Welcome to BILD62

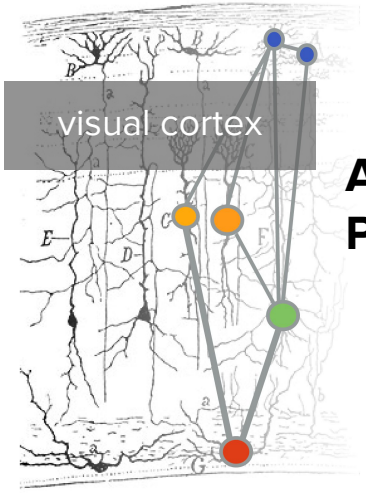
Dr. Juavinett
jah-vah-nett
(or, Dr. J)

Image: garetsworkshop/Shutterstock

Objectives for this morning

- Introduce the teaching staff, students, and class
- Motivate learning how to code as a biology student
- Discuss course logistics, expectations, & tools
- Start coding!

PhD in Neuroscience @ UCSD



multisensory processing & ethological behaviors

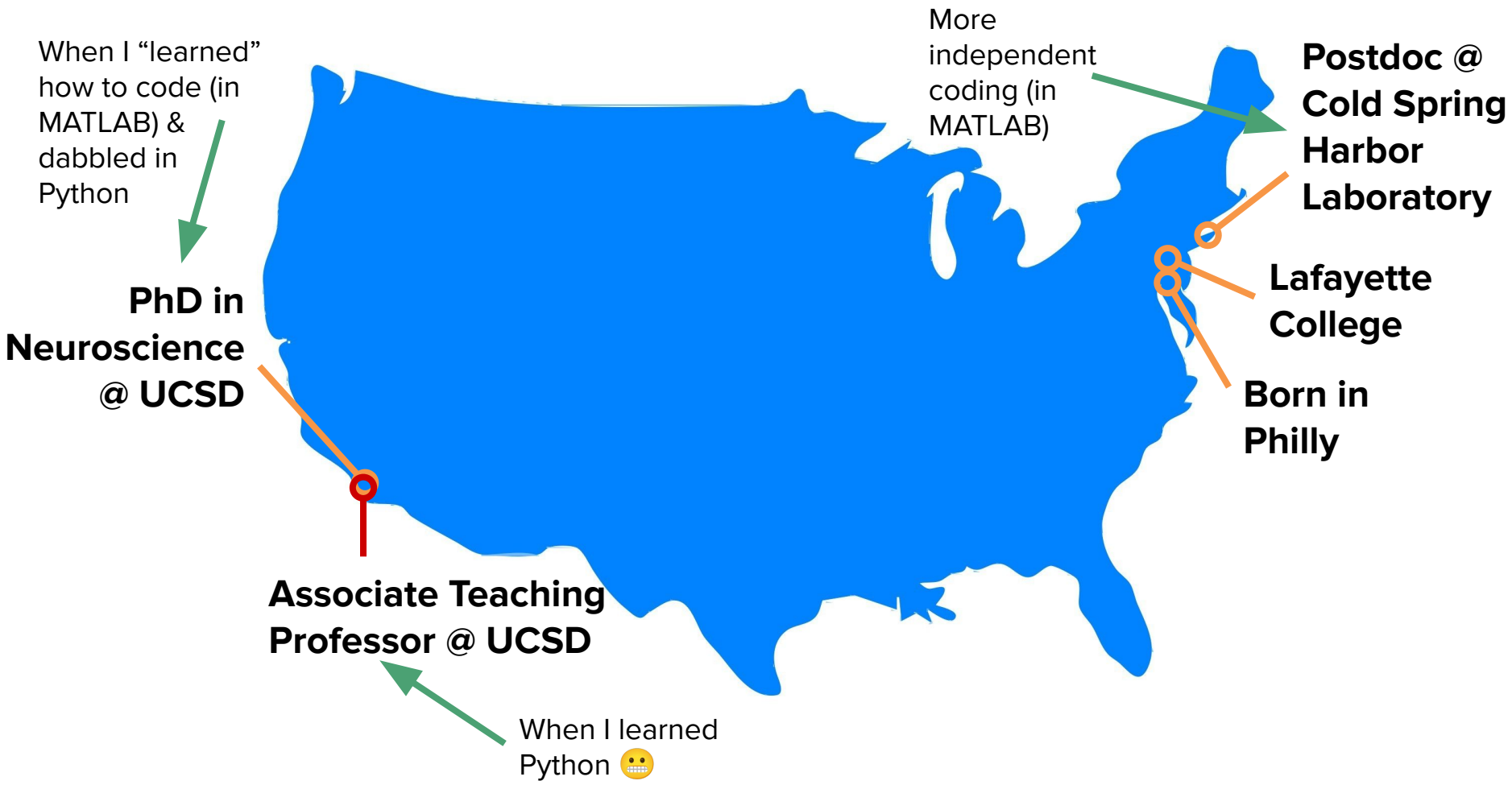
Postdoc @ Cold Spring Harbor Laboratory

Lafayette College

Born in Philly

Associate Teaching Professor @ UCSD

- Neuroscience careers & education
- Open-source data


When I “learned” how to code (in MATLAB) & dabbled in Python
PhD in Neuroscience @ UCSD
Associate Teaching Professor @ UCSD
When I learned Python 😬
More independent coding (in MATLAB)
Postdoc @ Cold Spring Harbor Laboratory
Lafayette College
Born in Philly

Introduction to our Instructional Assistants!

Let’s be human, for just a second.

With the folks next to you, share:

- Your name, major, and preferred pronouns
- Something that brought you joy over the break
- Why you’re taking this course

Objectives for this morning

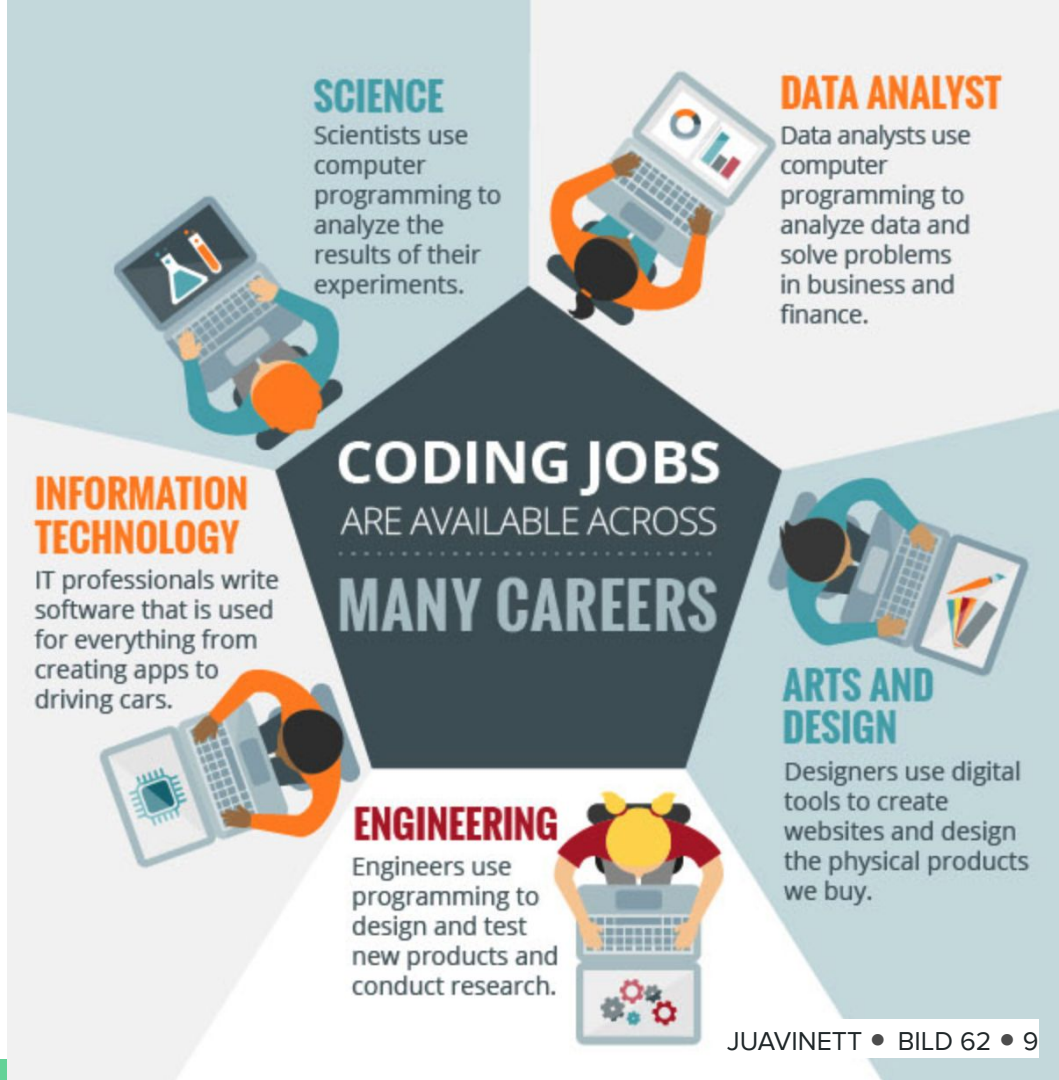
- Introduce the teaching staff, students, and class
- **Motivate learning how to code as a biology student**
- Discuss course logistics, expectations, & tools

What does coding have to do with *biology*?
Why *you, right now*?

Why should I learn how to code?

- Coding is useful for:
  - Data acquisition (controlling hardware, image acquisition, etc)
  - Data analysis & visualization
  - Computational modeling
- Beyond research, there are more and more jobs for software engineers, and they pay well

(see report by Burning Glass: https://www.burning-glass.com/research-project/coding-skills/)

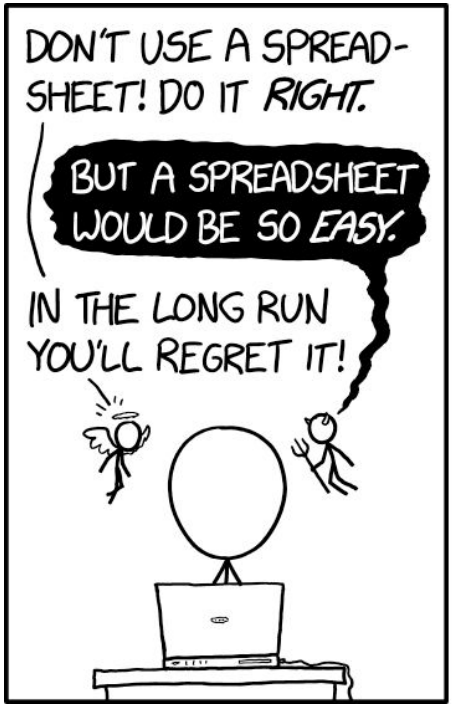


Excel can only handle datasets with **~1 million rows, and ~16,000 columns** — many datasets in biology are much larger than this!

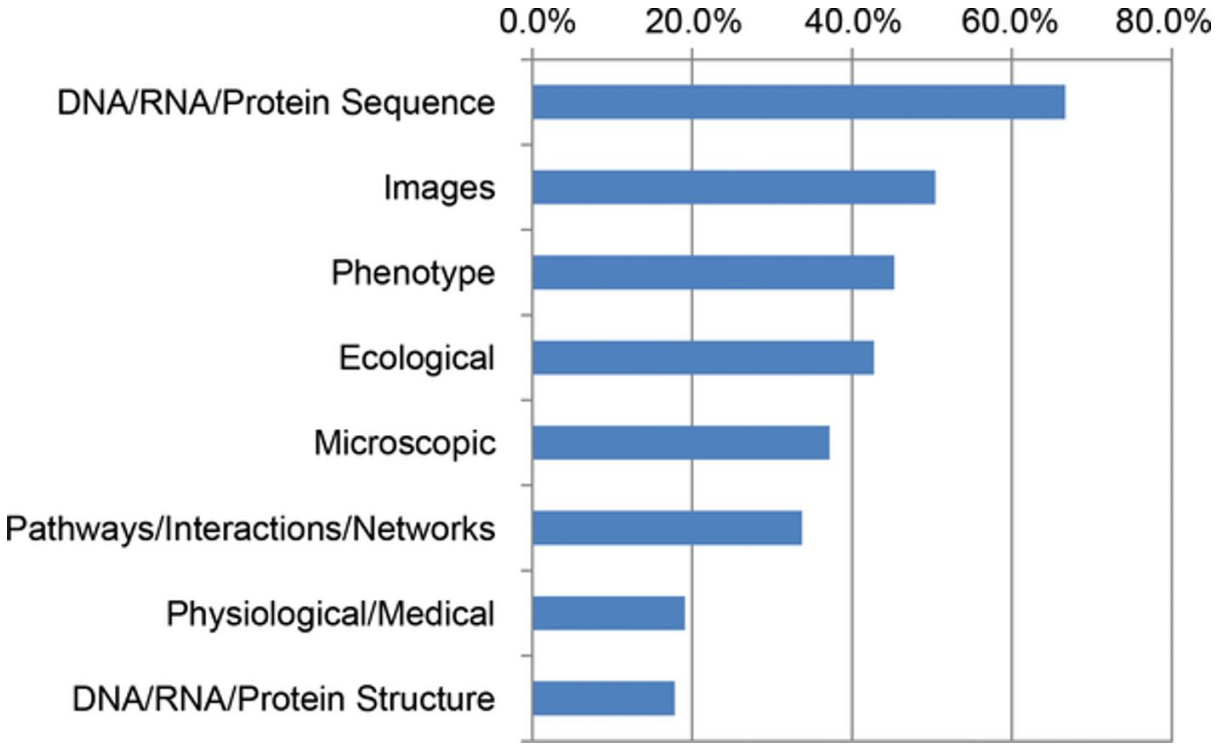
You can automate analyses in Excel, but this is quite limited.

There are also specialized biological data analysis software programs, but often these are limited in how much they can be customized.

Code is *infinitely* customizable.

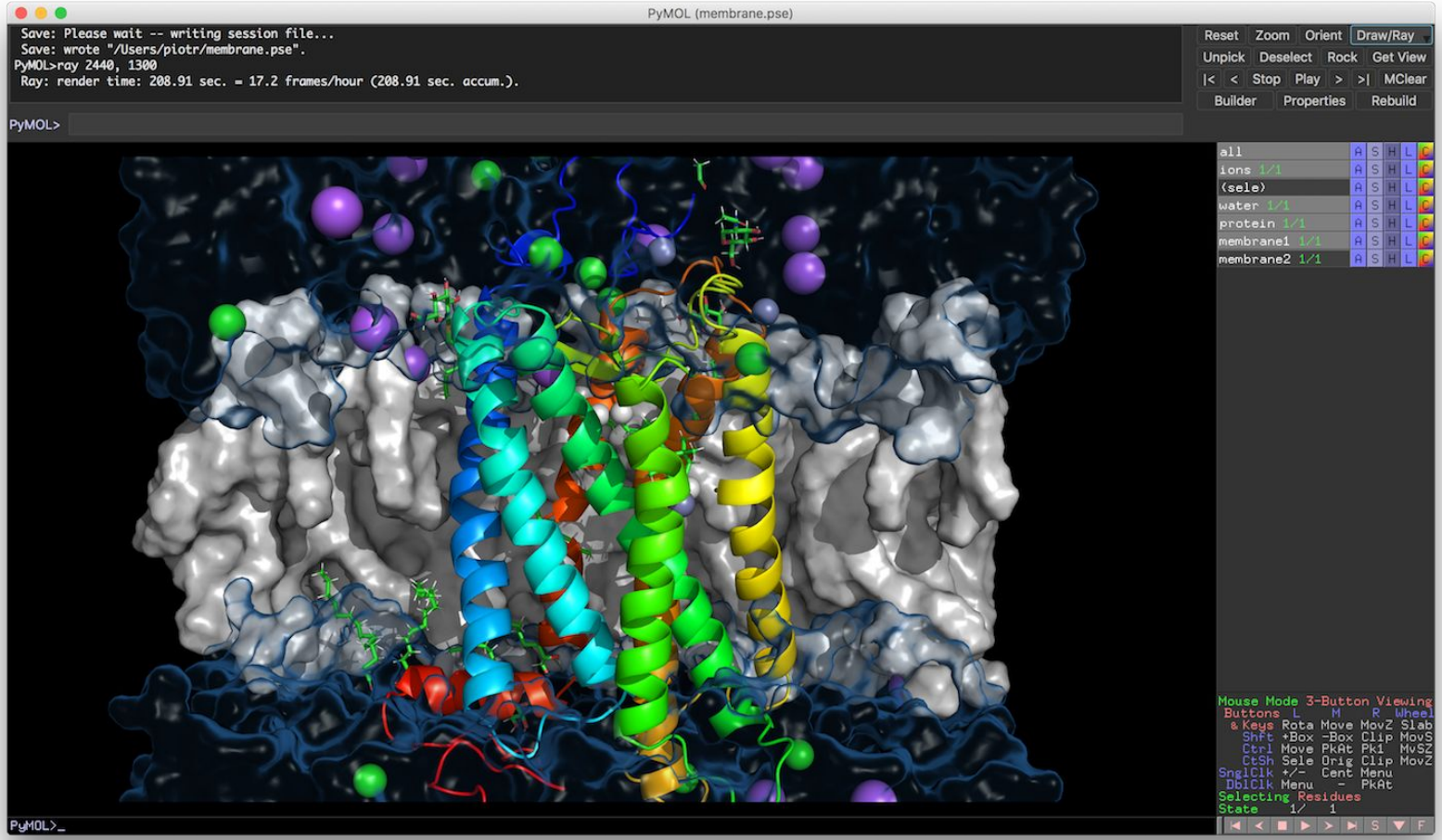


https://xkcd.com/2180/



Major data types used by National Science Foundation (NSF) Biological Sciences Directorate (BIO) principal investigators (PIs).

Barone L, Williams J, Micklos D (2017) Unmet needs for analyzing biological big data: A survey of 704 NSF principal investigators. PLOS Computational Biology https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005755


PyMOL (membrane.pse)
Save: Please wait -- writing session file...
Save: wrote "/Users/piotr/membrane.pse".
PyMOL>ray 2440, 1300
Ray: render time: 208.91 sec. = 17.2 frames/hour (208.91 sec. accum.).
PyMOL>
Reset
Zoom
Orient
Draw/Ray
Unpick
Deselect
Rock
Get View
|<
<
Stop
Play
>
>|
MClear
Builder
Properties
Rebuild
all
ions 1/1
(sele)
water 1/1
protein 1/1
membrane1 1/1
membrane2 1/1
Mouse Mode 3-Button Viewing
Buttons L M R Wheel
Keys Rota Move MovZ Slab
Shft +Box -Box Clip MovS
Ctrl Move PkAt Pkl MvSZ
CtSh Sele Orig Clip MovZ
SnglClk +/- Cent Menu
DblClk Menu - PkAt
Selecting Residues
State 1/ 1
PyMOL>_

pymol.org

pymol.org/2/

PyMOL by Schrödinger

DOWNLOAD SCREENSHOTS PRODUCTS SUPPORT CONTACT

Open-Source Philosophy

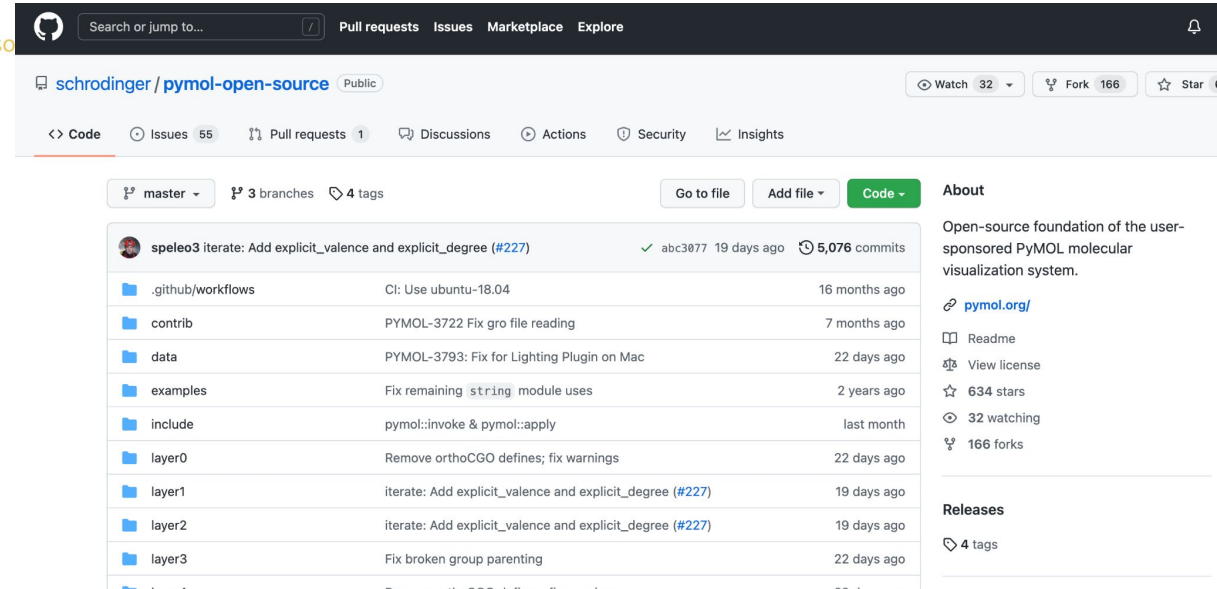
PyMOL is a commercial product, but we make most of its source code freely available under a permissive license. The open source project is maintained by Schrödinger and ultimately funded by everyone who purchases a PyMOL license.

Open source enables open science.
This was the vision of the original PyMOL author Warren L. DeLano.

Visit the Open-Source Project

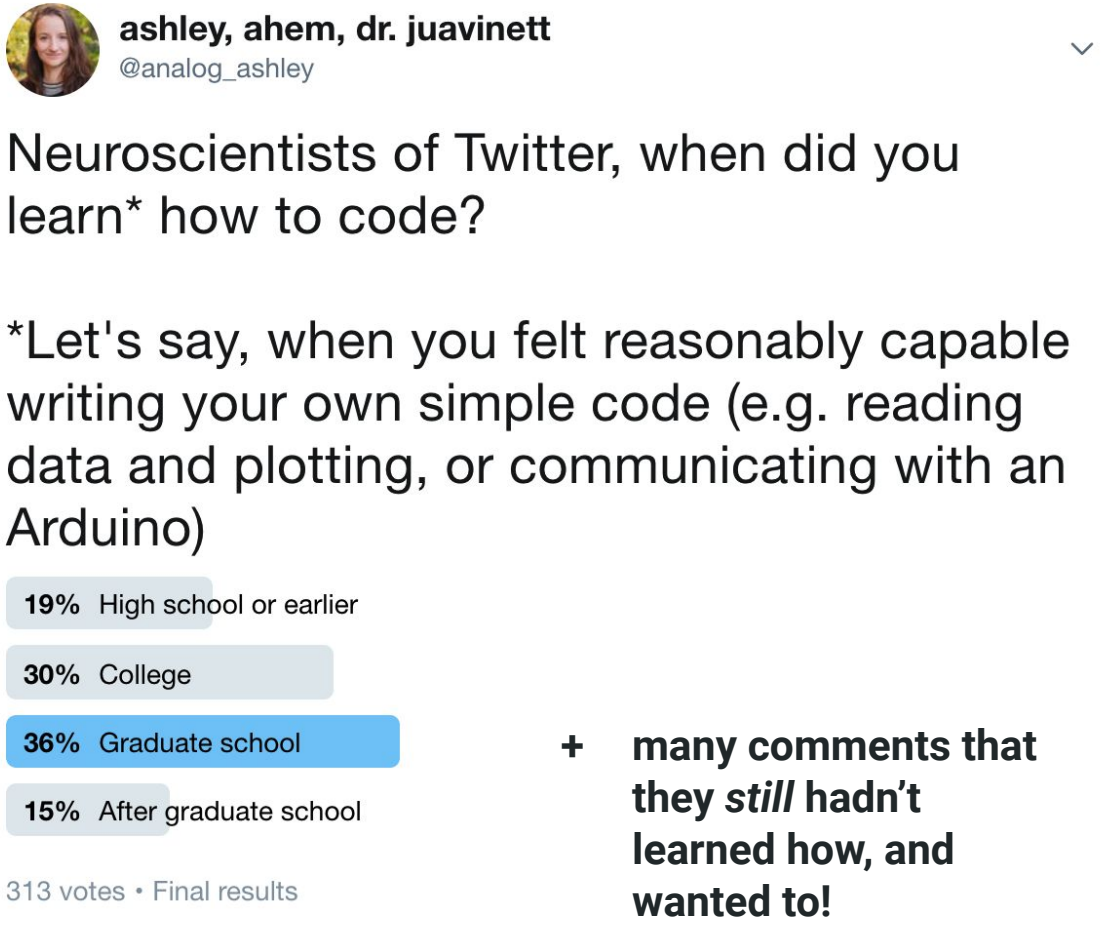
Become a sponso

AND many software packages for biologists can be modified… if you know how to code!



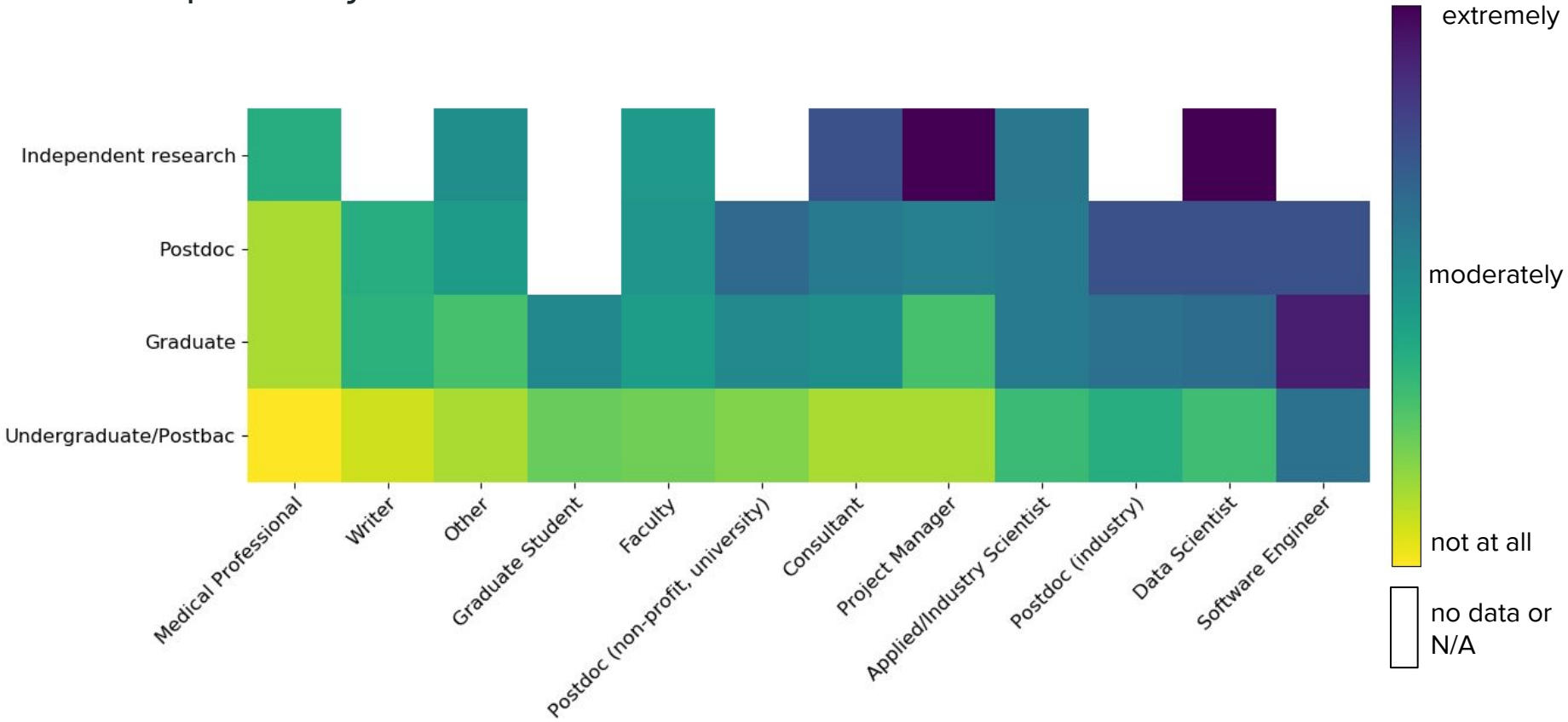
By taking this class, you're ahead of the game!

Many researchers learn to code really informally, and relatively late in their careers



\+ **many comments that they *still* hadn't learned how, and wanted to!**

How comfortable did/do you feel working with code at this point in your career?



Objectives for this morning

- Introduce the teaching staff, students, and class
- Motivate learning how to code as a biology student
- **Discuss course logistics, expectations, & tools**
- Start coding!

First step: let's drop our ideas of what it means to be a ***coder***.

Programming, like learning a language, *takes time*.

MASSIVE SCIENCE | Menu | Search | Account

Your language brain matters more for learning programming than your math brain

Christina Morillo on Wikimedia Commons.

New research contradicts long held assumptions about coding

Amy Nippert
Neuroscience
University of Minnesota

May 12, 2020 | 2 peer comments

https://massivesci.com/articles/programming-math-language-python-women-in-science/, summarizes this article: https://www.nature.com/articles/s41598-020-60661-8

Previous studies have shown that math and logic problems seem to rely mainly on the multiple demand regions in the left hemisphere, while tasks that involve spatial navigation activate the right hemisphere more than the left. The MIT team found that reading computer code appears to activate both the left and right sides of the multiple demand network, and ScratchJr activated the right side slightly more than the left. This finding goes against the hypothesis that math and coding rely on the same brain mechanisms.

https://news.mit.edu/2020/brain-reading-computer-code-1215
about this study: https://elifesciences.org/articles/58906

What will help you succeed in this course?

Things that predict success:

- How successful you *think* you'll be
- Completing assignments on time
- Asking questions when you have them
- Attending discussion sections & office hours

Things that ***do not*** predict success:

- Gender
- Age
- Personality
- Math ability

29A @StuxnetStudios · 14h

New programming student:

"I'm not very good at this. When I type out the code, I have to fix lots of errors. And I have to look up how to do most of it."

Instructor:

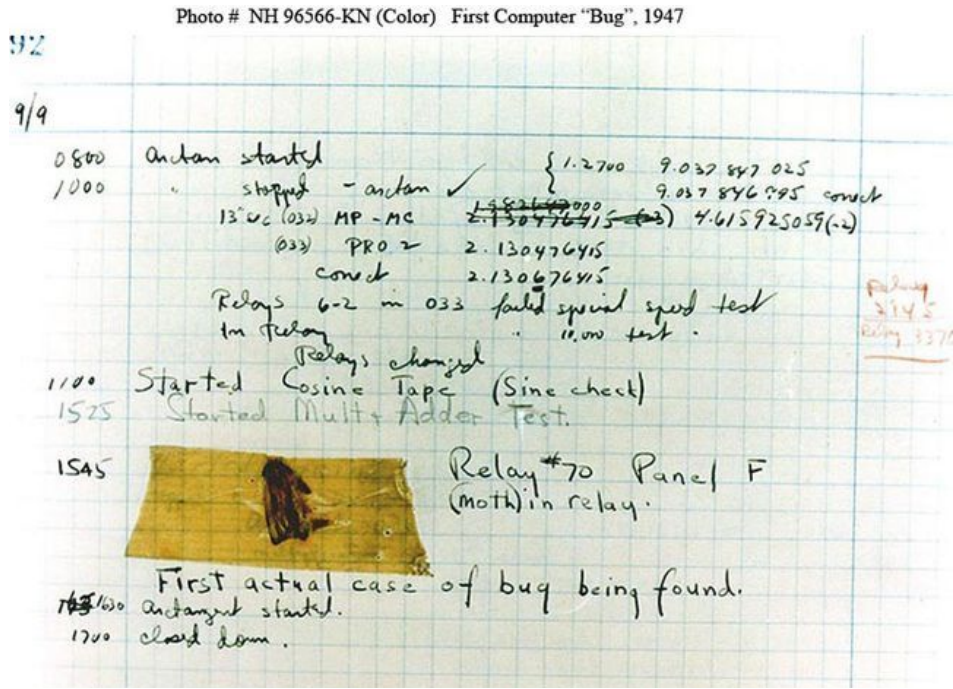
"You're doing it right."

29 275 1.4K

Historical sidenote: why is it called a **bug**?

In 1947, computer scientist & legend **Grace Hopper** found a *literal bug* in their computer, causing it to produce many errors.

Interview with Grace Hopper: https://www.youtube.com/watch?v=QA33wW5LaNY



https://www.nationalgeographic.org/thisday/sep9/worlds-first-computer-bug/

What is programming, anyway?

- Programming is the way humans communicate with **computers**
  - It’s a language!

Wait, what's a computer?

Hardware: the physical parts of the computer (CPU, hard drive, etc.)

Memory:

Primary: fast, temporary storage

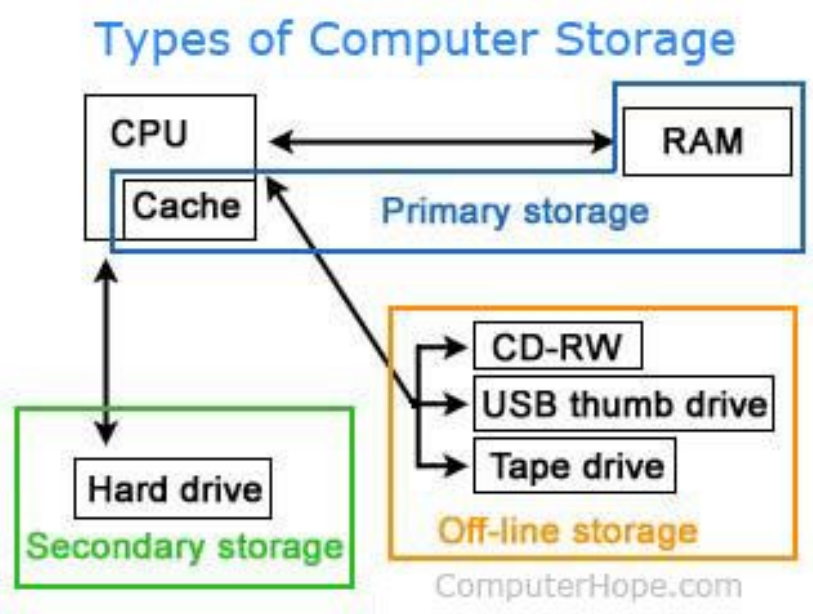
Secondary: slower, long-term storage

Tertiary: cloud storage

All computer memory is stored in **binary**.

More information about memory

Great explainer about binary memory



Wait, what's a computer?

Interface: software, the operating system, what you see

File structure

cats = files
Each has their own name.

boxes = folders/directories
Where you store the cats.
You can put a box into another box, but you can't put a box into a cat.

What is programming, anyway?

- Programming is the way humans communicate with computers
  - It's a language!
- The instructions we give the computer are taken **literally** and **sequentially**

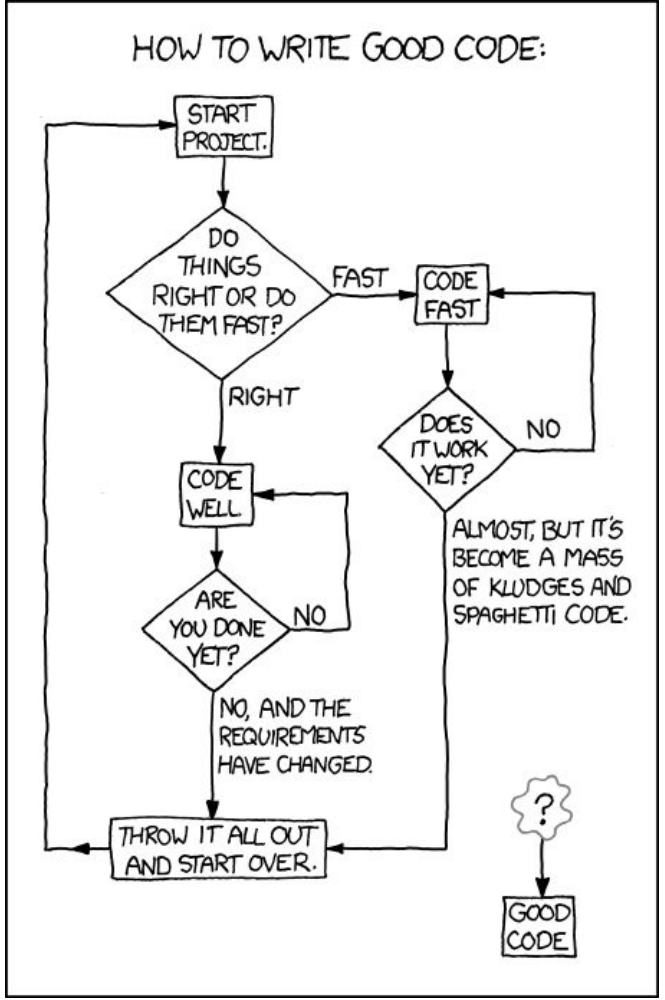
Capitalization matters:
`print() ≠ Print()`

```
b = a * 2
a = 2
```

computer: what is a?

The path to writing good, efficient code

1. Make it **work**
2. Make it **right**
3. Make it **fast**



XKCD, https://xkcd.com/844/

The path to writing good, efficient code

1. Make it **work**
2. Make it **right**
3. Make it **fast**

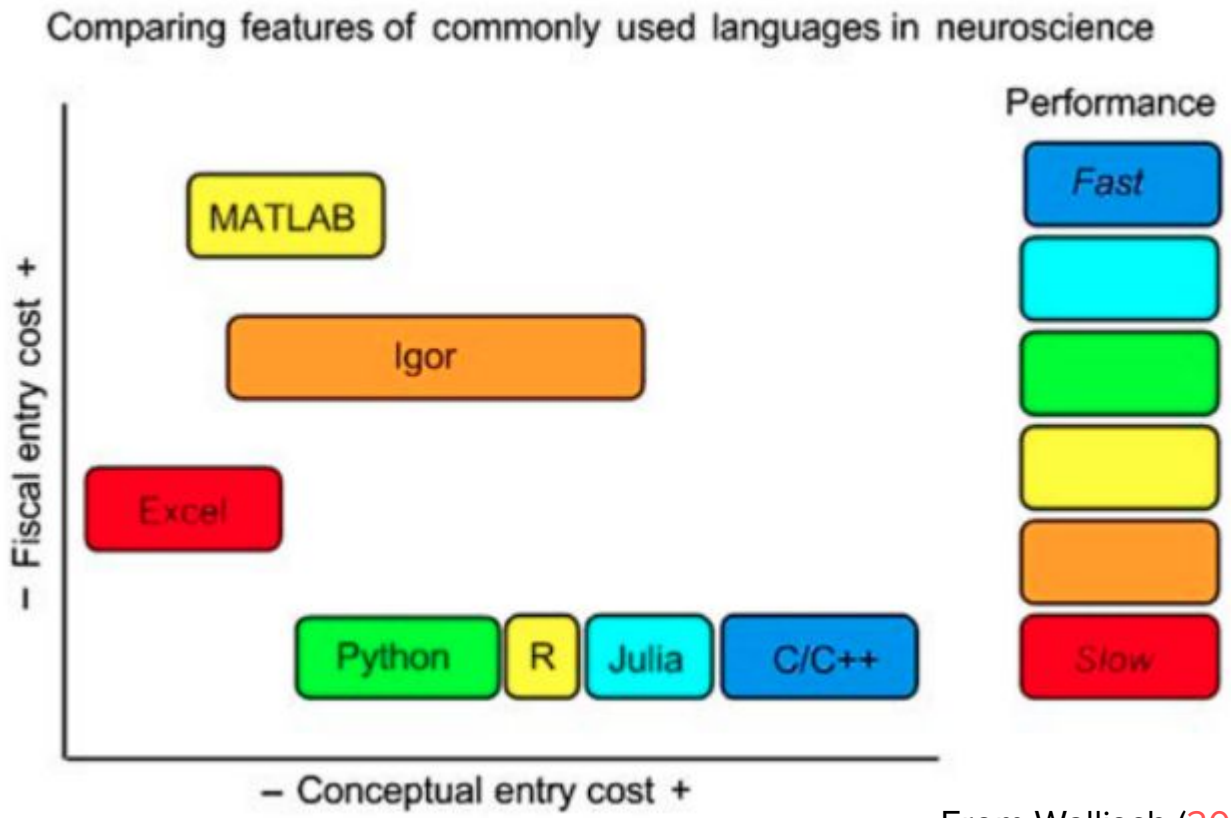
Our goal is to get to this step

If you ultimately became a back-end programmer, you'd care about step 3.

For most problems biologists face, step 3 isn't paramount.

Considerations for choosing a programming language

- Fiscal & conceptual entry
- Usage in particular field or profession



From Wallisch (2017)

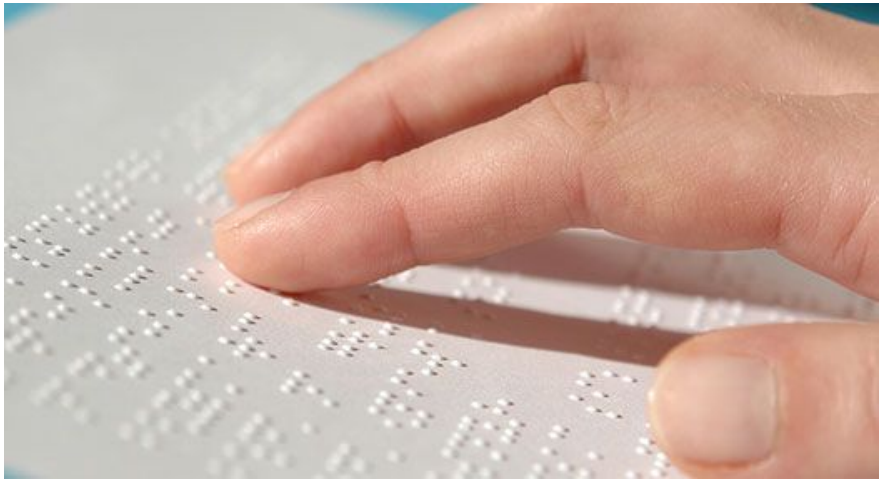
All coding languages eventually need to talk to the computer in binary:

`01001000 01100101 01101100 01101100 01101111 00100001`

(hello)

Learn How To Write Your Name In Binary Code

There are many types of binary code, beyond computers



Braille
https://www.afb.org/blindness-and-low-vision/braille/what-braille

A ●−	J ●−−−	S ●●●
B −●●●	K −●−	T −
C −●−●	L ●−●●	U ●●−
D −●●	M −−	V ●●●−
E ●	N −●	W ●−−
F ●●−●	O −−−	X −●●−
G −−●	P ●−−●	Y −●−−
H ●●●●	Q −−●−	Z −−●●
I ●●	R ●−●	

Morse code
https://www.discoveryworld.org/about/blog/discover_at_home/morse-code/

In this class, we'll use Python

- Programming language, development led by Python Software Foundation (www.python.org)
- Uses concise structure & wording similar to human language
- An **interpreted** language — it doesn't speak *directly* to the computer
- Can be used for many purposes, from web programming, to creating games, to analyzing & visualizing data
  - Extension: '.py'
- We'll also work in **Jupyter Notebooks**
  - Extension '.ipynb'

python™

Course logistics

“What is one thing you're hoping to gain from this course?”

Course Objectives

- Read and run basic Python programs, recognizing the structures used (i.e. variables, conditionals, loops, functions) and explaining how they work
- Manipulate and create objects in Python, including data structures and classes
- Write, edit, and execute Python code in Jupyter Notebooks as well as the command line
- Visualize and run hypothesis-testing on simple datasets in Python
- Implement common algorithms for analyzing biological data (e.g., time series, images) and determine when such computations are appropriate

Grading breakdown

In-class work & participation (10%)

Assignments (50%)

Lowest assignment grade dropped

Midterm (15%)

Final project (25%)

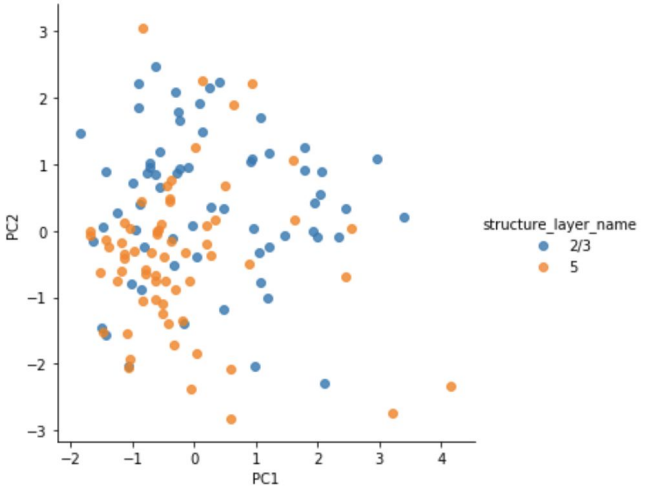
Assignments & project components **lose 10% each day they are late.**

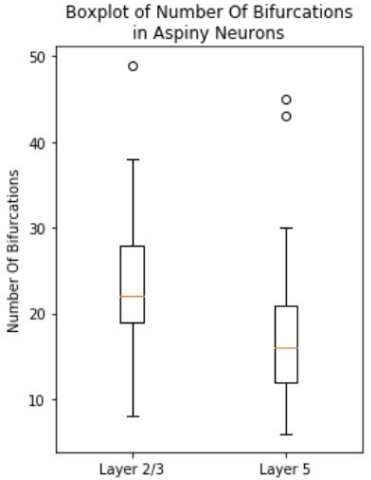
Assignments

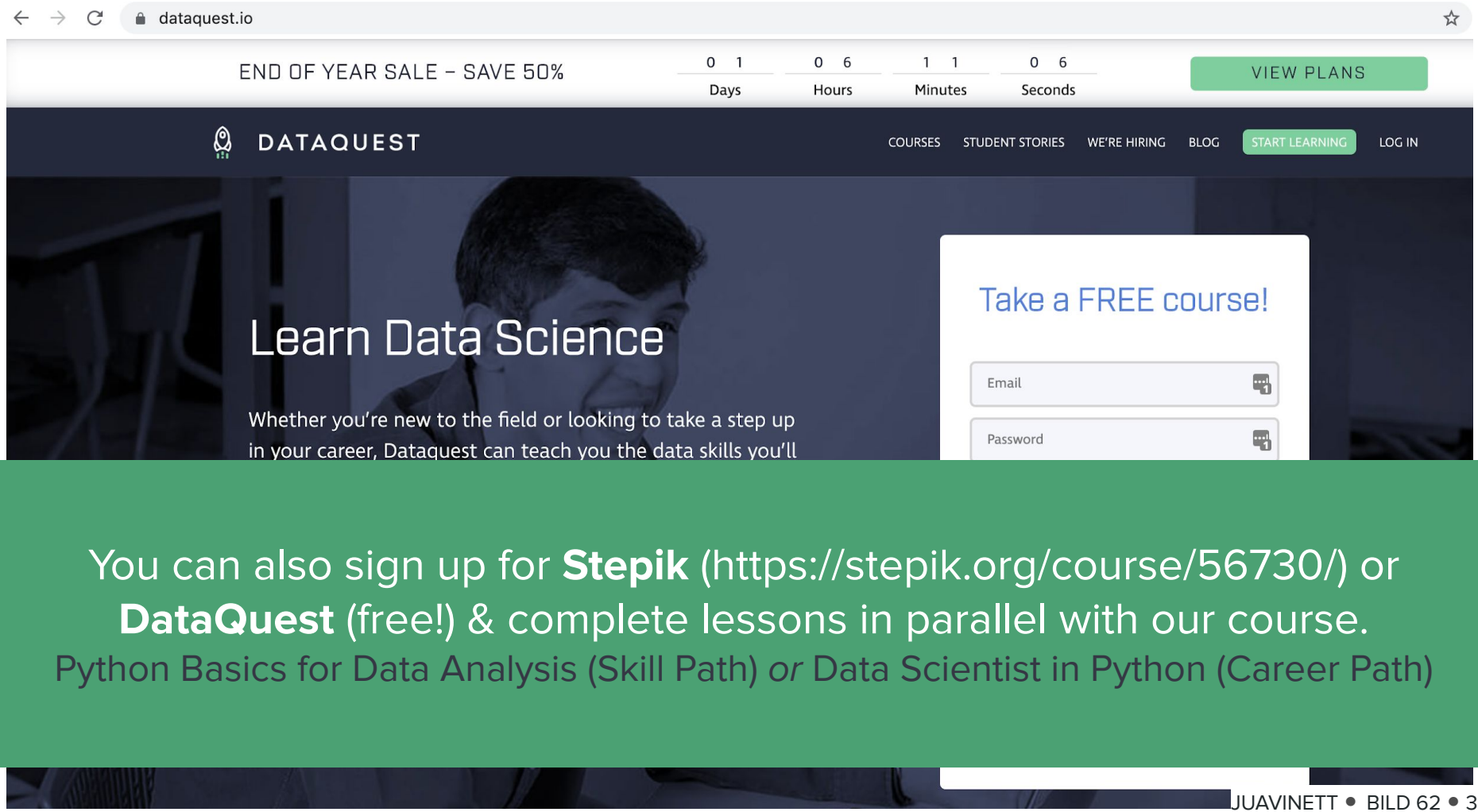
- Due **every Tuesday at 5 pm**
- Worth 2.5-10% each
- Completed individually
- Programmatically graded (via Datahub/NBGrader)
- Lowest assignment grade dropped.
- Next class, we'll walk through how to submit these.

Project, groups of 2-3

- Includes the project proposal, code, and presentation.
- Your final project will either:
  - Write a program to complete a task
  - Analyze & visualize data of your choosing
- We will discuss possibilities for your project as we move through the course.







You can also sign up for **Stepik** (https://stepik.org/course/56730/) or **DataQuest** (free!) & complete lessons in parallel with our course.

Python Basics for Data Analysis (Skill Path) *or* Data Scientist in Python (Career Path)

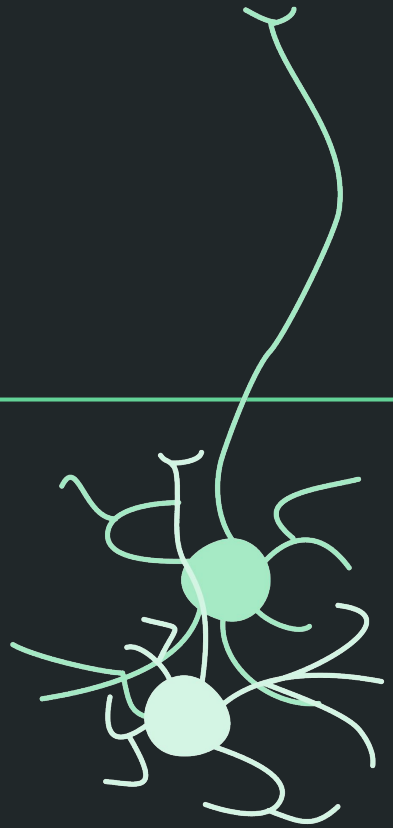
Office hours

On Zoom, where it's easier to troubleshoot!

Why should you come to office hours?

- You have clarifying questions about the course or its content
- You have concerns about the course and your progress
- You'd like to talk about career paths in biology or neuroscience

Tools for this class



Course Tools

canvas

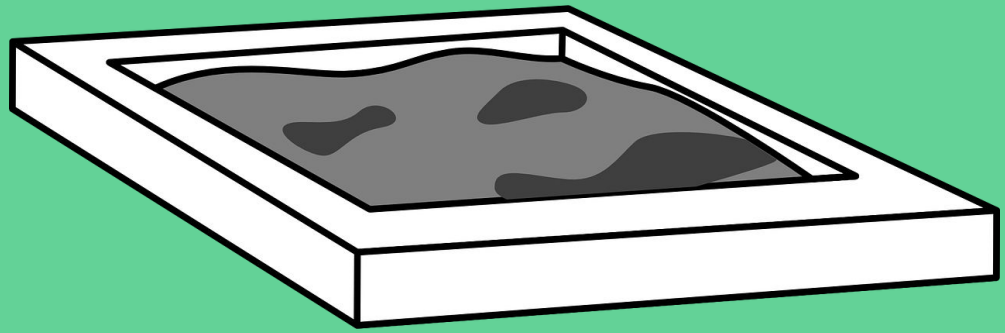
Submitting non-coding assignments & managing grades

Sharing private course content

Coding exercises & assignments

Sharing public course materials
https://github.com/BILD62

Interacting with course materials



You can find all of our course materials on either Canvas or the course GitHub: https://github.com/BILD62/BILD62_WI23

Lectures

In other words, PDF slides shown during class.

Hosted on GitHub in the Lectures folder

Materials

Jupyter Notebooks

You can pull these locally or to DataHub, or look at them online via GitHub or Colab/ Binder

launch binder

Assignments

Jupyter Notebooks, submitted through **Assignments** tab

Answers posted in the Assignments folder on Github

If I use both a PDF and a Jupyter Notebook during lecture, these numbers will match

💫 THE MAGIC LINK FOR THIS COURSE:

https://datahub.ucsd.edu/hub/user-redirect/git-sync?repo=https://github.com/BILD62/BILD62_WI23

💫 THE MAGIC LINK FOR THIS COURSE:

Sync with your datahub:

https://datahub.ucsd.edu/hub/user-redirect/git-sync?repo=https://github.com/BILD62/BILD62_WI23

Where our course content lives

To clone Materials to DataHub:

1. Click on the magic link.
2. Log in to DataHub as prompted.
3. You'll be in our course folder now!
4. If you want, save your own copy by adding your initials to the end of the file name. **DO NOT DO THIS FOR ASSIGNMENTS!**

5. Next time you click the link, you'll have a fresh copy, plus your copy.

To interact with Jupyter Notebooks on your computer

OPTIONAL

1. Install Anaconda with Python 3.7 for your operating system.
2. If you're using Windows, download git.
3. In Terminal (Mac) or the Anaconda Prompt (Windows), clone the repository by running the following command:

   `git clone http://www.github.com/BILD62/BILD62_WI23.git`
4. Open Jupyter Notebook. There are two ways to open:
   - In Terminal (Mac) or the Anaconda Prompt (Windows), type **`jupyter notebook`**
   - Open Anaconda Navigator and launch jupyter notebook
5. On the Jupyter landing page, navigate to the notebook and open it.
   - It will open in a browser but is *not* using an internet connection.

This is a new(ish) class — thanks for your patience!

We'll also be trying to learn about how this is going with some ***education research***.

Exempt Information Sheet

You are being invited to participate in a research study titled Coding Attitudes of Biology Students Coding Attitudes of Biology Students. This study is being done by Ashley Juavinett, PhD from UC San Diego. You were selected to participate in this study because you are a student in a class that is being studied.

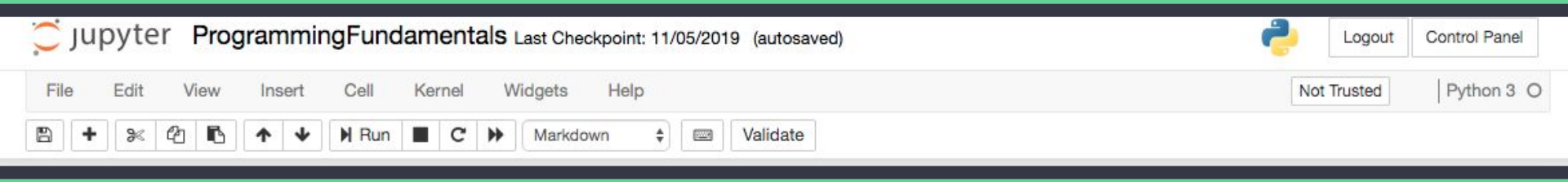
The purpose of this research study is to create knowledge that has the potential to improve the learning and educational experience of students at UC San Diego and beyond, and specifically to understand student attitudes towards coding

Your participation in this research should last approximately 15 minutes. If you agree to take part in this study, you will be asked to take pre- and post-course surveys containing about 20 multiple choice items and several open-ended questions.

Your participation in this study is completely voluntary and you can withdraw at any time. Choosing not to participate or withdrawing will result in no penalty or loss of benefits to which you are entitled, and will not adversely impact your standing with the instructional staff of your course. You are free to skip any question that you choose.

Complete both pre- and post-surveys on Qualtrics, gain 1% extra credit

Introduction to the UCSD DataHub & Jupyter Notebooks



About Jupyter Notebooks

- Jupyter is a loose acronym for Julia, Python, and R
- Run in a web browser but it's not *necessarily* online (it is when we use the DataHub)
- Usefully, it will show plots directly in the notebook as you work your way through, performing analyses in real-time
- Two main components:
  - **Kernel**: the engine that runs the code
  - **Dashboard**: landing page where you can see the notebooks you've created

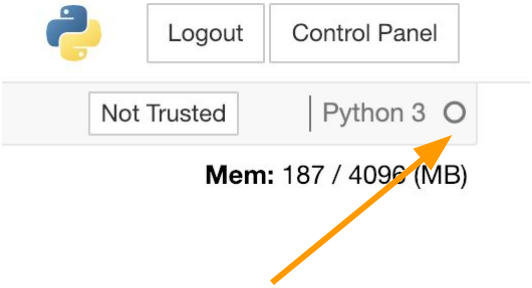
Using Jupyter Notebooks

- **Cell**: the main organizational structure of the notebook
  - Use **Shift+Enter** to run a cell (or press Run)
  - You can run cells out of order, and move cells around!
  - Cells can be **code** (the default) or **markdown** (descriptive text or images)
    - Code cells have `In[ ]:`
      - If there is a star (`In[]*:`), that means your cell is running
    - Change between code & markdown using dropdown menu (or keyboard shortcuts)
    - Turns **green** in edit mode

Using Jupyter Notebooks

- You can clear your **namespace** and get a fresh start by restarting the kernel

- **If you change anything in the cell, you need to re-run it.**
- For help:
  - Help > User Interface tour
  - Help > Keyboard Shortcuts



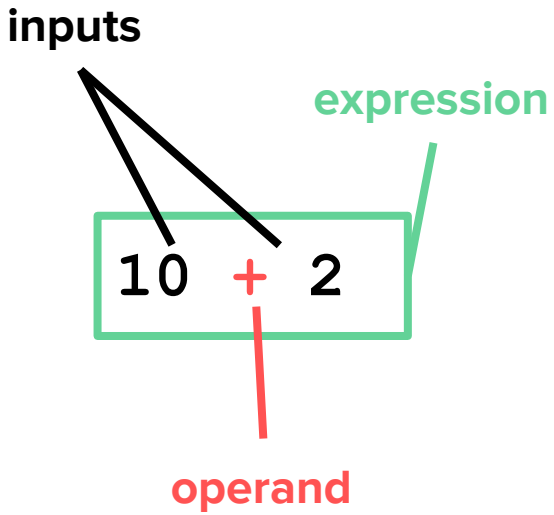
You can tell if the kernel is busy by whether or not the circle next to Python 3 (upper right corner) is filled or not. (filled = busy)

Expressions describe how to combine pieces of data (e.g., add them!)

SYMBOLS YOU WILL ENCOUNTER IN THIS COURSE		
Symbol	**Name**	**Sample Usage**
=	Equal sign	Assign variable
#	Pound sign; hashtag	Line comments
[ ]	Brackets	Indexing & Slicing
( )	Parentheses	Using functions
{ }	Curly Brackets	Defining dictionary
' '	Single quotes	Creating string
" "	Double quotes	Creating string
_	Underscore	In variable names
!	Explanation point	To test not equal (!=)
\	Back slash	Delineate line break
:	Colon	Indexing

Basic arithmetic operators in Python

Symbol	Operation	Usage
+	Addition	`10+2`
-	Subtraction	`10-2`
*	Multiplication	`10*2`
/	Division	`10/2`
**	Exponent	`10**2`
%	Modulo	`10%2`



If you want a whole number (floor division), use // instead.

Let’s get into a Jupyter notebook! Use the magic link to sync up your DataHub with our folder, and open notebook 01.

Before next class...

- Take the entry survey https://forms.gle/RPN26BKDbwqvyz9B9
- Take the computing attitudes survey https://ucsd.co1.qualtrics.com/jfe/form/SV_doRycCO2ZCnvBb0
- Access Canvas (canvas.ucsd.edu) & DataHub (datahub.ucsd.edu)
- (Optional) Sign up for Stepik and/or DataQuest
- (Optional) Install Python 3.7 (via the Anaconda distribution) on your computer (https://www.anaconda.com/distribution/)

You only *really* need access to the DataHub, but having the ability to use Python & Jupyter Notebooks on your local computer *may* be useful!