

Python Language

For teaching Computer Programming

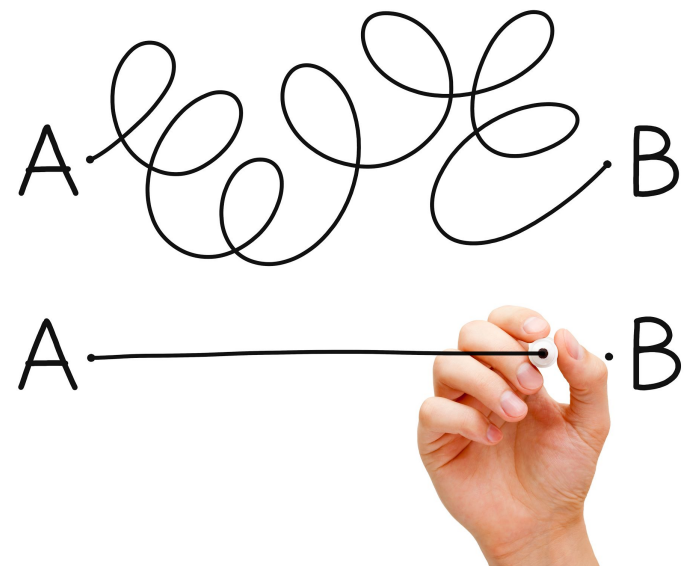


Important Questions

Why teach
Python?

Python is a simple language

- Simplified syntax
 - White space instead of brackets
 - Words used for logic instead of symbols
 - Dynamic typing for variables and functions
 - Flexible lists
 - Readable for loop structure
 - Main function not required
 - automatic namespaces
- Interpreter instead of a compiler
 - Run code as you write it
 - Line by line testing



Hello, World Comparison

C++

```
#include <iostream>
using namespace std;
int main() {
    cout << "Hello, World!";
}
```

Java

```
class myfirstjavaprogram {
    public static void main(String args[]) {
        System.out.println("Hello World!");
    }
}
```

Python

```
print("Hello, World!")
```

Python Promotes Readable Code

```
def example():  
    print("No semi colons, no brackets!")  
    for index in range(100):  
        if index == 21:  
            print("Indented code blocks")
```

- Less punctuation in syntax
- More keywords as operators
- Enforced indentation

Python is a powerful real-world language

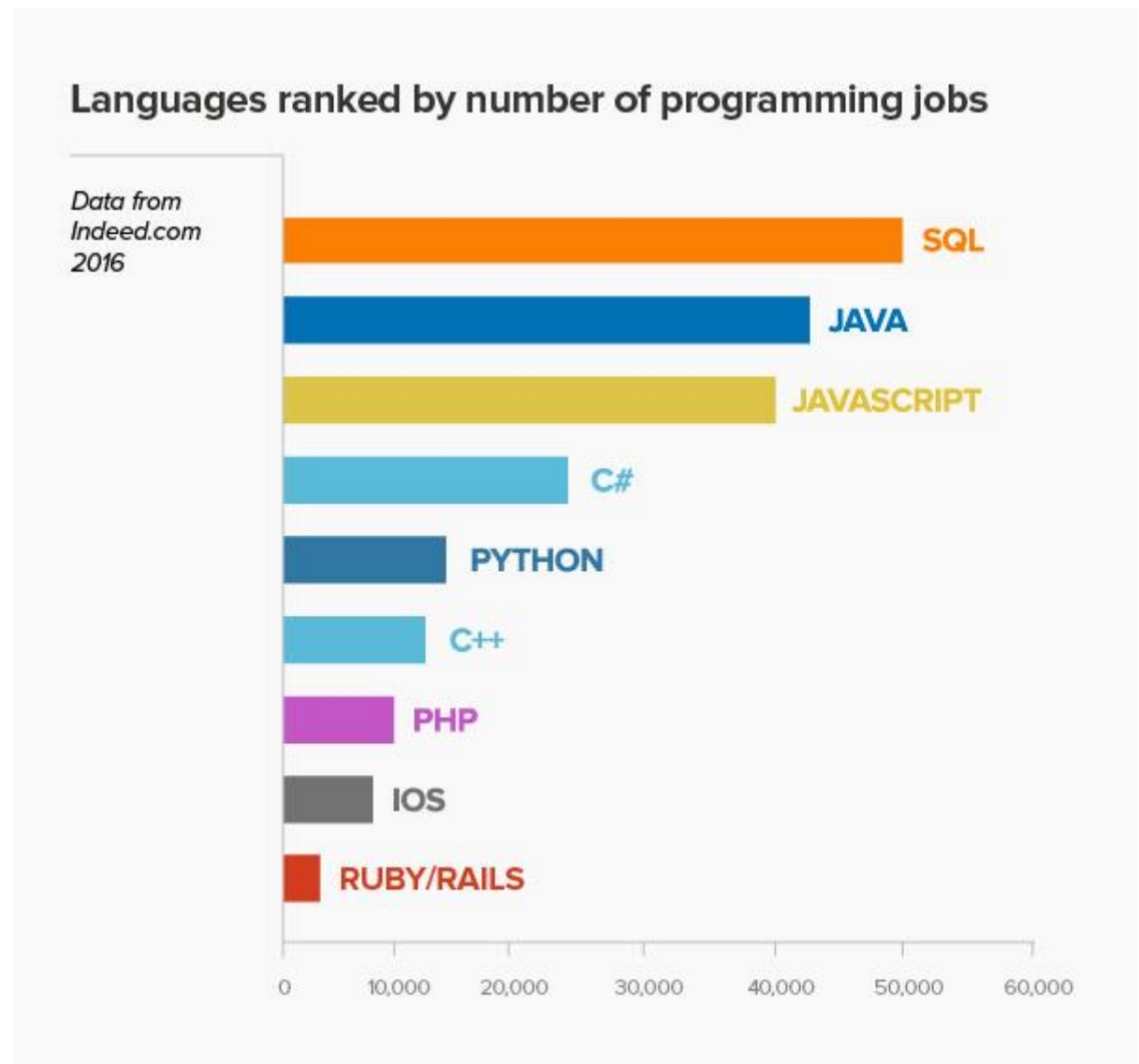
- Python is an [industry standard language](#)

- Google
- Youtube
- DropBox
- Maya
- Blender
- NASA
- NSA
- Industrial Light and Magic
- Pixar
- Cisco
- Intel
- Netflix
- etc.



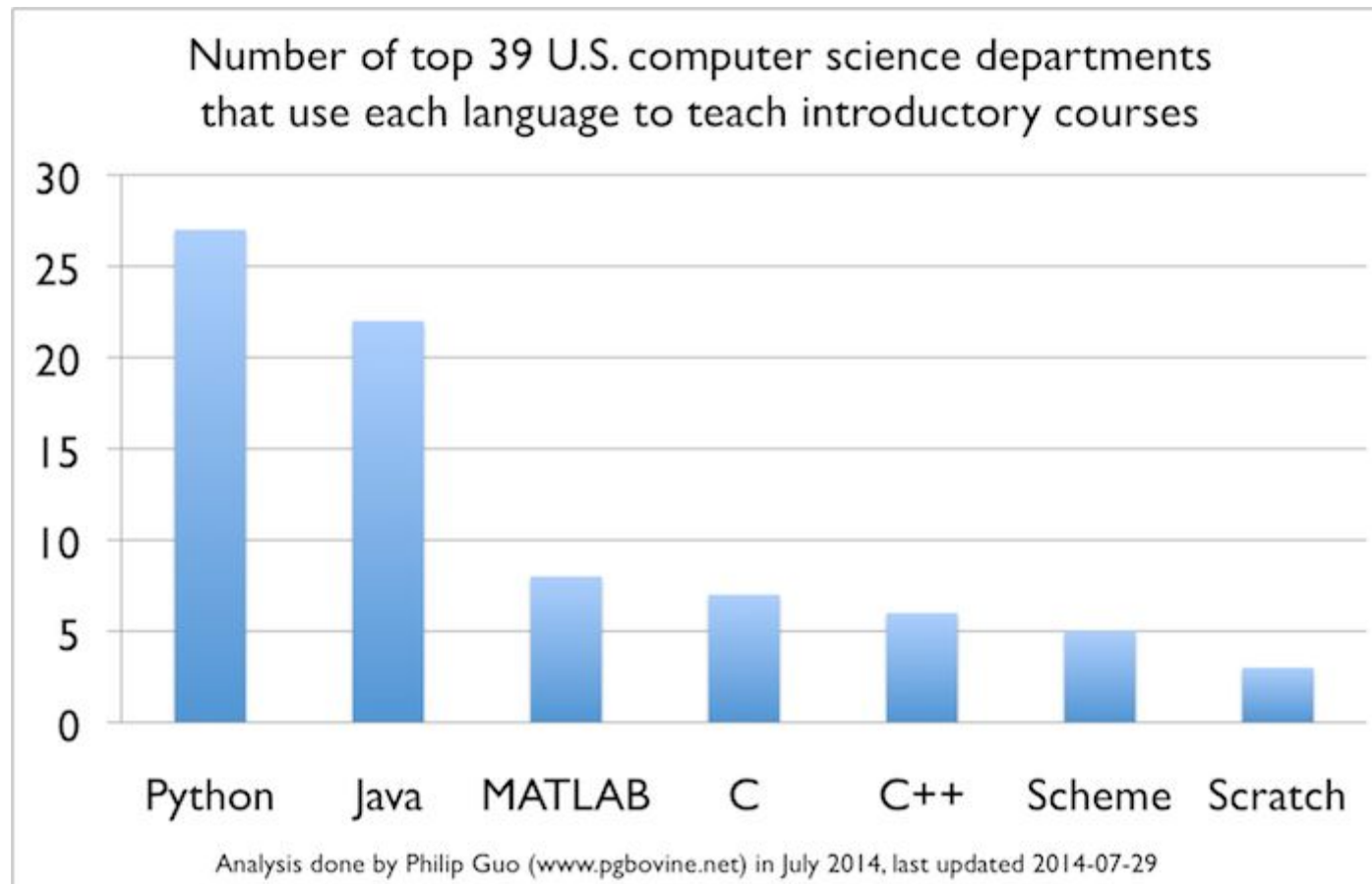
Python is in demand

- [A top five employable language](#)
- Popular for Web Development frameworks
 - Django
 - TurboGears,
 - web2py, etc.



Python is being adopted by Higher Education

- Python is the [most popular introductory language](#) with the top universities in the United States



Python is a fun language

Created by a guy named Guido
from the Netherlands.



Named after the comedy troupe Monty Python, very silly.

Active open source community

- [Python has an active, helpful community](#)
- Free to use, built in to most Operating Systems.
- The language of choice for raspberry pi
- Guido is still heading the development of the language, and it is getting more popular every year



How to learn
Python?

A good strategy

- Get familiar with the interpreter
- Successfully run a script
- Find a book you like and read it
- Experiment as you read
- Challenge yourself with problems to solve
 - 10 minute stuck limit
 - Ask questions on forums
 - Ask questions on stackoverflow
 - Email me questions

Which version to learn?

Python version 2 has been around the longest and has the most resources for learning

Python version 3 has officially been introduced and version 2 is no longer receiving updates and will be phased out.

Learn both. The basics are the same with minor exceptions. Learn version 3 where you can, but don't turn down a good learning tool that is in version 2.

The biggest difference between 2 and 3 is the print statement

Version 2: `print "Hello, World!"`

Version 3: `print("Hello, World!")`

Straight from the horse's mouth

“The Interpreter”

- Open the python interpreter
 - Try opening python2 and python3 interpreters
- Type `help()`
- Checkout the keywords, symbols, topics, and modules
- Open the official python tutorial

We need to write a script, Which editor to use?

Editors

Atom

Sublime

Text Wrangler / Notepad++

Emacs / Vim

IDLE

IDE

Pycharm

How to teach
Python?

Repetition, repetition, repetition

- Cover everything, then cover it all again, then again.
 - Each time through more complex
- Daily practice drills (computer and paper)
- Solve problems (individual and pairs)
- Use online tutorials for homework
- Build a portfolio on github

Books

A Byte of Python

<http://python.swaroopch.com/>

How to think like a computer scientist

<http://openbookproject.net/thinkcs/python/english3e/>

Interactive think python

<http://interactivepython.org/runestone/static/thinkcspy/toc.html>

Python for you and me

<http://pymbook.readthedocs.io/en/py3/>

Hands On Python

<http://anh.cs.luc.edu/python/hands-on/3.1/handsonHtml/index.html>

More books and tutorials

Video Lessons: Code101 by TWit (episodes 11-18)

<https://twit.tv/episodes?page=4&filter%5Bshows%5D=1680>

Big List of Resources: Hitch Hiker's Guide to Python

<http://docs.python-guide.org/en/latest/intro/learning/>

Notable resources

- Python Tutor <http://pythontutor.com/>
- Learn Python <http://www.learnpython.org/>
- Invent with Python <http://inventwithpython.com/>

Interactive learning options

- Code School: Try Python
<http://www.trypython.org/>
- Codecademy
<https://www.codecademy.com/learn/python>
- Code Wars
<https://www.codewars.com>
- Code Combat
<https://codecombat.com>

Do it! Exercises and Solving problems

- Basic Exercises

<http://cit.cs.dixie.edu/cs/1400/PythonLearning.pdf>

- Codingbat drills

<http://codingbat.com/python>

- Python Practice Book

<http://anandology.com/python-practice-book/index.html>

- Koans learn by testing

https://bitbucket.org/gregmalcolm/python_koans

- Project Euler

<https://projecteuler.net/archives>

- Rosalind Problems

<http://rosalind.info/problems/locations/>

Visual Programming - Turtle Graphics

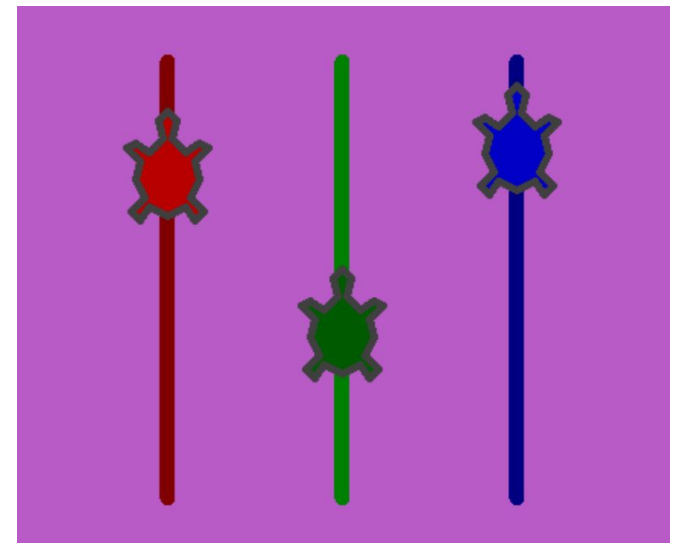
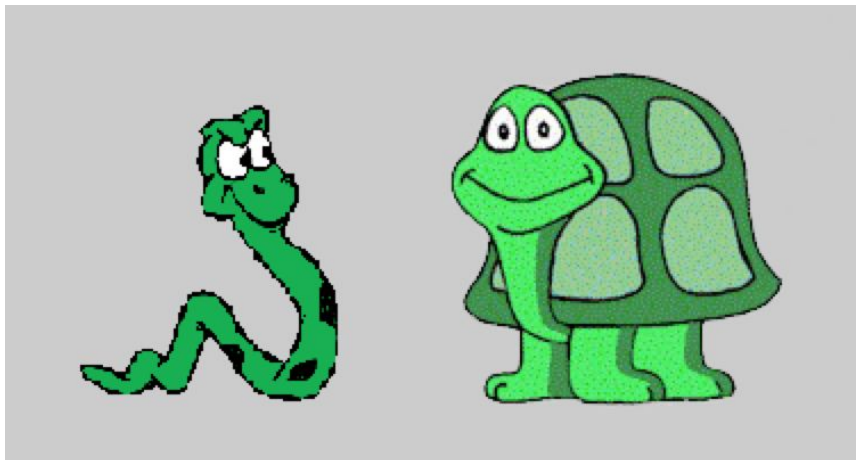
- **Helper App**

<http://pythonturtle.org/>

- **turtle** library built in with simple graphics computing

- import turtle
- <https://opentechschoool.github.io/python-beginners/en/index.html>
- ["Seven Ways to Use Python's New Turtle Module"](#)
- Exercises

<http://www.pythonforrookies.org/Answers/answersBook.pdf>



Visual Programming - Pygame Library

- Lots of exercises

<http://cit.cs.dixie.edu/cs/0000a/assignments.php>

- Tutorial

<http://programarcadegames.com/>

- Making games

<http://inventwithpython.com/>

- Pygame official site

<http://www.pygame.org/wiki/tutorials>

- Assignments

[Maze game](#)

[Virtual Pet](#)

[Hunt the Wumpus](#)

Tic Tac Toe



Visual Programming - Pyglet



- Tutorials

<http://steveasleep.com/pyglettutorial.html>

<http://www.novakeith.net/2013/02/26/asteroids-using-pyglet-python/>

- Official pyglet documentation

http://pyglet.readthedocs.io/en/latest/programming_guide/quickstart.html

<https://bitbucket.org/pyglet/pyglet/wiki/Home>

Assignments

- High Low Game
http://cit.cs.dixie.edu/cs/1400/labs/high_low_example.php
- Game of Pig
<http://cit.cs.dixie.edu/cs/1400/labs/pig.php>
- Encryption
<https://cit.dixie.edu/cs/1400/labs/encryption.pdf>
- Etc. (Thank you Dixie State University)
<http://cit.dixie.edu/directorylisting.php?u=cs/1400/labs/>

Online Courses

Google Campus

<https://developers.google.com/edu/python/>

MIT Open courseware Python

<http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-189-a-gentle-introduction-to-programming-using-python-january-iap-2011/index.htm>

Everything

Python Syntax Summary

<http://cit.cs.dixie.edu/cs/1400/PythonLanguageSummary-week9.pdf>

Everything about python3 in one file

<https://learnxinyminutes.com/docs/python3/>

Walk this way

Lets help each other out and share our resources.

Git on board my repo
[alveyworld/teachpython](https://github.com/alveyworld/teachpython)

