

Welcome to COGS 18:

Introduction to Python

Shannon E. Ellis, PhD

UC San Diego

Department of Cognitive Science

✉ sellis@ucsd.edu

COGS 18

Reminder: This (and all lectures) in COGS 18 are being **recorded**.

Slides available: https://cogs18.github.io/assets/intro/01_welcome.pdf







TAs

Paul
Sylvia

"IAs"

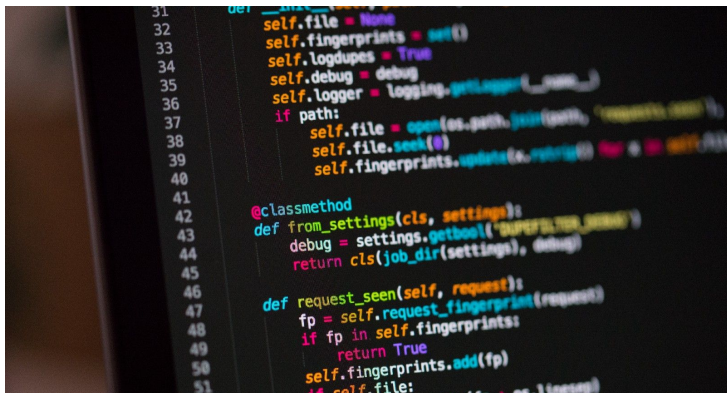
Jose
Taylor

Note: Prof Ellis will have office hours week 1. For all other staff, office hours begin Week 2.

Zoom links/passwords will be shared shortly.

Let's chat: Teaching &
Learning Programming

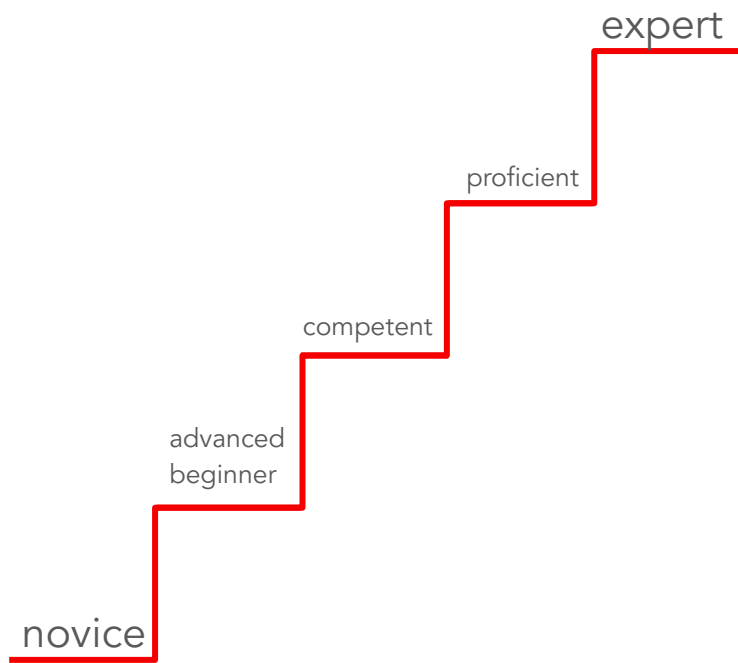
Intro Programming courses
are often **thought of as**
difficult and are courses with
the **highest dropout rates**



....yet, the only thing that is slightly
predictive of success in an intro
programming course is...*how successful*
the student thinks they will be

Things that do NOT predict success:

- gender
- age
- personality
- math ability



My goal is to have you all be able to **program at an introductory level**

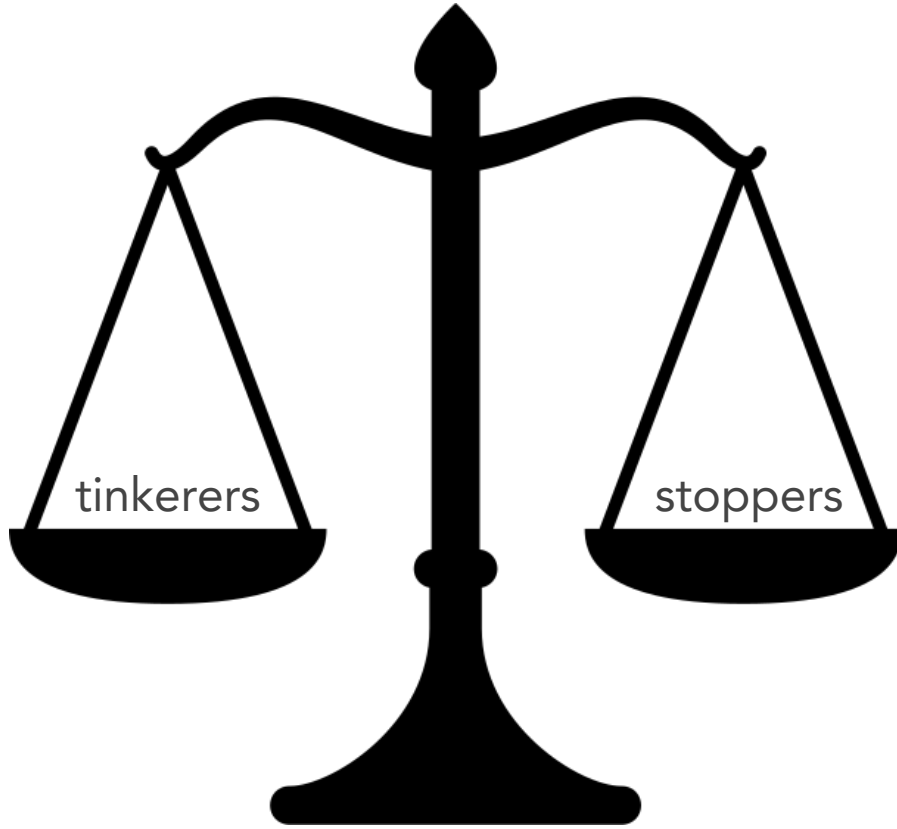
It's generally accepted that it takes people **10 years to move from novice to expert programmer**. But, there are lots of steps in between! We're working to move you further away from novice (& in the direction of expert) than you are right now.



Mixed Messages: We tell people learning to program will be tough and frustrating but that if you're not having fun, you're doing it wrong.



Building Blocks: Too often, we also tell people to “just try things out” without explaining basic concepts. Other courses aren’t taught this way...



Be a mover: Make forward progress. Strike a balance between just stopping and tinkering forever.

If you're not moving forward, consider the **2-hour rule**.

If you're trying to figure something out and struggling to move forward at all, consider the 2-hour rule. If you're stuck, **work on the problem for an hour**. If you're still stuck, walk away & **take a 30 min break**. Then, **try again for another 30 minutes** or so. If you're **still completely stuck, stop and contact us** (come to office hours, post on Piazza). If you're not even sure what your question is, include what information that you do have - what you're stuck on, what you've tried, error messages you've received, etc.

Why Python?

simple(r) syntax


widely-used

Jupyter Notebooks

*"It's not the best language for anything, but
it's the second best for everything"*

-Brad Voytek





COGS 18: How this course is going to work

To avoid the common pitfalls of intro programming courses, we're going to take the following approach:

1. First half of course: basic concepts
2. In-class practice
 - a. Zoom polls for comprehension
 - b. time to apply what was just explained
3. Coding Labs
 - a. Notebooks provided
 - b. Staff/classmates there to help
 - c. Checked for completion, not correctness
4. Assignments
 - a. Completed individually (*can* work together)
 - b. Programmatically graded

COGS 18: How You'll Be Evaluated

	% of Grade	Requirement
Coding Labs	15%	Participate In 5 Coding Labs
Assignments	35%	Complete 5 assignments
Midterms	30%	2 written take-home midterms (1-2 h)
Final Exam	20%	1 final exam (2-3 h)

CodingLabs:
apply concepts
discussed in
lecture using
coding labs
(15%). Practice
makes progress.

Attempt for full credit (3% each)

- Have to make a concerted effort to complete labs
- Coding Labs will be submitted on datahub
- Answers will be sent out the following week
- Can work with others

You should attend the section to which you're assigned. You can attend a different section. However, if one section becomes too crowded each week, we'll revisit this policy.

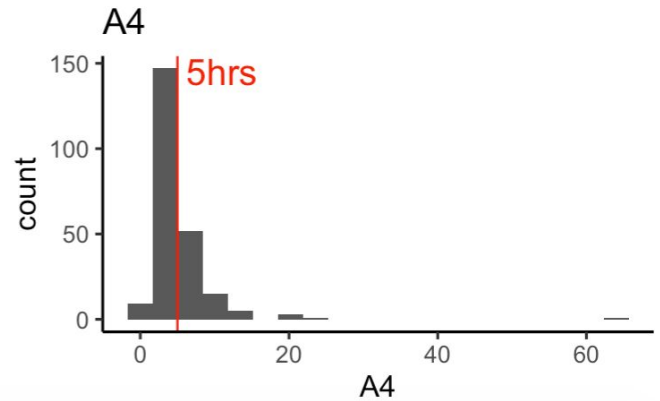
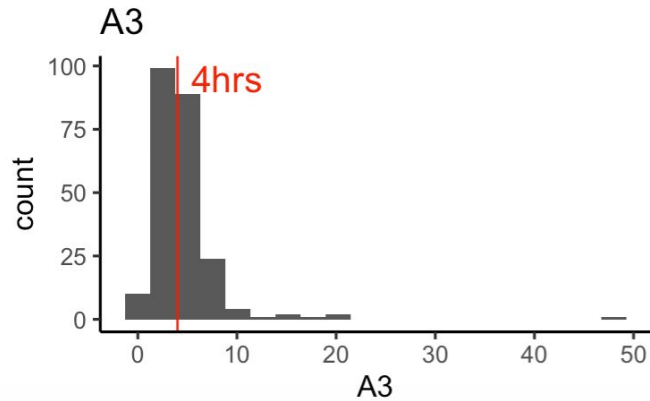
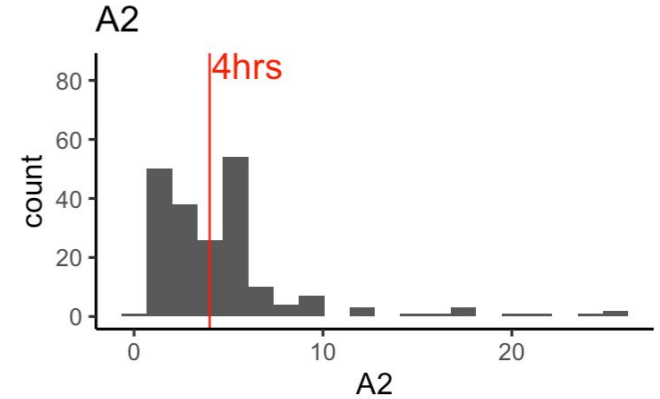
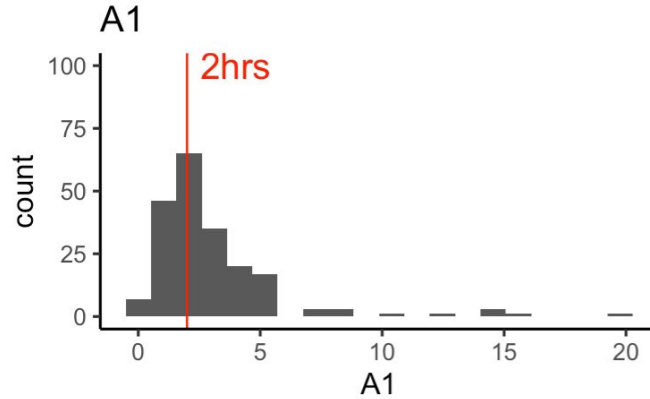
There will be a single zoom link for all coding labs, regardless of day of week or time.
Coding Labs start Wednesday.

(5) Assignments
(35%) : Jupyter
notebooks that are
completed
individually &
graded
programmatically.

Assignments will always be due on either Mon
or Thurs by 11:59 PM.

Assignment	Due Date (11:59 PM)	Day/Week
A1	7/6	Mon; wk2
A2	7/9	Mon; wk3
A3	7/20	Mon; wk4
A4	7/27	Mon; wk5
A5	7/30	Thurs; wk5

Assignments
get more
involved and
take more time
as the course
progresses



Assignment Submission @ Datahub: <https://datahub.ucsd.edu>

DATA SCIENCE / MACHINE LEARNING PLATFORM

UC San Diego

Information Technology Services - Educational Technology Services

Help Options ▾



Log In

Registered Users
"username@ucsd.edu"

UC San Diego Jupyterhub (Data Science) Platform

In technical
classes, **Piazza** is
a particularly
helpful resource

There are rules:

1. No duplicates.
2. Include Assignment & Question in Summary line.
3. Public posts are best.
4. Helping one another is encouraged.
5. No assignment code in public posts.
6. We're not robots.

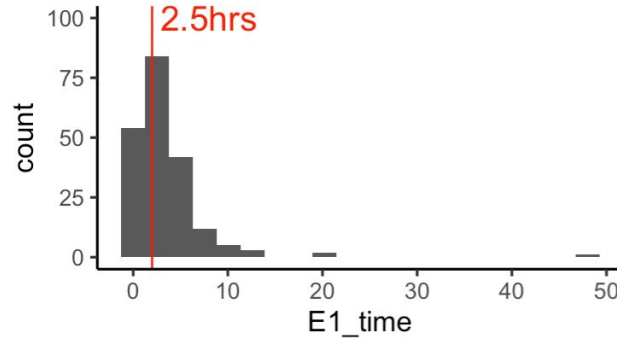
Sign up: <https://piazza.com/ucsd/summer2020/cogs18>

(2) Exams (30%):
will be open-book
and completed on
7/13 and 7/23.

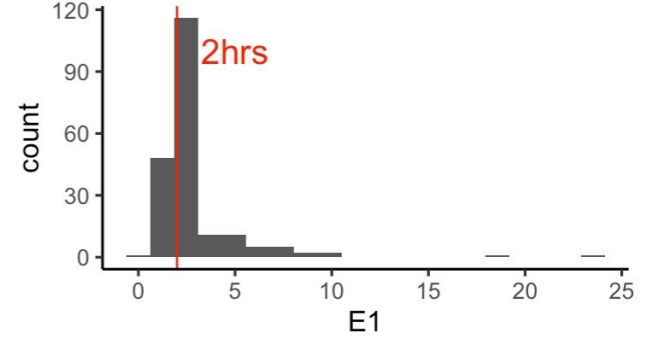
Exams are open-book/open Google but completed on your own. These will be taken on datahub. Each will include a combination of types of questions. There will be a flexible time window when these exams can be taken/submitted. You can expect this exam to take you 1-2 hours to complete.

Exams are
designed to be
completed in
1hr...but many
students took 2hr
to complete their
exams

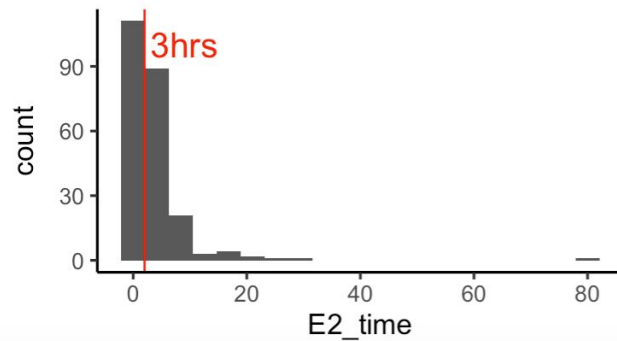
E1: Time Studying



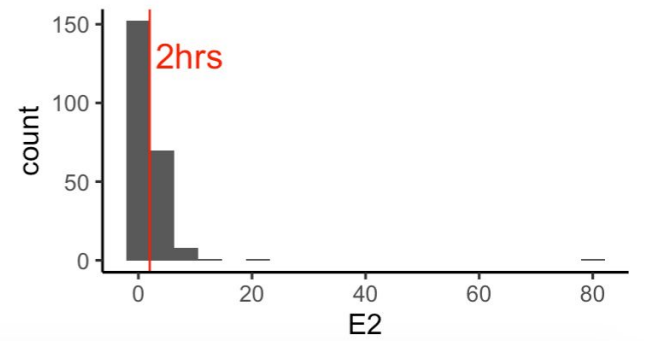
E1: Time Completing



E2: Time Studying



E1: Time Completing



(1) Final Exam (20%):
will be open-book
and completed on
Friday 7/31.

The same as midterms...but more
information and longer. You can
expect this exam to take you
~2-3 hours to complete.



Your point of contact for COGS 18
will be the course website:

<https://cogs18.github.io>

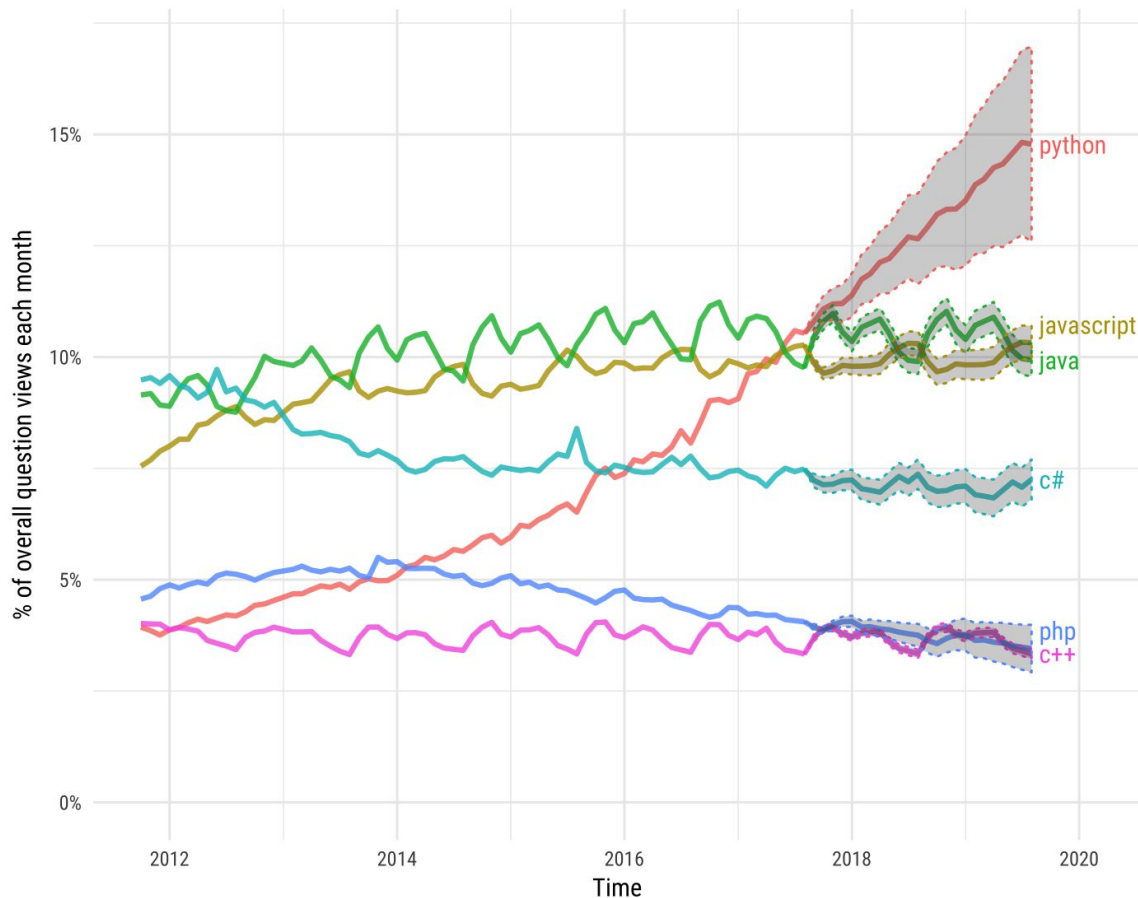
Any questions about
course logistics?

Why even learn to
program in Python?

Python is only
gaining in
popularity as a
programming
language

Projections of future traffic for major programming languages

Future traffic is predicted with an STL model, along with an 80% prediction interval.





The University of St Andrews, founded in 1411, is Scotland's oldest university ([Source](#))



Introduction

The IT Services department at the University of St Andrews, Scotland, develops and maintains software systems used in a variety of capacities throughout the university.

I had several years of experience working with Perl when I took my first serious look at Python back in 1999. Our team's projects were becoming bigger and more complex, and it was obvious that we needed to bring to them more structure and clarity. I had been looking at Java for some time, but its potential benefits seemed to come at the cost of a steep learning curve, and an overall increase in development time. In contrast, Python appeared to offer the prospect of having both clarity *and* productivity at the same time. And if we ever needed to make use of Java's class libraries there was always Jython, an implementation of Python for the JVM. The increasing number of Python books being published testified to the language's growing popularity, and the number of available libraries was beginning to rival Perl's. This convinced me to give Python a try.

<https://www.python.org/about/success/st-andrews/>



In 1999, Python becomes mainstay of IT systems

Use Case: Matching students with a class

- Select preferences each semester
- Validate who is in which course
- Handles concurrent users well

Why Python:

- Reduced amount of programming needed
- Quickly learned by staff





INDUSTRIAL
LIGHT & MAGIC

VISUAL EFFECTS

ART DEPARTMENT

SERVICES

ILMxLAB

COMPANY

BLOG

Background


Industrial Light & Magic (ILM) was started in 1975 by filmmaker George Lucas, in order to create the special effects for the original Star Wars film. Since then, ILM has grown into a visual effects powerhouse that has contributed not just to the entire Star Wars series, but also to films as diverse as Forrest Gump, Jurassic Park, Who Framed Roger Rabbit, Raiders of the Lost Ark, and Terminator 2. ILM has won numerous Academy Awards for Best Visual Effects, not to mention a string of Clio awards for its work on television advertisements.

In 1996, Python unifies ILM

- Used Python to code (and re-code) the programs that controlled the production pipeline
 - Ease-of use
 - Quickly learnable
 - Replace Unix shell scripting
- What Python accomplished
 - Streamlined production
 - Provided the needed flexibility
 - Reduced hardware costs
 - Stay on top of the competition
- Since 1996, no better system found



*Entering a new era in
vascular and cardiac
regeneration research*

Read more 



Annual Report 2018

What science can do



Sustainability Report 2018

Leading with health

We believe in what science can do

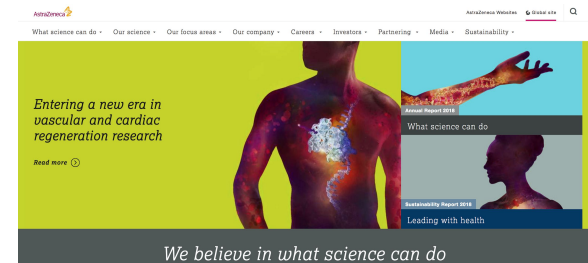
Introduction

AstraZeneca is one of the world's leading pharmaceutical companies. With over 54,000 employees worldwide, it provides innovative, effective medicines designed to fight cancer, provide pain control, heal infection, and fight diseases of the cardiovascular, central nervous, gastrointestinal, and respiratory systems.

To save time and money on laboratory work, experimental chemists use computational models to narrow the field of good drug candidates, while also verifying that the candidates to be tested are not simple variations of each other's basic chemical structure.

At AstraZeneca, Python enabled collaboration

- Drug Discovery is key
- Computational chemists weren't the only ones who could do analysis
 - web-based interface called H2X, initially written in Perl
 - Experimentalists could make predictions on their own
 - In 2001, further developed in Python: PyDrone
 - 3 months of development time + 3 months of QA + 3 weeks of documentation time to produce about 5,600 lines of finished Python code.
 - Chosen for its ease-of-use & interpretability
 - Can solve real-world problems
 - Error handling made product more robust



AIR TRAFFIC MANAGEMENT

Keeping 3 billion passengers safe each year

VOICE COMMUNICATIONS

NETWORKS

REMOTE VIRTUAL TOWER

ATC TOWER

SURVEILLANCE

AIM

AMHS

ATM/UTM INTEGRATION

Keeping 3 billion passengers safe each year

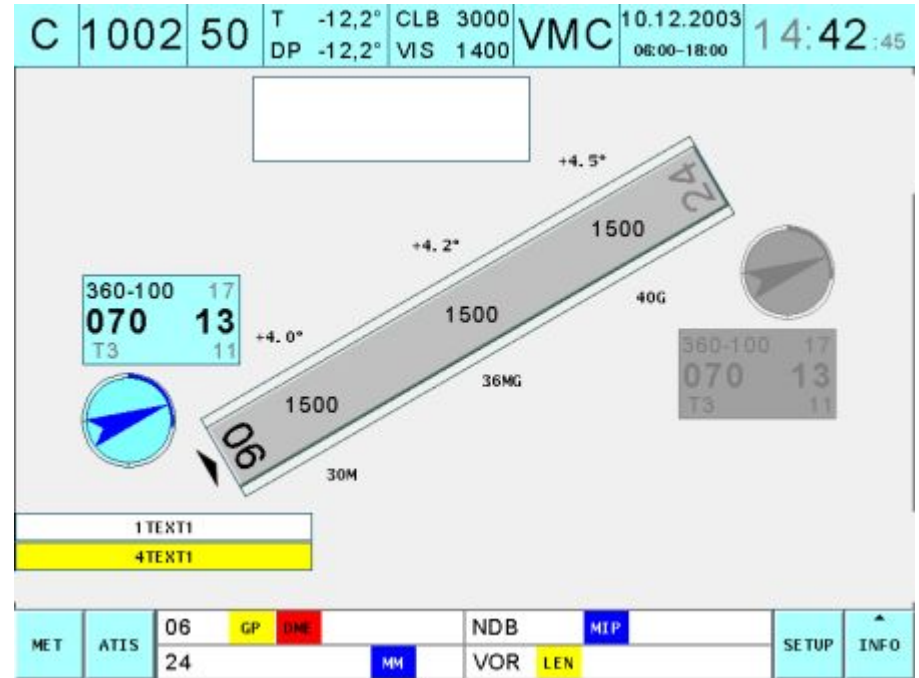
Air Traffic Management (ATM) solutions leverage more than seventy years of ATC experience showcasing true leadership in this market, with deep cross-industry experience to provide functionality in support of the needs of today's Air Navigation Service Providers (ANSP). Embracing digitalisation, virtualisation and innovation, these solutions are an essential enabler to lead today's Air Navigation Service Providers through future evolutions of their infrastructure.

Frequentis is one of the world's leading providers for safety-critical solutions in the field of Air Traffic Management and Public Safety & Transport. With over 500 employees world-wide, it provides innovative, user-centered solutions to its customers.

Frequentis has been using Python in its TAPtools[®] product family, which focuses on the *Tower and Airport Tools* segment of Air Traffic Control. These tools are used by air traffic controllers to track weather conditions, control runway lighting, and to monitor and control navigational aid instruments.

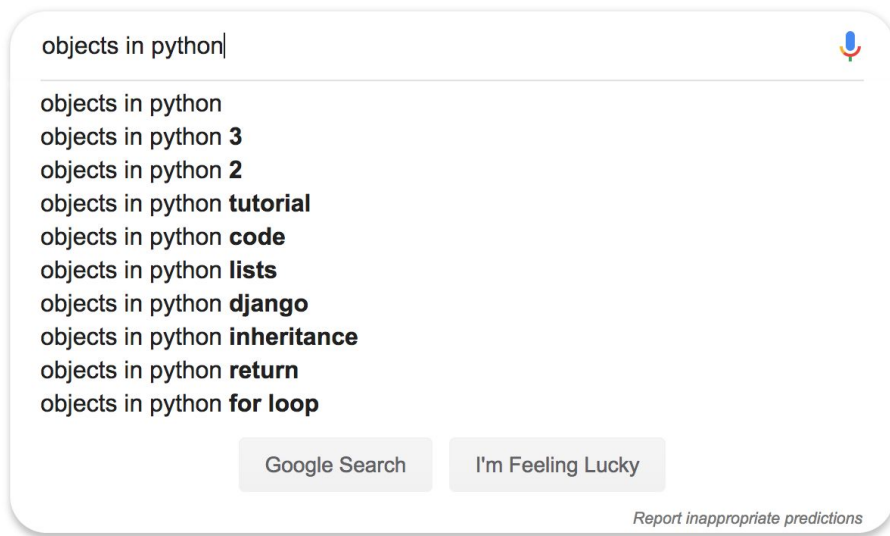
Python enables tool to work in front of the customer

- Each airport is unique
- TAPtools enables interface unique to airport to be built
 - Rapidly developed in front of the user
 - Minimizes back and forth
- Python advantages
 - Easy to read and maintain
 - Flexible
 - Faster to write (3X faster)



Where to turn for **help**
and practice when
learning to program?

Including “in python” in your
Google search can be magic

A mockup of a Google search interface. At the top, a search bar contains the text "objects in python|". To the right of the search bar is a microphone icon. Below the search bar, a list of search suggestions is displayed: "objects in python", "objects in python 3", "objects in python 2", "objects in python **tutorial**", "objects in python **code**", "objects in python **lists**", "objects in python **django**", "objects in python **inheritance**", "objects in python **return**", and "objects in python **for loop**". At the bottom of the suggestions list are two buttons: "Google Search" and "I'm Feeling Lucky". Below these buttons is a link that says "Report inappropriate predictions".

objects in python|

objects in python
objects in python 3
objects in python 2
objects in python **tutorial**
objects in python **code**
objects in python **lists**
objects in python **django**
objects in python **inheritance**
objects in python **return**
objects in python **for loop**

Google Search I'm Feeling Lucky

[Report inappropriate predictions](#)

StackOverflow probably has the answer to your question



21 ● 3



Home

PUBLIC

Stack Overflow

Tags

Users

Jobs

Teams
Q&A for work



Learn More

Tags

A tag is a keyword or label that categorizes your question with other, similar questions. Using the right tags makes it easier for others to find and answer your question.

Popular

Name

New

python × 1137913

a multi-paradigm, dynamically typed, multipurpose programming language, designed to be quick (to learn, to use, and to

1085 asked today, 6241 this week

python-3.x × 151128

For questions about Python programming that are specific to version 3+ of the language. Use the more generic [python] tag

273 asked today, 1641 this week

python-2.7 × 89413

the last major version in the 2.x series. Do not use this tag simply to convey the version of Python you're using, unless the question

40 asked today, 219 this week

python-requests × 9229

a full-featured Python HTTP library with an easy-to-use, logical API.

8 asked today, 81 this week

wxpython × 6191

a Python wrapper for the cross-platform C++ GUI API wxWidgets.

20 asked this week, 52 this month

ipython × 6036

a feature-rich interactive shell for Python, and provides a kernel for frontends such as IPython Notebook and Jupyter Notebook.

15 asked this week, 65 this month

python-imaging-library × 4495

The Python Imaging Library (PIL) provides the Python language with a de-facto standard foundation for image work. PIL's

23 asked this week, 113 this month

python-3.6 × 3882

Version of the Python programming language released in December 2016. For issues specific to Python 3.6. Use more

10 asked today, 43 this week

python-3.5 × 3260

The version of the Python programming language released on September 13, 2015. For issues that are specific to Python 3.5.

9 asked this week, 34 this month

python-import × 3150

For questions about importing modules in Python

18 asked this week, 58 this month

python-3.4 × 2594

The version of the Python programming language released on March 16, 2014. For issues that are specific to Python 3.4. Use

6 asked this month, 126 this year

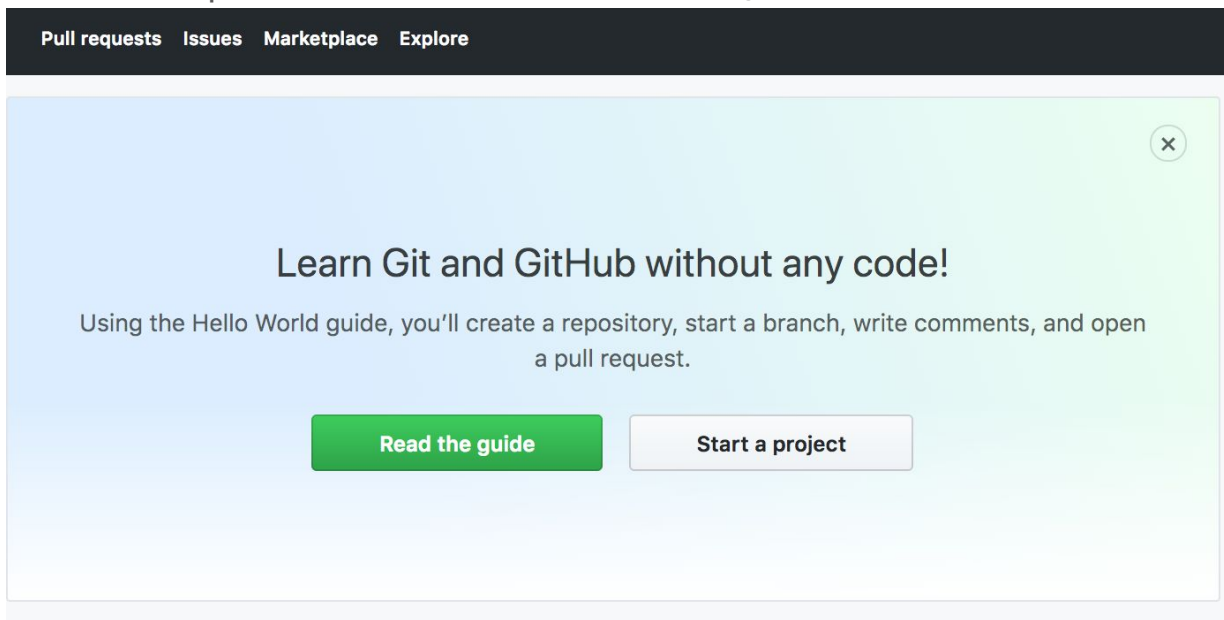
python-sphinx × 2365

a tool that makes it easy to create intelligent and beautiful documentation. Sphinx is especially suitable for Python

9 asked this week, 34 this month

GitHub: programmers' social media platform

Code is shared on GitHub. In the beginning, it may be intimidating, but I encourage you to familiarize yourself with the platform and share code you write on GitHub.



There are also
**COGS18-specific
avenues** when
looking for help

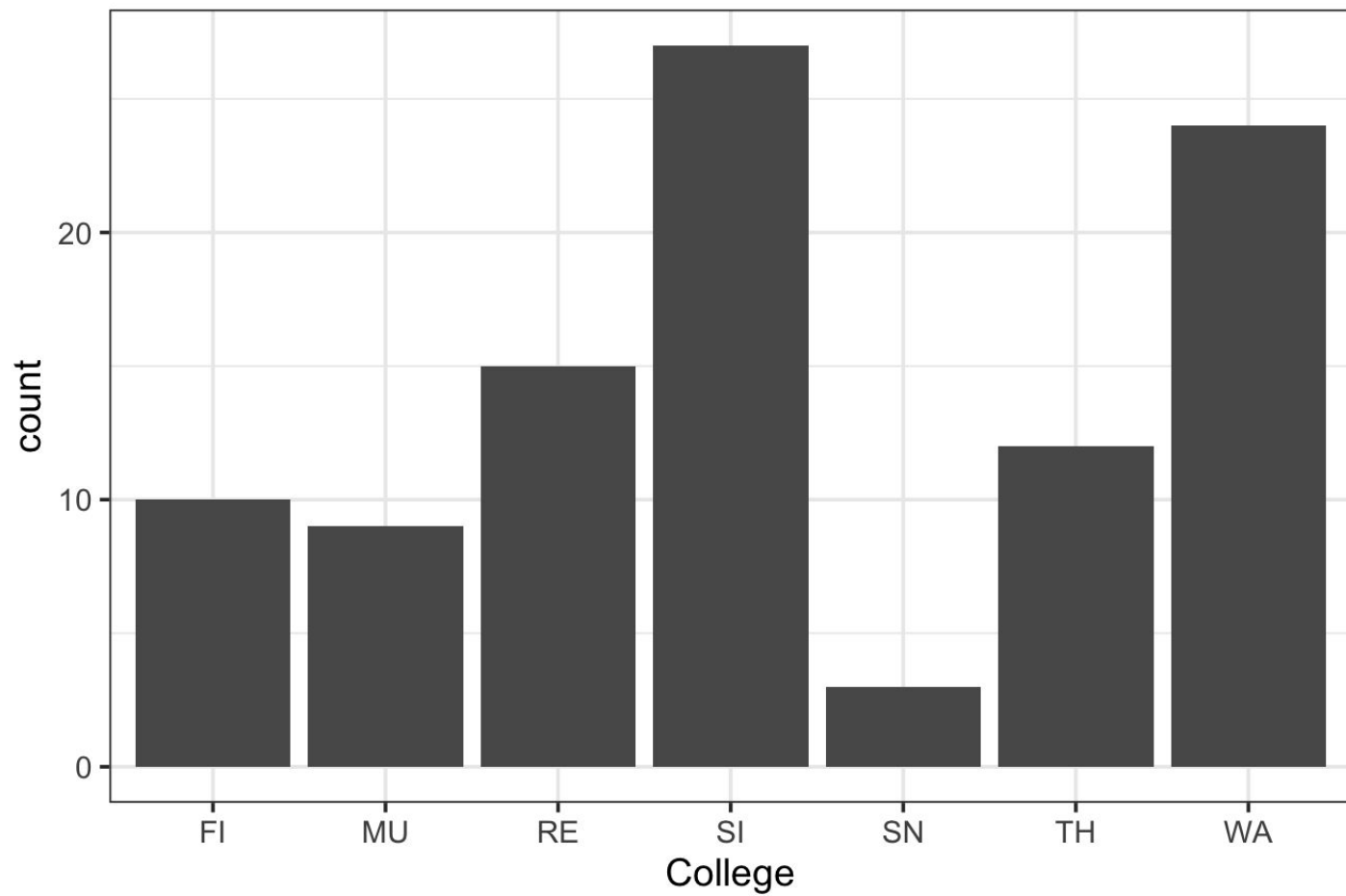
Questions in **CodingLabs**, coming to
office hours, talking to your **classmates**,
or reaching out for help on **Piazza** are
all options for you. You're encouraged
to help one another on **Piazza**!



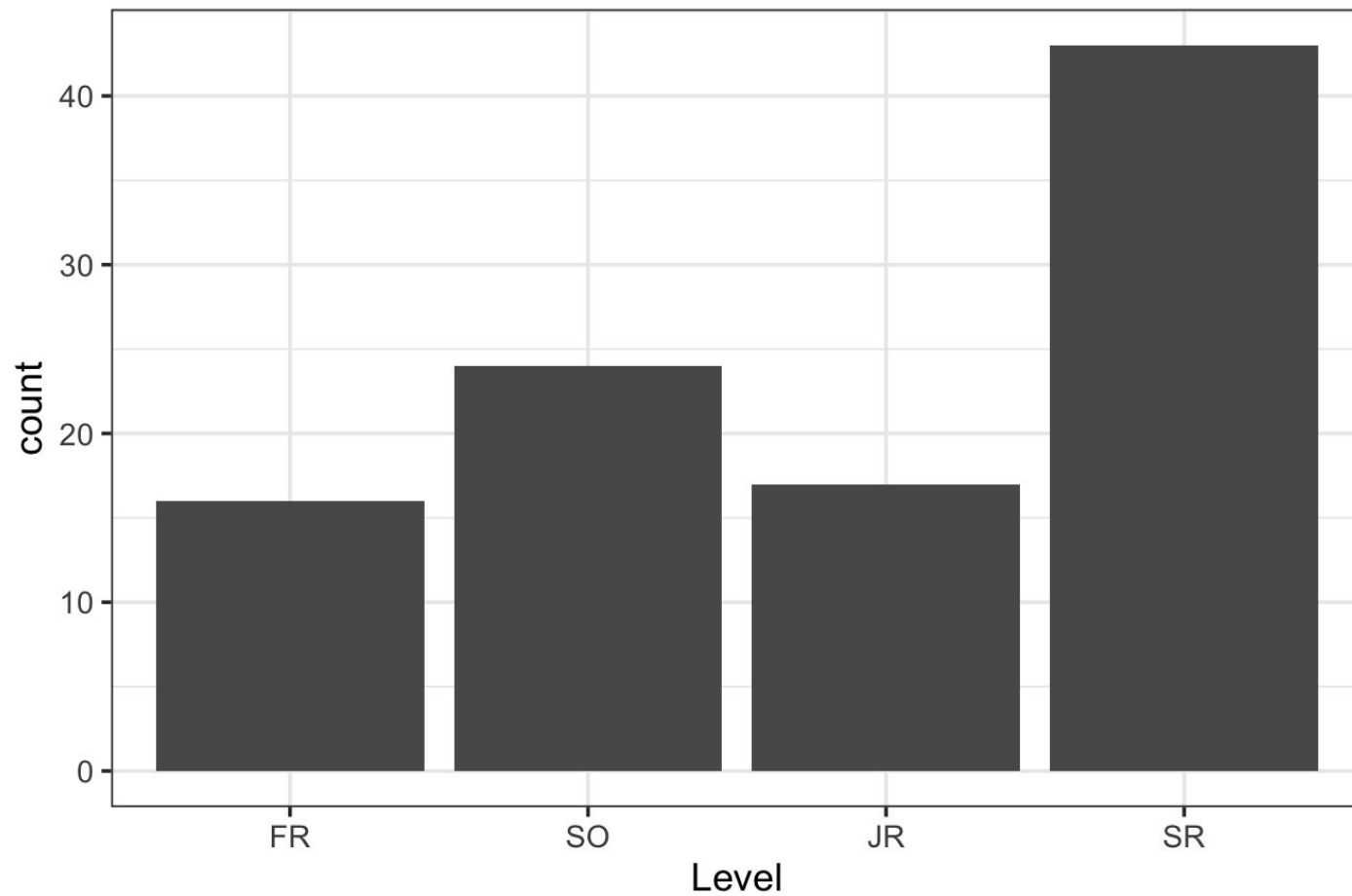
Today I used a PDF slideshow,
but every other day of class,
lecture notes will be presented
in a **Jupyter notebook**



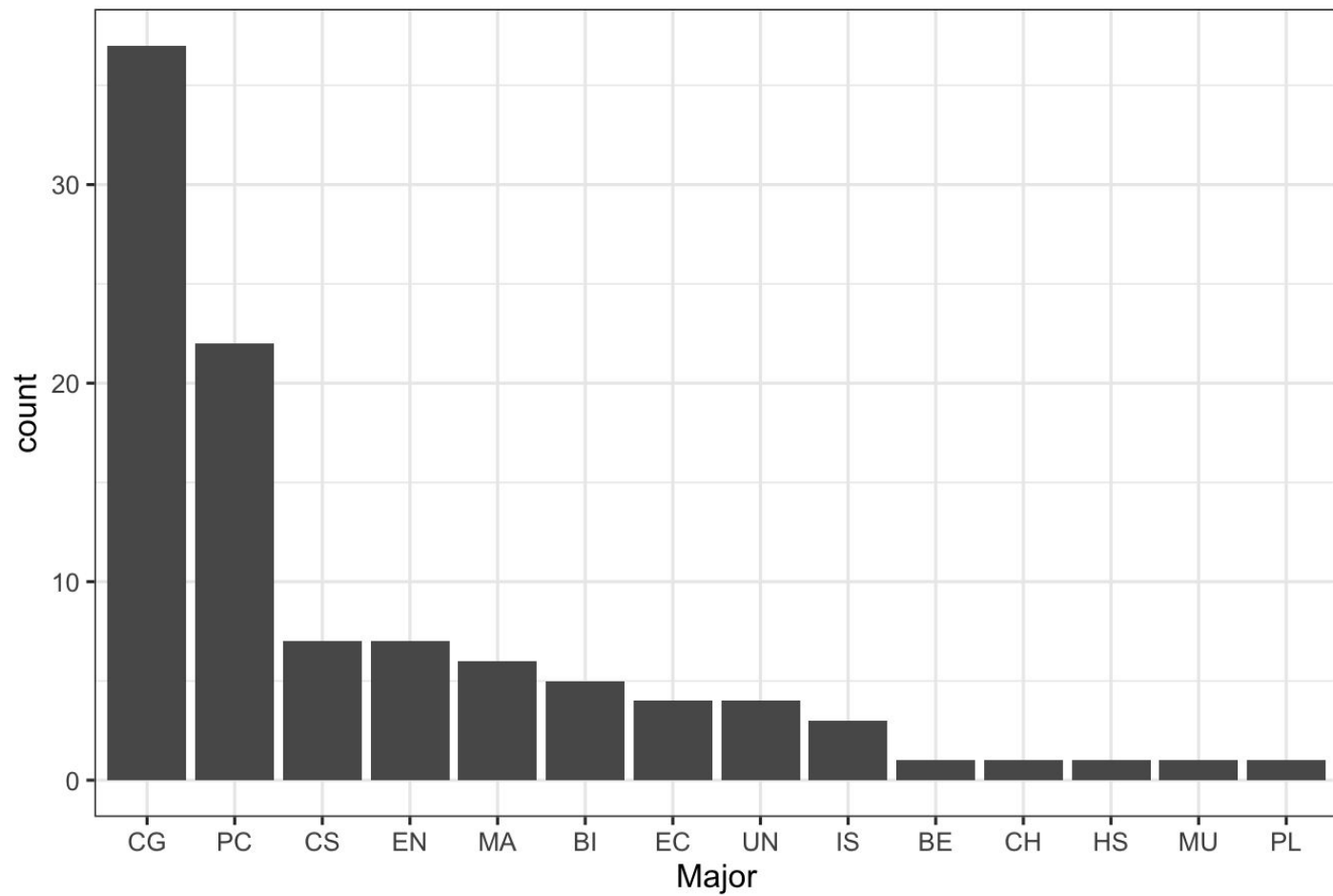
COGS 18



COGS 18



COGS 18





I'm excited to have you all in
COGS 18 this quarter & I'd love
to learn more about you:

http://bit.ly/cogs18_survey_su20