



# Learning to Program With Python – Part 1



## Computers, Languages, and Tools: Saying “Hello”

*Based on the book:*

Snake Wrangling for Kids, Learning to Program with Python  
by Jason R. Briggs

*(Version 0.7.7-python2.7, modified by SJL)*

*Presented by*

*Steve Arnold, Principal Scientist VCT Labs*

*Stephanie Lockwood-Childs, President VCT Labs*

*(we are also open source Gentoo Linux / Yocto developers)*



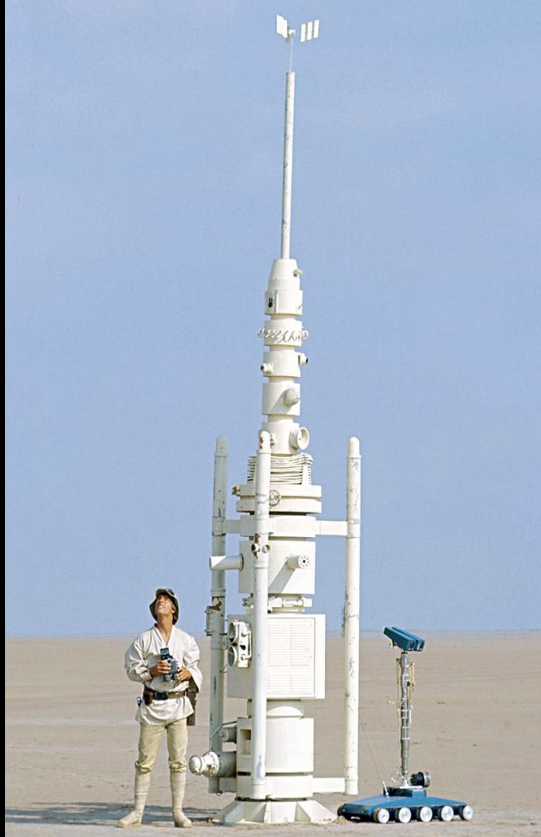
# Computers and Languages



- What is a computer?
  - Abacus? Calculator? Watch? Phone? Tablet?
  - Game Console? Laptop? Desktop? HAL 9000?
- Languages
  - Humans and Computers
  - Binary language of moisture “vaporators”
  - Number systems and representations
  - Bits and bytes, text and binary
  - Machine and Programming Languages
- Computers and Intelligence
  - Can animals think?
  - Can computers think?

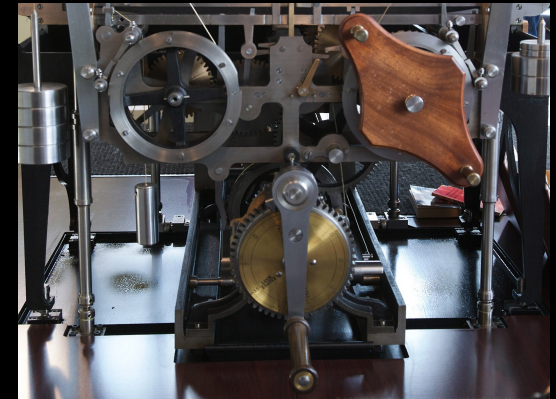
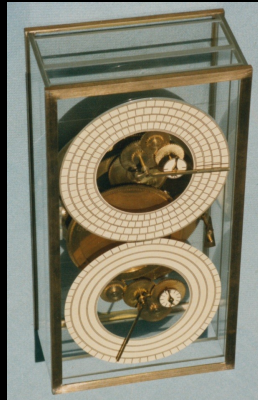
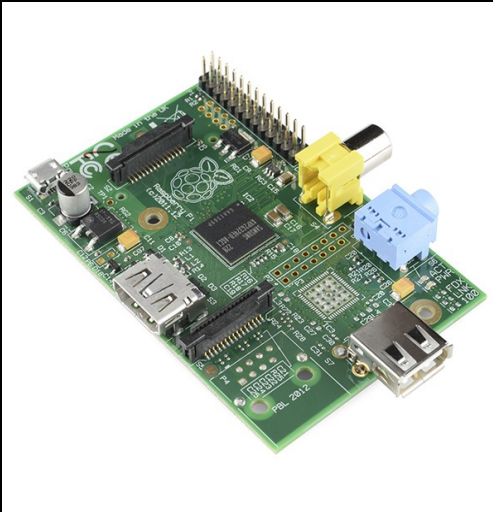
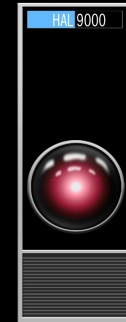


# Moisture Vaporators



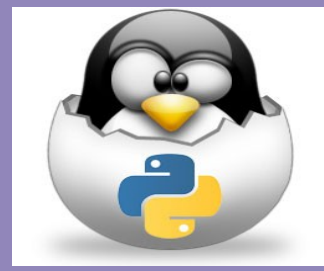


# Which of these is a computer?





# Computers and Languages cont.



- Files and Encodings
  - ASCII, Binary, Octal, Decimal, Hexadecimal
- Computer Languages and “Flavors”
  - Machine and Assembly Languages
    - Processors and Architecture
  - High Level Languages
    - Compiled vs. Interpreted
    - 1<sup>st</sup> “modern” language: FORTRAN (1954)
    - ALGOL, LISP (1958)
    - Then A, B, C, D, C++, C#, perl, Java, SmallTalk
    - COBOL, SNOBOL, ruby, lua, vala, scala, clojure...
  - “Human” Programming Languages
    - Ada (1<sup>st</sup> standard published 1983)
    - Python (1<sup>st</sup> implementation 1989)





# Programming Language Tools



- The Language Files
  - Compiler/interpreter, libraries, other components
  - /us/lib/python/... (core python, site-packages, etc)
- Programmer's Editor
  - Text editor with special features for software development (not a word processor)
  - Examples: **gedit**, nano, kate, gvim, emacs, nedit, scite
- Shell / Terminal Emulator
  - A place to type commands, like a DOS prompt only better (some shells are special, eg: python, pycrust)
  - Open **Terminal Emulator**, type "python" then Ctrl-D
- Integrated Development Environment (IDE)
  - Like a fancy editor, with project interface, object browsing
  - Examples: **geany**, code-blocks, eclipse, anjuta



# Writing Your First Program in Python



- Open Terminal Emulator (bash shell)

- At the “\$” prompt, type:

```
$ python <enter>
```

- At the “>>>” prompt, type:

```
>>> print ("Hello, world!") <enter>
```

```
Hello, world!
```

- Now write it as a Python program

- Open your editor and type:

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

- Save the file as “hello.py”
  - At the “\$” prompt, type”

```
$ python hello.py
```

- Congratulations, you just wrote a Python program!



# What Does /. Say?



What programming language will earn you the best salary over time?

- According to Burning Glass and a Brookings Institution economist, Ruby on Rails, Objective-C, and **Python** are all programming skills that will earn you more than \$100,000 per year.
- Tech-industry analyst firm RedMonk's list of most-used languages: Java/JavaScript, PHP, **Python**, C#, and C++/Ruby
- **Python** was the one programming language to appear on Dice's recent list of the fastest-growing tech skills
- **Python** is a staple language in college-level computer-science courses, and has repeatedly topped the lists of popular programming languages as compiled by TIOBE Software and others.

<http://developers.slashdot.org/story/14/12/03/177214/which-programming-language-pays-the-best-probably-python>





This work is an original work by Stephen Arnold  
<[stephen.arnold@acm.org](mailto:stephen.arnold@acm.org)>

<<http://www.vctlabs.com>>

Portions copyright 2014 Stephen L Arnold. Some rights reserved.

The Gentoo Linux logo is Copyright 2006 Gentoo Foundation, used with permission.



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike License. To view a copy of this license, visit <<http://creativecommons.org/licenses/by-nc-sa/1.0>> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

Please contact Stephen Arnold <[stephen.arnold@acm.org](mailto:stephen.arnold@acm.org)> for commercial uses of this work.