

# GitHub upskilling session

Mishka Nemes – 18 January 2022

Based on the Turing Way *Collaboration chapter* and the Turing Way *workshop at the EEI summer conference*

<https://the-turing-way.netlify.app/collaboration/github-novice/>

<https://zenodo.org/record/5070361#.YeaE5i-l3T8>

# Intro, background and motivation

- Collaborative tool that provides global file hosting and version control
- Useful for working collaboratively on a file, and for project / community management

## Main uses

- it can be used to **store** documentation, data and make web pages for projects.
- it provides an easy-to-use interface for **version control** that allows all activities to be recorded so you can revisit past versions and you know who made each contribution to the project.
- it has many options for **automating** repeated **project management tasks**.

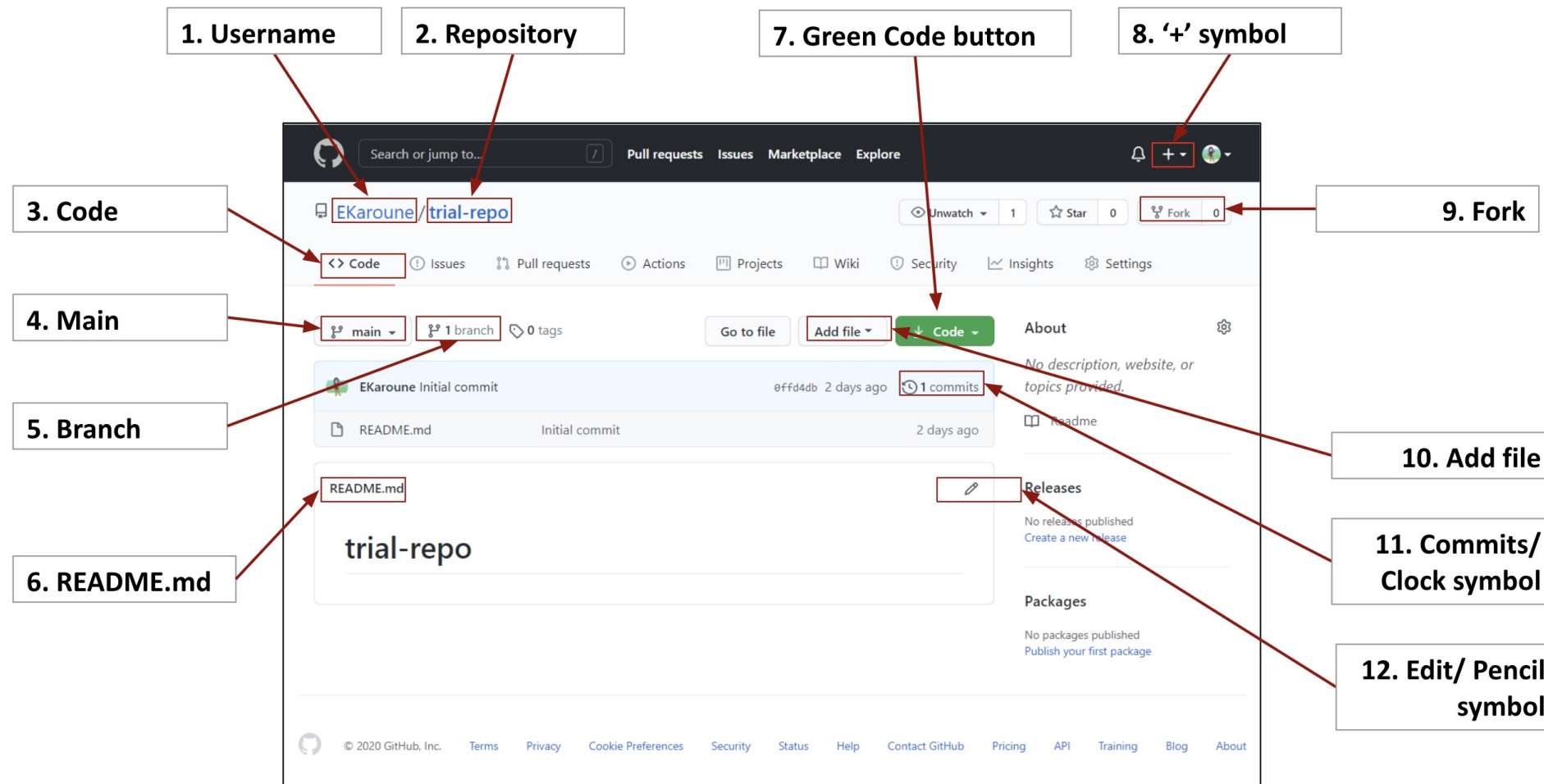
# Exercise 1 – Create your own repository

When you have created a new account and you are logged in, you need to create a new repository.

A repository or repo is the online space where you store all of the documents, data and other files for your project.

- To create the new repository, you need to click on the + sign in the top right corner (in the black band at the top of your window) and then click New repository. This will take you to a page that looks like a form.
- You will see the name of your account and you need to fill in a repository name next to it.
- Also, leave the box ticked for “public” (so your repository is open to all) and then tick the box to create a “README file”.
- Then click the green create repository button at the bottom.

# Exercise 1 – Create your own repository



# Exercise 1 – Create your own repository

## Things to note

- You must save your edits and leave a commit message
- You cannot have empty folders

## Other useful features

- License
- Code of conduct
- Contributor guide

# Markdown – the language of GitHub

Markdown (.md) is a simple text “markup” language made for quickly writing formatted text  
Great for blogs, documentation, and even writing papers.

*You are already using it in the HackMD.*

Markdown cheat sheet: <https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet>

# Exercise 2 – Write your readme file in .md

Create a README file

- Landing page for your repo
- Add about your project
- List tasks of varying expertise clearly
- Add names/ids of your collaborators
- Invite others with specific skills
- But let's write it in style with Markdown

Add a checklist

Tag the Turing institute Github

Add a list

# Version Control – language of GitHub

- Commit



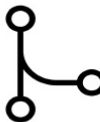
- Branch & Fork



- Pull request



- Merge

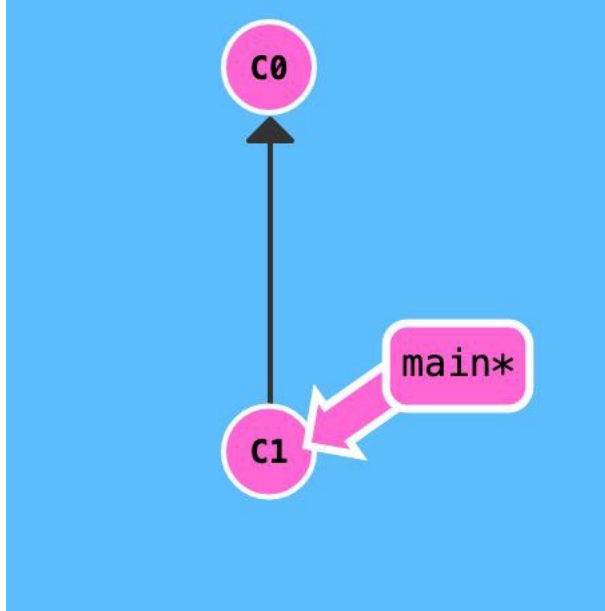




# Version Control – Commit

Saving a version of file(s)

- ‘commit’ changes with message (i.e. ‘My first commit’)



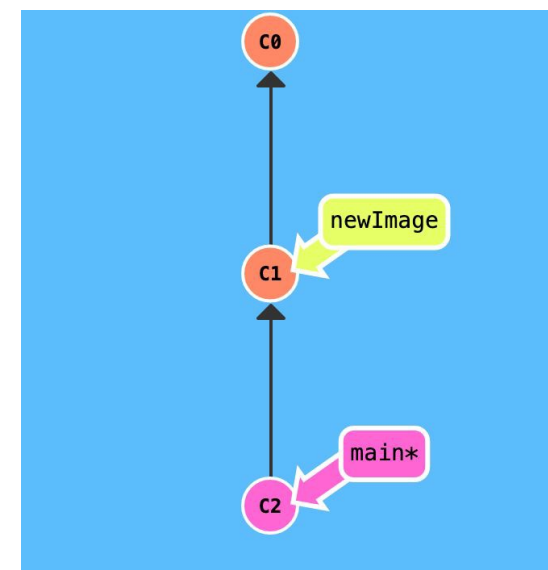
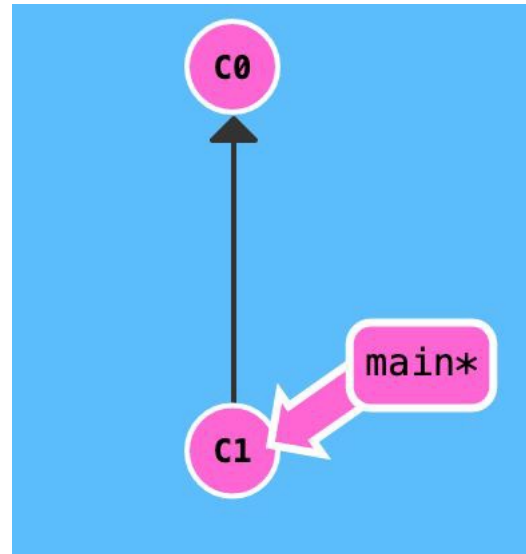
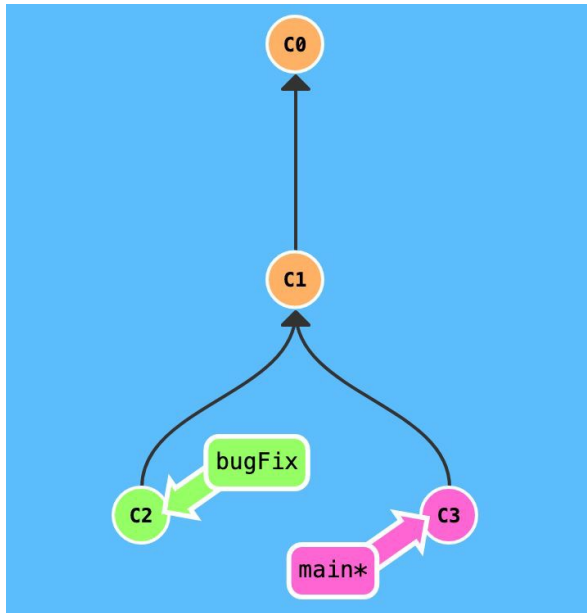
The screenshot shows the GitHub commit history for the repository 'alan-turing-institute/the-turing-way'. The URL in the browser is <https://github.com/alan-turing-institute/the-turing-way/commits/>. The page displays a list of commits grouped by date.

- Commits on Apr 26, 2021**
  - Update contributors.md** ...  
EstherPlomp committed 16 days ago ✓
- Commits on Apr 24, 2021**
  - Update hyperlink so it renders in Netlify build (#1823)**  
inwaves committed 18 days ago ✓
  - Improve grammar in compendia.md (#1822)** ...  
inwaves and malvikasharan committed 18 days ago ✓
- Commits on Apr 22, 2021**
  - Update project\_management/turing\_mybinder\_role.md**  
sgibson91 committed 20 days ago ✓

# Version Control – Branch & Fork

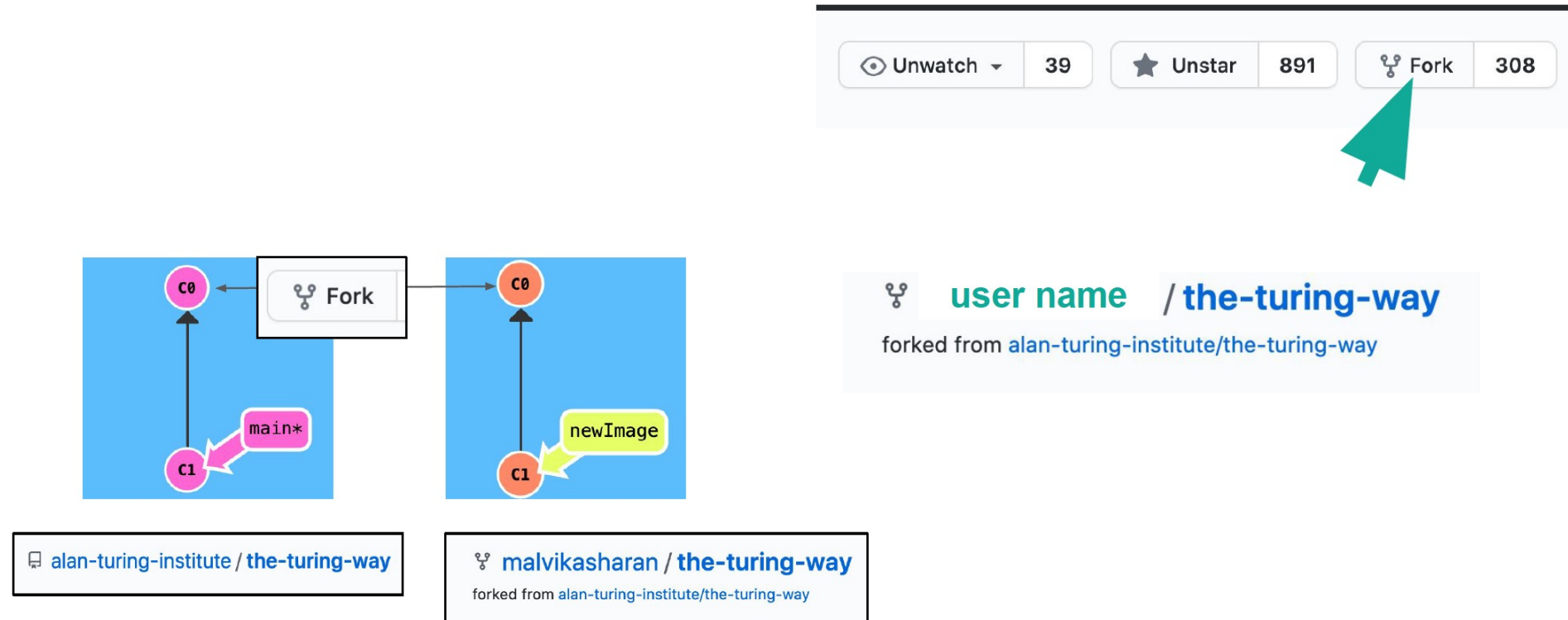
**Branch:** pointers to a specific commit in your repo, temporary places to work through a feature ← you have write access

**Fork:** copy of repo that is an entirely independent repo ← no need of write access or intent to merge with the main



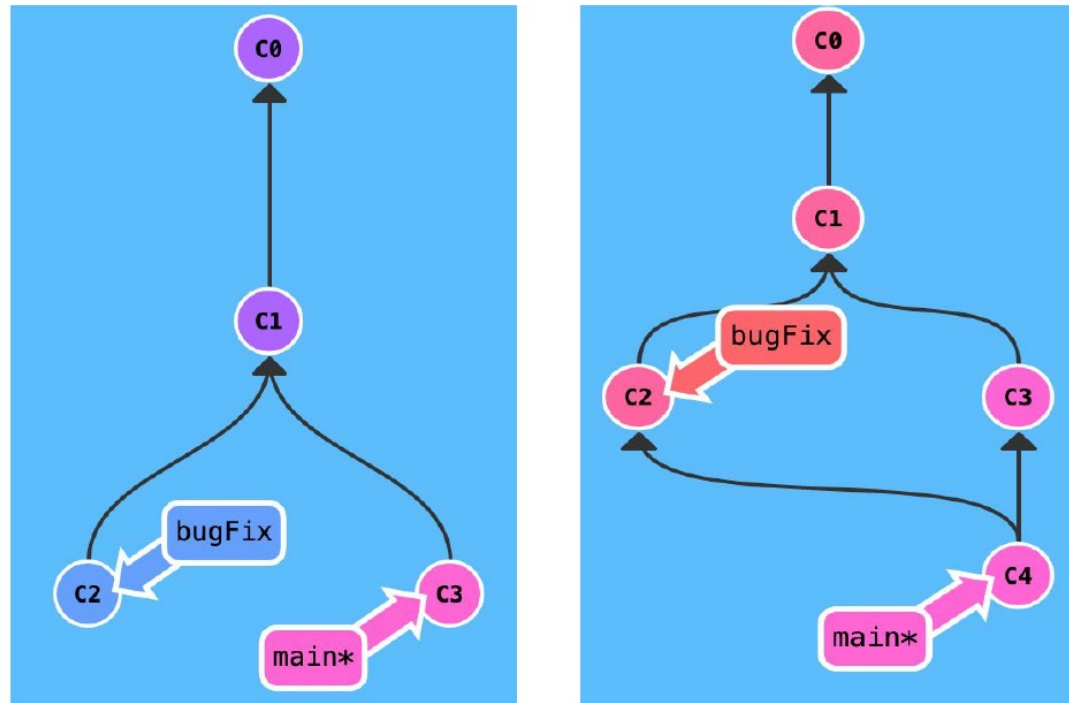
# Exercise 3 – Fork the Learning at the Turing repository

<https://github.com/alan-turing-institute/learning-at-the-turing/>

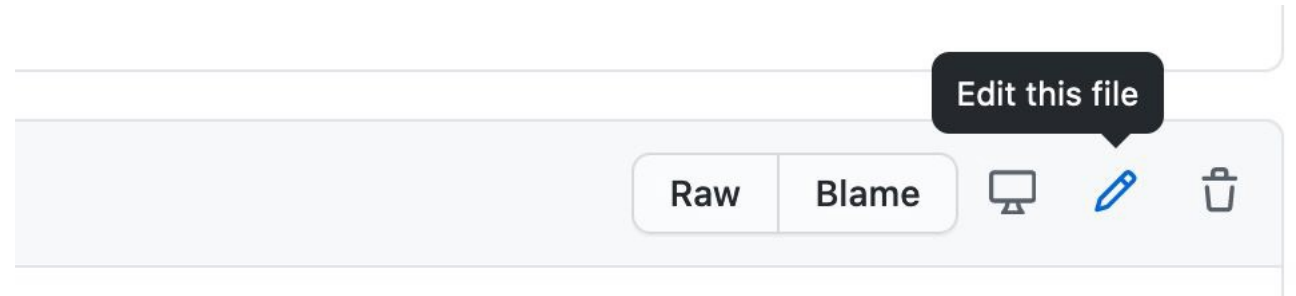


# Version Control – Pull Request (PR)

PR: Requesting to **combine** the work from a forked repo or two branches to the main repo



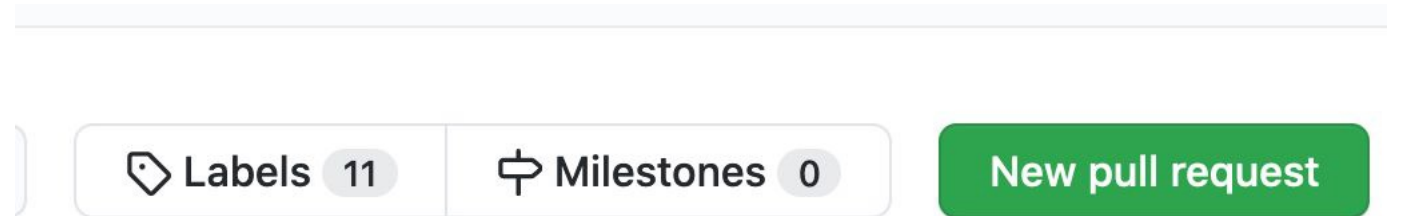
# Exercise 4 – Make a Pull Request



Edit the file by completing the fields at

<https://github.com/alan-turing-institute/learning-at-the-turing/blob/main/upskilling-workshop.md>

# Exercise 4 – Make a Pull Request

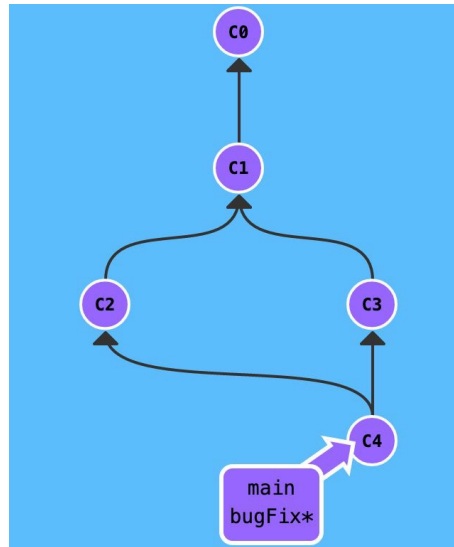
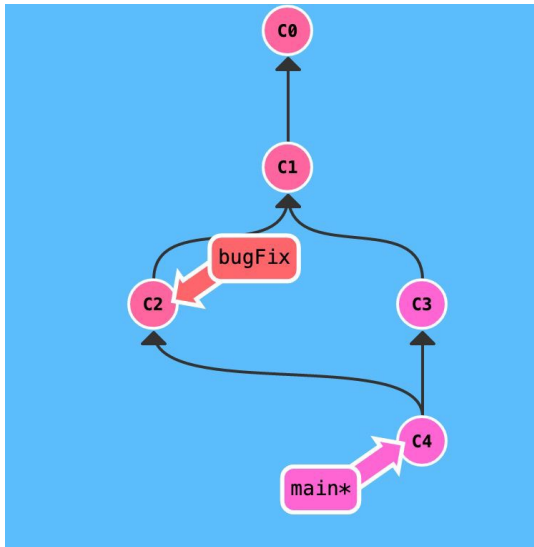


Edit the file by completing the fields at

<https://github.com/alan-turing-institute/learning-at-the-turing/blob/main/upskilling-workshop.md>

# Version Control – Merge

Merge: combining the work from two different branches together after review



# Demo: how to approve a PR and merge changes

Review changes ▾

Finish your review

Write

Preview

H

B

I

≡

<>

↗

≡

≡

☑

@

↗

↶

This looks correct.

G

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

M

☒ **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.

Submit review

NB: Users with **write access** can **approve** and merge PRs



Break 10 min

# Project management

## Projects

Visually track issues, pull requests, and notes as cards that you can arrange to suit your workflow.

[Learn more >](#)

## Labels

Organize and prioritize your work. Apply labels to issues and pull requests to signify priority, category, or any other information you find useful.

[Learn more >](#)

## Milestones

Track progress on groups of issues or pull requests in a repository, and map groups to overall project goals.

[Learn more >](#)

# Project management

## Issues

Track bugs, enhancements, and other requests, prioritize work, and communicate with stakeholders as changes are proposed and merged.

[Learn more >](#)

## Unified Contribution Graph

See all of your contributions to GitHub Enterprise and GitHub.com in one place: your profile's contribution graph.

[Learn more >](#)

## Org activity graph

See visualizations of your entire organization or specific repositories, including issue and pull request activity, top languages used, and member activity data

[Learn more >](#)

# Project management

## Org dependency insights

With dependency insights you can view vulnerabilities, licenses, and other important information for the open source projects your organization depends on.

[Learn more >](#)

## Repo insights

Use data about activity and contributions within your repositories, including trends, to make data-driven improvements to your development cycle.

[Learn more >](#)

## Wikis

Host documentation for projects in a wiki within your repository. Contributors can easily edit documentation on the web or locally.

[Learn more >](#)

# Project management: Issues & Exercise 5 – create new issue

Issues are like posts/messages

- Report bugs and errors, propose a new idea, save notes or invite discussions
- Give a descriptive header, provide details

CreatedAssignedMentioned

Q is:open is:issue author:mishkanemes archived:false

9 Open ✓ 3 Closed

Visibility Organization Sort

alan-turing-institute/learning-at-the-turing automation for issues, tags and pull request

#11 opened on 27 May 2021 by mishkanemes MVP: platform ...

alan-turing-institute/learning-at-the-turing website

#8 opened on 14 Apr 2021 by mishkanemes Peer-review th...

alan-turing-institute/learning-at-the-turing develop automated acknowledgement section

#7 opened on 14 Apr 2021 by mishkanemes MVP: platform ...

alan-turing-institute/learning-at-the-turing write contributing guidelines

#6 opened on 14 Apr 2021 by mishkanemes MVP: platform ...

# Issue vs Pull Request (PR)

With a issues, you can propose ideas for possible changes, and invite discussions before you make a PR (or do something else).

With a pull request, you can propose changes, discuss, and iterate on before you merge the changes into the project.

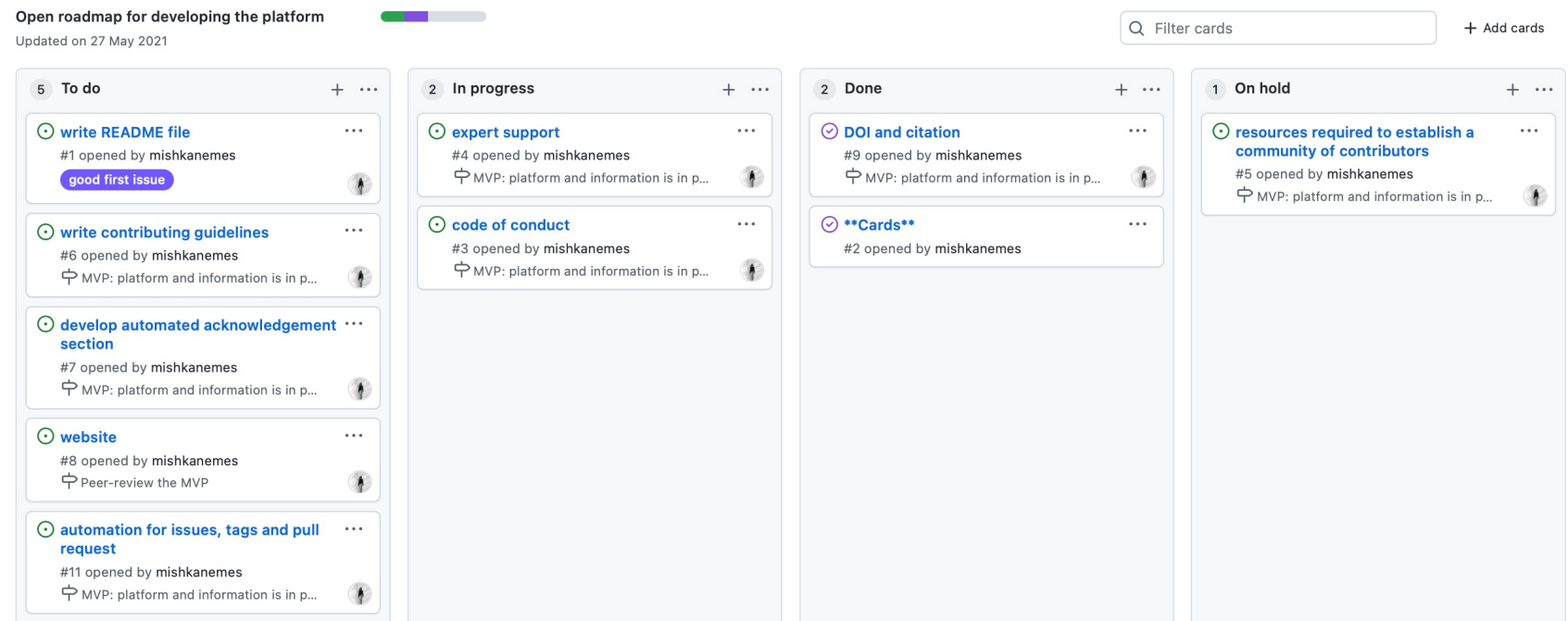
## **Using Issue & PR Features**

Different contributors can see what you are working on and offer help, share thoughts or learn from your work.

Provide sufficient details, respond patiently, accept constructive feedback and invite collaboration.

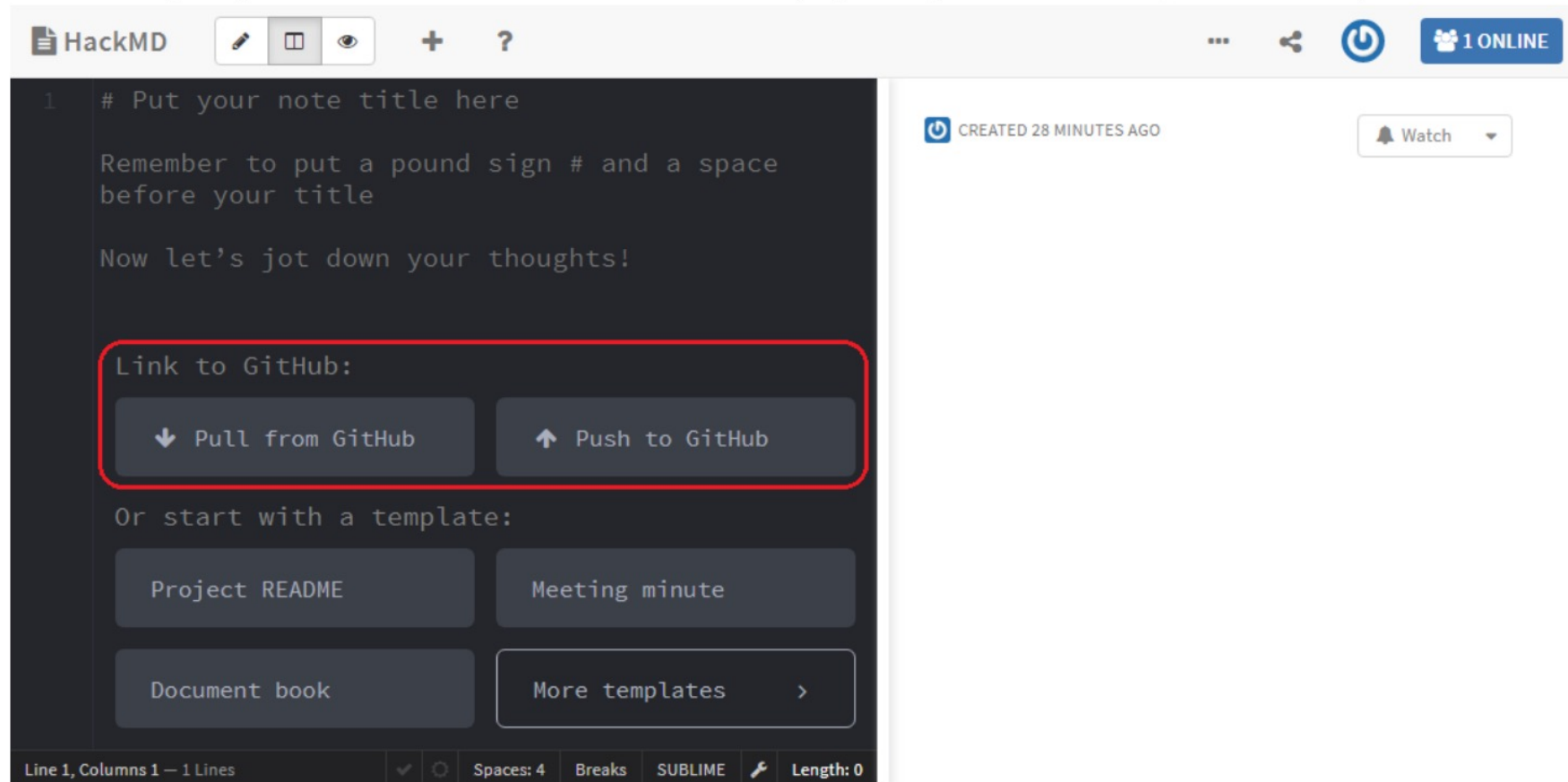
# Project management demo

- Issues
  - Turn issues into tasklist
  - Labels
  - Milestones
  - Project boards
  - Adding collaborators
  - Setting up different access levels
- 
- Download Zip
  - Clone repository



# HackMD integration with GitHub

You can sync your note from the editor of an empty note, or from the  **Versions** panel.



The screenshot displays the HackMD web application interface. At the top, a header bar includes the 'HackMD' logo, a toolbar with icons for editing, viewing, and adding content, and a status indicator showing '1 ONLINE'. The main area is split into two panels. The left panel is a dark-themed code editor containing a note with the following text: '# Put your note title here', 'Remember to put a pound sign # and a space before your title', and 'Now let's jot down your thoughts!'. Below the text, a red rectangular box highlights a section titled 'Link to GitHub:' which contains two buttons: 'Pull from GitHub' (with a downward arrow icon) and 'Push to GitHub' (with an upward arrow icon). Below this section, there is a prompt 'Or start with a template:' followed by four buttons: 'Project README', 'Meeting minute', 'Document book', and 'More templates >'. The right panel is a light-themed sidebar titled 'Versions' with a 'CREATED 28 MINUTES AGO' timestamp and a 'Watch' button. The bottom status bar shows 'Line 1, Columns 1 — 1 Lines', a checkmark icon, a circular refresh icon, 'Spaces: 4', 'Breaks', 'SUBLIME' editor mode, a pencil icon, and 'Length: 0'.



# HackMD integration with GitHub

The image shows the HackMD web application interface, which is a collaborative note-taking tool. The interface is split into two main panels: a code editor on the left and a preview on the right.

**Code Editor (Left Panel):**

- Header:** Includes the HackMD logo, a toolbar with icons for editing, viewing, and adding content, and a status bar at the bottom showing "Line 15, Columns 52 — 15 Lines", "Spaces: 4", "Breaks", "SUBLIME", and "Length: 307".
- Content:** The editor contains a document titled "# Hello World". The text is as follows:

```
1 # Hello World
2
3 ---
4
5 Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna
aliqua.
6
7 ---
8
9 - [Ut enim](@My-username/rJvCElApE)
10 - [quis nostrud](/hello-world)
11 - [exercitation](/hello-world)
12
13 ---
14
15 Make some noise!! I mean, changes, for
testing out.
```

**Preview (Right Panel):**

- Header:** Shows a status bar with "CHANGED 2 MINUTES AGO", a "Watch" button, and a "1 ONLINE" indicator.
- Content:** The preview displays the rendered version of the document. It features a large heading "Hello World", a paragraph of Lorem ipsum text, a list of three items (Ut enim, quis nostrud, and exercitation), and a final paragraph "Make some noise!! I mean, changes, for testing out.".

# HackMD integration with GitHub

Useful for:

- Storing meeting notes, templates, course / event pages in one place
- Track changes to previous versions
- Recover previous versions particularly when a HackMD file is open to all for editing

Demo!

# GitHub pages

Useful for:

- Build a website from scratch
- Create a project page / website

Demo!

<https://mishkanemes.github.io/learning-at-the-turing-demo/>

# How we would use GitHub in the Skills team

## **Project management and community engagement**

Learning at the Turing repository

- Offers a platform for others to contribute or make use of the training materials (static)
- Links to other course repositories
- Complements the Moodle platform (storage, notebook maintenance)
- A way to acknowledge contributors and maintainers

*All data scientists and related researchers are on GitHub, so hopefully knowing how to use GitHub provides a way to engage with them and learn how to use the platform to browse other people's repositories!*