



Unit 1

Basic Python programs, functions

Special thanks to Roy McElmurry, John Kurkowski, Scott Shawcroft, Ryan Tucker, Paul Beck for their work.

Except where otherwise noted, this work is licensed under:

<http://creativecommons.org/licenses/by-nc-sa/3.0>

Python!

- Created in 1991 by Guido van Rossum (now at Google)
 - Named for Monty Python
- Useful as a **scripting language**
 - **script**: A small program meant for one-time use
 - Targeted towards small to medium sized projects
- Used by:
 - Google, Yahoo!, Youtube
 - Many Linux distributions
 - Games and apps (e.g. Eve Online)



Installing Python

Windows:

- Download Python from <http://www.python.org>
- Install Python.
- Run **Idle** from the Start Menu.

Note: For step by step installation instructions, see the course web site.

Mac OS X:

- Python is already installed.
- Open a terminal and run `python` or run Idle from Finder.

Linux:

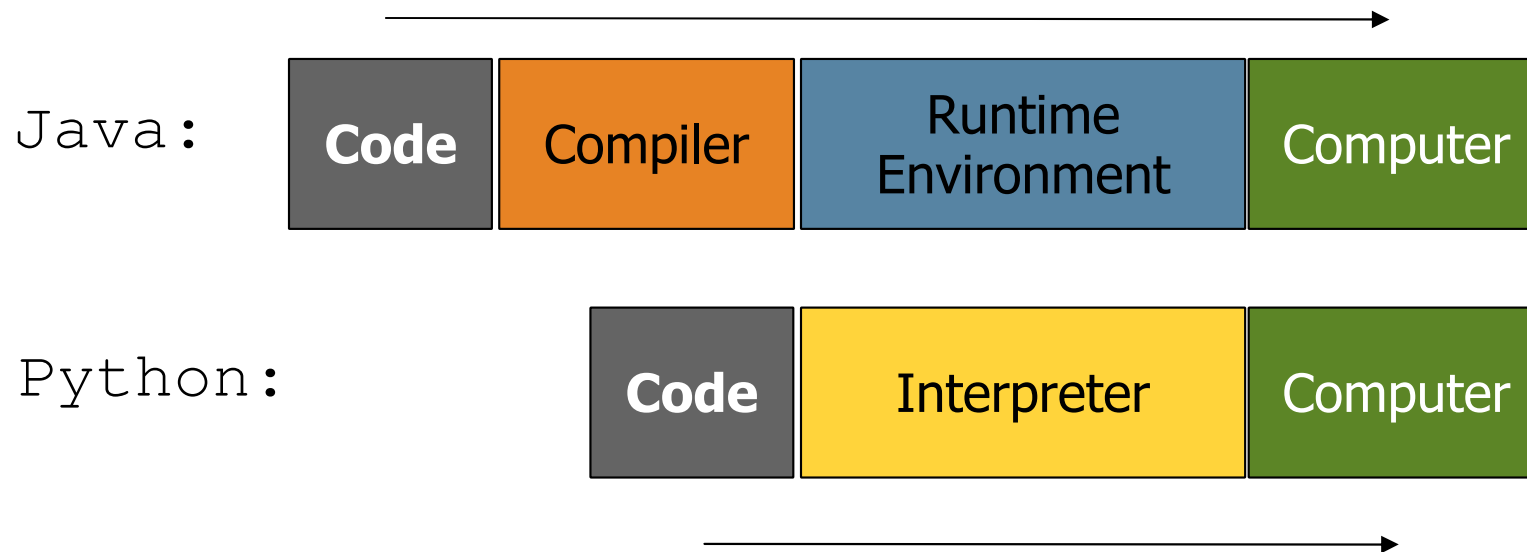
- Chances are you already have Python installed. To check, run `python` from the terminal.
- If not, install from your distribution's package system.



Interpreted Languages

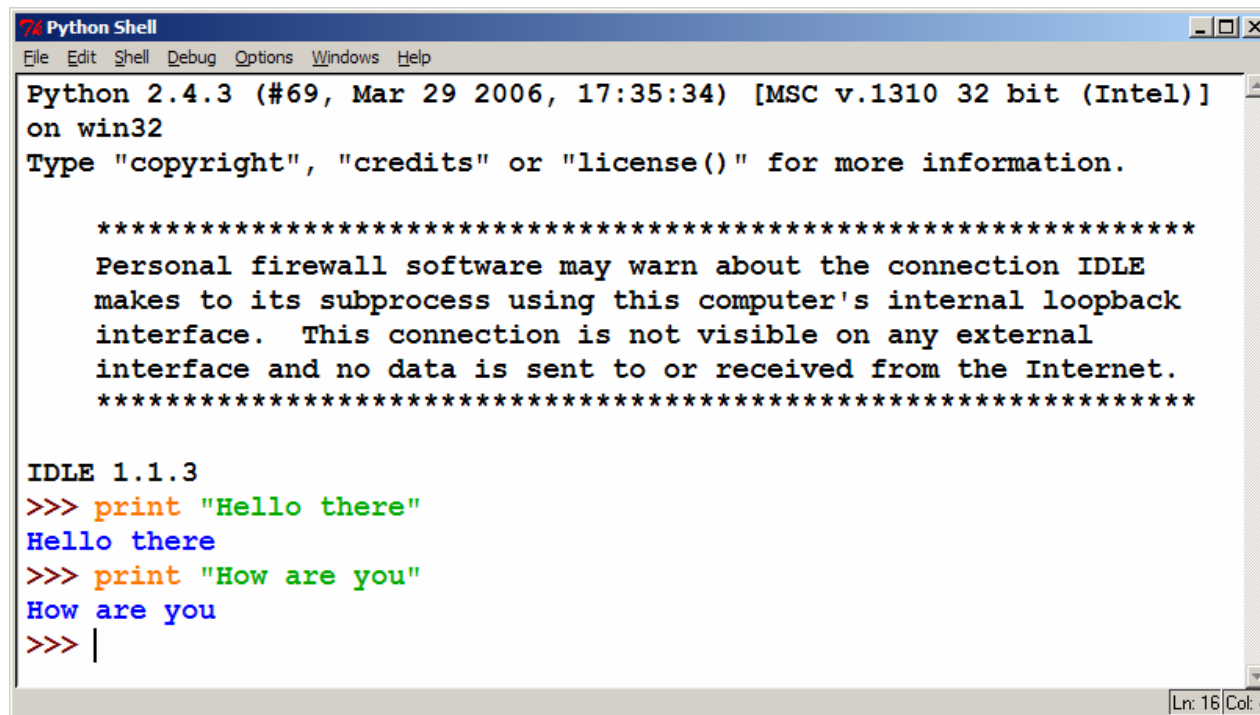
- **interpreted**

- Not compiled like Java
- Code is written and then directly executed by an **interpreter**
- Type commands into interpreter and see immediate results



The Python Interpreter

- Allows you to type commands one-at-a-time and see results
- A great way to explore Python's syntax
 - Repeat previous command: Alt+P



```
Python Shell
File Edit Shell Debug Options Windows Help
Python 2.4.3 (#69, Mar 29 2006, 17:35:34) [MSC v.1310 32 bit (Intel)]
on win32
Type "copyright", "credits" or "license()" for more information.

*****
Personal firewall software may warn about the connection IDLE
makes to its subprocess using this computer's internal loopback
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****

IDLE 1.1.3
>>> print "Hello there"
Hello there
>>> print "How are you"
How are you
>>> |
```



Our First Python Program

- Python does not have a `main` method like Java
 - The program's main code is just written directly in the file
- Python statements do not end with semicolons

hello.py

```
1 print("Hello, world!")
```



The `print` Statement

```
print("text")
```

```
print()           (a blank line)
```

- Escape sequences such as `\` are the same as in Java
- Strings can also start/end with `'`

swallows.py

```
1 print("Hello, world!")
2 print()
3 print("Suppose two swallows \"carry\" it together.")
4 print('African or "European" swallows?')
```



Comments

- Syntax:
 # **comment text (one line)**

swallows2.py

```
1 # Suzy Student, CSE 142, Fall 2097
2 # This program prints important messages.
3 print("Hello, world!")
4 print()                # blank line
5 print("Suppose two swallows \"carry\" it together.")
6 print('African or "European" swallows?')
```



Functions

- **Function:** Equivalent to a static method in Java.

- Syntax:

```
def name () :  
    statement  
    statement  
    . . .  
    statement
```

hello2.py

```
1  # Prints a helpful message.  
2  def hello():  
3      print("Hello, world!")  
4  
5  # main (calls hello twice)  
6  hello()  
7  hello()
```

- Must be declared above the 'main' code
- Statements inside the function must be indented



Whitespace Significance

- Python uses indentation to indicate blocks, instead of { }
 - Makes the code simpler and more readable
 - In Java, indenting is optional. In Python, you **must** indent.

hello3.py

```
1  # Prints a helpful message.
2  def hello():
3  → print("Hello, world!")
4  → print("How are you?")
5
6  # main (calls hello twice)
7  hello()
8  hello()
```



Exercise

- Rewrite the Figures lecture program in Python. Its output:

