

Introduction to Python 0 – How to run Python

Chang Y. Chung

Office of Population Research

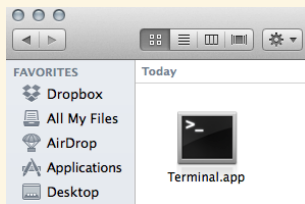
May 2015

Schedule

from	to	topic	note
9:30am	- 10:30am	How to run Python; Comments; Variables; Integers and Floating point numbers; Strings; None; Operators	
10:30am	- 11:00am	Break	
11:00am	- noon	Flow Control and Compound statements; File I/O; Defining and Calling a Function; Local and Global Variables; Importing a module	
noon	- 1:30pm	Lunch break	#242
1:30pm	- 2:30pm	List; Dictionary; Data Structure	
2:30pm	- 3:00pm	Break and Optional Evaluation	
3:00pm	- 4:00pm	Tuples; Class; Exception Handling; Regular Expression	

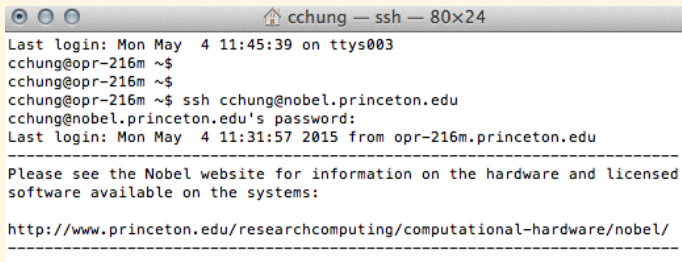
Running Python Read-Eval-Print Loop 1

- ▶ For those who are using UNIX-like systems, including Apple Mac, Nobel, or Adroit.
- ▶ On Mac, open the terminal window.



Running Python Read-Eval-Print Loop 2

- ▶ Skip this step, if you are running locally installed Python. Continue, if you are to run Python on Nobel or Adroit.
- ▶ Register for an account at: <http://www.princeton.edu/researchcomputing/computational-hardware/>
- ▶ Secure Shell (ssh) into Nobel (or Adroit).



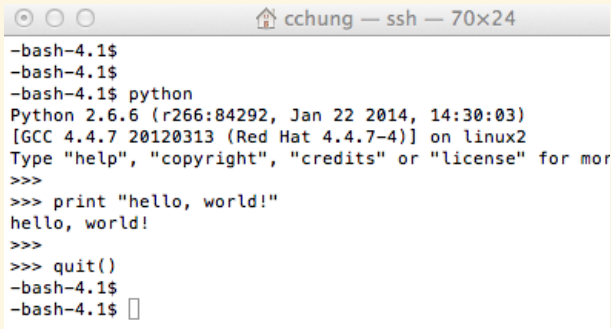
```
cchung — ssh — 80x24
Last login: Mon May  4 11:45:39 on ttys003
cchung@opr-216m ~$
cchung@opr-216m ~$
cchung@opr-216m ~$ ssh chung@nobel.princeton.edu
cchung@nobel.princeton.edu's password:
Last login: Mon May  4 11:31:57 2015 from opr-216m.princeton.edu

-----
Please see the Nobel website for information on the hardware and licensed
software available on the systems:

http://www.princeton.edu/researchcomputing/computational-hardware/nobel/
-----
```

Running Python Read-Eval-Print Loop 3

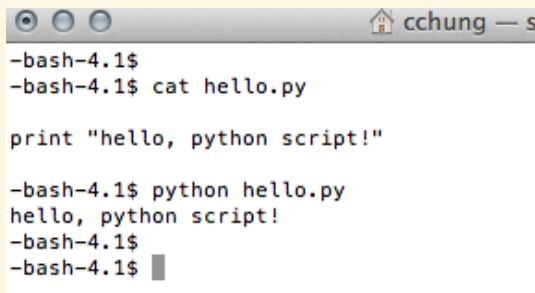
- ▶ Run Python REPL at the shell prompt.

A terminal window titled 'cchung — ssh — 70x24' showing a sequence of commands and their outputs. The user enters 'python' at the shell prompt, which starts the Python 2.6.6 interpreter. The user then enters 'print "hello, world!"' and 'quit()' to demonstrate the Read-Eval-Print Loop.

```
-bash-4.1$  
-bash-4.1$  
-bash-4.1$ python  
Python 2.6.6 (r266:84292, Jan 22 2014, 14:30:03)  
[GCC 4.4.7 20120313 (Red Hat 4.4.7-4)] on linux2  
Type "help", "copyright", "credits" or "license" for mor  
>>>  
>>> print "hello, world!"  
hello, world!  
>>>  
>>> quit()  
-bash-4.1$  
-bash-4.1$
```

Running a Python Script File (.py)

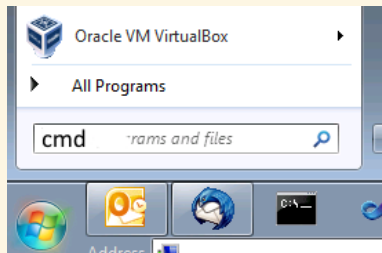
- ▶ Create a python script file using a text editor (nano, vim, emacs, ...).
- ▶ Type "python" followed by the script file name at the shell prompt.

A terminal window with a title bar containing a home icon, the name 'cchung', and a partial window title '— s'. The terminal shows a sequence of commands and their outputs:

```
-bash-4.1$  
-bash-4.1$ cat hello.py  
  
print "hello, python script!"  
  
-bash-4.1$ python hello.py  
hello, python script!  
-bash-4.1$  
-bash-4.1$ █
```

Running a Local Python REPL on Windows 1

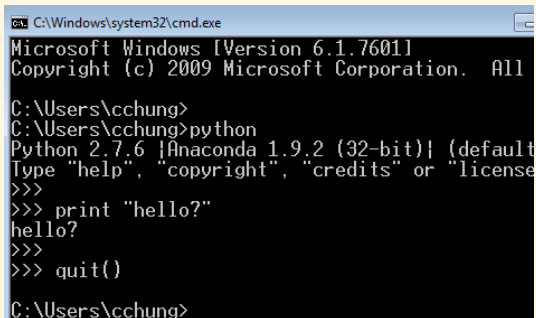
- ▶ (MS Windows before 8) Open a cmd window.



- ▶ (Ms Windows 8 and 8.1) Swipe up to show the Apps screen. Swipe or scroll to the right and click on the Command Prompt under the Windows System section.

Running a Local Python REPL on Windows 2

- ▶ Start Python REPL.



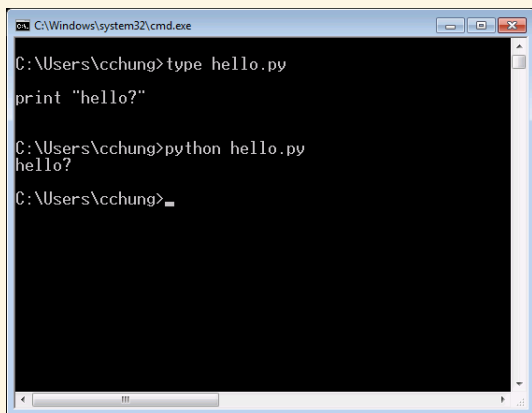
```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\cchung>
C:\Users\cchung>python
Python 2.7.6 |Anaconda 1.9.2 (32-bit)| (default)
Type "help", "copyright", "credits" or "license()" for more
>>>
>>> print "hello?"
hello?
>>>
>>> quit()

C:\Users\cchung>
```


Running a Python Script File (.py) on Windows

- ▶ Create a python script file using a text editor (notepad, nano, vim, emacs, ...).
- ▶ Execute python command with the script file name at the shell prompt.



```
C:\Windows\system32\cmd.exe

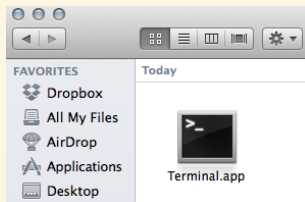
C:\Users\cchung>type hello.py
print "hello?"

C:\Users\cchung>python hello.py
hello?

C:\Users\cchung>_
```

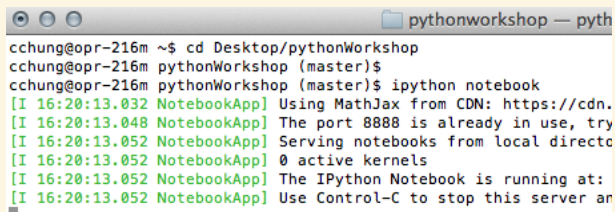
Starting IPython Notebook 1

- ▶ For those who are using Apple Mac.
- ▶ Open the terminal window.



Starting IPython Notebook 2

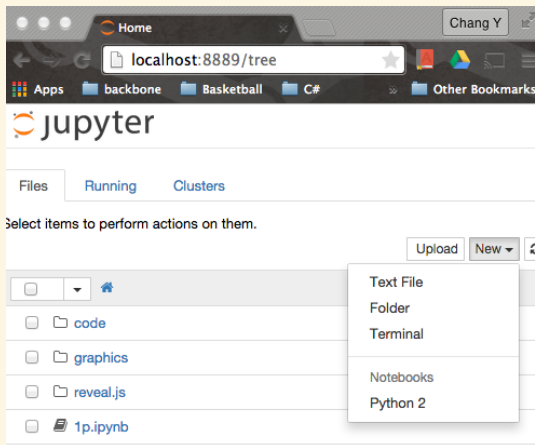
- ▶ (Optional) Change directory to the desired sub-directory.
- ▶ Execute "ipython notebook" command. The default browser should open up showing the current working directory.



```
pythonworkshop — pyth
cchung@opr-216m ~$ cd Desktop/pythonWorkshop
cchung@opr-216m pythonWorkshop (master)$
cchung@opr-216m pythonWorkshop (master)$ ipython notebook
[I 16:20:13.032 NotebookApp] Using MathJax from CDN: https://cdn.
[I 16:20:13.048 NotebookApp] The port 8888 is already in use, try
[I 16:20:13.052 NotebookApp] Serving notebooks from local directo
[I 16:20:13.052 NotebookApp] 0 active kernels
[I 16:20:13.052 NotebookApp] The IPython Notebook is running at:
[I 16:20:13.052 NotebookApp] Use Control-C to stop this server an
```

Starting IPython Notebook 3

- ▶ Either open an existing ipython notebook (.ipynb) or create a new one (click on Python 2 under New > Notebooks)



Quiz

- ▶ Print out a "HELLO" in your environment.
- ▶ Print out "HELLO" 20 times in your environment.
- ▶ Print out "HELLO" *vertically*, that is, the printed output should look like below (line numbers are not required).

```
1 H  
2 E  
3 L  
4 L  
5 O
```