CIS 192: Python Programming 🐍

Course Wrap Up!

Arun Kirubarajan
(kiruba@seas)
@kirubarajan
kirubarajan.com
Game Plan

1. What we learned!
2. Next steps for Python!
3. Next steps for programming!
4. Go home early or play Super Smash Bros!
NOT SURE IF PSEUDOCODE
OR JUST PYTHON
"You can't just copy-paste pseudocode into a program and expect it to work."

That's where you're wrong kiddo.
Python: Dad, can I have $x++$?

: No, we have $x++$ at home.

$x++$ at home: $x+=1$
PYTHON

IS THIS A DATA TYPE?
PYTHON

THIS IS PLAGIARISM.
YOU CAN'T JUST "IMPORT ESSAY."

JAVA

I'M TWO PAGES IN AND I STILL
HAVE NO IDEA WHAT YOU'RE SAYING.
How am I so good at this? I've never written code before!
When you try to define constants

*PYTHON*

We don't do that here
Why does Python live on land?
Because it is above C level!
He's a special boy
Python is the easier language to learn. No brackets, no main.

You get errors for writing an extra space.
Other languages:
*proper description of the error*

Python:

is that like invalid syntax or something
So, what did we actually learn in this class?
1. Python Syntax
2. Python Data Structures
3. Object Oriented Programming
4. File Input/Output
5. Project Architectures
6. REST API (Flask)
7. Full Stack Development (Django)
8. Database Interactions (SQL)
9. Machine Learning
10. Deep Learning (Keras)
11. Natural Language Processing
Now what?
Where to go from here?

If you’re interested in learning more advanced Python:

- Stanford's Python class
- PyCon lectures
- RealPython tutorials
- Obey the Testing Goat (TDD in Python + Web Dev)
- r/Python community
- Learn Python for automation
- Make games with PyGame
- Full Stack Python
Where to go from here?

If you’re interested in learning more about building large-scale systems and infrastructure:

- **NETS 212**
  - Build a social network
- **CIS 331**
  - Several fun security projects
- **CIS 341**
  - Build a compiler
- **CIS 380**
  - Build an OS
- **CIS 455/555**
  - Build a search engine
Where to go from here?

If you’re interested in learning more about machine learning and deep learning:

- Stanford ML courses
  - CS221 Artificial Intelligence: [http://cs221.stanford.edu/](http://cs221.stanford.edu/)
  - CS231n ConvNets for Visual Recognition: [https://cs231n.github.io/](https://cs231n.github.io/)

- fast.ai courses
  - More applied than Stanford courses, less intuition

- CIS 419/519/520/521/530/545/581/680
  - 419/519 is more applied, 520 is more mathematical
  - 521 covers general AI, 530 covers NLP
  - 545 will likely be too basic for you after 192
  - 581/680 cover deep learning and computer vision
Where to go from here?

- **What every computer science major should know**
- Use the skills you’ve learned from 192 to work on side projects
  - Side projects are the best way to pick up new skills and keep learning things that will be relevant in industry.
  - Courses will almost never teach you what you need to do your job.
- **Consider doing research**
  - Data analysis and ML skills are valued deeply by the research community.
  - Not interested in CS research? No problem!
    - Consider professors in Wharton, Annenberg, Economics, etc.
- **Contribute to open source projects**
  - [Trending GitHub Python repositories](https://github.com/trending/python)
Considering a career as a software engineer?

- Learn how to use the Python Standard Library well
  - Understand data structures, algorithms, and systems
- Interview Prep
  - Practice interview questions on Leetcode
  - Read this book
- Read about the tradeoffs between startups and big tech companies
  - Understand how equity compensation works
- Learn how to negotiate
- Learn how to evaluate a company’s culture
Reminders

1. Take the staff out to lunch!
2. Complete course evaluations!
3. Submit final project proposals!
4. Sign up for demo slots!
See you around campus! 🐍