

---

# Housekeeping

- Please respect the **Code of Conduct**
- Handouts available in:  
<https://github.com/alan-turing-institute/github-introduction>
- Use the **post-its** provided to let us know if you have a question or need help
- Please ask questions on anything related to these topics!

# The Alan Turing Institute



Online repo: [github.com/alan-turing-institute/github-introduction](https://github.com/alan-turing-institute/github-introduction)

All illustrations by Scriberia for The Turing Way community used under CC-BY 4.0 License

Zenodo at DOI: <https://doi.org/10.5281/zenodo.8221283>

## Git Good: an Introduction to GitHub for Collaboration



**Sophia Batchelor**  
Research Community Manager, AIM RSF



**Malvika Sharan**  
Senior Researcher - Open Research, Co-Lead The Turing Way, Lead RCM Team



**Cassandra Gould van Praag**  
Senior Research Community Manager, TRIC:DT



**Emma Karoune**  
Senior Research Community Manager, Health



**Eirini Zormpa**  
Research Community Manager, AIM RSF



**Gabin Wilfried Kayumbi**  
Senior Research Community Manager, DCE



**Anne Lee Steele**  
Research Community Manager, The Turing Way

---

# Let's Git Good:



- **Part 1: Why and basics**
- Exercise 1: Finding an Issue and contributing to a project.
  - You will practice: Using issues to contribute and document decisions
- **Part 2: Advanced**
- Exercise 2: Editing a file and making a Pull Request.
  - You will practice: Markdown; contributing, pull requests
- **Bonus Exercise:** Make your own repo, edit your readme

# Collaboration

Working together, often not at the same time or in the same place (distributed)

Challenges:

- People editing the same line
- Incorporating different ideas
- Different versions of the doc



# What is version control?

An approach to **recording** changes in a file over time so that you can:

- track history
- review specific changes
- go to earlier/later versions  
(back up!)



---

# Git & GitHub



Git is one of the most widely used **software** for version control. It is free and open source!

You can use git **on your computer** through the command line or a GUI.



GitHub is a popular **website** for hosting and sharing projects you have been tracking with git.

Flexible and securely control of access to materials, and open publishing.



And version controlled documentation!

# Where the world builds software

Millions of developers and companies build, ship, and maintain their software on GitHub—the largest and most advanced development platform in the world.

Email address

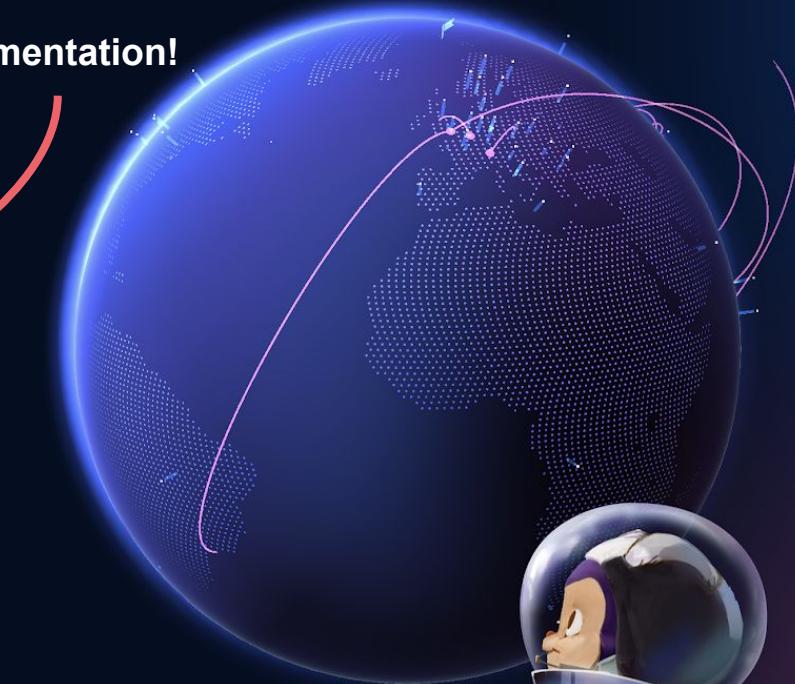
Sign up for GitHub

56+ million  
Developers

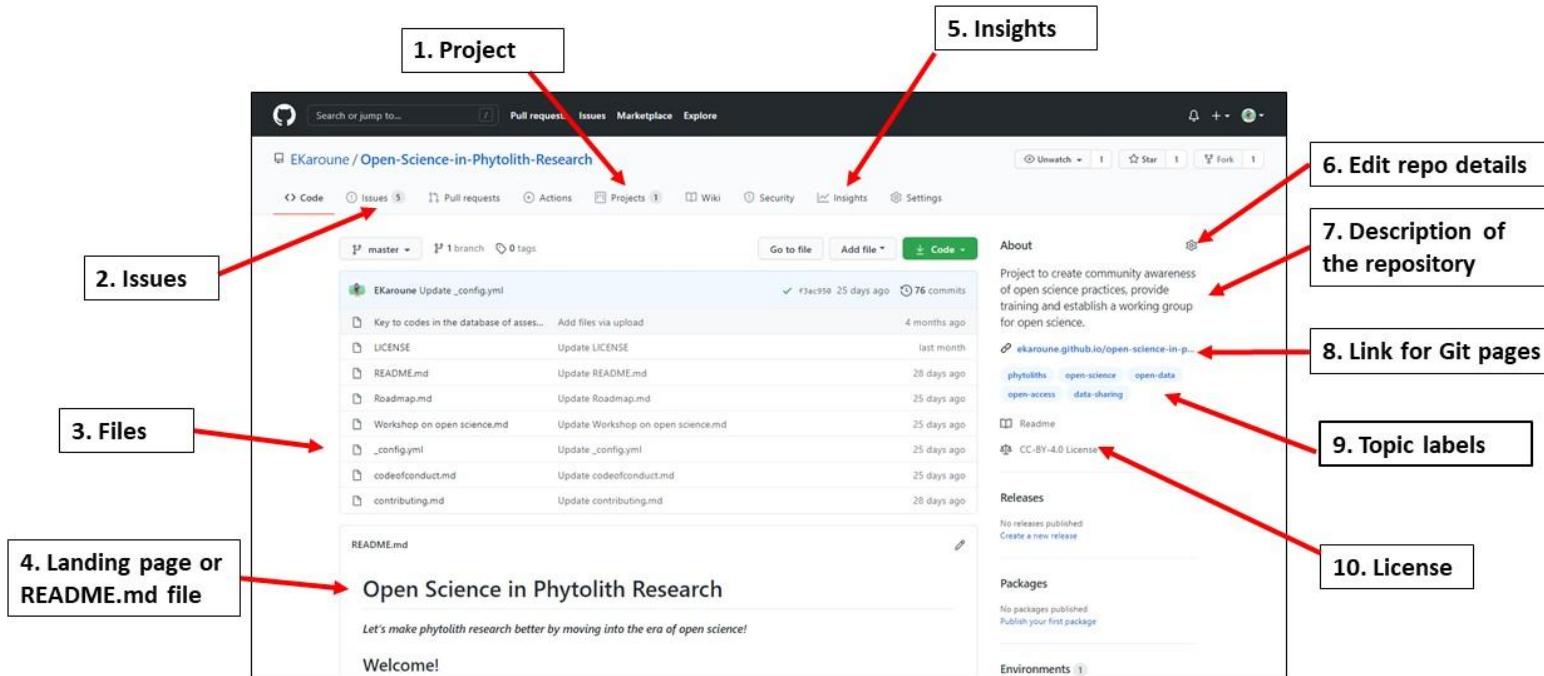
3+ million  
Organizations

100+ million  
Repositories

72%  
Fortune 50

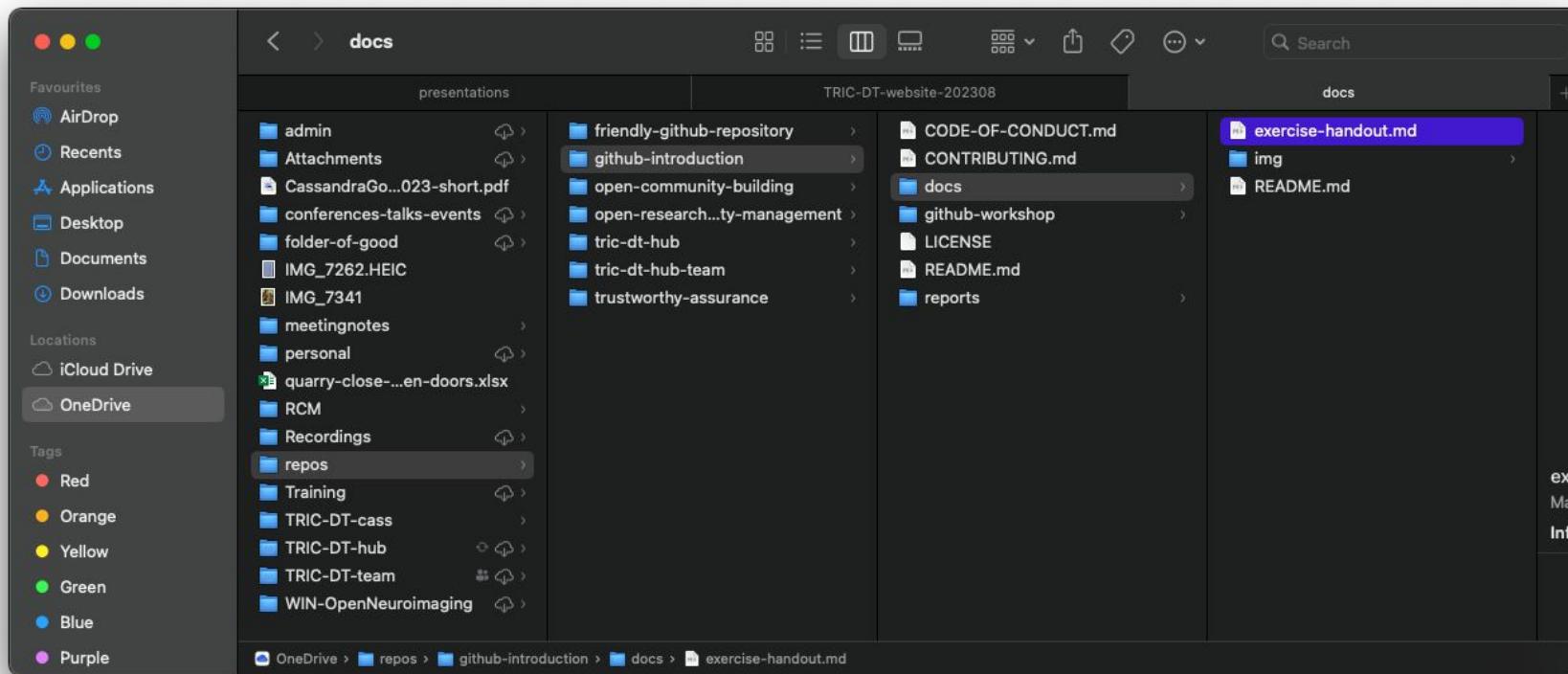


# GitHub - online (“remote”)



[github.com/alan-turing-institute/github-introduction](https://github.com/alan-turing-institute/github-introduction) ; [github.com/alan-turing-institute/open-research-community-management](https://github.com/alan-turing-institute/open-research-community-management) ;  
[github.com/alan-turing-institute/the-turing-way](https://github.com/alan-turing-institute/the-turing-way)

# GitHub - local



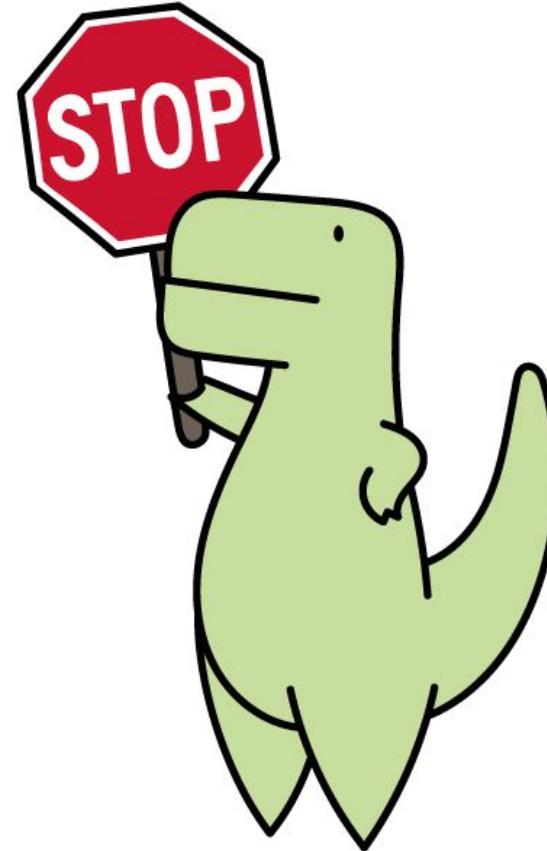
Online repo: [github.com/alan-turing-institute/github-introduction](https://github.com/alan-turing-institute/github-introduction)

---

# GitHub Vocabulary: Repository

- Repository (repo)  = filing cabinet / directory
- Commit  = save point
- Issue  = feedback and “to-do” list
- Branch & Fork  = different “streams” of work
- Pull request  = ask to incorporate your changes
- Merge  = incorporate changes

# Questions?



LoofandTimmy.com

---

# Exercise 1: Commenting on an issue.

- Go to the repository we've set up for this session.
- Comment on the issue titled "**Exercise 1: Getting started**".

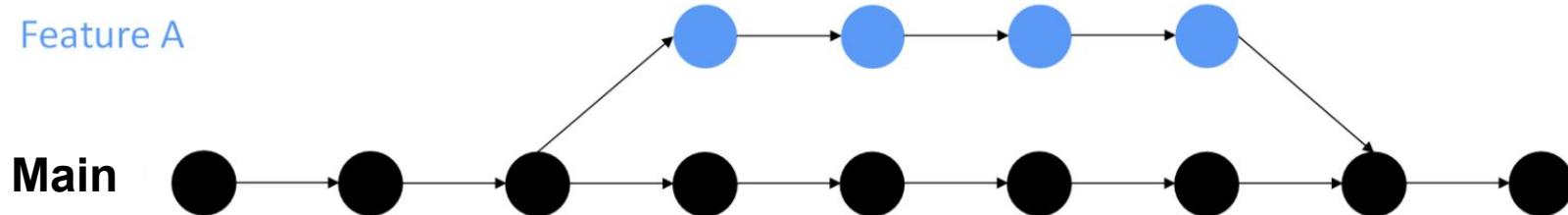


---

# ⌚ Collaborating in GitHub

When working with others, we want to:

- draft ideas NOT on main branch
- reviews and editing from team members before finalising new additions



*Fig. 18* An illustration of a development and main branch in git.

---

# Branch vs Fork

A temporary place in **your** repo pointing to a commit

Is used to work on an idea  
(like a solution to a problem)  
→ *requires ‘write’ access to the original repo*

**Fork:** An **independent** copy of a repo

Can be used for independent work or collaboration →  
*does not require write access to original repo*

# Branch

A temporary place in **your** repo pointing to a commit

Is used to work on an idea  
(like a solution to a problem)  
→ *requires ‘write’ access to the original repo*

The screenshot shows a GitHub repository interface. At the top, there are navigation buttons: 'main' (with a dropdown arrow), '4 branches' (highlighted with a red box), and '0 tags'. To the right are buttons for 'Go to file', 'Add file', and 'Code'. Below this, a list of commits is shown, starting with a merge from 'BrainonSilicon' 18 hours ago. The commits are: 'Update issue templates' (20 hours ago), 'Update glossary formating' (18 hours ago), 'docs: update .all-contributorsrc [skip ci]' (23 days ago), 'Add a .gitignore file and ignore the \_build directory' (27 days ago), and 'fixed typo' (yesterday). The 'README.md' file is open, displaying the content: 'Welcome', 'all contributors 4', and 'About the Glossary of Terms'. A note at the bottom states: 'The AIM Research Support Facility (RSF), based at The Alan Turing Institute with collaborators from Swansea University and the University of Edinburgh, offers AI and advanced data science support to the research teams part of the NIHR's Artificial Intelligence for Multiple Long-Term Conditions (AIM) project.' Another note says: 'Our community has compiled a [Glossary](#) of common terms relating to Health, Medicine, Artificial Intelligence,'.

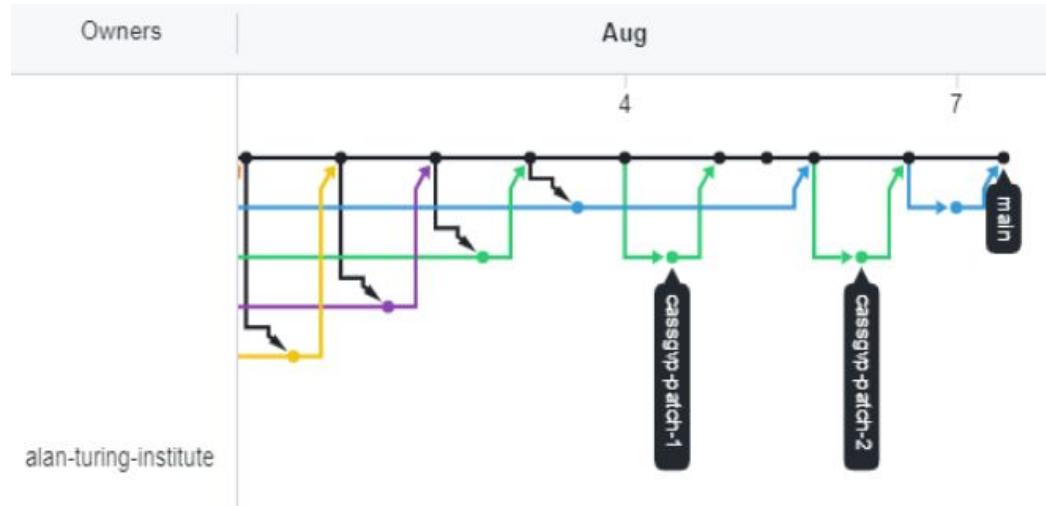
# Pull Request & Merge

## Pull request

request to add your changes from a branch back into main.

## Merge

act of incorporating new changes (commits) from one branch to another.



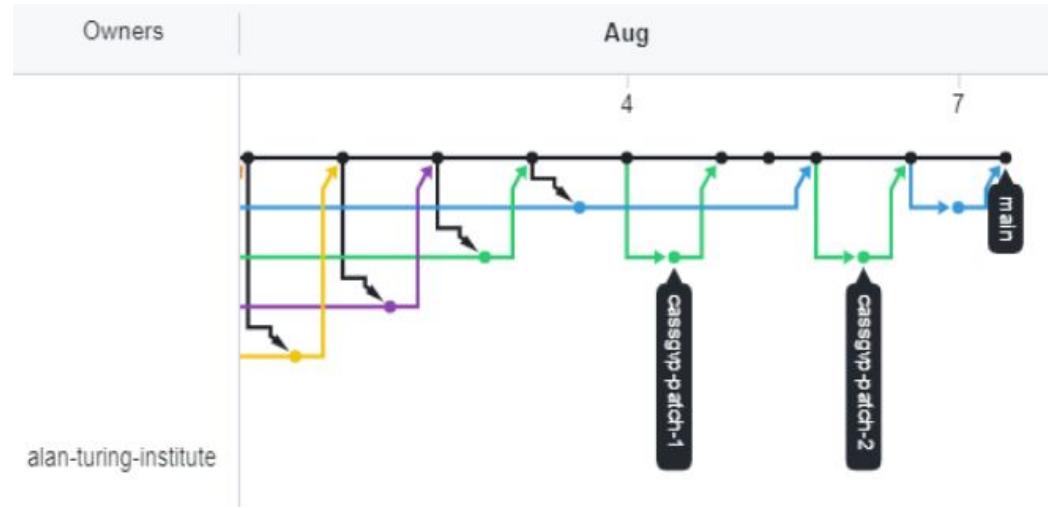
# Pull Request & Merge

**Pull request**

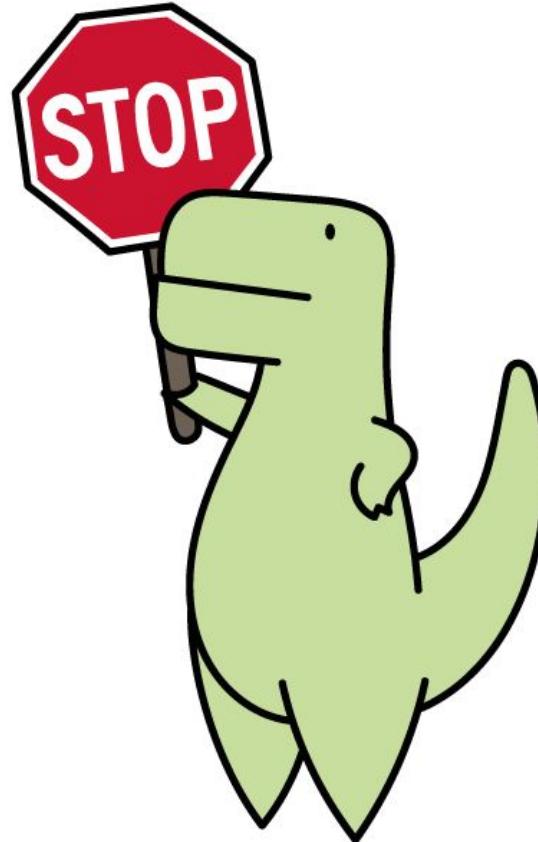
request to add your changes from a branch back into main.

**Merge**

act of incorporating new changes (commits) from one branch to another.



# Questions?



LoofandTimmy.com

---

# GitHub Exercise 2: Edit a file and make a Pull Request.

1. Go to the file labelled **introductions**
2. Edit the file and write about your **favourite pizza topping**
3. **Commit** the change and submit a **Pull Request**.



---

# GitHub Stretch Exercise: Create a new repo

1. Create a repo
2. Edit your README.md



---

# ⌚ Branches, PRs, and Merging

If you own a repo or are an official collaborator:

- a. you will **Create a new branch**
- b. work on the new branch
- c. **create a Pull Request** for the changes you've made
- d. a collaborator will decide to merge them in.

---

# ⌚ Forks, PRs, and Merging

If you **do not** own a repo and aren't an official collaborator:

- a. you will **Create a fork**
- b. work on your own fork
- c. **create a Pull Request** for the changes you've made
- d. the repo maintainer will decide to merge them in.

---

# GitHub Skills Summary

- Repository to store your project 
- Issues to track tasks 
- Writing in Markdown to add links and content
- Branches and Pull requests to collaborate 

---

# Acknowledgements

- The Brilliant Club, support staff, and Juliette Sexton.
- Kirstie Whitaker, Malvika Sharan, *The Turing Way* community, friends & collaborators
- Sophia Batchelor, Malvika Sharan, Cass Gould van Praag, Emma Karoune, Eirini Zormpa, Gabin Kayumbi.
- GitHub workshop based on [Friendly GitHub Intro by Kirstie Whitaker](#), [GitHub Collaborating Document by Malvika Sharan](#) & [GitHub for Collaboration](#) by Sophia Batchelor.
- Original artwork by Scriberia: <https://doi.org/10.5281/zenodo.3332807>

THANK YOU !!

