Develop Intelligence

Learning Solutions to Attract, Retain, and Grow your top technical talent.

Who Uses Python?

- Google
- Facebook
- Dropbox (Guido works there!)
- Quora
- Reddit
- Netflix
- Bank of America (porting their entire backend)
- etc ...

Why Python?

- Ease of use
- Simplicity, readability
- **Immense** open-source community
- Broad, broad applicability
  - Web
  - Scripting
  - Data
  - Scientific
  - ML

Categorization of Python

- No formal public, private methods
- Strongly, dynamically ("duck") typed
- Interpreted language
- Function & classes are first class objects
- Pass by value (but non-primitives' "values" are like references...more on this later)

Isn't Python too slow?

- Depends - but probably not!
  - In terms of execution? _sometimes_
  - Development time? _fast_
  - Using underlying libraries? _super fast!_
  - Cython? _fast, compiles to C_
  - PyPy? _faster than normal Python_
  - Realtime embedded applications? _slow!_
  - HFT? _wayyyyyy to slow_

What can I do with Python?

- Literally everything.
  - OS actions (*os, sys, shutil, fabric*)
  - Serialization, parsing (*pandas, yaml, json*)
  - Machine learning (*scikit, tensorflow*)
  - Numerics & plotting (*numpy, matplotlib*)
  - Data analytics (*hadoop, spark*)
  - Cloud automation (*boto, openstack*)

What's the future of Python?

- ◎ Very bright future in Python 3
- ◎ Some folks switching to Go, but only certain use cases
- ◎ Even today, the most foward-thinking and advanced scientific libraries are being designed Python-first or with Python as the first bindings (tensorflow, scikit, most all APIs)
- ◎ Python will always be the higher-level glue language
- ◎ Bet on C/C++, Python, Java (in my opinion)