Activity 1: Make a repo

Modern DevOps for Systems Biologists

Jun Allard

A. Make a repository

i. Go to github.com/new. Give it a name like `my-first-repo`. Keep it **Private**.

ii. Tick **Add a README file**.

iii. In the **Add .gitignore** dropdown, choose the template for whatever language you actually work in — Python, R, Julia, MATLAB. Then click **Create repository**.

iv. (Optional) Open the `.gitignore` it made for you and inspect it.

v. Click `README.md`, click the pencil icon, delete what is there, and paste this in its place:

```
# Rotation project

## Results so far

Nothing yet.

## Conditions

| Condition | Temperature |
| control | 25 C |
| heat shock | 42 C |

## Overview

Measuring how fast the reporter turns over under heat stress.
```

Commit it.

B. Editing and committing

Perform each of the following edits, with each one being a separate commit, with a semantic commit message. Use the **Preview** tab in the editor as you go, so you can see what your Markdown actually renders as.

i. Add a column to the table that records a rate k.

ii. But the conditions table in `README.md` does not render properly. It comes out as one long line with pipes in it, instead of a table. Fix it.

A Markdown table needs a row of dashes under its header, separating the header from the body. Without it, the whole thing is just a paragraph that happens to contain | characters.

iii. Add a **Data** section to `README.md`, saying where the large files for this project live — a Dropbox folder, a Google Drive, a server — and why they are not in this repository.

iv. Put the sections in a more standard order: Put **Overview** at the top.

C. Read your own history

i. Go to your repository's front page and click on the commit count — the little clock icon with a number next to it.

ii. Six months from now you want the commit where you recorded k. Can you find it from the messages alone, **without opening a single commit**?

D. Optional: the hash that cannot cite itself

Every commit has a hash — the string like `e4f19a2` beside it in the list. It is a fingerprint of everything in the project at that moment. It is how the three places in the lecture — data, code, manuscript — each record which version of the others they go with.

i. Copy the hash of your most recent commit. Add a line to `README.md` reading **This version:** `e4f19a2`, with your own hash, and commit it.

Your README now cites the version before it. A figure caption in a manuscript does exactly this, which is what lets you regenerate that figure years later.

ii. Are you able to make the `README.md` state the current commit hash? *Hint: do not try for too long.*