Introduction

SURP 2022 Python Bootcamp

Ohio State Astronomy

Slides by: James W. Johnson

Hello!

Fifth/Sixth year PhD student in the Astronomy Department
Research Interests

- Galactic chemical evolution (David Weinberg, Jennifer Johnson)
- [Type Ia] Supernovae (Chris Kochanek, Kris Stanek)
- Previously: dark matter halos and their environments

Versatile Integrator for Chemical Evolution (VICE: https://pypi.org/project/vice)

- ~89,200 lines (Python & C) in latest release (1.3.0) supporting Python >= 3.6.

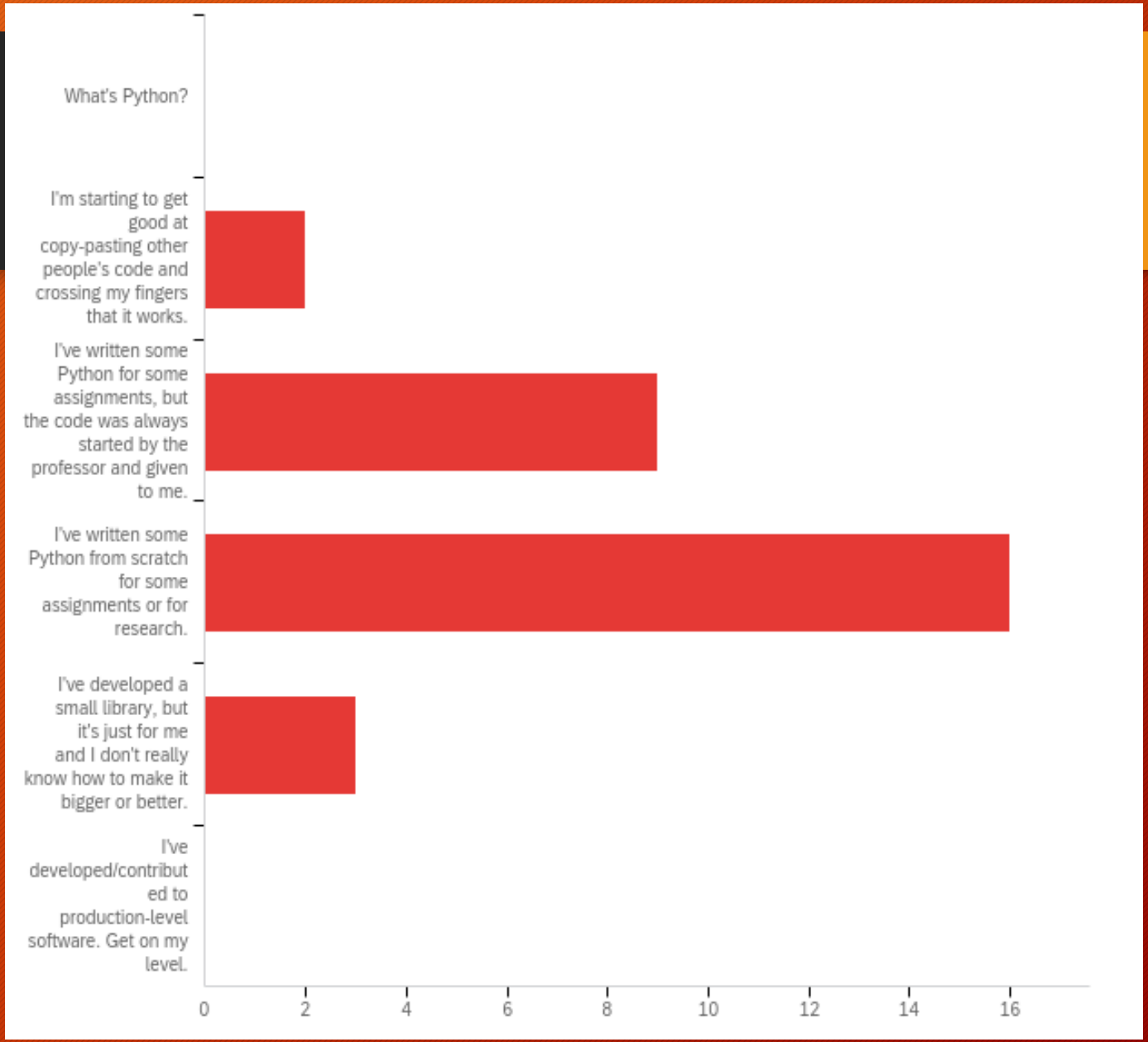
We'll do a mix of slides and exercises – I won't be collecting and grading anything. This isn't a course. What you get out of this will reflect what you put into it.

Survey Responses

Q1: How would you describe your skill level in Python?

A wide range of skills – some quite new, some have made small libraries

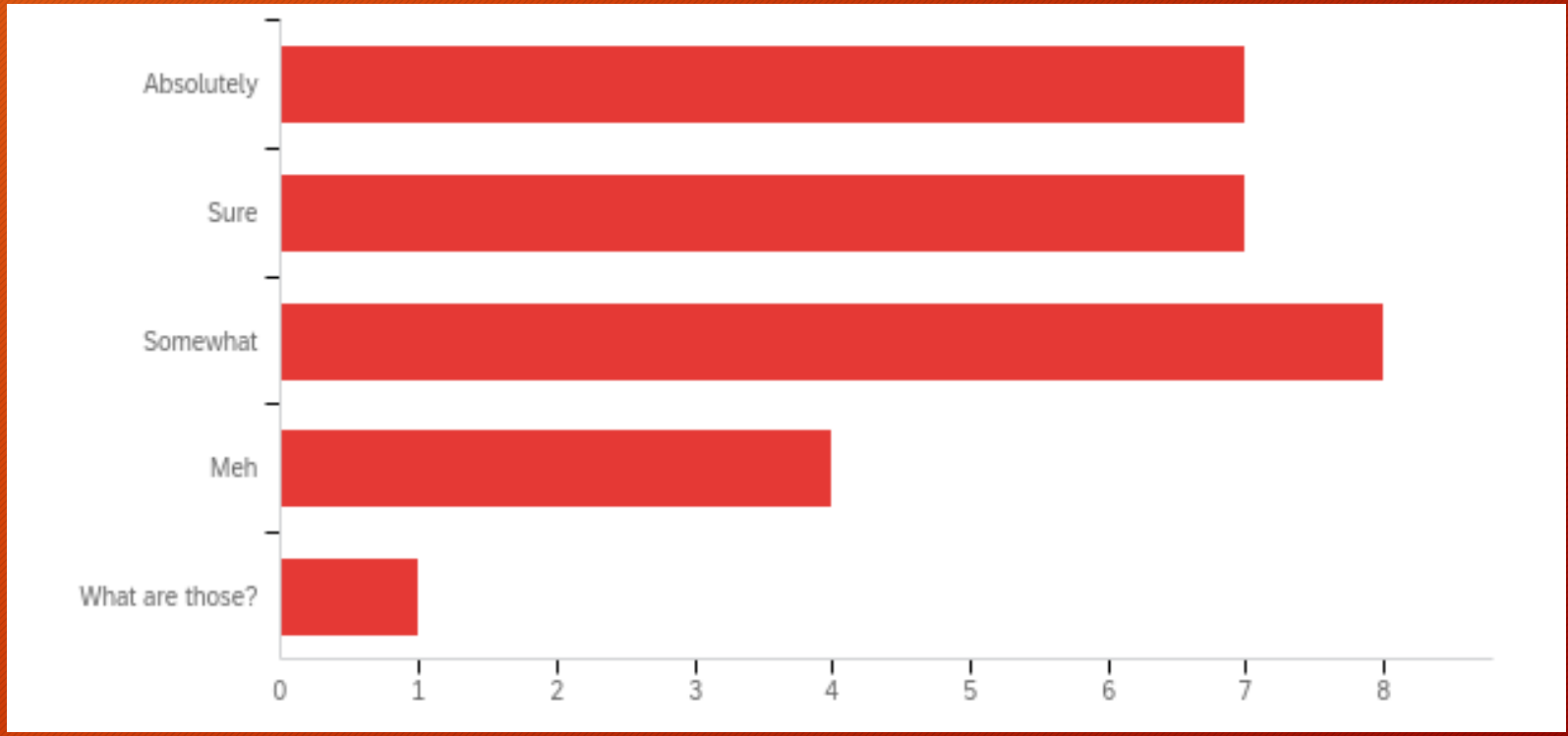
We're aiming to get you up to the second-to-last option



Survey Responses

Q2: Are you comfortable with if/else conditions, for- and while-loops, and def statements?

A wide range of skills in functional programming

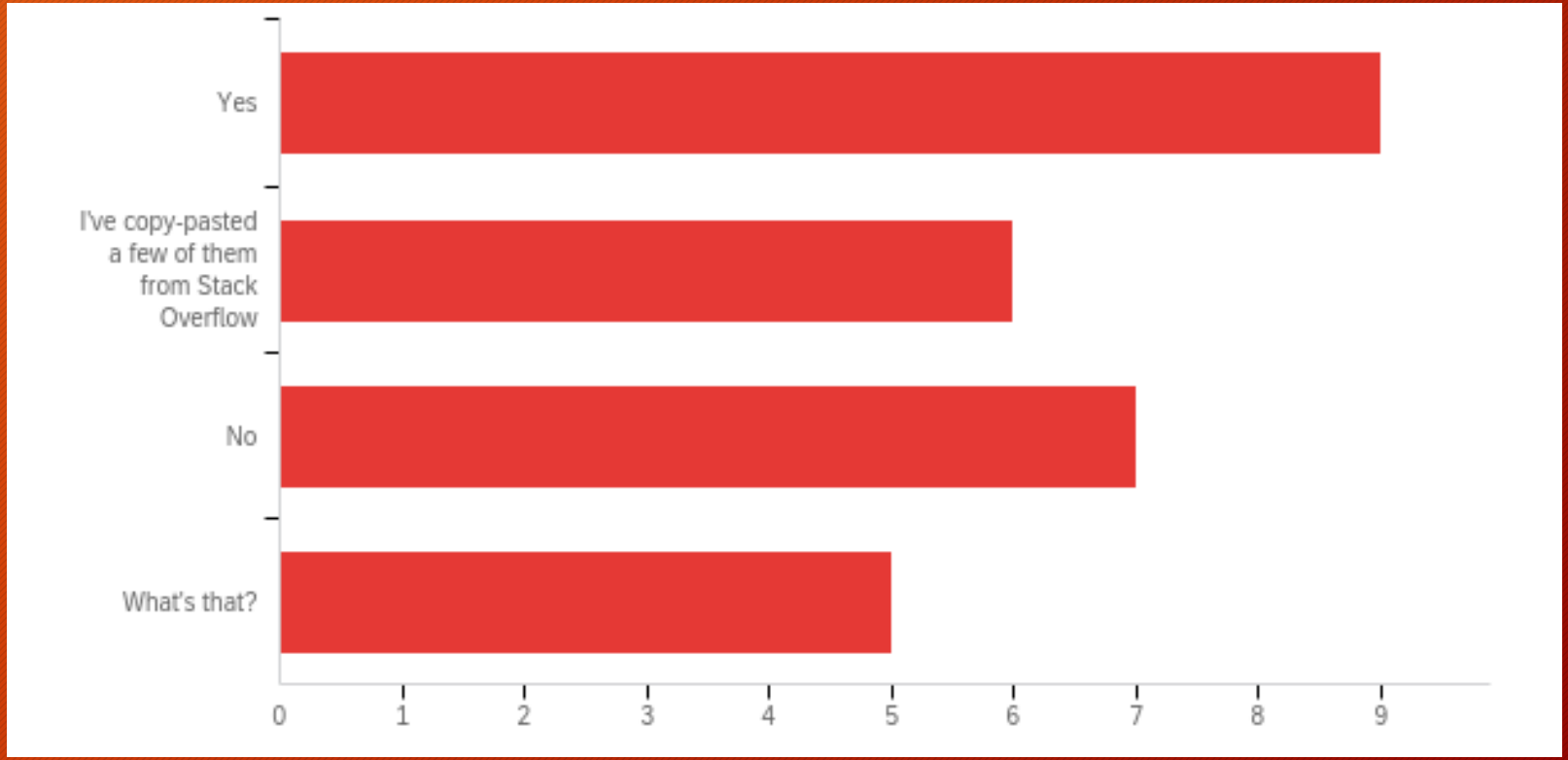


Survey Responses

Q3: Have you ever written a class?

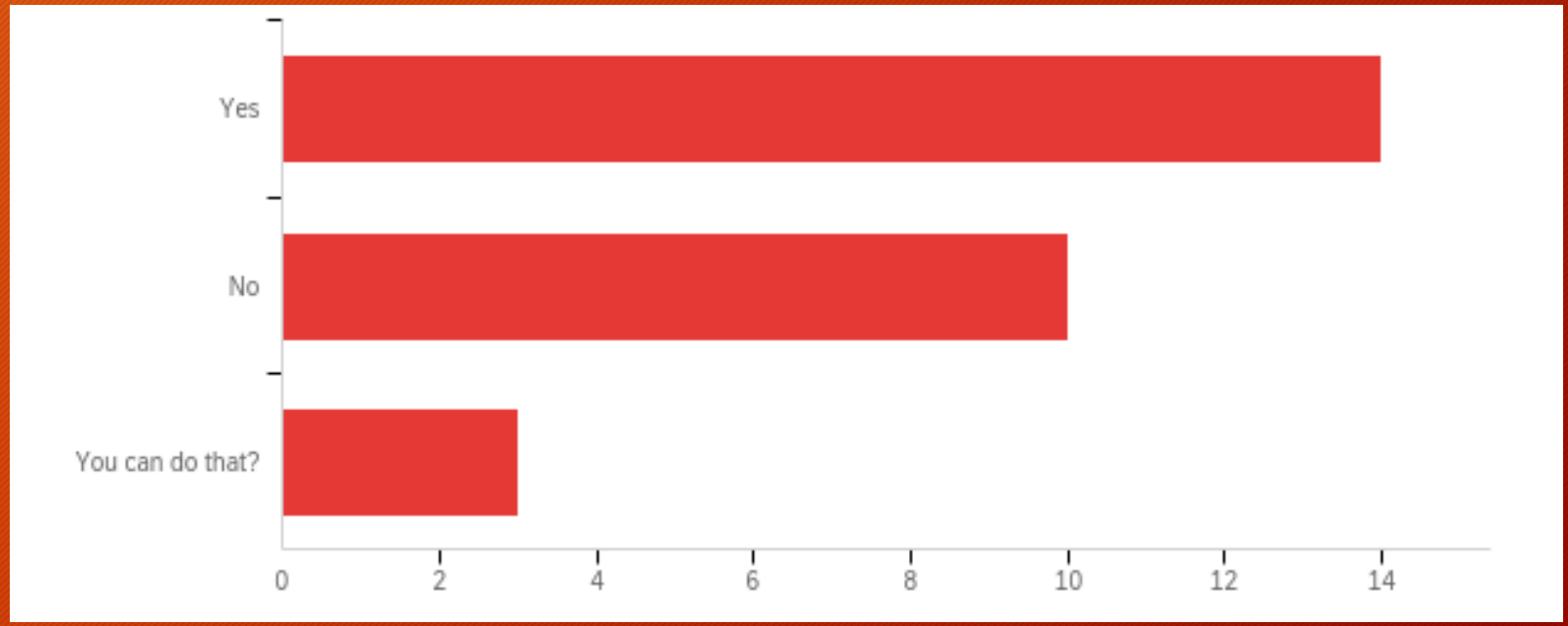
Many of you have, many of you haven't

This will be the focus of sessions 4 and 5



Survey Responses

Q4: Do you know how to, or have you already imported your own code from another file?

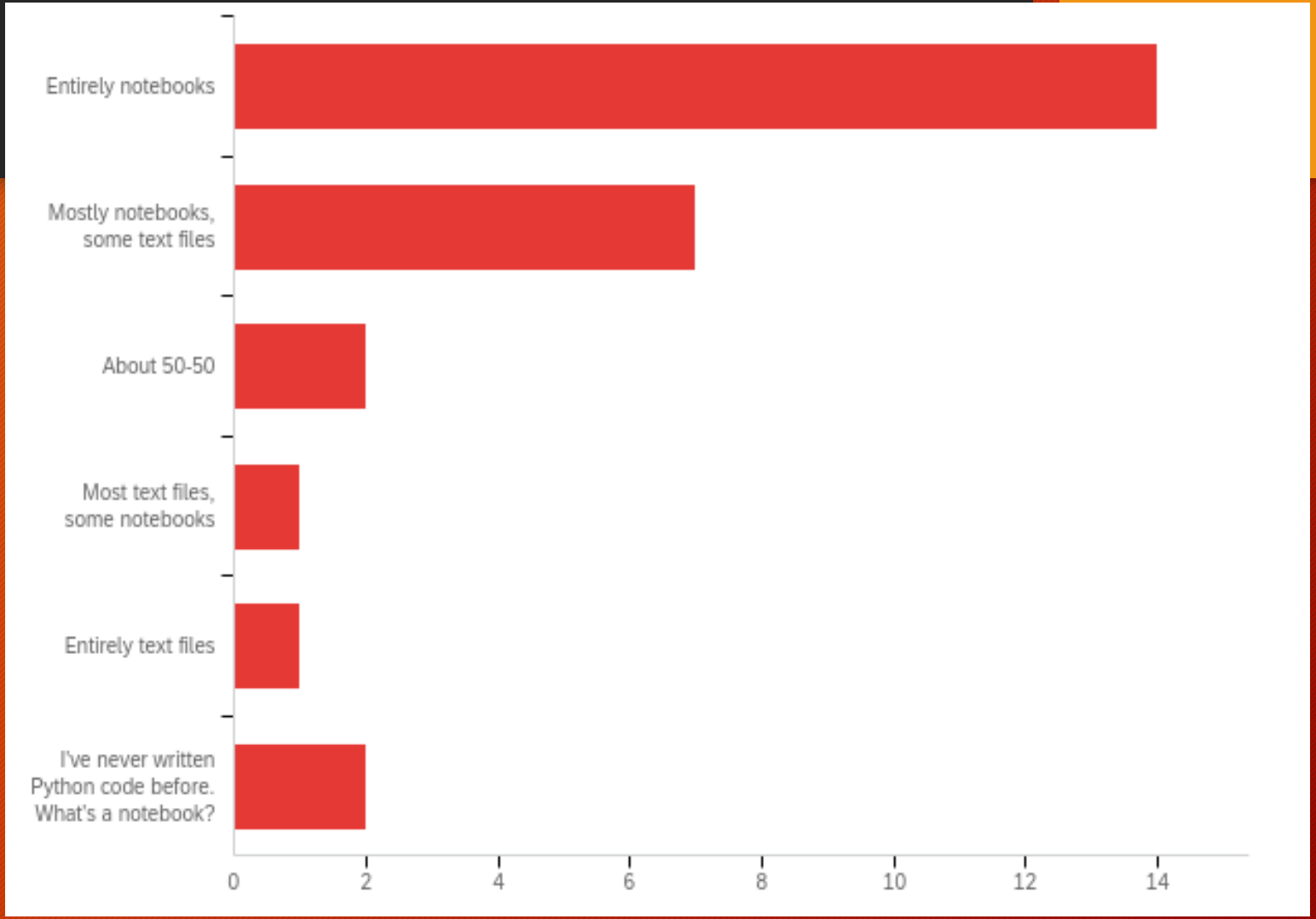


Many of you have, many of you haven’t

Survey Responses

Q5: Do you write code in text files or jupyter notebooks (or some equivalent)?

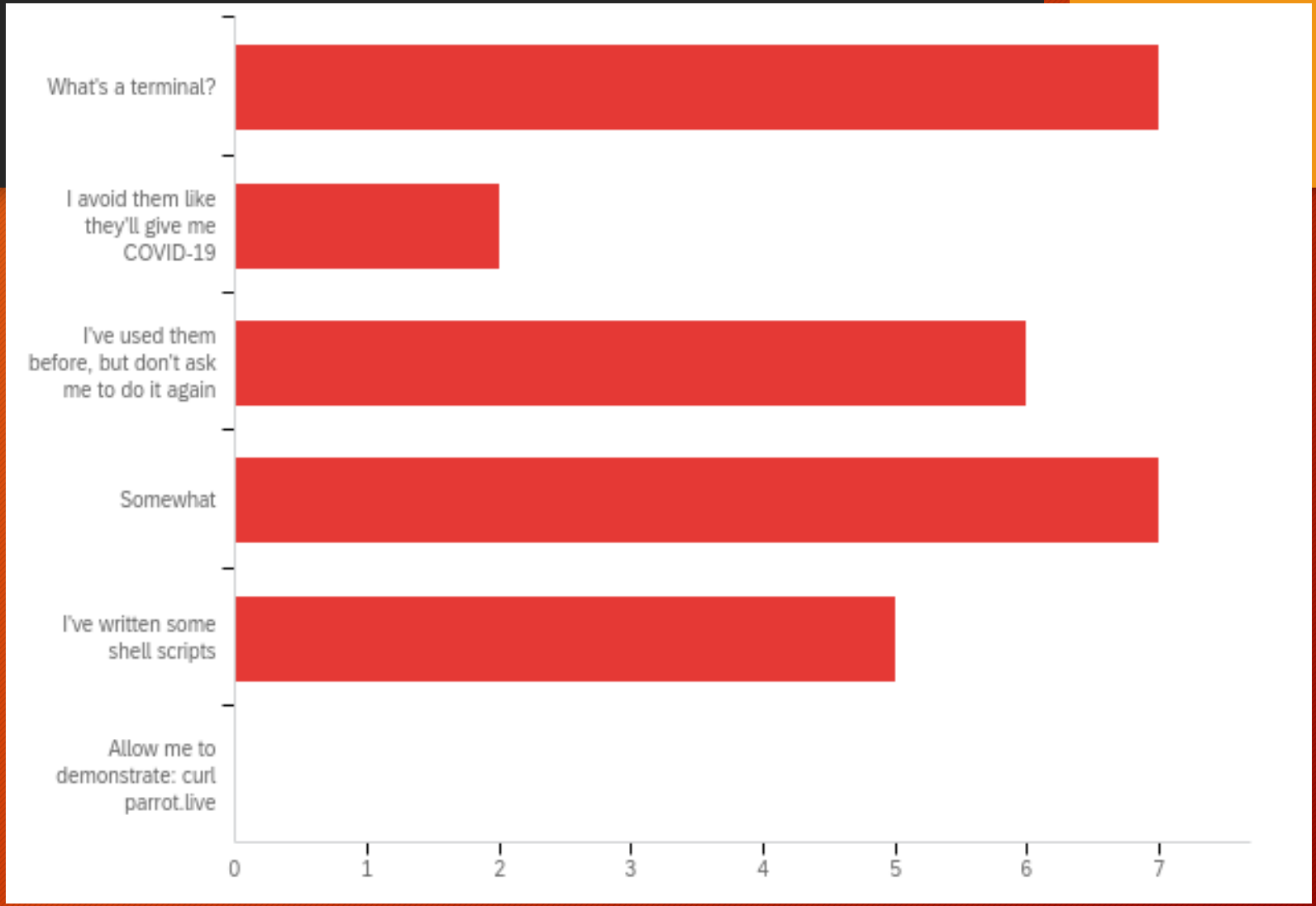
A wide range of preferences for your programming medium. Those who have received code from instructors probably ran it on SciServer, which uses notebooks



Survey Responses

Q6: How comfortable are you using a terminal/command-line?

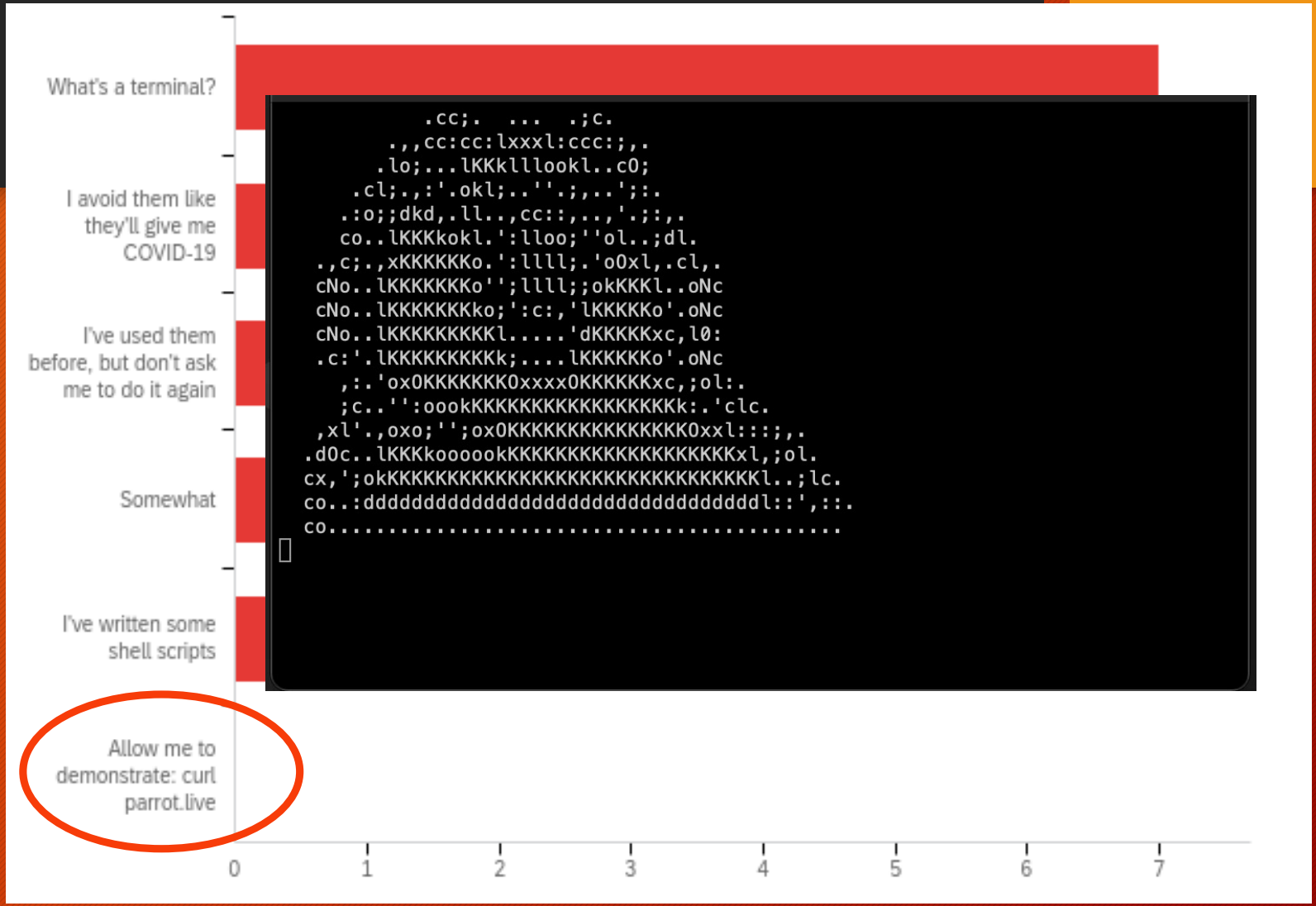
Some of you have never used them, others are rather experienced



Survey Responses

Q6: How comfortable are you using a terminal/command-line?

Some of you have never used them, others are rather experienced



Survey Responses

New material:

- For many of you: the terminal
- For about half of you:
  - How to import your own code from elsewhere in your computer (packaging may be new to some)
  - Object oriented programming (some of the material may still be new)

Some of the material is either impossible or quite difficult when coding in a notebook. I advise all of you to use this bootcamp as practice for working in text files.

A few of you are students from last year's SURP program who have already gone through this bootcamp – come talk to me if you'd like to go beyond this material!

Survey Responses

New material:

- For many of you: the terminal
- For about half of you:
  - How to import your own code from elsewhere in your computer (packaging may be new to some)
  - Object oriented programming (some of the material may still be new)

Some of the material is either impossible or quite difficult when coding in a notebook. I advise all of you to use this bootcamp as practice for working in text files.

Those of you new to Python – Wednesday is going to be a crash course. I'd strongly advise going through some of the additional beginner-level material linked on GitHub

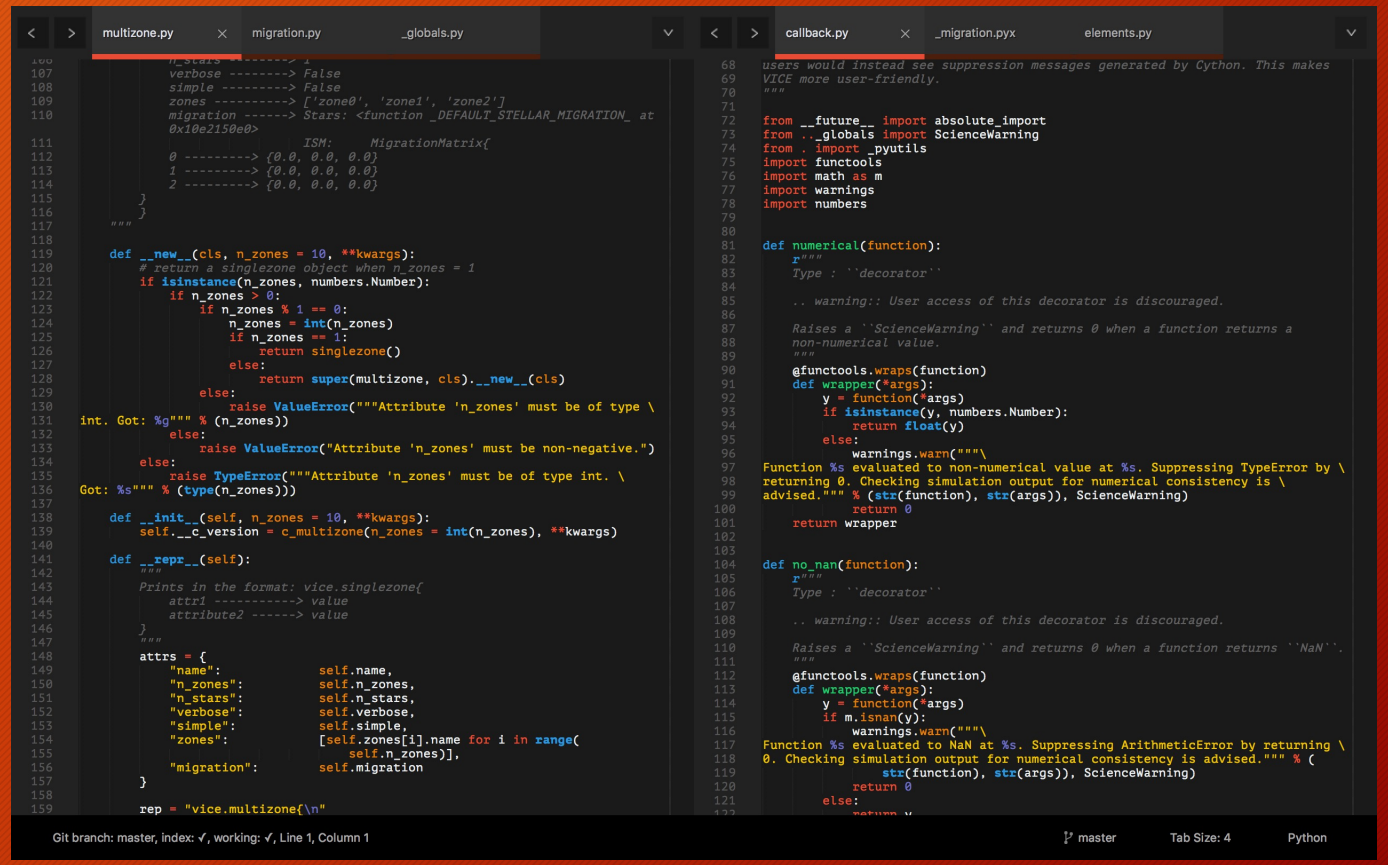
Tools: A Text Editor

Differs from an *Integrated Development Environment* (IDE) in that IDEs will *run* the code – all they do is open, create, edit, etc. plain text files

- pycharm, spyder

I recommend Sublime Text

https://www.sublimetext.com/



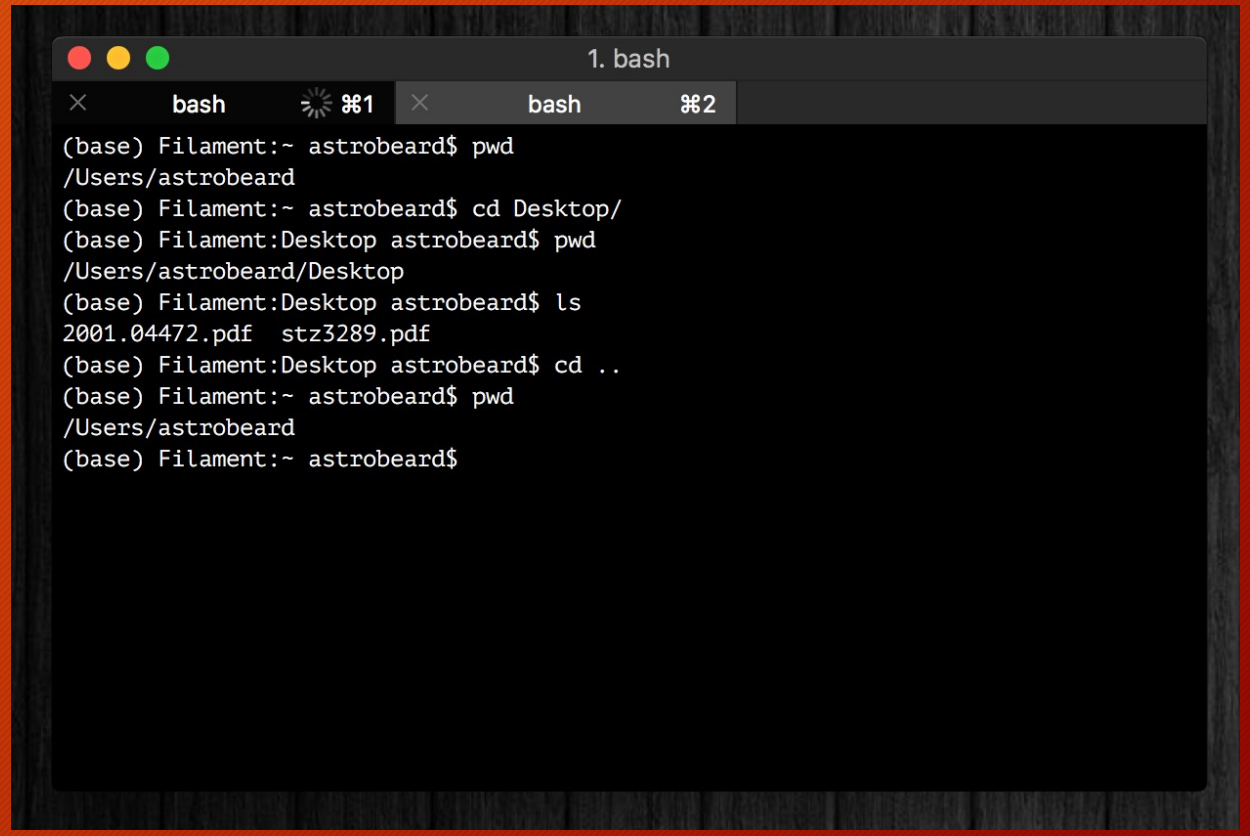
Tools: A Terminal

I recommand iTerm2

- Terminal *replacement*
- https://www.iterm2.com/

You should think of a terminal as just a different interface on a Finder window with some extra programs built-in

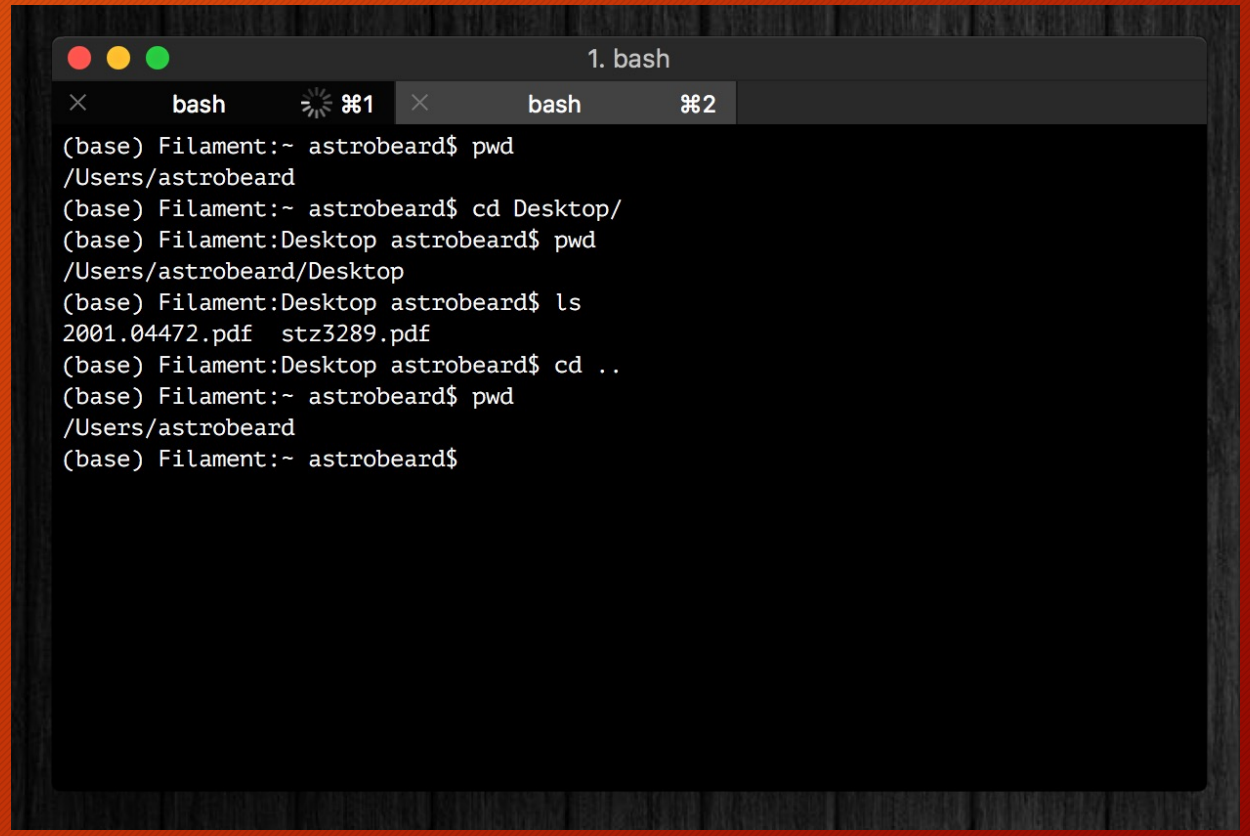
Tip: you can run python line-by-line in a terminal (*python* or *ipython*)



Tools: A Terminal

If you're running Windows, your terminal will be different than some of the notes and exercises here, unless you take some extra steps at the beginning to set up a bash interpreter.

Talk to me if you need help with this!



Tools: Cloud Computing

Allows you to run python on a remote server

SciServer is a popular platform across many STEM fields: https://www.sciserver.org/

In the long run you should choose the tools that you're most comfortable with

If You Haven't Already

https://www.anaconda.com/products/individual

- This will install Python, Anaconda, and Jupyter Notebooks

Latest version of python: 3.10.4

- Most libraries now require >= 3.7
- NumPy's latest version requires >= 3.8

Python 2.7 is *deprecated*

Other Resources

Python Foundation's Beginner's Guide: https://www.python.org/about/gettingstarted/

https://www.learnpython.org/ - There is also an iOS app for this

Codecademy: https://www.codecademy.com/learn/learn-python-3

Goals

What we'll aim to cover:

- How to use a terminal
- Review of the basics: control structures, data types, functions, import, etc.
- How to read documentation
- How to import your own code, and how to set up a directory tree to organize it
- Classes: how to make new objects
  - Inheritance and Composition
- Some basic software engineering principles (i.e. good habits)

Monday Motivation

Every expert coder was once a novice.

You can't improve your coding practices without first criticizing what you once thought was great code.

Since we're not professional developers, scientists have to *actively* create and foster good coding habits if they want them. Being early career researchers, you have the option to make this decision now. It will only become more difficult to do this.