ECON 370 Quantitative Economics with Python

Lecture 4: Python Fundamentals (Part 2)

Zhen Huo [zh335@nyu.edu]
Matthew McKay [mm8390@nyu.edu]
Thomas Sargent [thomas.sargent@nyu.edu]

Spring 2016

Agenda ...

Part 1

1. Terminal
   - OS X Terminal
   - Windows Powershell
2. GitHub
   - Review Course Page
   - GitHub Notifications
   - Markdown Resources
   - LaTex Resources
3. Reading Material - `http://quant-econ.net/`
4. Assignment #1

Terminal

Review Terminal on OS X and Windows

The Terminal application allows you to access a powerful command environment.

OS X	Use Spotlight <command>+<spacebar> then type terminal + <enter>
Windows	Use Command Key and then type powershell*

*__Note:__ Powershell provides similar commands to a linux environment for Windows.

Basic Terminal Commands

There are different terminal environments but these commands are fairly similar across platforms

ls	List Files
cd	Change Directory
cp	Copy File
mv	Move File
jupyter notebook	Launch Jupyter notebook
conda update conda	Update conda package manager
conda update anaconda	Update the anaconda packages to the latest “official” version

If you used cmd you will need to use dir to list files etc.

GitHub

Course GitHub Page:

`https://github.com/mmcky/nyu-econ-370`

GitHub Notifications

If you have a GitHub account you can receive notifications when the repository is updated.

Click on Watch

Alternatively the time information gives you an indicator of when a document is revised.

I will try and keep a list of important updates in the Updates section.

Markdown Resources

Simple Markup Language for Formated Text used in Jupyter Markdown Cells

Full Specification:
`http://daringfireball.net/projects/markdown/`

GitHub Flavored Markdown: `https://help.github.com/articles/basic-writing-and-formatting-syntax/`

You can also find a sample notebook here

You can download this notebook from nbviewer using the top-right hand icon

LaTeX Math Resources

LaTeX is a typesetting language for producing scientific documents.

You might like to checkout the LaTeX Mathematics page

`https://en.wikibooks.org/wiki/LaTeX/Mathematics`

In Jupyter you can use $ <math-here> $ for inline math (i.e. in sentences).

Alternatively you can have math expressions on their own line using $$ <math-here> $$

Reading Material

These lecture notes are complementary to the Reading assignments

`http://quant-econ.net/`

Assignments

Assignment #1 is due:

Tuesday 09th February 2016 at the beginning of class.

Please bring a hard copy to submit in the box as you walk in.

Assignment #2 will be released this weekend and will be due:

Tuesday 16 February 2016

Python Fundamentals ... continued

Part 2 - Python Fundamentals

1. Review of Python Fundamentals
2. Dictionaries, Sets, and Tuples
3. Conditional Logic
4. Functions

Review of Python Fundamentals

1. Variables
2. Boolean Values
3. Numerics - Integers, Floats, Complex Numbers
4. Strings
5. Lists

Questions?

Dictionaries, Sets and Tuples

See notebook **intro-to-python.ipynb**

Conditional Logic

Using Boolean expressions to control the flow of a program

Relational Operators

```
x == y    # x is equal to y
x != y    # x is not equal to y
x > y     # x is greater than y
x < y     # x is less than y
x >= y    # x is greater than or equal to y
x <= y    # x is less than or equal to y
```

References: http://quant-econ.net/py/python_essentials.html#comparisons-and-logical-operators

Conditional Logic

Three main ways to write conditional logic expressions

```
if x > 0:
    print(”x is > 0”)
```

```
if x > 0:
    print(”x is > 0”)
else:
    print(”x is <= 0”)
```

```
if x > 0:
    print(”x is > 0”)
elif x == 0:
    print(”x is = 0”)
else:
    print(”x is < 0”)
```

Functions

We have seen a few of these already `int('2')` converts the string representation “2” to the integer 2.

Functions are useful as a collection of computations that takes input and produces some output

They take the general form:

```
def function_name(<arguments>):
    """
    Docstring
    """
    # Some Computation Goes Here
    return something_useful
```

Functions

A simple example:

```
def hello(name):
    """
    This function returns a greeting for a person given a name

    Parameters
    ----------
    name    str
            Specify a name for the greeting

    Returns
    -------
    greeting    str
                Customised Greeting
    """
    return "Hello! %s"%name     #This is a pretty silly function
```

Additional Resources:

1. `http://quant-econ.net/py/python_essentials.html`
2. “Think Python”, Allen B. Downey, Oreilly Media