Python, An Introduction

Lecture 02

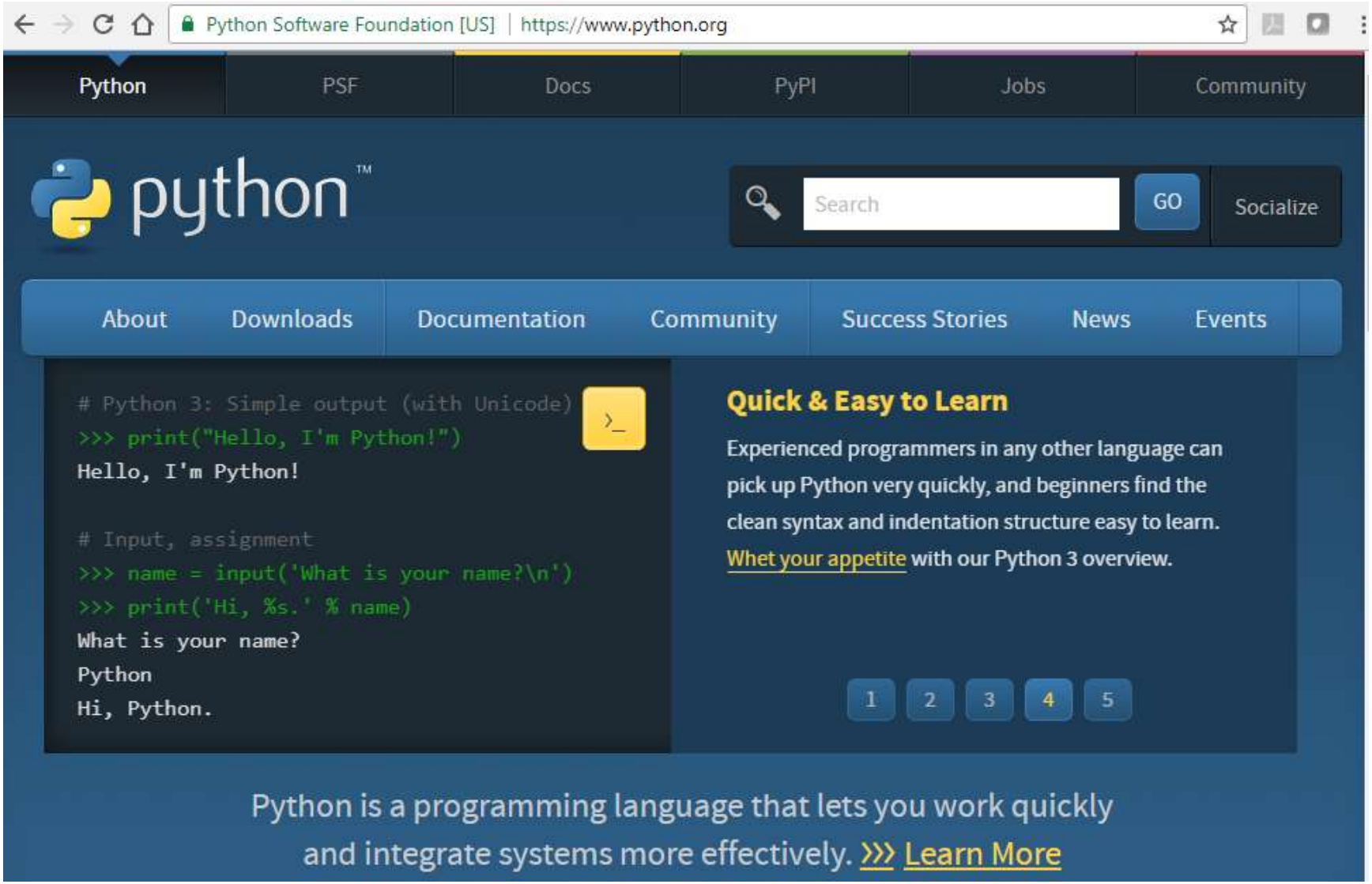
Deep Azure @ McKesson

Zoran B. Djordjević

Python

- **From Wikipedia. Org: Python** is a widely used high-level language for general-purpose programming, created by Guido van Rossum and first released in 1991.
- An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java.
- The language provides constructs intended to enable writing clear programs on both a small and large scale.
- Python features a dynamic type system and automatic memory management and supports multiple programming paradigms, including object oriented, imperative, functional programming, and procedural styles. It has a large and comprehensive standard library.
- Python interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations.
- CPython is managed by the non-profit Python Software Foundation.

http://www.python.org



So What

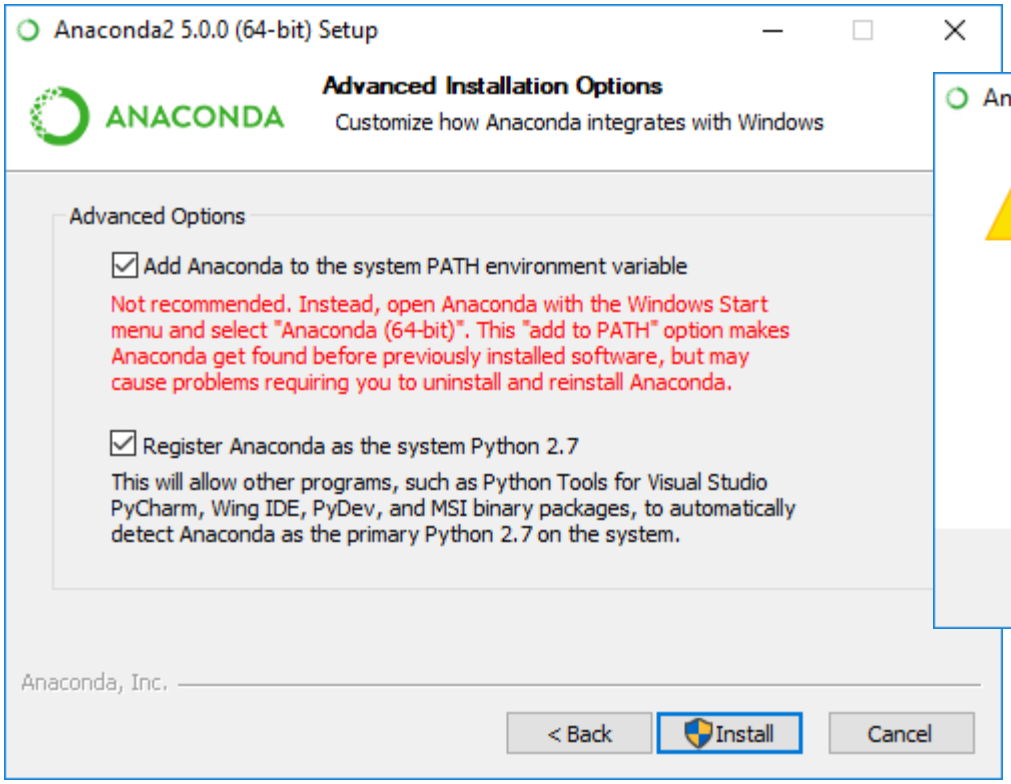
- If you are a Java, C# or C++ developer you can do practically everything in your favorite language others could do in Python.
- Java, C# or C++ have rich libraries for almost everything that Python does and in some areas perhaps richer.
- Unfortunately, Python is becoming the major language of Machine Learning and Artificial Intelligence. Many more books are published in Python on those subjects than in the older languages. Why is that?
- Python's syntax is more efficient. Python code is much more concise (2-3 times fewer lines of code) than in Java.
- Also, you do not have to use semi-colons (";") to end your statements. They advertise that feature a lot. It must be important.
- Hopefully, once you learn Python, you will be more productive. You should take a chance.
- For your PC-s we will use so called Anaconda distribution maintained by Anaconda. They used to call themselves Continuum Analytics. Apparently, that is a very large , comprehensive collection of most or many scientific, mathematical and machine learning packages.
- There are two active branches of Python, 2.7 and 3.6. We will stick with 2.7 for a while. Eventually, we will learn to operate both on the same machine.

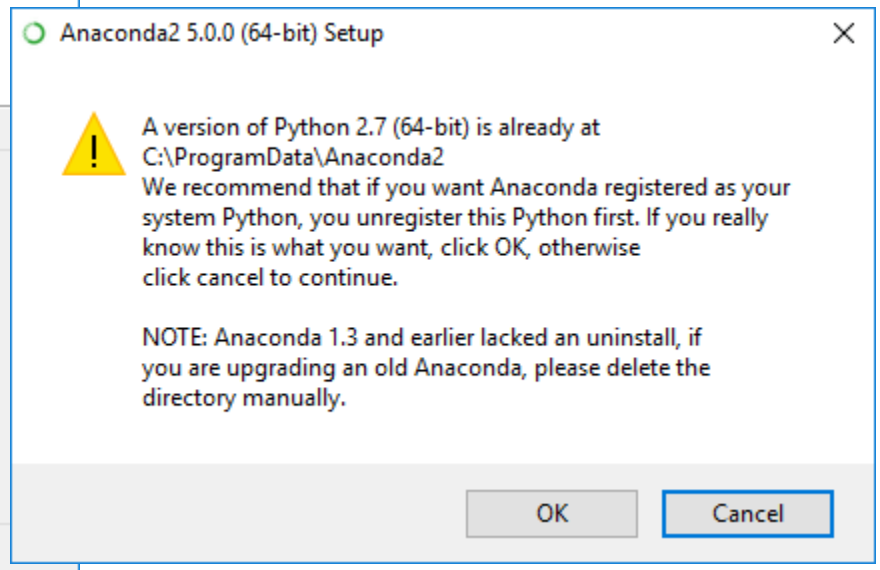
https://www.anaconda.com/download/

- Under Python 2.7 version, choose 64-Bit graphical Installer
- You will download: Anaconda2-5.0.0-Windows-x86_64.exe. Run it as an administrator.
- Pay attention to which directory Anaconda is installed.

Add Anaconda to your Path, Register Anaconda

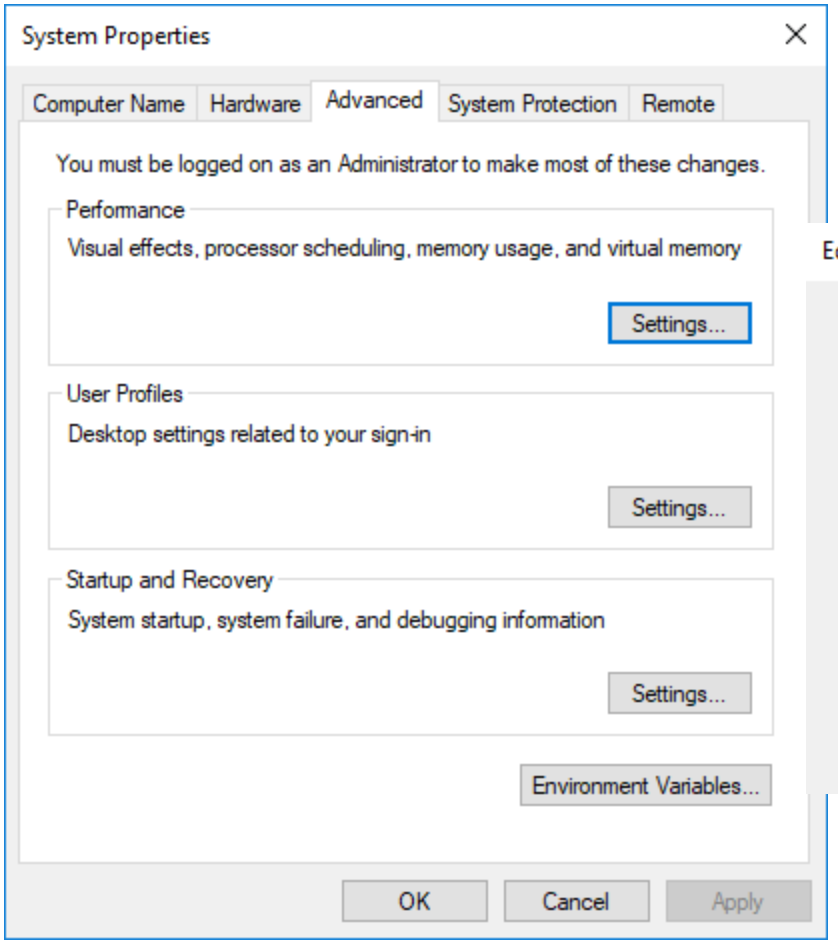
- During the installation, you always say YES. You also pay attention what is happening. Either choose or record the directory where Anaconda is installed.
- Add Anaconda to your PATH. Select both check boxes on Advanced Installation Options, including: Register Anaconda as you Python 2.7. Click `OK`, then `Install`



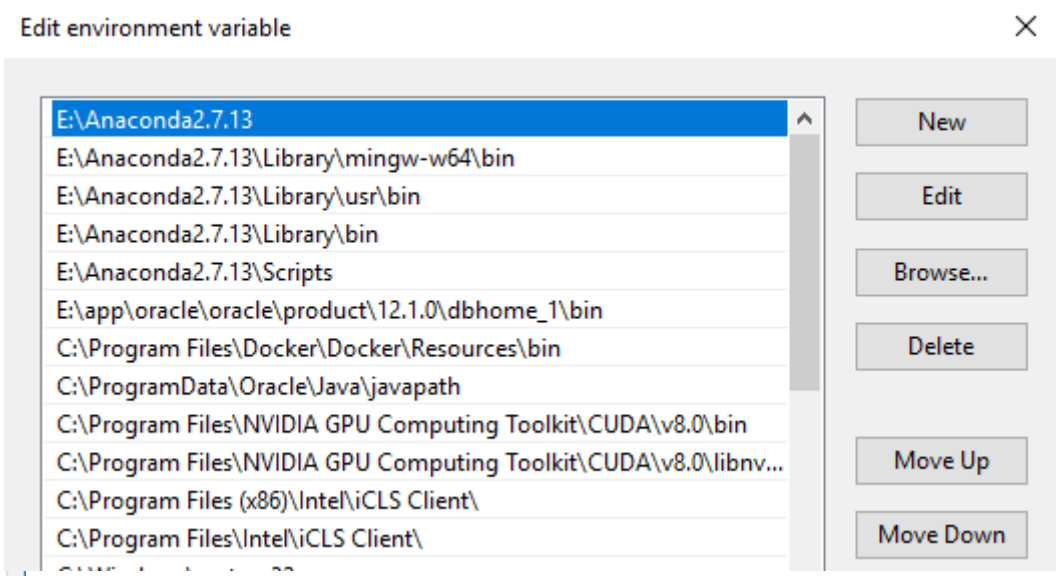


Check Your PATH

- In the bottom leftmost search field on your PC, type Environment. Select "`Edit the system environment variables`".
- On `System Properties` widget, select `Environment Variables.`



- On the next screen, select `Path` in the `System Variables`, then `Edit`.
- You should see something like:



- You are done, most probably.

If your `Path` is too long

- Windows 10 tolerates Environmental Variables shorter of 2047 characters. If your new Path happens to have more, do the following:
- **Solution 1:**
  - Create a new system variable say 'NEWPATH'
  - Assign the bin directory location to 'NEWPATH'
  - Now append '**; %NEWPATH%**' to the PATH variable
  - If this still doesn't work then try to copy some part of PATH variable already existing values to the 'NEWPATH' and then append the 'NEWPATH'
- **Solution 2:**
  - Check the value of PATH variable if you can group and shorten the paths. For example,
  - C:\Program Files\Microsoft SQL Server\102\Tools\Binn\;C:\Program Files\Microsoft SQL Server\102\DTS\Bin\;
  - can be combined to
  - C:\Program Files\Microsoft SQL Server;
  - In this way you can build more space into your fixed length PATH variable and finally adjust your bin directory location into PATH.

Final Check

- Open Windows command prompt and type these four commands:

```
C:\Users\zdjor>python
Python 2.7.13 |Anaconda, Inc.| (default, Sep 19 2017, 08:25:59)
[MSC v.1500 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more
information.
>>> exit()

C:\Users\zdjor>conda --version
conda 4.3.27

C:\Users\zdjor>pip --version
pip 9.0.1 from E:\Anaconda2.7.13\lib\site-packages (python 2.7)

C:\Users\zdjor>jupyter --version
4.3.0
```

- Those are 4 important programs we will be using: `python` interpreter, `conda` management utility, `pip` installation utility and `jupyter`, software development environment.

Jupyter Notebook App

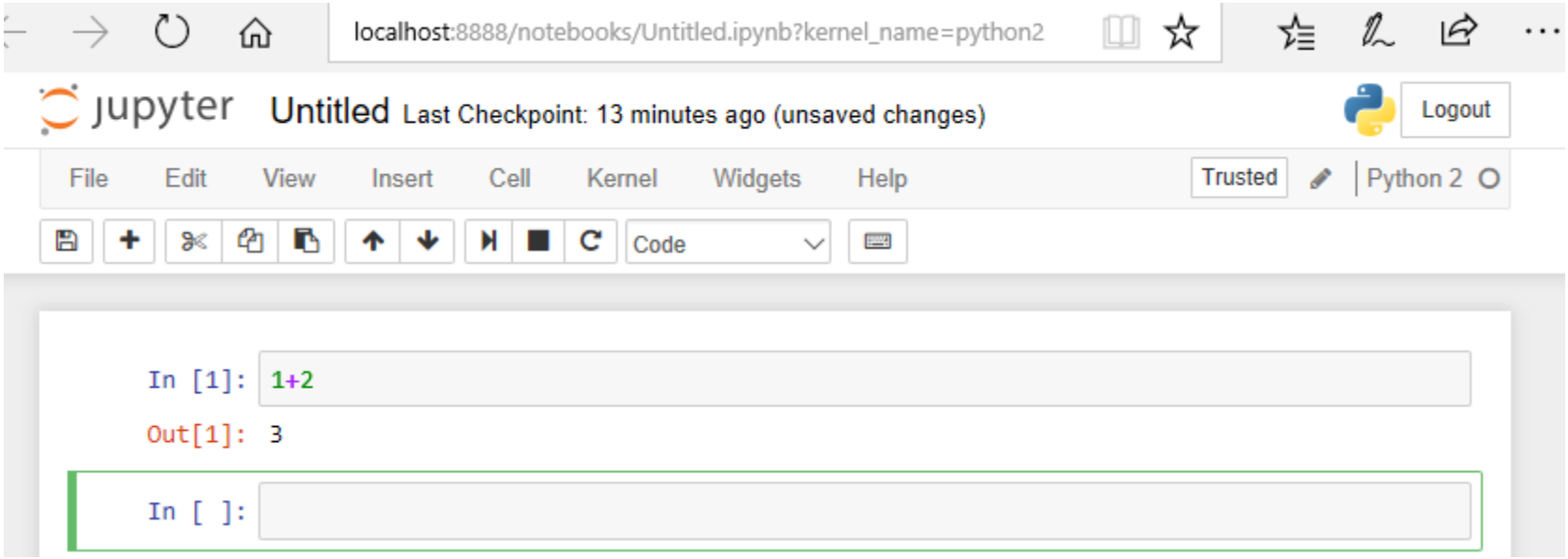
- The planet is Jupiter. Pluto is not a planet any more. Pluto is a former planet.
- The *Jupyter Notebook App* is a server-client application that allows editing and running notebook documents via a web browser. The *Jupyter Notebook App* can be executed on a local desktop requiring no internet access (as described in this document) or can be installed on a remote server and accessed through the internet.
- notebook documents (or “notebooks”, all lower case) are documents produced by the Jupyter Notebook App, which contain both computer code (e.g. python) and rich text elements (paragraph, equations, figures, links, etc...). Notebook documents are both human-readable documents containing the analysis description and the results (figures, tables, etc..) as well as executable documents which can be run to perform data analysis.
- A notebook *kernel* is a “computational engine” that executes the code contained in a notebook document. The *ipython kernel*, referenced in this guide, executes python code. Kernels for many other languages exist (official kernels).
- When you open a notebook document, the associated *kernel* is automatically launched. When the notebook is *executed* (either cell-by-cell or with menu *Cell -> Run All*), the *kernel* performs the computation and produces the results. Depending on the type of computations, the *kernel* may consume significant CPU and RAM. Note that the RAM is not released until the *kernel* is shut-down.

Open a jupyter notebook

- On your command prompt, type:

```
C:\..> jupyter notebook
```

- A Web browser will open on port 8888. If that port is taken, it will open on an adjacent port.
- On the rightmost part of the screen you see New. Select New and Python 2.
- The following screen will appear.



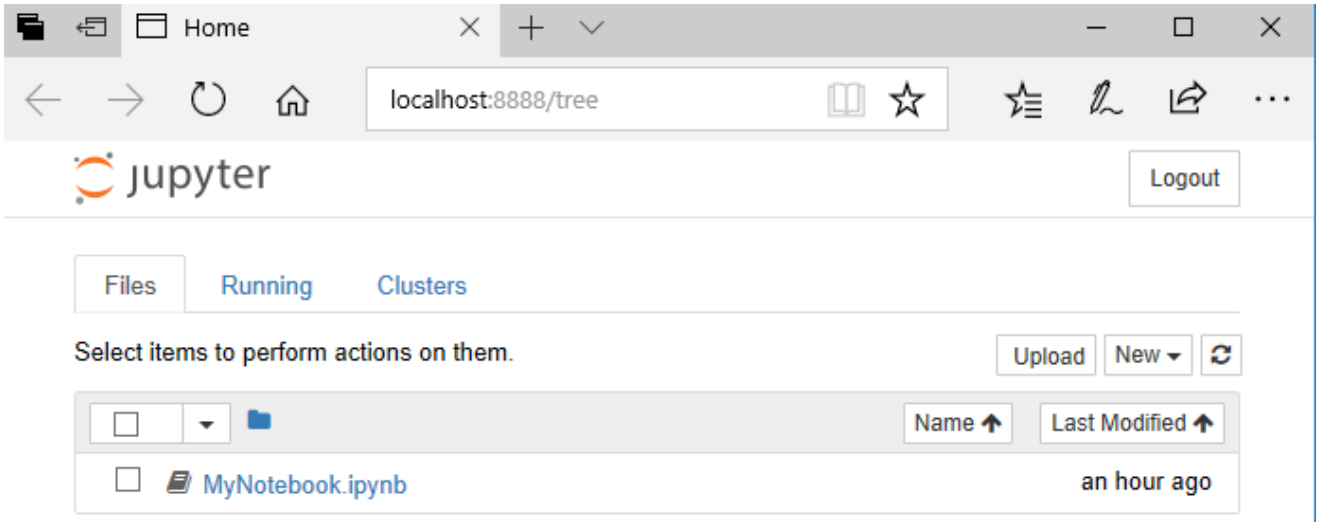
- The field `In[1]` is an input cell. Type `1 + 2` and then either Run symbol `>|` or `Ctrl-Enter` you will get an output cell with the result `3`.

Save `notebook`, reopen `notebook`.

- Your notebook is called `Untitled.ipynb`. You do not like that name.
- Select `Files > Rename`. Give your notebook a new name: `MyNotebook`.
- Then do: `Files > Save and Checkpoint`.
- Finally do: `Files > Close and Halt`. You are done.
- You can kill the browser. Kill `jupyter notebook` command on the command prompt with `Crtl C`.
- If you look at your directory, you will see file `MyNotebook.ipynb`
- If you again type:

```
C:\..> jupyter notebook
```

- You will be able to open your notebook.

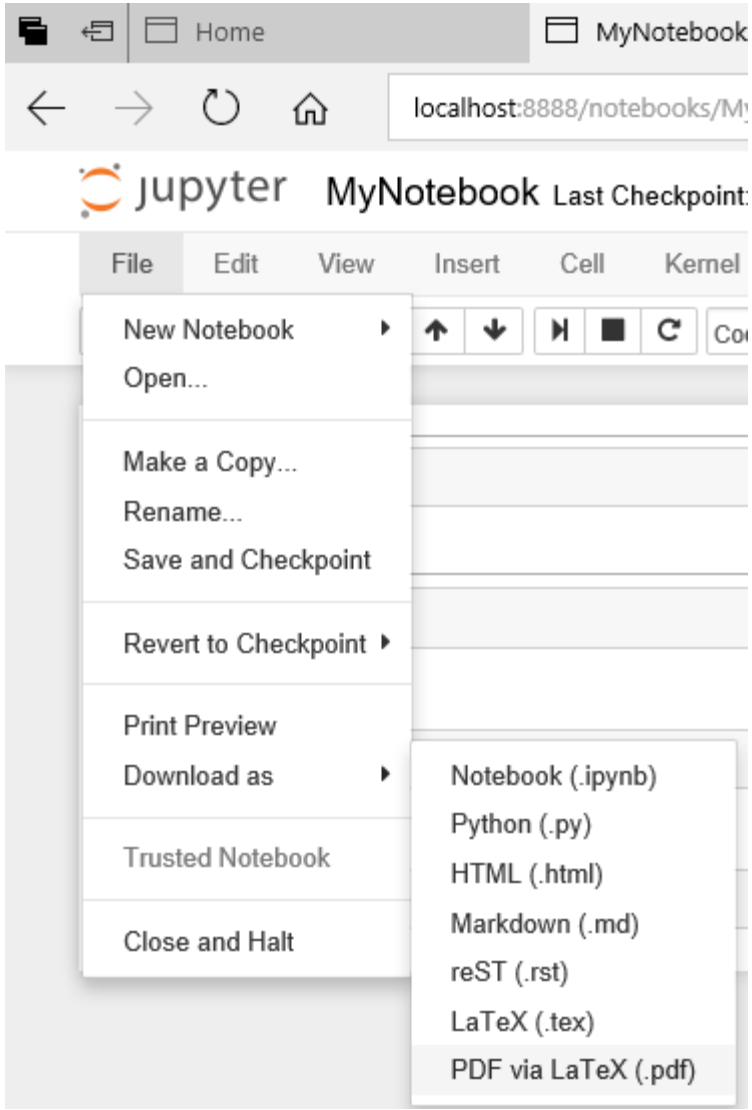


Upload another `notebook`, Download as PDF

- If you select `Upload` button, you will be able to navigate to any other notebook saved in any other directory, open it and continue to work in it.

- Once your work is done, you might want to Download content of your notebook as: PDF, Python, HTML, or in some other form.
- PDF download might require installation of `Pandoc`, a universal document converter.
- Please find Windows msi installer at:

`http://pandoc.org/installing.html`



Python Language Fundamentals

- In what follows we will open a jupyter notebook `python-fundamentals.ipynb` and walk through its content.
- The notebook is based on the content presented in Anaconda tutorials.