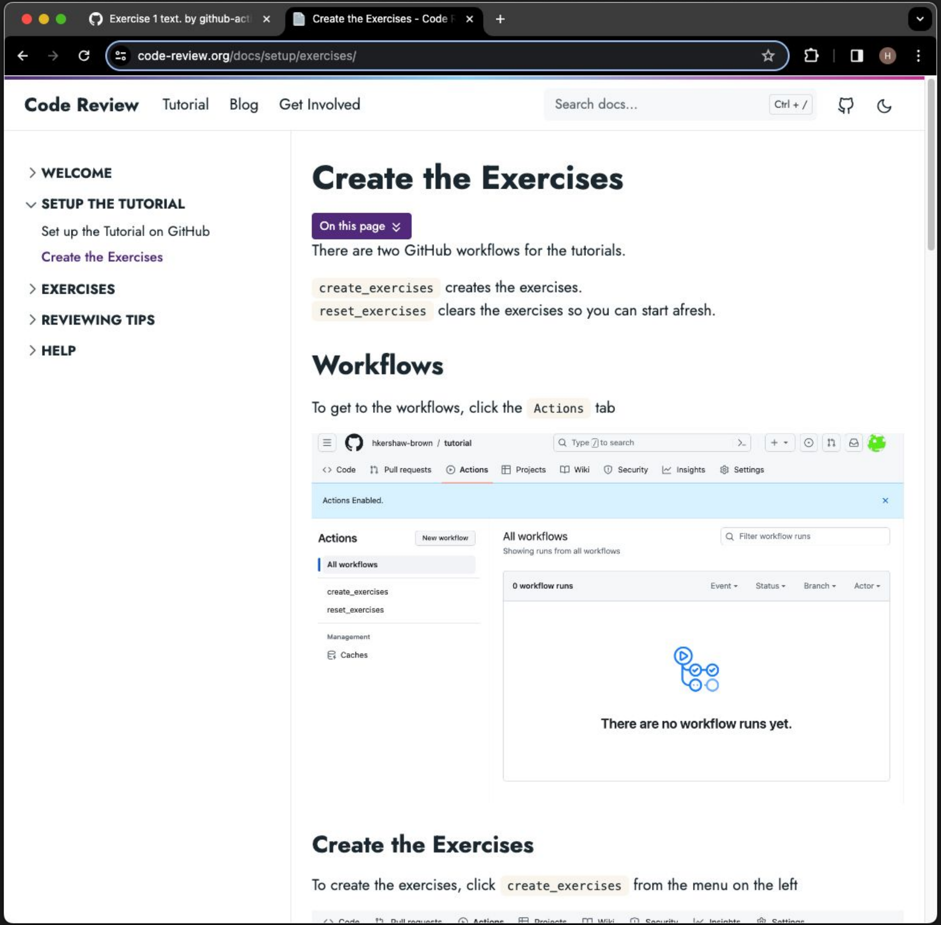
README.md

Welcome to code review tutorial GitHub repository

Uncheck `Copy the main branch only` and click `Create fork`

- Fork
- All branches
- Enable workflows
- Run workflows

Setting up the tutorial on GitHub



Create the Exercises

There are two GitHub workflows for the tutorials.

`create_exercises` creates the exercises.
`reset_exercises` clears the exercises so you can start afresh.

Workflows

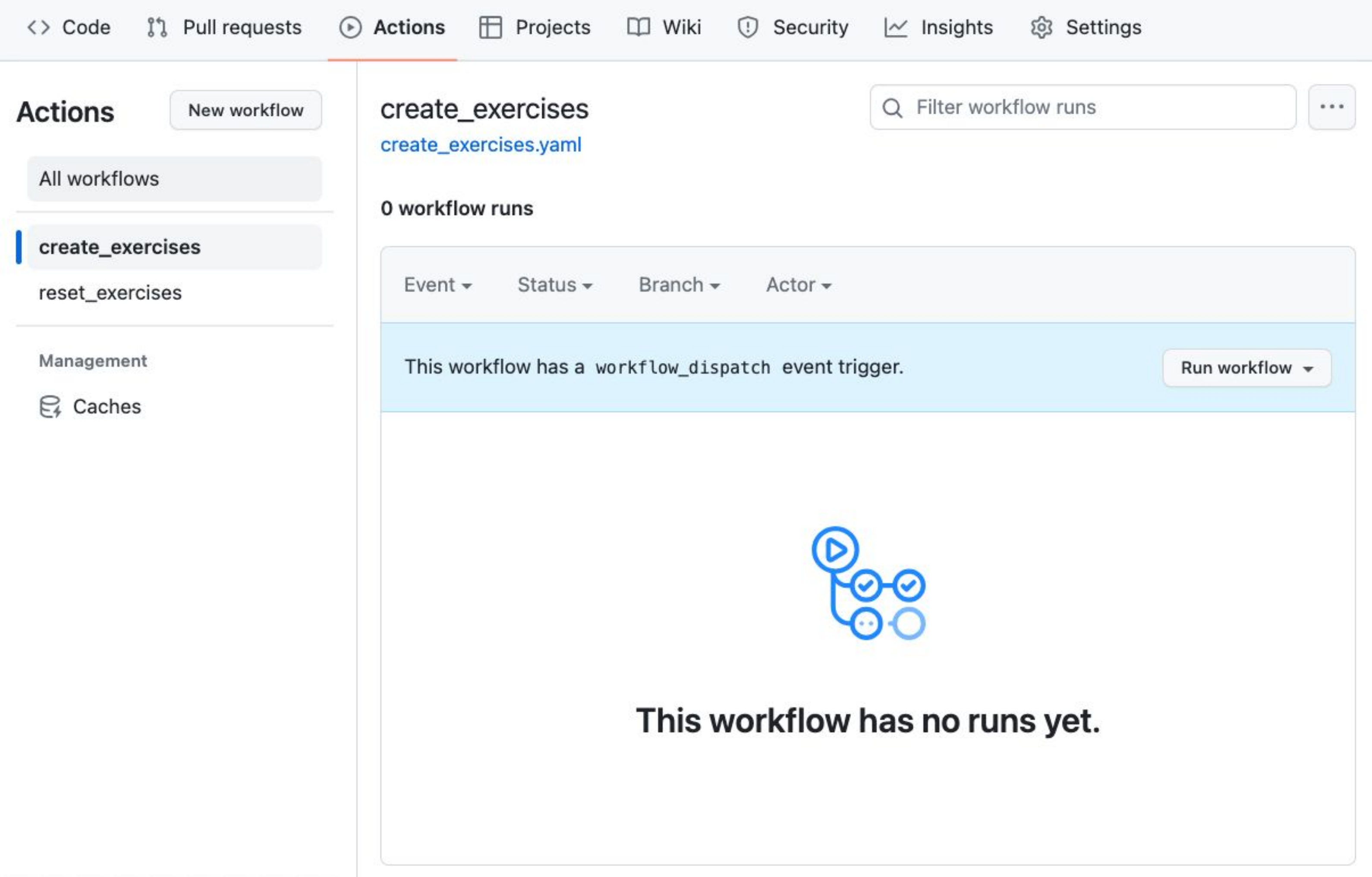
To get to the workflows, click the `Actions` tab

Create the Exercises

To create the exercises, click `create_exercises` from the menu on the left

- Fork
- All branches
- Enable workflows
- Run workflows

Create the exercises



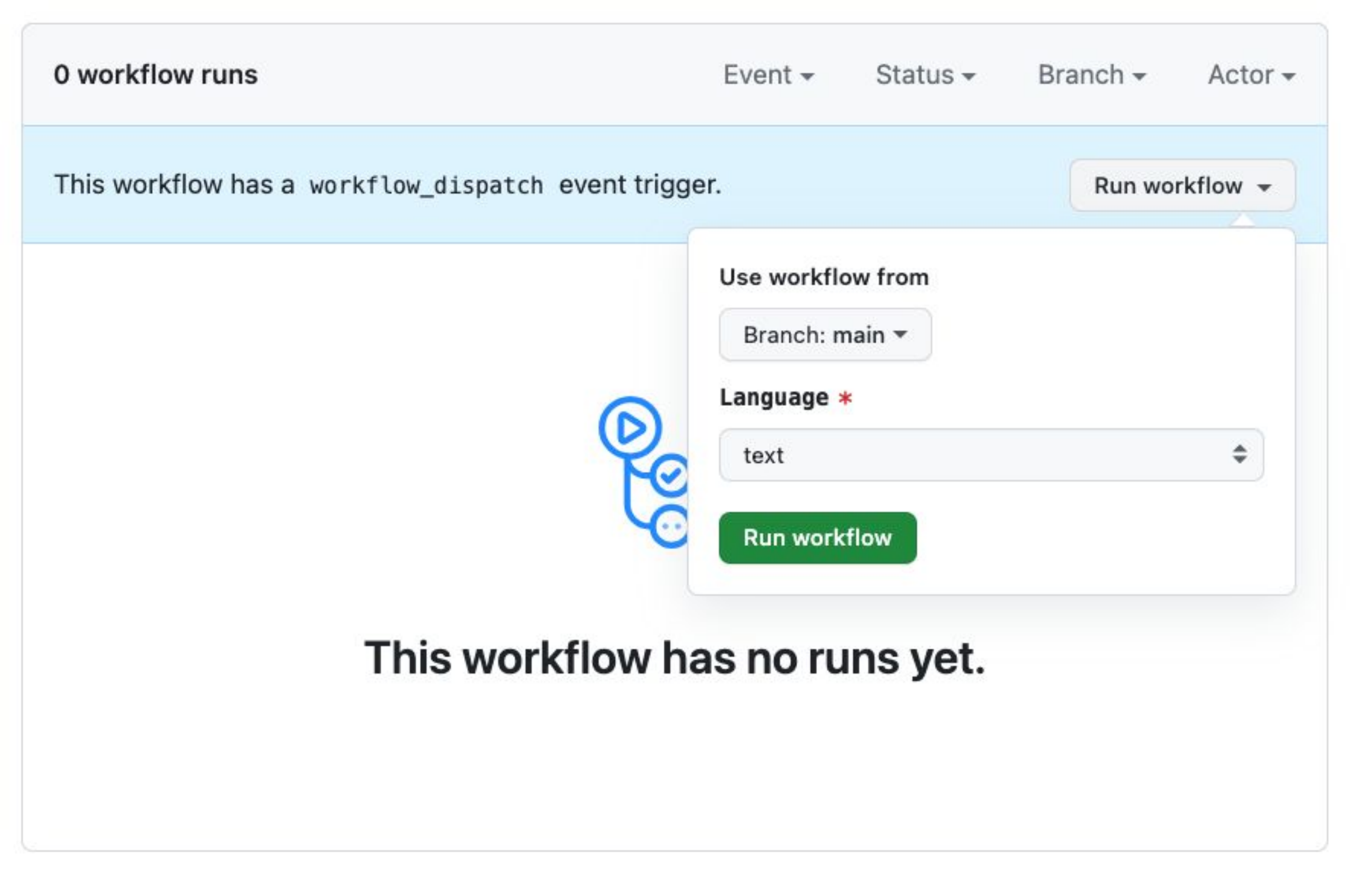
create_exercises

create_exercises.yaml

0 workflow runs

This workflow has a `workflow_dispatch` event trigger.

This workflow has no runs yet.



This workflow has a `workflow_dispatch` event trigger.

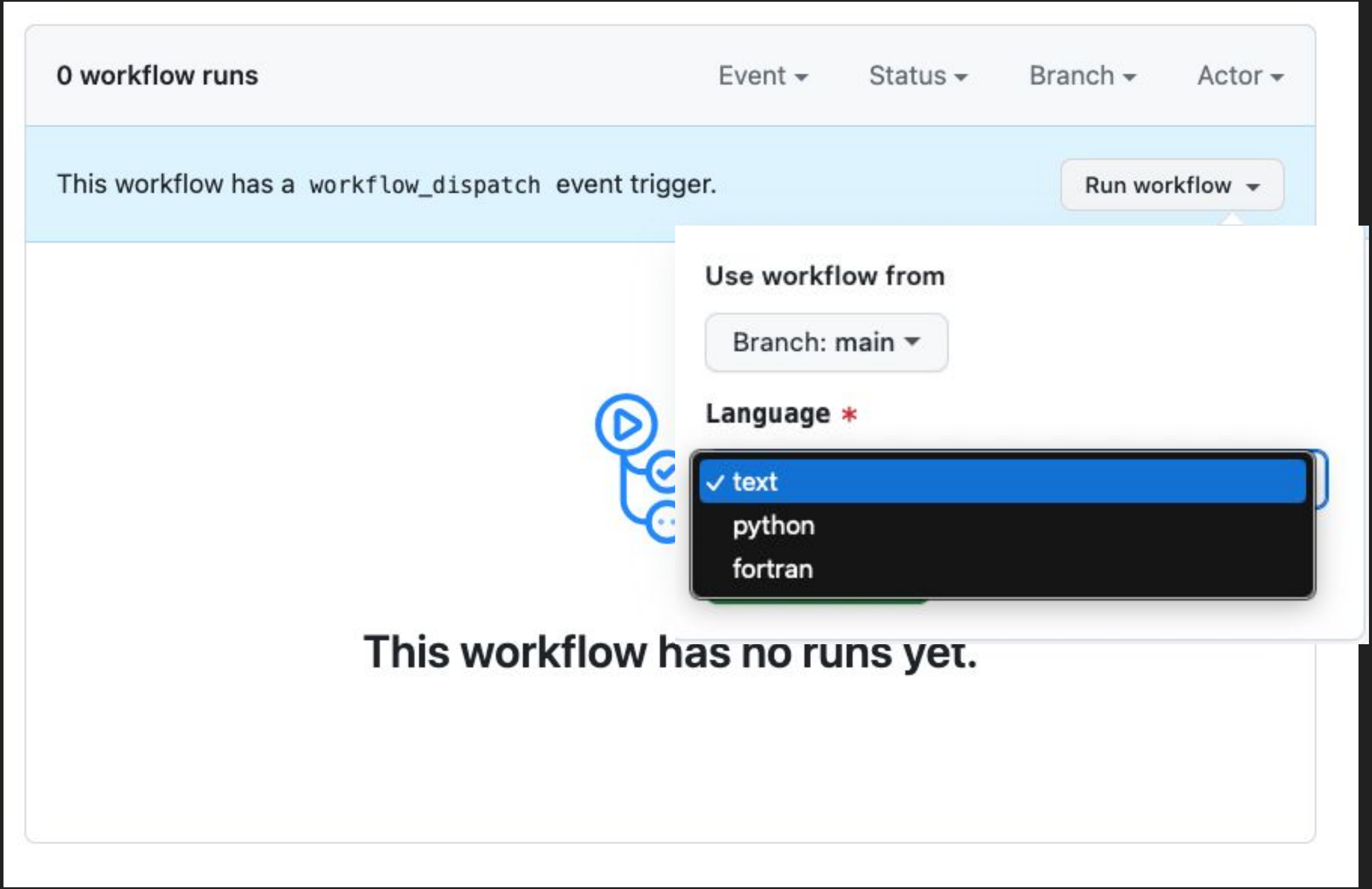
Use workflow from

Branch: main

Language *

text

This workflow has no runs yet.



0 workflow runs

This workflow has a `workflow_dispatch` event trigger.

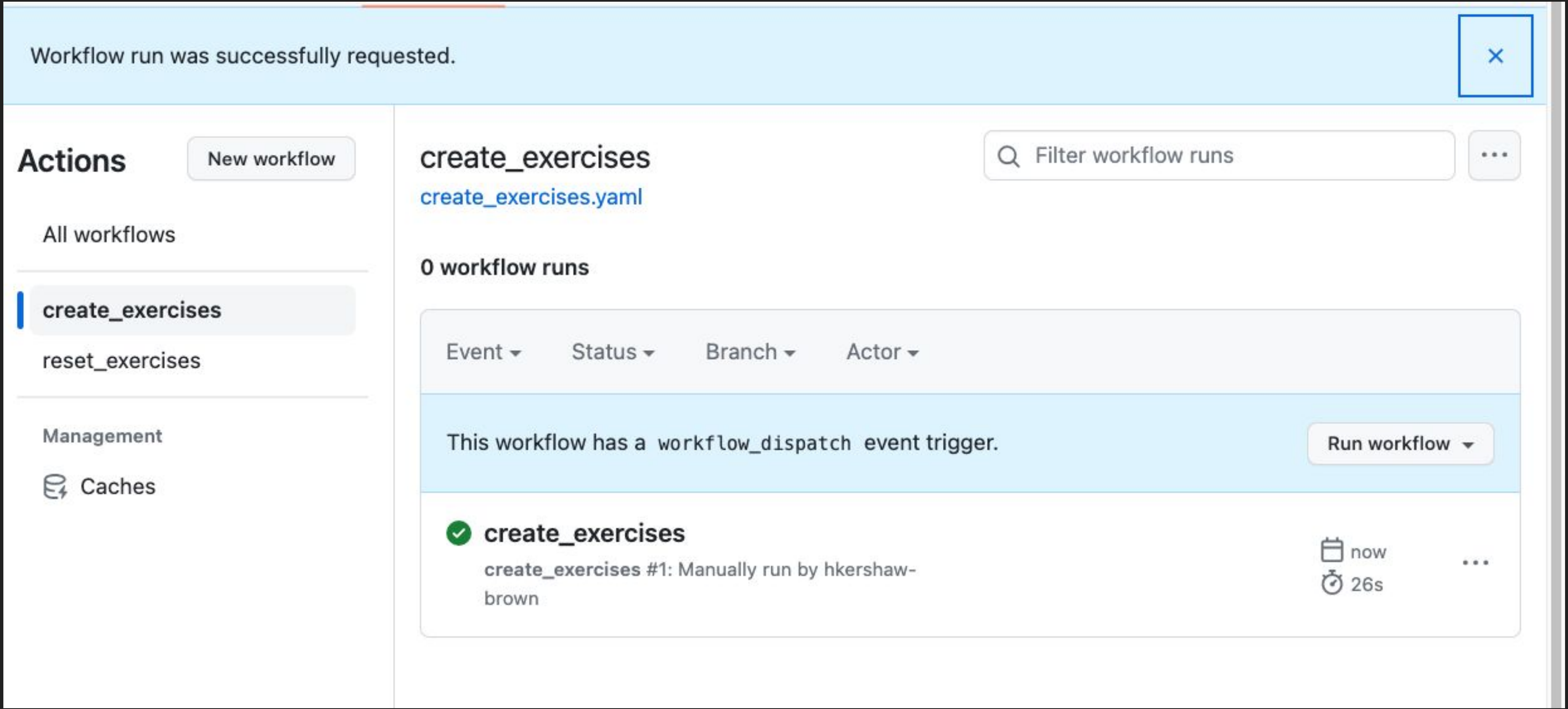
Use workflow from

Branch: main

Language *

✓ text

This workflow has no runs yet.



Workflow run was successfully requested.

create_exercises

create_exercises.yaml

0 workflow runs

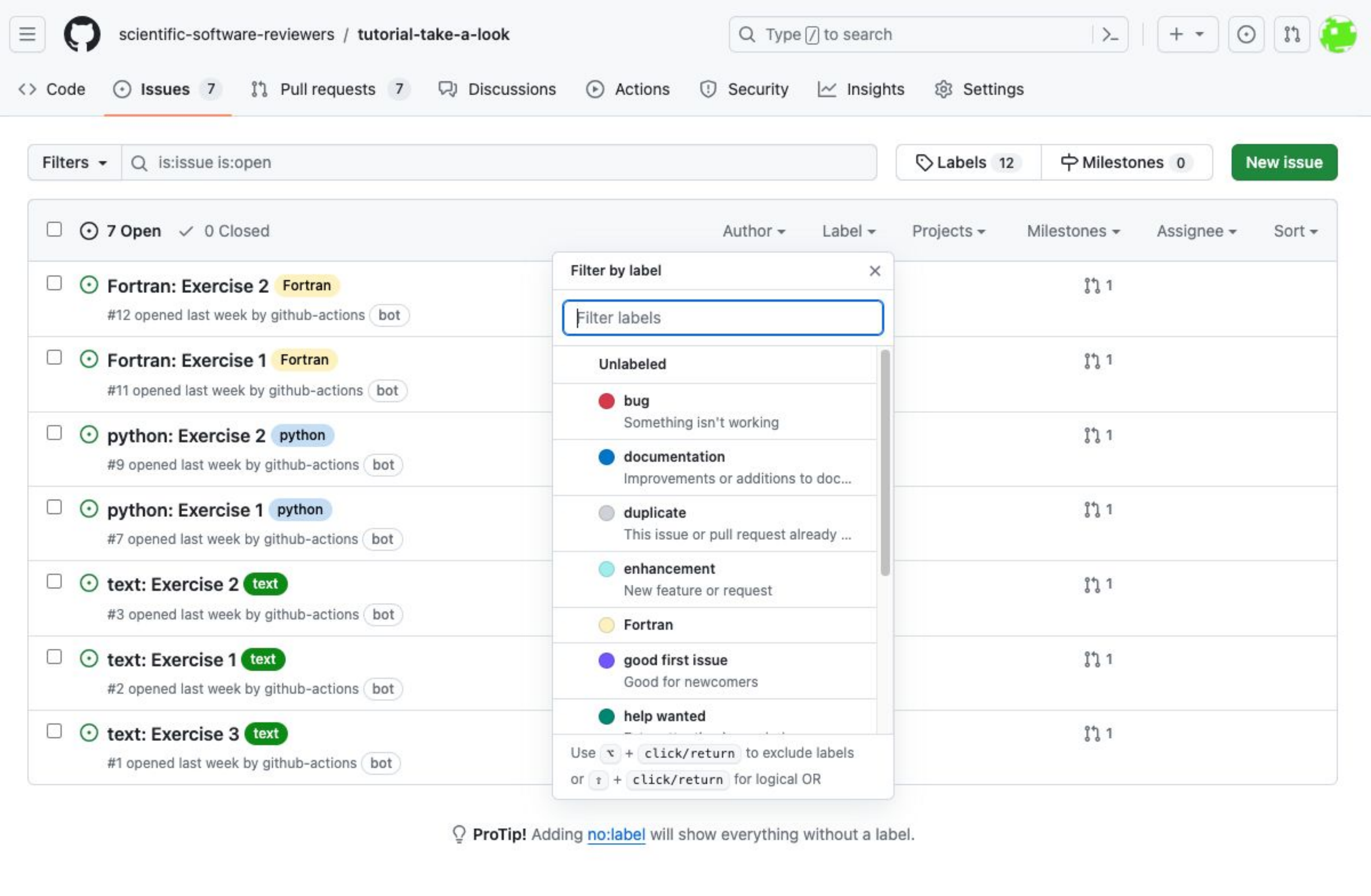
This workflow has a `workflow_dispatch` event trigger.

✓ create_exercises

create_exercises #1: Manually run by hkershaw-brown

now

26s



scientific-software-reviewers / tutorial-take-a-look

Type / to search

Code · Issues 7 · Pull requests 7 · Discussions · Actions · Security · Insights · Settings

Filters ▾ · is:issue is:open

Labels 12 · Milestones 0 · New issue

□ 7 Open ✓ 0 Closed — Author ▾ · Label ▾ · Projects ▾ · Milestones ▾ · Assignee ▾ · Sort ▾

- □ Fortran: Exercise 2 `Fortran` — #12 opened last week by github-actions `bot` — 1
- □ Fortran: Exercise 1 `Fortran` — #11 opened last week by github-actions `bot` — 1
- □ python: Exercise 2 `python` — #9 opened last week by github-actions `bot` — 1
- □ python: Exercise 1 `python` — #7 opened last week by github-actions `bot` — 1
- □ text: Exercise 2 `text` — #3 opened last week by github-actions `bot` — 1
- □ text: Exercise 1 `text` — #2 opened last week by github-actions `bot` — 1
- □ text: Exercise 3 `text` — #1 opened last week by github-actions `bot` — 1

Filter by label ✕

Filter labels

Unlabeled

- 🟥 **bug** — Something isn't working
- 🟦 **documentation** — Improvements or additions to doc...
- ● **duplicate** — This issue or pull request already ...
- ● **enhancement** — New feature or request
- ● **Fortran**
- 🟪 **good first issue** — Good for newcomers
- 🟩 **help wanted**

Use `⌥` + `click/return` to exclude labels
or `⇧` + `click/return` for logical OR

ProTip! Adding no:label will show everything without a label.

Navigating the exercises

Issues

Pull Requests

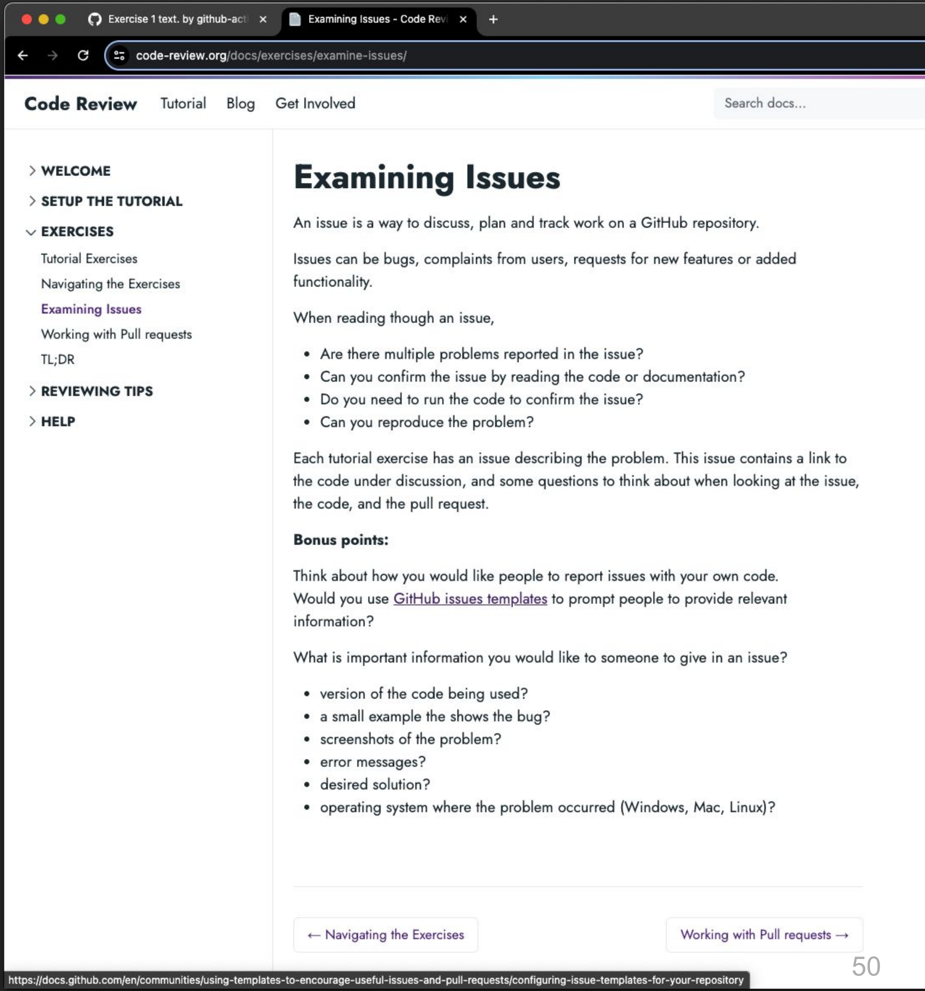
Issues

Problem

Pull Requests

Solution

Issues



Examining Issues

An issue is a way to discuss, plan and track work on a GitHub repository.

Issues can be bugs, complaints from users, requests for new features or added functionality.

When reading though an issue,

- Are there multiple problems reported in the issue?
- Can you confirm the issue by reading the code or documentation?
- Do you need to run the code to confirm the issue?
- Can you reproduce the problem?

Each tutorial exercise has an issue describing the problem. This issue contains a link to the code under discussion, and some questions to think about when looking at the issue, the code, and the pull request.

Bonus points:

Think about how you would like people to report issues with your own code. Would you use GitHub issues templates to prompt people to provide relevant information?

What is important information you would like to someone to give in an issue?

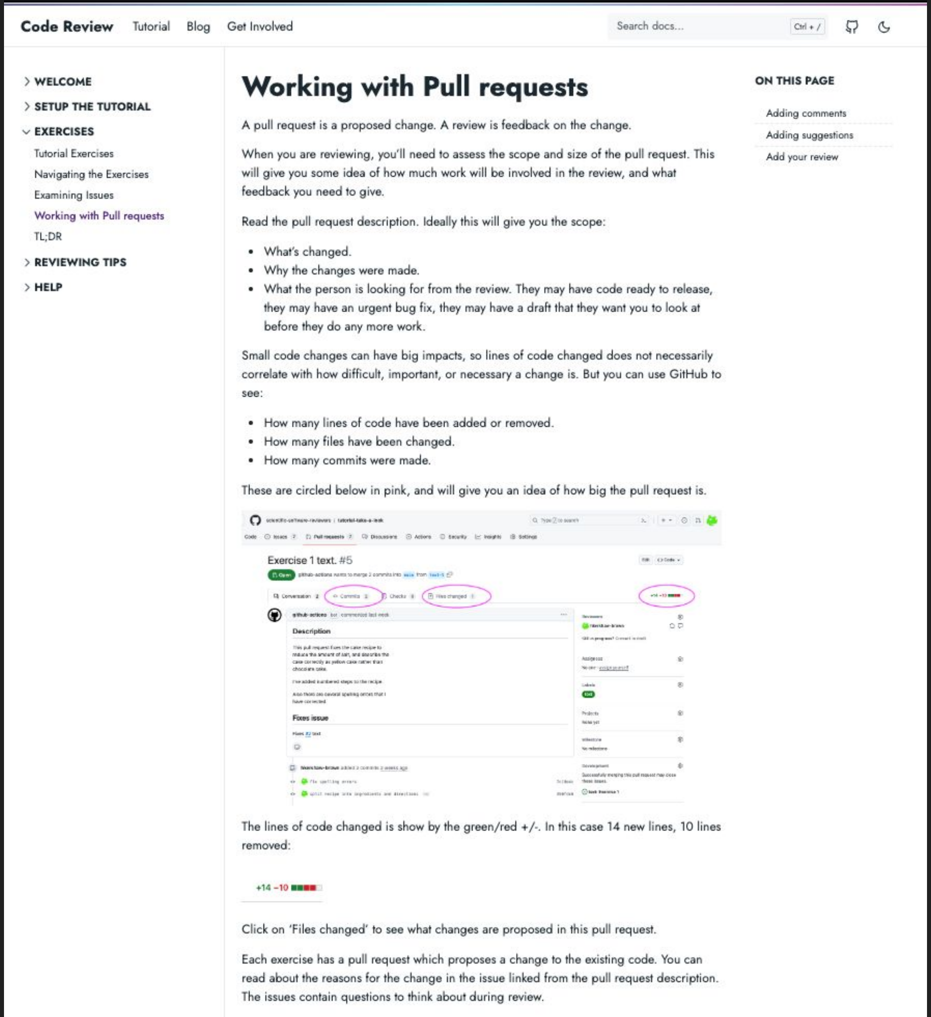
- version of the code being used?
- a small example the shows the bug?
- screenshots of the problem?
- error messages?
- desired solution?
- operating system where the problem occurred (Windows, Mac, Linux)?

← Navigating the Exercises

Working with Pull requests →

https://docs.github.com/en/communities/using-templates-to-encourage-useful-issues-and-pull-requests/configuring-issue-templates-for-your-repository

Pull Requests



- WELCOME
- SETUP THE TUTORIAL
- EXERCISES
  - Tutorial Exercises
  - Navigating the Exercises
  - Examining Issues
  - Working with Pull requests
  - TL;DR
- REVIEWING TIPS
- HELP

Working with Pull requests

A pull request is a proposed change. A review is feedback on the change.

When you are reviewing, you'll need to assess the scope and size of the pull request. This will give you some idea of how much work will be involved in the review, and what feedback you need to give.

Read the pull request description. Ideally this will give you the scope:

- What's changed.
- Why the changes were made.
- What the person is looking for from the review. They may have code ready to release, they may have an urgent bug fix, they may have a draft that they want you to look at before they do any more work.

Small code changes can have big impacts, so lines of code changed does not necessarily correlate with how difficult, important, or necessary a change is. But you can use GitHub to see:

- How many lines of code have been added or removed.
- How many files have been changed.
- How many commits were made.

These are circled below in pink, and will give you an idea of how big the pull request is.

The lines of code changed is show by the green/red +/-. In this case 14 new lines, 10 lines removed:

+14 −10 🟩🟩🟥🟥⬜

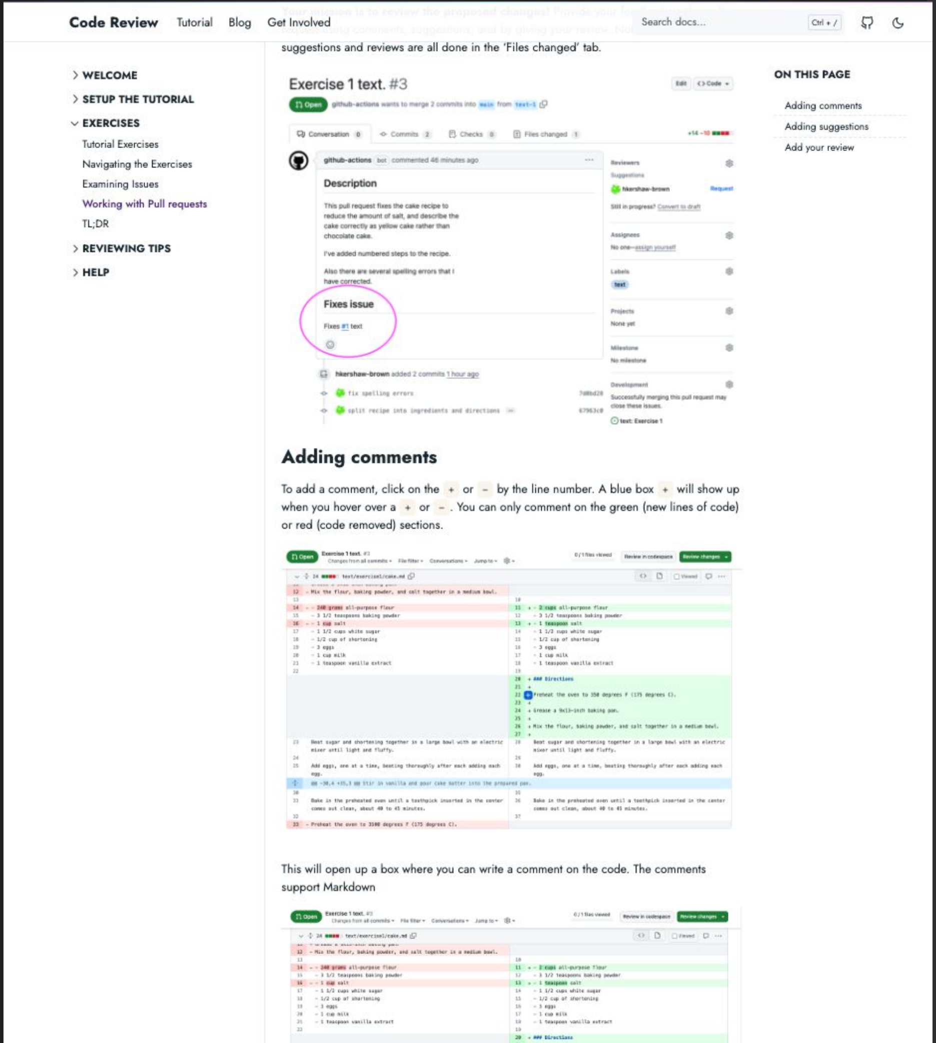
Click on 'Files changed' to see what changes are proposed in this pull request.

Each exercise has a pull request which proposes a change to the existing code. You can read about the reasons for the change in the issue linked from the pull request description. The issues contain questions to think about during review.

ON THIS PAGE

- Adding comments
- Adding suggestions
- Add your review

Pull Requests



suggestions and reviews are all done in the 'Files changed' tab.

Exercise 1 text. #3

github-actions wants to merge 2 commits into main from text-1

github-actions bot commented 46 minutes ago

Description

This pull request fixes the cake recipe to reduce the amount of salt, and describe the cake correctly as yellow cake rather than chocolate cake.

I've added numbered steps to the recipe.

Also there are several spelling errors that I have corrected.

Fixes issue

Fixes #1 text

hkershaw-brown added 2 commits 1 hour ago

```
fix spelling errors                               7d8bd28
split recipe into ingredients and directions      67963c0
```

Reviewers: Suggestions: hkershaw-brown Request. Still in progress? Convert to draft

Assignees: No one—assign yourself

Labels: text

Projects: None yet

Milestone: No milestone

Development: Successfully merging this pull request may close these issues. text: Exercise 1

Adding comments

To add a comment, click on the + or − by the line number. A blue box + will show up when you hover over a + or −. You can only comment on the green (new lines of code) or red (code removed) sections.

Exercise 1 text. #3

```
24 text/exercise1/cake.md

- Mix the flour, baking powder, and salt together in a medium bowl.

- - 240 grams all-purpose flour                      + - 2 cups all-purpose flour
  - 3 1/2 teaspoons baking powder                      - 3 1/2 teaspoons baking powder
- - 1 cup salt                                       + - 1 teaspoon salt
  - 1 1/2 cups white sugar                             - 1 1/2 cups white sugar
  - 1/2 cup of shortening                              - 1/2 cup of shortening
  - 3 eggs                                             - 3 eggs
  - 1 cup milk                                         - 1 cup milk
  - 1 teaspoon vanilla extract                         - 1 teaspoon vanilla extract

                                                     + ### Directions
                                                     +
                                                     + Preheat the oven to 350 degrees F (175 degrees C).
                                                     +
                                                     + Grease a 9x13-inch baking pan.
                                                     +
                                                     + Mix the flour, baking powder, and salt together in a medium bowl.
                                                     +
  Beat sugar and shortening together in a large bowl with an electric    Beat sugar and shortening together in a large bowl with an electric
  mixer until light and fluffy.                                          mixer until light and fluffy.

  Add eggs, one at a time, beating thoroughly after each adding each     Add eggs, one at a time, beating thoroughly after each adding each
  egg.                                                                   egg.
@@ -30,4 +35,3 @@ Stir in vanilla and pour cake batter into the prepared pan.

  Bake in the preheated oven until a toothpick inserted in the center    Bake in the preheated oven until a toothpick inserted in the center
  comes out clean, about 40 to 45 minutes.                               comes out clean, about 40 to 45 minutes.

- Preheat the oven to 3500 degrees F (175 degrees C).
```

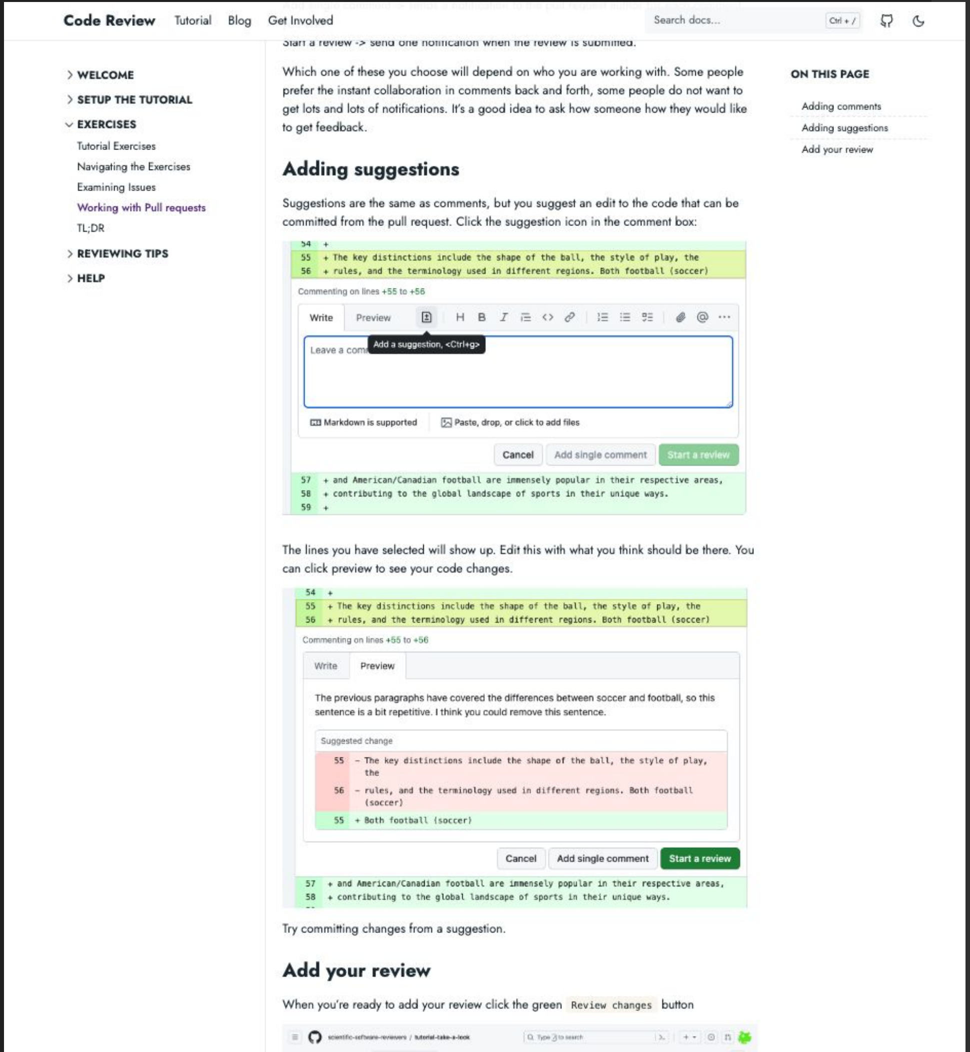
This will open up a box where you can write a comment on the code. The comments support Markdown

Exercise 1 text. #3

```
24 text/exercise1/cake.md

- Mix the flour, baking powder, and salt together in a medium bowl.

- - 240 grams all-purpose flour                      + - 2 cups all-purpose flour
  - 3 1/2 teaspoons baking powder                      - 3 1/2 teaspoons baking powder
- - 1 cup salt                                       + - 1 teaspoon salt
  - 1 1/2 cups white sugar                             - 1 1/2 cups white sugar
  - 1/2 cup of shortening                              - 1/2 cup of shortening
  - 3 eggs                                             - 3 eggs
  - 1 cup milk                                         - 1 cup milk
  - 1 teaspoon vanilla extract                         - 1 teaspoon vanilla extract

                                                     + ### Directions
```

Navigating Pull Requests

size and scope



Start a review -> send one notification when the review is submitted.

Which one of these you choose will depend on who you are working with. Some people prefer the instant collaboration in comments back and forth, some people do not want to get lots and lots of notifications. It's a good idea to ask how someone how they would like to get feedback.

Adding suggestions

Suggestions are the same as comments, but you suggest an edit to the code that can be committed from the pull request. Click the suggestion icon in the comment box:

```
54 +
55 + The key distinctions include the shape of the ball, the style of play, the
56 + rules, and the terminology used in different regions. Both football (soccer)
```

Commenting on lines +55 to +56

```
57 + and American/Canadian football are immensely popular in their respective areas,
58 + contributing to the global landscape of sports in their unique ways.
59 +
```

The lines you have selected will show up. Edit this with what you think should be there. You can click preview to see your code changes.

```
54 +
55 + The key distinctions include the shape of the ball, the style of play, the
56 + rules, and the terminology used in different regions. Both football (soccer)
```

Commenting on lines +55 to +56

The previous paragraphs have covered the differences between soccer and football, so this sentence is a bit repetitive. I think you could remove this sentence.

Suggested change

```
55 - The key distinctions include the shape of the ball, the style of play,
     the
56 - rules, and the terminology used in different regions. Both football
     (soccer)
55 + Both football (soccer)
```

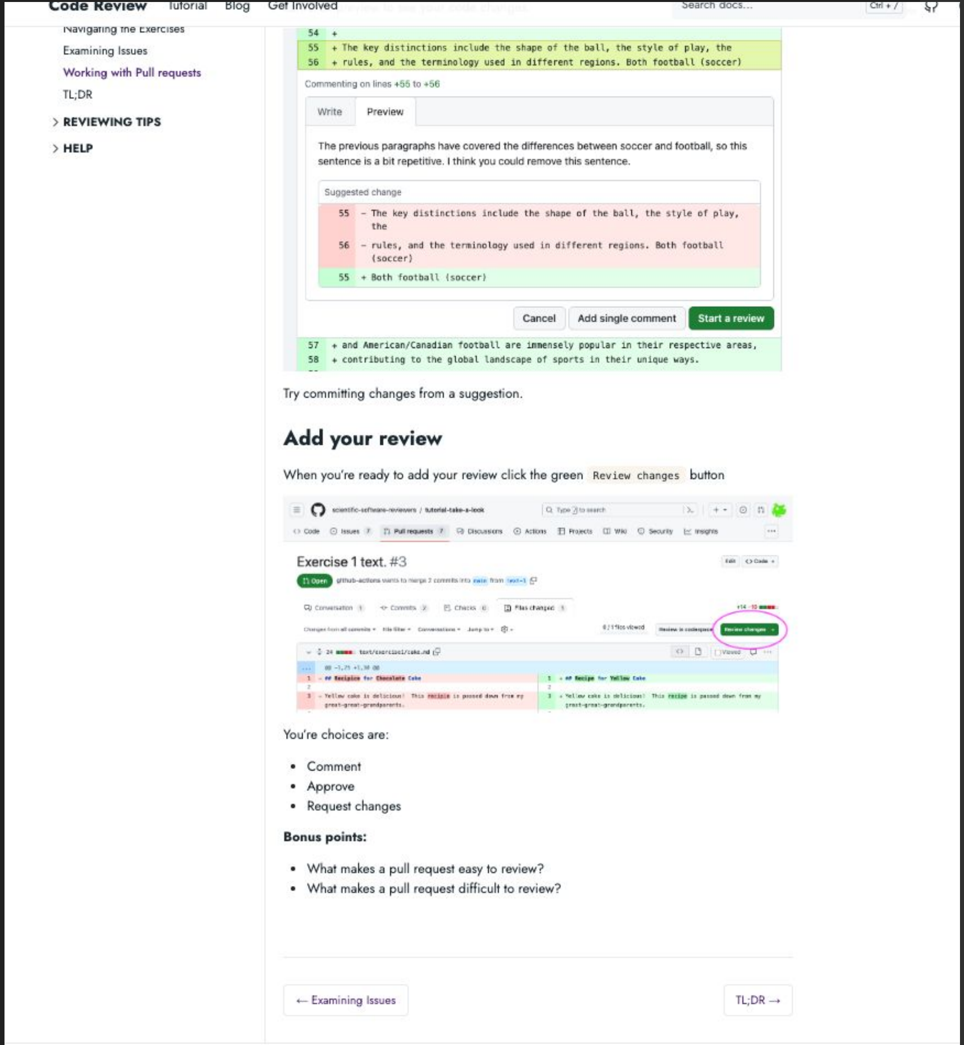
```
57 + and American/Canadian football are immensely popular in their respective areas,
58 + contributing to the global landscape of sports in their unique ways.
```

Try committing changes from a suggestion.

Add your review

When you're ready to add your review click the green `Review changes` button

Adding suggestions



Code Review Tutorial Blog Get Involved Search docs... Ctrl + /

Navigating the Exercises
Examining Issues
Working with Pull requests
TL;DR

> REVIEWING TIPS

> HELP

```
54 +
55 + The key distinctions include the shape of the ball, the style of play, the
56 + rules, and the terminology used in different regions. Both football (soccer)
```

Commenting on lines +55 to +56

Write | Preview

The previous paragraphs have covered the differences between soccer and football, so this sentence is a bit repetitive. I think you could remove this sentence.

Suggested change

```
55 - The key distinctions include the shape of the ball, the style of play,
     the
56 - rules, and the terminology used in different regions. Both football
     (soccer)
55 + Both football (soccer)
```

Cancel | Add single comment | Start a review

```
57 + and American/Canadian football are immensely popular in their respective areas,
58 + contributing to the global landscape of sports in their unique ways.
```

Try committing changes from a suggestion.

Add your review

When you're ready to add your review click the green `Review changes` button

scientific-software-reviewers / tutorial-take-a-look
Type / to search
Code
Issues 7
Pull requests 7
Discussions
Actions
Projects
Wiki
Security
Insights
Exercise 1 text. #3
Edit
Code
Open
github-actions wants to merge 2 commits into main from text-1
Conversation 1
Commits 2
Checks 0
Files changed 1
+14 -10
Changes from all commits
File filter
Conversations
Jump to
0 / 1 files viewed
Review in codespace
Review changes
24 text/exercise1/cake.md
Viewed
@@ -1,25 +1,30 @@
1 - ## Recipice for Chocolate Cake
3 - Yellow cake is delicious! This recipice is passed down from my great-great-grandparents.
1 + ## Recipe for Yellow Cake
3 + Yellow cake is delicious! This recipe is passed down from my great-great-grandparents.

You're choices are:

- Comment
- Approve
- Request changes

Bonus points:

- What makes a pull request easy to review?
- What makes a pull request difficult to review?

← Examining Issues

TL;DR →

Adding your review

Reviewing

Reviewing

Being reviewed

Reviewing

Does the pull request address the issue?

Are there any deal breakers that would stop you accepting the changes?

Can you suggest any improvements?

What is a good way to phrase your suggested improvements?

Is the solution overly complicated? Are the comments up to date, necessary, helpful?

Would you accept the pull request as it is now? Are your suggested changes must-do? nice-to-have? nitpicks? How would you communicate this?

Do you spend a lot of time reviewing the code style? Is it worth having a style guide for contributors? Can you make use of an existing style guide? Or a linter?

Being reviewed

Reviewing

Does the pull request address the issue?

Are there any deal breakers that would stop you accepting the changes?

Can you suggest any improvements?

What is a good way to phrase your suggested improvements?

Is the solution overly complicated? Are the comments up to date, necessary, helpful?

Would you accept the pull request as it is now? Are your suggested changes must-do? nice-to-have? nitpicks? How would you communicate this?

Do you spend a lot of time reviewing the code style? Is it worth having a style guide for contributors? Can you make use of an existing style guide? Or a linter?

Being reviewed

When putting in a pull request, how can you make it easy for a reviewer to understand what you have done?

What makes a good pull request, what makes a bad pull request?

Can you commit code in a way that lets someone review your code more easily? Should you separate functional changes from style changes?

Would you use a tool such as commitizen to prompt yourself at commit time? Why? Why not?

Mechanics of the tutorial

Adding exercises

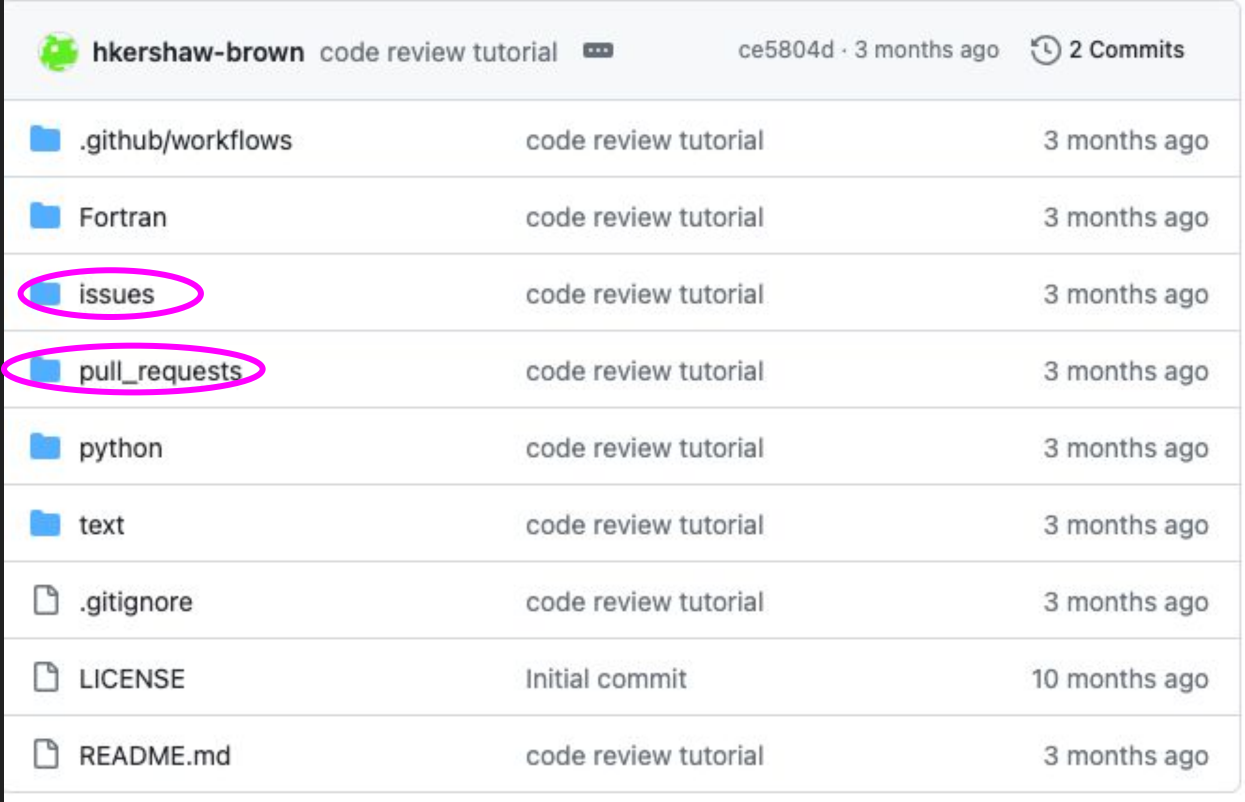
Adding exercises

Two GitHub workflows:

create_exercises	create_exercises.yaml
reset_exercises	close_issues_and_pulls.yaml

https://github.com/scientific-software-reviewers/tutorial

Adding exercises



hkershaw-brown code review tutorial ··· ce5804d · 3 months ago 2 Commits

.github/workflows	code review tutorial	3 months ago
Fortran	code review tutorial	3 months ago
issues	code review tutorial	3 months ago
pull_requests	code review tutorial	3 months ago
python	code review tutorial	3 months ago
text	code review tutorial	3 months ago
.gitignore	code review tutorial	3 months ago
LICENSE	Initial commit	10 months ago
README.md	code review tutorial	3 months ago

Adding exercises

issues/{Language}-ex{#}-issue.md

pull_requests/{Language}-ex{#}-pull_body.md

Branch: {Language}-{#}

Adding exercises

issues/{Language}-ex{#}-issue.md

pull_requests/{Language}-ex{#}-pull_body.md

Branch: {Language}-{#}

.github/workflows/create_exercises.yaml is the action that takes ‘Language’, and for each exercise {1..n}:

Adding exercises

issues/{Language}-ex{#}-issue.md

pull_requests/{Language}-ex{#}-pull_body.md

Branch: {Language}-{#}

.github/workflows/create_exercises.yaml is the action that takes ‘Language’, and for each exercise {1..n}:

- creates any issues {Language}-{1…n}.

Adding exercises

issues/{Language}-ex{#}-issue.md

pull_requests/{Language}-ex{#}-pull_body.md

Branch: {Language}-{#}

.github/workflows/create_exercises.yaml is the action that takes ‘Language’, and for each exercise {1..n}:

- creates any issues {Language}-{1…n}.
- creates pull requests {1..n} for branches {Language}-{1..n} using text from {Language}-pull_body.md

Adding exercises

issues/{Language}-ex{#}-issue.md

pull_requests/{Language}-ex{#}-pull_body.md

Branch: {Language}-{#}

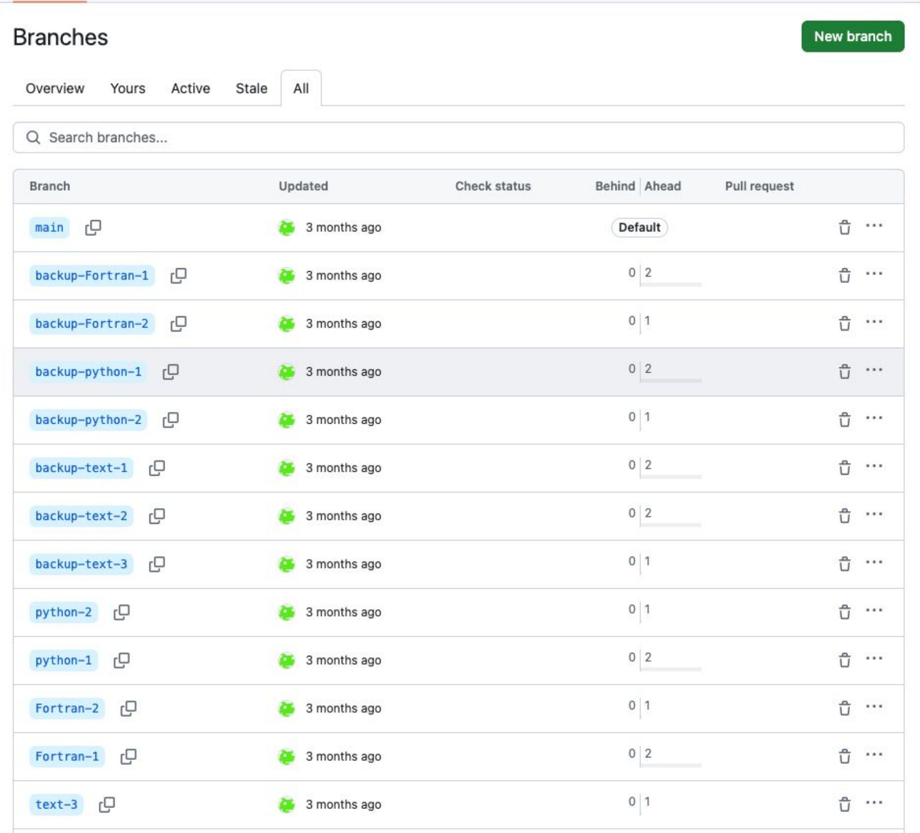
Code is in the directories:

{Language}/exercise{#}

.github/workflows/create_exercises.yaml is the action that takes ‘Language’, and for each exercise {1..n}:

- creates any issues {Language}-{1…n}.
- creates pull requests {1..n} for branches {Language}-{1..n} using text from {Language}-pull_body.md

Adding exercises



Branch	Updated	Check status	Behind	Ahead	Pull request
main	3 months ago		Default		
backup-Fortran-1	3 months ago		0	2	
backup-Fortran-2	3 months ago		0	1	
backup-python-1	3 months ago		0	2	
backup-python-2	3 months ago		0	1	
backup-text-1	3 months ago		0	2	
backup-text-2	3 months ago		0	2	
backup-text-3	3 months ago		0	1	
python-2	3 months ago		0	1	
python-1	3 months ago		0	2	
Fortran-2	3 months ago		0	1	
Fortran-1	3 months ago		0	2	
text-3	3 months ago		0	1	

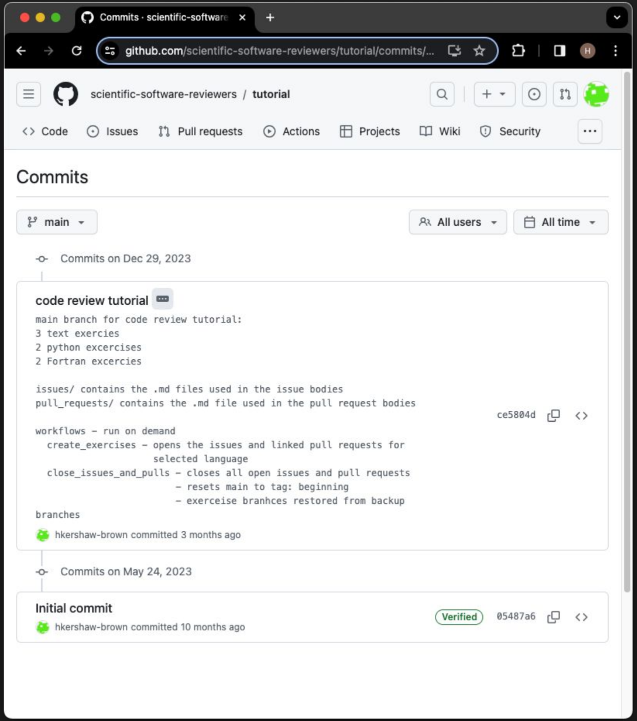
.github/workflows/close_issues_and_pulls.yaml

Resets the exercises:

Roll back the repo with git reset hard

Restores the {Language}-{#} branch from a corresponding backup-{Language}-{#} branch

Squashing git history


Commits · scientific-software
github.com/scientific-software-reviewers/tutorial/commits/...
scientific-software-reviewers / tutorial
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
main
All users
All time

Commits

Commits on Dec 29, 2023

code review tutorial

```
main branch for code review tutorial:
3 text exercies
2 python excercises
2 Fortran excercies

issues/ contains the .md files used in the issue bodies
pull_requests/ contains the .md file used in the pull request bodies

workflows - run on demand
  create_exercises - opens the issues and linked pull requests for
                     selected language
  close_issues_and_pulls - closes all open issues and pull requests
                         - resets main to tag: beginning
                         - exerceise branhces restored from backup
branches
```

hkershaw-brown committed 3 months ago — ce5804d

Commits on May 24, 2023

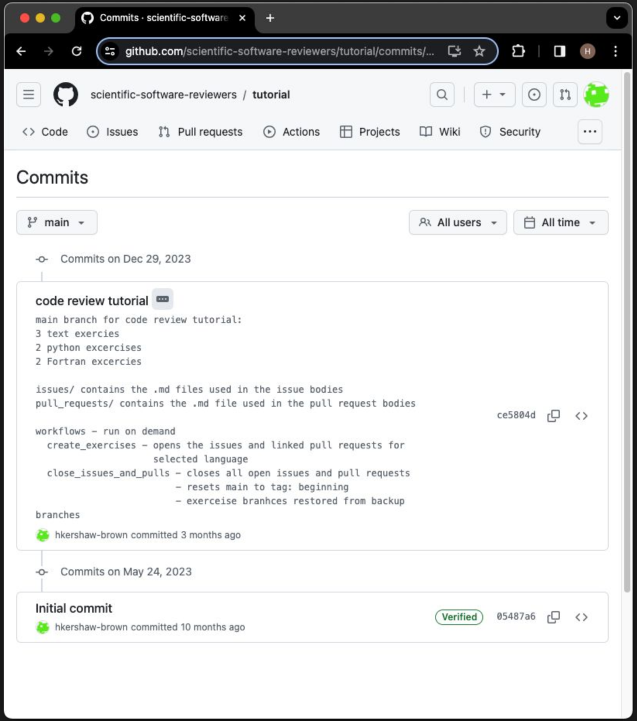
Initial commit

hkershaw-brown committed 10 months ago — Verified — 05487a6

main branch has only two commits:

- Initial commit
- Code review tutorial

Squashing git history


Commits · scientific-software
github.com/scientific-software-reviewers/tutorial/commits/...
scientific-software-reviewers / tutorial
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
main
All users
All time

Commits

Commits on Dec 29, 2023

code review tutorial

```
main branch for code review tutorial:
3 text exercises
2 python excercises
2 Fortran excercises

issues/ contains the .md files used in the issue bodies
pull_requests/ contains the .md file used in the pull request bodies

workflows - run on demand
  create_exercises - opens the issues and linked pull requests for
                     selected language
  close_issues_and_pulls - closes all open issues and pull requests
                         - resets main to tag: beginning
                         - exerceise branhces restored from backup
branches
```

ce5804d

hkershaw-brown committed 3 months ago

Commits on May 24, 2023

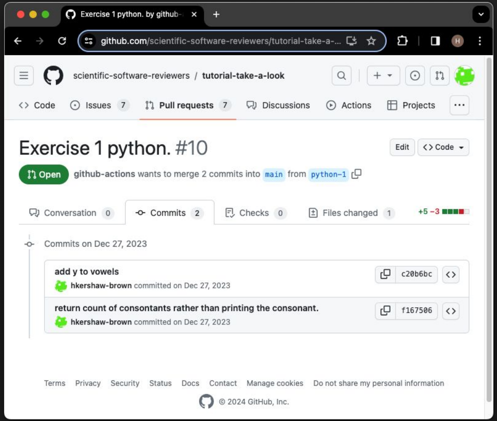
Initial commit

Verified 05487a6

hkershaw-brown committed 10 months ago

main branch has only two commits:

- Initial commit
- Code review tutorial



Exercise 1 python. #10

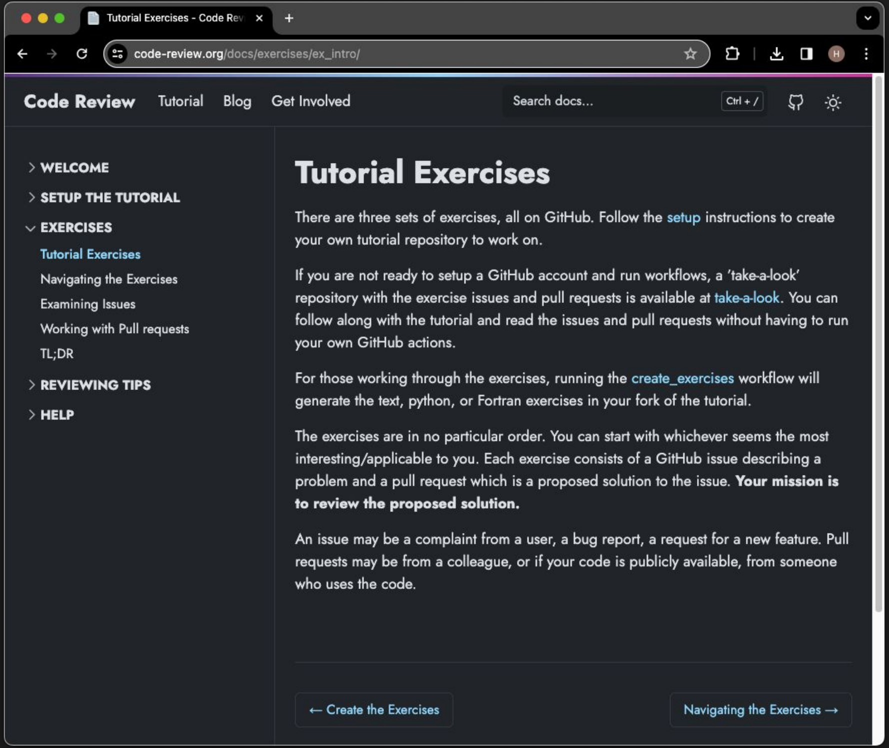
Open github-actions wants to merge 2 commits into `main` from `python-1`

Commits on Dec 27, 2023

add y to vowels
hkershaw-brown committed on Dec 27, 2023 c20b6bc

return count of consontants rather than printing the consonant.
hkershaw-brown committed on Dec 27, 2023 f167506

Examine commit history in exercises



Tutorial Exercises

There are three sets of exercises, all on GitHub. Follow the setup instructions to create your own tutorial repository to work on.

If you are not ready to setup a GitHub account and run workflows, a 'take-a-look' repository with the exercise issues and pull requests is available at take-a-look. You can follow along with the tutorial and read the issues and pull requests without having to run your own GitHub actions.

For those working through the exercises, running the create_exercises workflow will generate the text, python, or Fortran exercises in your fork of the tutorial.

The exercises are in no particular order. You can start with whichever seems the most interesting/applicable to you. Each exercise consists of a GitHub issue describing a problem and a pull request which is a proposed solution to the issue. **Your mission is to review the proposed solution.**

An issue may be a complaint from a user, a bug report, a request for a new feature. Pull requests may be from a colleague, or if your code is publicly available, from someone who uses the code.

code-review.org

https://github.com/scientific-software-reviewers/tutorial

Software Engineering is
“programming integrated over time”

Winters, T., Manshreck, T., & Wright, H. (2020). *Software engineering at google: Lessons learned from programming over time*.
O'Reilly Media

Experiences from UCAR SEA
Software Engineering Assembly

What is the SEA?

NCAR

Operated by UCAR

UCAR Software Engineering Assembly

- Foster community for software engineering professionals within UCAR
- Facilitate effective participation
- Advocate for Software Engineers

SEA Improving Scientific Software Conference April 15th-18th

https://sea.ucar.edu/conference/2024

NSF | NCAR Operated by UCAR

Code Review
Experiences from UCAR SEA
Software Engineering Assembly

NCAR

Operated by UCAR

Experiences from UCAR SEA

Join the UCAR Software Engineering Assembly for a lightly-moderated discussion on code review across UCAR.

Any discussion topics are welcome, as are all experience levels. We encourage you to share your good and bad experiences with code review.

- Do you use code review in your group? Who does the reviewing? Have you used code review to transfer knowledge between team members?
- Reviewing is hard. Being reviewed can be difficult. How do you give and receive constructive and actionable criticism?
- Do you do in person code reviews? Offline code-reviews? What works, what doesn’t?
- Do you spend too much time in review, and have ideas to improve the process?

Experiences from UCAR SEA

GitHub made it much easier to code review.

Code review feels like someone works with me and we learn from each other

Downside:Back/forth that happens, especially since the code review is not #1 priority. Can slow down the process.

When people do not know much about what others do in the code, review gives an opportunity to learn about what is going on in the project

Getting very burned out with code reviews generally
e.g. Do a review, wait ~2 weeks, can feel really negative sometimes
Recently got more negative on it but would love to hear positive experiences about it

Experiences from UCAR SEA

Used to do code reviews in person years ago. Finding bugs and avoiding problems down the line works great. Can't imagine deploying code without reviews. Couldn't maintain the code without reviews.

Experience mostly getting my code being reviewed rather than reviewing others'. Need to coordinate with each other to find the time. Trick is that it'd be helpful to walk the reviewer through the code first.

The objectives can be communicated well beforehand using a pull request template to reduce the overhead of back & forth and expectations for a due date for the pull request can be set.

Communicating what to look at in the code is really important.

Experiences from UCAR SEA

A lot of friction points about code review. Ethics around code review is not clear. Code review is a lot of times not equitable, e.g. more pushback for women's code.

Code style actions, automation could be helpful with the code reviewing process to reduce unwanted reviewing (code styling, etc.)

Systemic Gender Inequities in Who Reviews Code
The Pushback Effects of Race, Ethnicity, Gender, and Age in Code Review – Communications of the ACM
Presentation by Dr. Kelly Blincoe about code review as a socio-technical activity. Includes relevant data and potential policy implications on code review processes and impact.

Pick the most impactful aspects of the code to comment on, no need to mention everything. Impact can include functionality, quality, maintainability, readability, testability.

Experiences from UCAR SEA

Submitting changes without sufficient descriptions is less helpful.

Make it clear about the asynchronous aspect of the PRs. Also use “why would you do that?” for asking the reasoning (?)

Sometimes reviews have a lot of back & forth, and can get political. Try to keep it very non-personal. The thing being reviewed is not the person but the code that will benefit an entire project/organization.

Having been in both scientist and developer perspectives, set expectations and convey what the goals are for each group, collective set of expectations. And, things may differ from person to person, even if they are all one kind (e.g. scientist).

It’s a joint responsibility.

Consistency. Type of code you are working on (pure research vs. operational product/deliverable) and how you set expectations is also very important.

Encourage "the code" and not "your code". We are not our code

Experiences from UCAR SEA

1:1 code review in person is a bit different than remote.

Code review as an onboarding task

Do onboarding by working side-by-side rather than a remote pull request review process. Some form of pair programming.

When getting someone new to our code contributions, reach out individually with an email that clarifies some important points about the process.

How Microsoft do code reviews mentions the use of emojis to describe things like nitpick, thinking out loud, take it or leave it, etc.

👍❓❌🔧🙃💭🤡

Finding Community

Finding Community

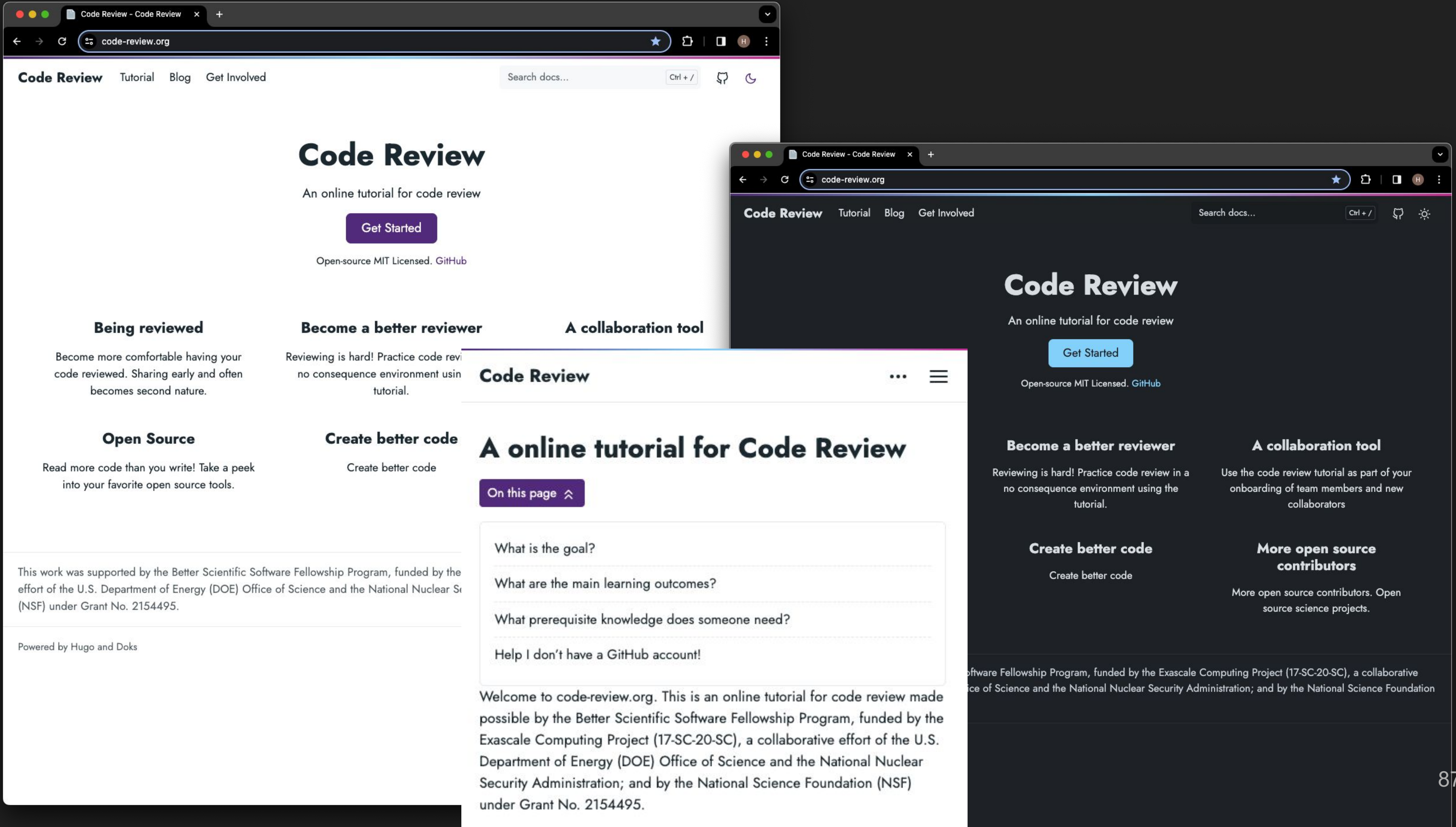
- US-RSE. A community of people who make research software happen.
- Society of Research Software Engineering which emerged from the successful grass-roots RSE movement and is the successor to the UK RSE Association.
- Better Scientific Software. A hub for scientific software development resources.
- Campus Champions. Uniting Research Computing Facilitators
- …

Finding Community

- US-RSE. A community of people who make research software happen.
- Society of Research Software Engineering which emerged from the successful grass-roots RSE movement and is the successor to the UK RSE Association.
- Better Scientific Software. A hub for scientific software development resources.
- Campus Champions. Uniting Research Computing Facilitators
- …

code-review.org

hkershaw@ucar.edu
USRSE slack



Code Review

An online tutorial for code review

Open-source MIT Licensed. GitHub

Being reviewed

Become more comfortable having your code reviewed. Sharing early and often becomes second nature.

Become a better reviewer

Reviewing is hard! Practice code revi
no consequence environment usin
tutorial.

A collaboration tool

Open Source

Read more code than you write! Take a peek into your favorite open source tools.

Create better code

Create better code

This work was supported by the Better Scientific Software Fellowship Program, funded by the
effort of the U.S. Department of Energy (DOE) Office of Science and the National Nuclear Se
(NSF) under Grant No. 2154495.

Powered by Hugo and Doks

Code Review

An online tutorial for code review

Open-source MIT Licensed. GitHub

Become a better reviewer

Reviewing is hard! Practice code review in a no consequence environment using the tutorial.

A collaboration tool

Use the code review tutorial as part of your onboarding of team members and new collaborators

Create better code

Create better code

More open source contributors

More open source contributors. Open source science projects.

ftware Fellowship Program, funded by the Exascale Computing Project (17-SC-20-SC), a collaborative
ice of Science and the National Nuclear Security Administration; and by the National Science Foundation

A online tutorial for Code Review

- What is the goal?
- What are the main learning outcomes?
- What prerequisite knowledge does someone need?
- Help I don't have a GitHub account!

Welcome to code-review.org. This is an online tutorial for code review made possible by the Better Scientific Software Fellowship Program, funded by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy (DOE) Office of Science and the National Nuclear Security Administration; and by the National Science Foundation (NSF) under Grant No. 2154495.