# dizon_ipythonexercise_part3

February 12, 2015

## 0.1 James Nikko U. Dizon - Ipython Part 3

Introduction 1. The Ipython notebook contains a web-based interactive interpreter as well as notebook documents that represent all content visible in the browser interpreter.

Main Features of the Web Application 1. In-browser editing for code, indentation, tab completion, automatic highlighting of syntax, and introspection. Code can be executed from the browser and results are attached to the same code blocks that generated them. Computational results generated using the matplotlib library can be included inline, as well as HTML files, LaTeX, PNG, SVG images, as well as Youtube videos may be embedded.

2.

```
In [ ]: from IPython.display import YouTubeVideo
        YouTubeVideo('VaV1OVNZCLA')
```

Notebook Documents

1. JSON files are Javascript Object Notation files; they store information in an organized manner.
2. Yes, they can be version-controlled, because they are editable files.
3. nbviewer (ipython notebook viewer) loads the notebook URL and renders it as a static web page; think of it as nbconvert for web.

Notebook Server 1. ipython notebook http://localhost 8888 2. –port 3. –no-browser

Creating a new Notebook 1. Click on the new notebook button.

Opening Notebooks 1. A kernel is a program that runs and introspects a user's code. For example, Ipython includes a python kernel. 2. The shutdown button is for notebooks whose kernel is active. On the other hand, the delete button is for notebooks without an active kernel. 3.

```
In [ ]: %connect_info
```

4. ipython qtconsole –existing cee0aaa0

Notebook User Interface

1. Notebook name - the file name, as well as the name of the notebook displayed on the upper part of the page. Menu bar - presents options for manipulating the notebook. Toolbar - shows icons that represent a quick way of performing operations in the notebook. Code cell - the default cell type initialized in the notebook.

Structure of a notebook document 1. A cell is a multi-line text input field. 2. You execute a cell by using the Play button or the Shift+Enter shortcut. 3. Code cells - where code is entered (with full syntax highlighting and tab completion); markdown cells - allows richly formatted text; raw cells - allows unformatted text; heading cells - used for entering headings. 4.

```
In [ ]: from IPython.display import Image
        i = Image(filename='p3-img1.png')
        i
```

```
In [ ]: from IPython.display import Image
        i = Image(filename='p3-img2.png')
        i
```

Basic Workflow 1. Kernel -> Interrupt ; Ctrl-M I 2. Kernel -> Restart ; Ctrl-M

Keyboard shortcuts 1. Shift-Enter: Run cell - executes the code inside the cell and moves to the next cell. Ctrl-Enter: Run cell in place - executes the code similarly to the terminal. Cursor remains inside the cell. Alt-Enter: Run cell, insert below - Runs the cell and inserts a new cell below Esc: command mode Enter: inserts a new line

Plotting 1. %matplotlib 2. The matplotlib backend 3. A user interface for matplotlib 4. The inline backend

Configuring the Ipython Notebook 1. ipython locate 2. ipython profile create

Converting Notebooks to Other Formats 1. ipython nbconvert –to FORMAT notebook.ipynb 2. HTML 3. latex, slideshow, markdown, restructuredtext, pythonscript

Running a Notebook Server 1. Set the NotebookApp.password configurable 2. Ipython.lib.security.passwd(): 3. By setting NotebookApp.password 4. So that the password is not sent unencrypted by the user's browser. 5. openssl req -x509 -nodes -days 365 -newkey rsa:1024 -keyout mycert.pem -out mycert.pem This command writes the certificate to the same file in which it is run.

1. ipython profile create nbserver

Security in Ipython Notebooks 1. Unauthorized clients, engines and controllers. 2. The notebook server may be protected with a simple pas

```
In [ ]:
```

```
In [ ]:
```