

Git/Hub for Collaborative DataSci & DataEng



With the AI Club for Gender Minorities, London

26/02/2019

Actually, you're here to learn:

~~Git/Hub for Collaborative DataSci & DataEng~~

Good Collaboration in Nearly Anything File-Based, using Github

By Alessia Visconti, Ester Ramos & Fei Phoon

Good collaboration...

- makes a safe space for constructive and honest discussion.
- makes room and time for teaching and learning.
- accepts mistakes & mishaps - and that we'll help each other fix and learn from them.
- does not blame and shame.
- gives credit and recognition where it is due.
- is first and foremost about working with and respecting other people.

Good collaboration...

- is NOT a "soft", optional skill for engineers - it's essential!

Git & Github are only tools to help us focus on contribution and discussion.

Git vs Github

- **Git != Github**
- **Github:** a web application that allows people to store and collaborate on a project. Today we'll learn to use this!
- **Git:** the underlying magic/the engine under the hood, that manages contribution history and order.

Basic Git Concepts

1. **Repository:** a project.
2. **Branch:** a collection of changes.
3. **Master:** the "default" branch, also the most recent, fully-approved version of the project.
4. **Commit:** a saved change on a branch. So branches are groups of changes with a common objective.
5. **Pull request (PR):** a request to add your branch of changes to `master`.
6. **Merge:** the action that accepts your branch into the master branch.

Today's learning objectives

+++++

- Use these basic Git concepts on a project, using the Github web UI.

Workshop Time

See: <https://github.com/feiphoon/github-workshop/>

Reminders for collaboration:

- Be kind (you can still be firm and opinionated!)
- Leave constructive comments
- If you feel an online conversation is getting out of control, retry the discussion in person.
- If you're lost, ask for help! There are no stupid questions, only lost people who are feeling sad that they didn't ask for help sooner.

A Recap +++++

- Git concepts
- High-level workshop tasks by sections

Github in the Wild +++++

- Today's process - what many people use, from individuals and small teams, to developers & engineers at companies
- Other patterns for collaboration with Git, such as git-flow (a fancier version of what we did today,), open source and forking

Github in the Wild +++++

- Examples of non-code projects on Github:
 - A crowdsourced travel itinerary:
<https://github.com/dylanegan/travel>
 - A novel:
<https://github.com/gregorygershwin/Benjamin-Buckingham-And-The-Nightmares-Nightmare-Novel>
- Open source libraries with collaborators all over the world, e.g. <https://python-sprints.github.io/>
- Other interfaces for Git - workflow preferences: Bitbucket, Gitlab, Github Desktop, Git command

Your future lies in:

- finishing these exercises with your teammates (if you haven't already!)
- starring this repository
(<https://github.com/feiphoon/github-workshop>)
for reference - we've added some light recap reading
- looking at the history of the workshop repository - we collaborated remotely to create it!
- exploring Github further & using it for a real project

Question Time

Thank you! +++++