

Reuven M. Lerner, PhD • reuven@lerner.co.il http://lerner.co.il/

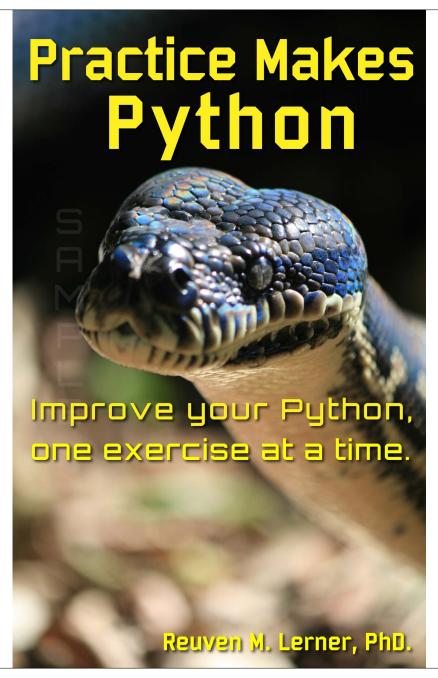
Who am I?

- Programmer, consultant, developer
- Long-time Python (+Ruby/Git/PostgreSQL) user
- Linux Journal columnist
- PhD in Learning Sciences from Northwestern

2

My stuff

- Newsletter: http://lerner.co.il/newsletter
- Blog: http://blog.lerner.co.il/
- Daily Tech Video: http://dailytechvideo.com/
 - Or @DailyTechVideo on Twitter
- Mandarin Weekly: http://MandarinWeekly.com
 - Or @MandarinWeekly on Twitter





Course objectives

- Understand programming in Python
- Familiarity with data types
- Object-oriented and functional styles
- Know enough to learn more by yourself

6

Python history

- Guido van Rossum (BDFL)
- Language around since 1991
- Latest version is 3.5.0
 - We're using 2.7.11
- Name from Monty Python
 - But snakes are common icons



What version?

\$ python --version

Python 2.7.11

\$ python3 --version

Python 3.5.1

Changes to Python

- Anyone can suggest a change!
- PEP Python Enhancement Proposal
- http://www.python.org/dev/peps/

Implementations

- Python (i.e., CPython): http://python.org
- PyPy: http://pypy.org
- Jython: http://jython.org/
- IronPython: http://ironpython.net/

Why Python?

- Powerful a real language, with objects
- Dynamic, so it's easy to jump in
- Easy to read and maintainable
- Many packages with useful extensions
- Large user community

Online resources

- http://docs.python.org/reference/
- Stack Overflow (of course)
- http://pythonweekly.com/
- http://pycoders.com/

"Hello world"

• Interactive Python shell:

```
print 'Hello world'
```

We can also get user input:

```
name = raw_input('Your name: ')
print 'Hello, ' + name
```

Interactive Python

- A great part of Python programming!
 - Saves on debugging (and your sanity)
 - We will use it extensively
- Any Python expression is valid
- Definitions disappear upon exit

IPython

- You can download it from <u>ipython.org</u>
- Works on all platforms
- Provides a superior interactive shell
- Provides IPython Notebook

IPython Notebook

• Installing:

pip install 'ipython[notebook]'

• Running:

ipython

Running IPython Notebook

ipython notebook

Variables

- name in our program was a variable
 - Strict, dynamic typing
- Type is for data, not variables
- Variable names begin with letters or _
- Delete variables with del

More sophistication

```
name = raw_input("Enter name: ")
if name == 'Reuven':
    print 'Hello, master!'
    print "I missed you!"
else:
    print 'Hello, ' + name + '.'
                    19
```

Whitespace

- Anything you want, but be consistent
- Seems weird, but easy to get used to
- No more missing (or extra) braces

End of line

- Statements end at the end of a line
- You can optionally put a semicolon (;) at end
 - Don't do it!
 - Also: Avoid more than one command per line
- If you open parentheses, then the line extends until the parentheses end

Comments

- Begin with #
- Extend to the end of the line
- Use them! (Or not...)

22

Conditionals

```
#!/usr/bin/env python
name = raw_input("Enter your name: ")
if name == 'Reuven':
    print 'Hello, master!'
    print "I missed you!"
elif name == 'Shikma':
    print 'Hello, other master!'
else:
    print 'Hello, ' + name + '.'
                               23
```

One-line conditionals

You can do this, too:

```
x = 5
```

'five' if x == 5 else 'not five'

- It must have an "else," and be an expression
- Not considered very readable or Pythonic

Equality

- Check value equality with == and !=
- Check object equality with "is"

```
>>> a = 'hello'
>>> b = a
>>> a == b

True
>>> a is b

True
```

Boolean operators

• Use words, not symbols: and, or, not

```
if (first == 'Reuven' and
    last == 'Lerner'):
    print "Total match!"

if first != 'Reuven':
    print "Who are you?"
```

Examples of and, or

```
\Rightarrow\Rightarrow x = 'a'
\Rightarrow\Rightarrow y = 'b'
\Rightarrow\Rightarrow if x == 'a' and y == 'b':
          print "They're both what I want"
They're both what I want
if x == 'q' or y == 'b':
     print "One is what I want"
One is what I want
```

not

• A bit different than C-like languages

```
x = False
if not x:
  print "x is false!"
else:
  print "x is true!"
```

Python in files

- Programs go in files with a .py extension
 - .pyw, .pyc, .pyd, .pyo are also for Python, but not for us to worry about
- Use any text editor
 - IDLE, Emacs, vi it's up to you (but I really like Emacs)
- #!/usr/bin/env python on 1st line

PyCharmCE!

- PyCharm is a commercial IDE
- A community edition (PyCharmCE) was released as an open-source project in September 2013
- Try it out I've heard good things
- http://www.jetbrains.com/pycharm/

Python in files

```
#!/usr/bin/env python
name = raw_input("Enter your name: ")
if name == 'Reuven':
    print 'Hello, master!'
else:
    print 'Hello, ' + name + '.'
```