Introductions

COGS 108 Spring 2025
<TA Place your name here>
D1

<Email>

OH: <OH>

Discussion slides and materials adapted from Sam Lau (TA: WI20)

Section Philosophy

- Attendance is not required
- But we recommend to come for
  - Review and guidance
  - Work on the assignment and projects
  - Asking questions directly (to TA/IA, and your classmates!)

Project

- Form a group of 4-5 students
- Feel free to talk to others right now! Chat with your classmates about your interests, region, skills etc.
- Use Campuswire
- Start working towards the project as soon as possible
- https://github.com/COGS108/Projects

Programming

This course assumes basic programming knowledge

- But not much!

Programming

Resources:

- Codeacademy
- Start Here: https://github.com/COGS108/Tutorials/blob/master/01-Python.ipynb
- Python in detail: https://jakevdp.github.io/PythonDataScienceHandbook/
- Pandas: https://www.dataschool.io/python-pandas-tips-and-tricks/
- Git: https://guides.github.com/activities/hello-world/

Programming

Cheatsheets

- Google: ‘python cheatsheet’, ‘pandas cheatsheet’, ‘git cheatsheet’ (find one that’s good for you)

Policy on AI Generated Content

Policy on using AI programming assistance

I believe that using large language models (LLMs) or other kinds of AIs can help a good programmer work more efficiently. I also believe that using AI assistance will probably slow down the development of a beginning programmer into a good programmer.

My advice: if you struggle to conceptualize how you want to write a program you should probably NOT use an LLM. The beginning or intermediate programmer needs to practice their craft... just like you will never get to be great at a video game by just watching other people's speed runs. I think its fine to use AI assistance if you can immediately imagine how to solve the problem, but you just want help with boring implementation details, or to see alternative algorithms you could use, or help writing it faster.

You can use AI to help you program as long as you:

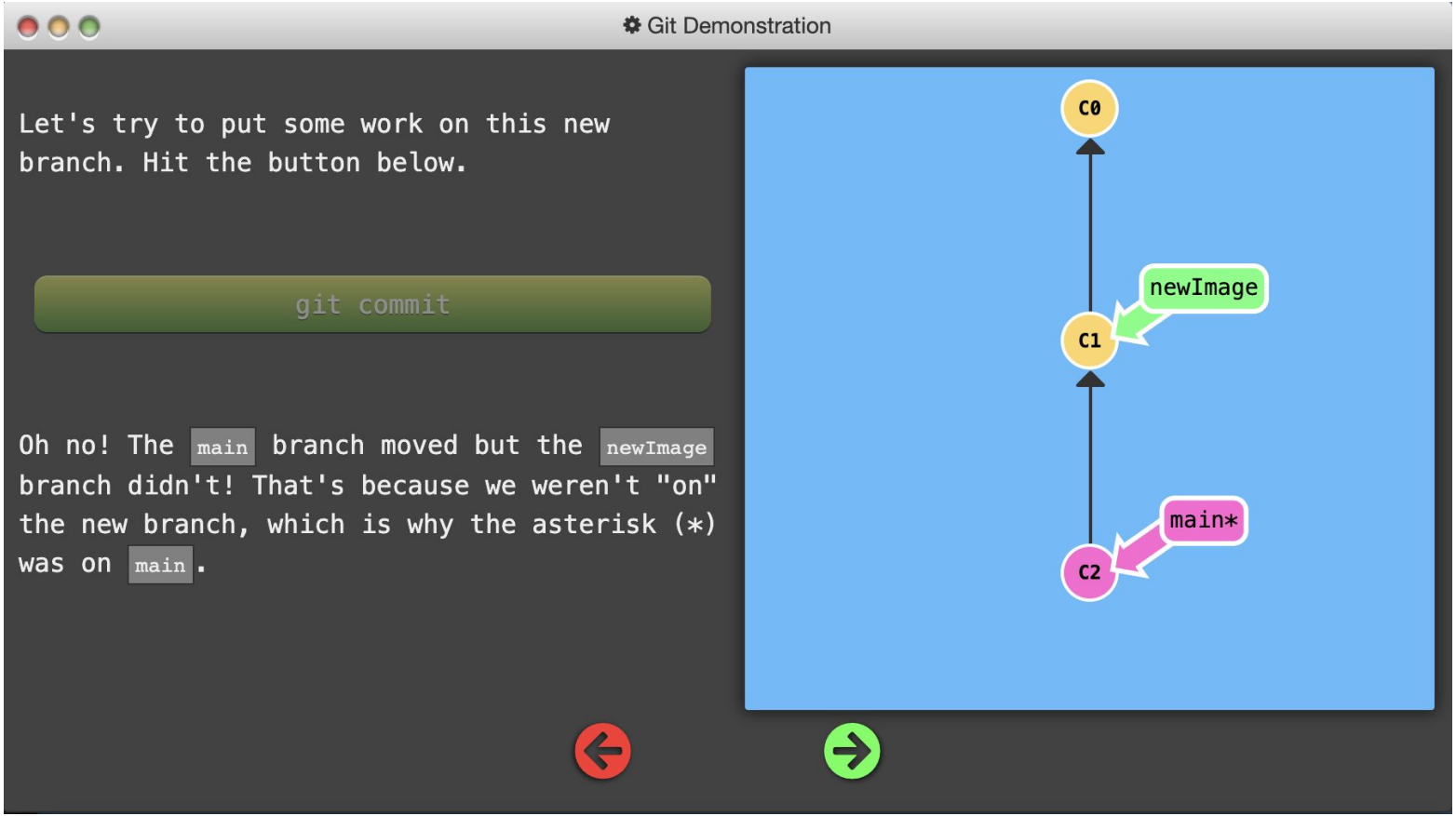
- make a code comment that cites the AI used, and provides an estimate of how much code in a given block is machine generated. For instance you might write this `# The (code/design) of this function is (completely/mostly/partially) generated by Github Copilot from the prompt "write a python function to bubble sort a list"` Feel free to include a description of any specific changes you made from the machine generated code... it was edited to reduce execution time, to deal with edge cases, to deal with an empty data file, etc..
- don't assume LLM code is working and just hand in without checking. You are always responsible for functionality and understanding how something works.
- understand that programming with LLMs still requires you to do programming. But instead of creating code from scratch (the part many people enjoy) you will need to do debugging and unit testing (the part many people don't like)
- you understand that you may be asked to explain your code at any time. If you can't explain how your code works and why the design is that way you may lose points

Git

Version control system!

- Go to https://git-scm.com/downloads
- Choose your Operating System (Windows/OS X/Linux)
- Follow the steps specific to your OS
- Verify installation: In terminal type “git –-version”

learngitbranching.js.org



https://about.gitlab.com/images/press/git-cheat-sheet.pdf

A Git installation

For GNU/Linux distributions, Git should be available in the standard system repository. For example, in Debian/Ubuntu please type in the **terminal**:

```
$ sudo apt-get install git
```

If you need to install Git from source, you can get it from git-scm.com/downloads.

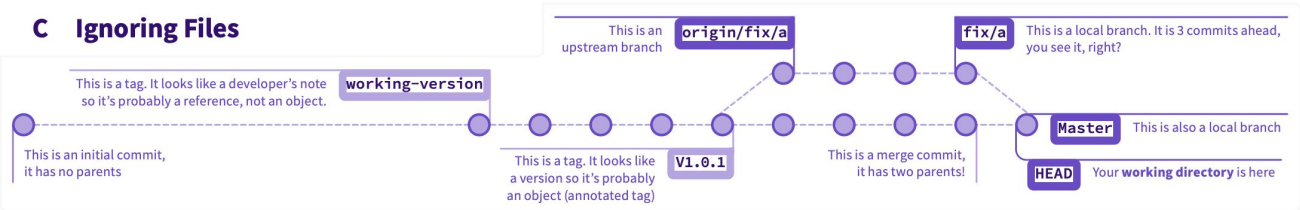
An excellent Git course can be found in the great **Pro Git** book by Scott Chacon and Ben Straub. The book is available online for free at git-scm.com/book.

B Ignoring Files

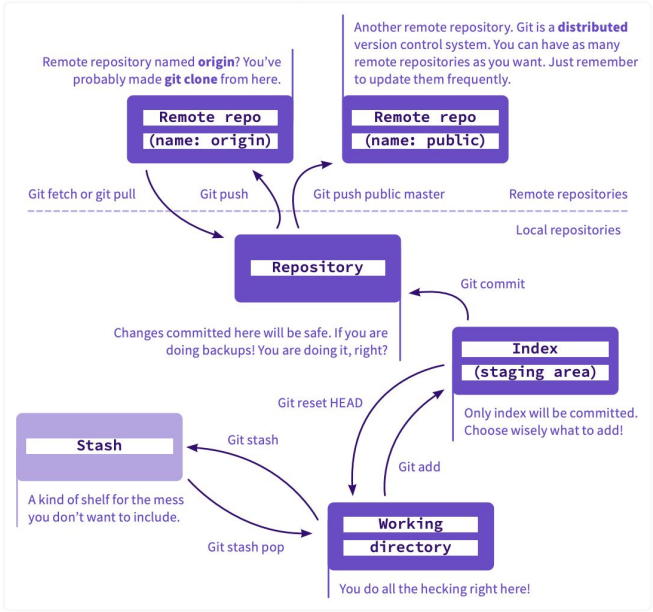
```
$ cat .gitignore
/logs/*
!logs/.gitkeep
/tmp
*.swp
```

Verify the .gitignore file exists in your project and ignore certain type of files, such as all files in **logs** directory (excluding the **.gitkeep** file), whole **tmp** directory and all files ***.swp**. File ignoring will work for the directory (and children directories) where **.gitignore** file is placed.

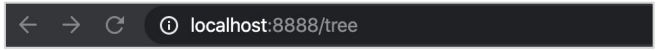
C Ignoring Files



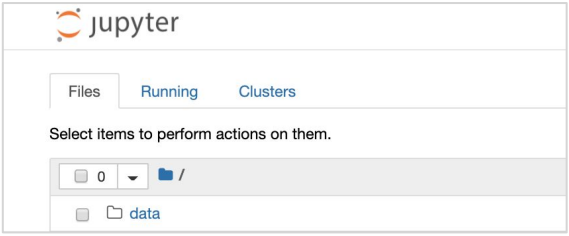
D The zoo of working areas

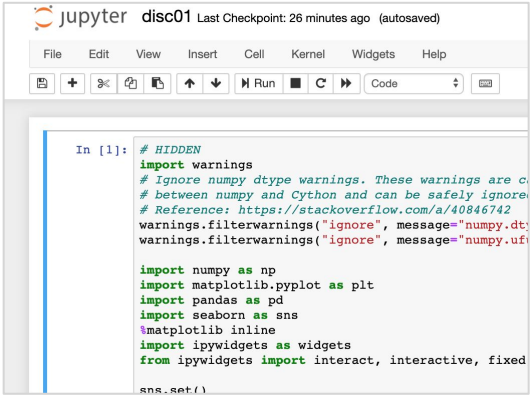


Jupyter



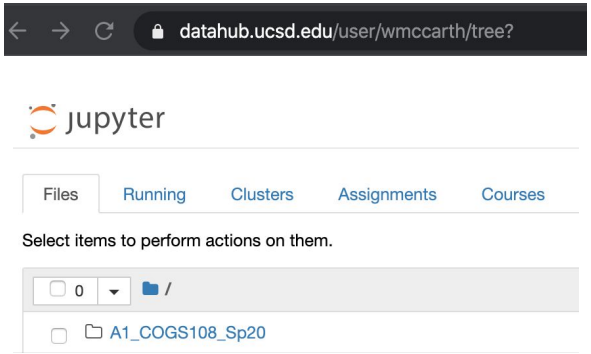
- Python code is run on a python interpreter
- Jupyter is a program that creates an interface for typing python code in a browser, that also runs that code in a python interpreter
- What does this mean?!
  - Jupyter is a way of running python programs from a browser (like chrome) (hooray!)





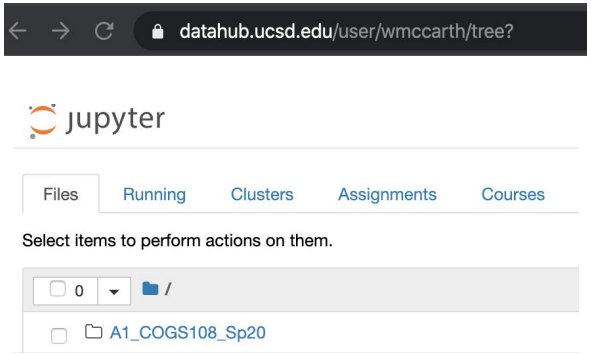
datahub.ucsd.edu

- Jupyter runs python code in a browser.
  - But Jupyter is itself just a program that's running on a computer somewhere.
- datahub lets you interact with Jupyter that's running somewhere else.



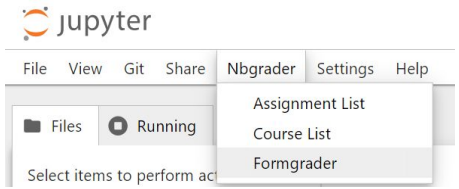
datahub.ucsd.edu

- What does this mean?!
  - You don't need to worry about installing Jupyter
  - You can use datahub to create and run python programs (online)
  - You can use this interface to fetch and submit assignments



Working on your assignments

- Log into datahub.ucsd.edu
- Go to Assignments tab (or Nbgrader->Assignment List if you are using the new container)
- ‘fetch’ assignments you have access to -> Submit after completion
- Demo of this workflow



Your time to …

- Talk to your classmates to find potential teammates!

Python Review Notebook

https://github.com/COGS108/Lectures-Sp25/blob/main/section/01_pythonreview.ipynb