

# Python Primer 1

## Language Basics

# What is Python

- Scripting language
- Developed by Guido van Rossum in the 90s
- Focus on code readability and ease of programming
- Two major versions:
  - 2.7
  - 3.xx
- Talk about 2.7 since that is what I know

# Why should I use it?

- Freely available
- Cross platform
- Open source
- Easy to learn and fast to code in
- Full language (GUI, web, networking, process control)
- Tons of libraries and bindings
  - More on this later

# What is it good for?

- Scripting
- Data analysis
  - Scipy and Numpy (arrays, linear algebra, statistics)
- Code prototyping
- Plotting in 2D and 3D (including movie generation)
  - Matplotlib
- Code autogeneration
  - Cheetah

# What is it not good for?

- Anything that needs to be parallel and/or highly scalable
  - This is starting to change
- Looping over large data structures
  - Use vectorized libraries such as numpy
    - Focus of next week's talk
- Anything where performance is the central concern

# Language Basics

- Interpreted language
  - Can also be compiled into stand-alone programs
- Dynamically typed language
  - Type errors give run-time warning/error
- Syntax similar to C
- Indentation delineates code blocks
- Classes, errors, I/O, exceptions, modules, standard library, etc.

# Data Types

- Has ints, float, booleans
- Strings
  - Immutable
- Main Complex Data Structures
  - Tuples
  - Lists
  - Dictionaries

# Lists

- General arrays
- Indexed starting from 0
- Can hold ANY items in any combination
- Lots of useful built-in functions
  - Append, pop, remove, index, etc.
- Can be iterated over
- Can be sliced, concatenated, etc.



# Tuples

- Immutable lists
- Useful for returning multiple values from a function
- Web tutorial break!

# Dictionaries

- Hash tables
  - Sets of key:value pairs
- Indexed by keys
  - Any immutable type (numbers, strings, tuples)
- Web tutorial break!

# Control Structures

- Main ones:
  - Branching: if, else, elif
  - Loops: for, while
- Modifiers: Break/Continue

# Loops

- For
  - Loop over numerical ranges
    - Use xrange/range
  - Looping over iterable structures
    - Lists, tuples, dictionaries (keys, values, key-value pairs)
- While
- Break
- Continue

# Branching

- If
- Else
- Elif
- Web tutorial break!

# Functions

- Def
- Pass by reference
- Return
- Pass
- Web tutorial break!

# Resources

- <http://docs.python.org/tutorial/>
- Recommend learning Python from websites- books just end up outdated, are hard to copy code from, etc.
- Me!

# Future Talks

- Talk 2 will focus on Numpy/Scipy/Matplotlib
  - Workhorse of “useful” python for atmospheric science
- Talk 3 will focus on other useful modules and libraries
  - Config file parsing, system calls, templating, etc.
- What do you guys want to hear about?