



# **Python: Introduction to AI, Python and Colab**

**AAA-Python Edition**



school of a

# Plan

- 1- What is AI ?
- 2- AI Branches
- 3- AI Applications
- 4- Introduction to Python
- 5- Jupyter Notebook
- 6- Google Colab



school of ai

## 1- What is AI ?

- The definition of AI (Artificial Intelligence) depends on how it has been approached by the researchers through history.
- These approaches concern developing intelligent software taking into consideration **2** aspects: “**how do we think**” and “**how do we act**”.
- So in AI, we develop software that let machines be able to:
  - Think like a **human** intelligibly
  - Or Act like a **human** intelligibly
  - Or Think **Rationally**
  - Or Act **Rationally**



school of ai

## 1- What is AI ?

- If we want a more formal definition, we can select these 4 definitions :

	Human	Rational
Think	“[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning ...” (Bellman, 1978)	“The study of the computations that make it possible to perceive, reason, and act.” (Winston, 1992)
Act	“The art of creating machines that perform functions that require intelligence when performed by people.” (Kurzweil, 1990)	“Computational Intelligence is the study of the design of intelligent agents.” (Poole et al., 1998)

Selected from (Stuart J Russell and Peter Norvig. Artificial intelligence: a modern approach. Pearson Education, Inc, 2010.).



school of ai

## 1- What is AI ?

- For example, we can find these involved techniques:

	Human	Rational
Think	Cognitive Science	Logic Programming
Act	Natural Language Processing	Rational Agents

- It doesn't mean that each technique relate to only one category. For example, agents may need natural language processing skills.
- The categorization means, that the research will focus in a particular aspect related to the category it belongs to.



school of ai

## 2- AI Branches

- From the AI categories, many branches emerged:

Machine Learning

Learn from data

Logic Based AI

Logic programming paradigm

Search

Select the optimal solution

Knowledge Representation

Efficient formal representation of Knowledge

Planning

Generate the most optimal plan to achieve a goal



school of ai

## 2- AI Branches

- From the AI categories, many field emerged:

### Heuristics

Making estimations from the knowledge of a specific problem

### Genetic Programming

Use biological processes in software to find answers to a large problem



school of ai

### 3- AI Applications

- We can use AI in:

Expert Systems

Speech recognition

Games

Robotics

Computer vision

Natural Language Processing





school of ai

## 4- Introduction To Python

- Python

Programming  
language

( The syntax Rules )

Interpreter

( Perform source code  
instructions)

### The programming language

- Is easy to learn and powerful
- It has efficient high-level data structures
- It has a simple approach to object-oriented programming.
- It is ideal for scripting and rapid application development in many areas on most platforms.

### The interpreter

- Is freely available in source or in binary form.
- It is available for all major platforms



school of ai

## 4- Introduction To Python

- Major versions
  - 2.\* (many project compatibles only with python2)
  - 3.\* (not backward compatible with python 2)
- In major of our code we will use **Python 3** (unless we specify an other version)
- Right now, (the time we are writing this course), the latest version is **3.7.2**



school of ai

## 5- Jupyter Notebook

- It is :
  - An “Open-Source” web application
  - Allows you to create and share documents that contain :
    - Live code
    - Equations
    - Visualizations
    - Narrative Text
- It runs code in many programming languages
- For installation, it requires: python  $\geq 3.3$  ,  $\geq 2.7$
- To run with additional languages, you have to install additional Kernels
- The list of all available Kernels can be found in Jupyter website (<https://github.com/jupyter/jupyter/wiki/Jupyter-kernels>)



school of ai

## 5- Jupyter Notebook



Logout

Files

Running

Clusters

Select items to perform actions on them.

Upload

New



/ Desktop



..

Notebook list empty.

Notebook:

Python 3

Other:

Text File

Folder

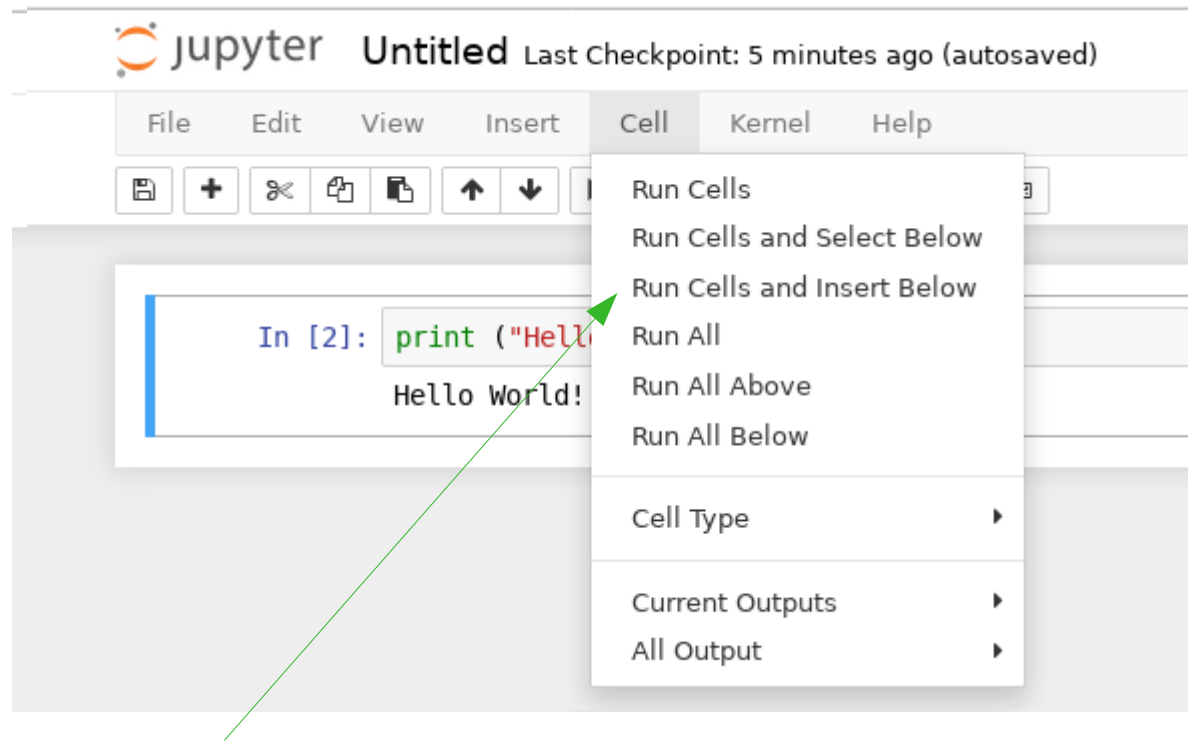
Terminal

To write code, you have to create a new Notebook



school of a

## 5- Jupyter Notebook



To run the code, you have to Run the Cells



school of a

## 5- Jupyter Notebook

jupyter Untitled Last Checkpoint: 10 minutes ago (autosaved)

File Edit View Insert Cell Kernel Help

New Notebook  
Open...  
Make a Copy...  
Rename...  
Save and Checkpoint  
Revert to Checkpoint  
Print Preview  
Download as  
Trusted Notebook  
Close and Halt

Notebook (.ipynb)  
Python (.py)  
HTML (.html)  
Markdown (.md)  
reST (.rst)  
LaTeX (.tex)  
PDF via LaTeX (.pdf)

Code

Your Notebook can be converted in any of these formats

But some formats require certain preinstalled libraries



school of ai

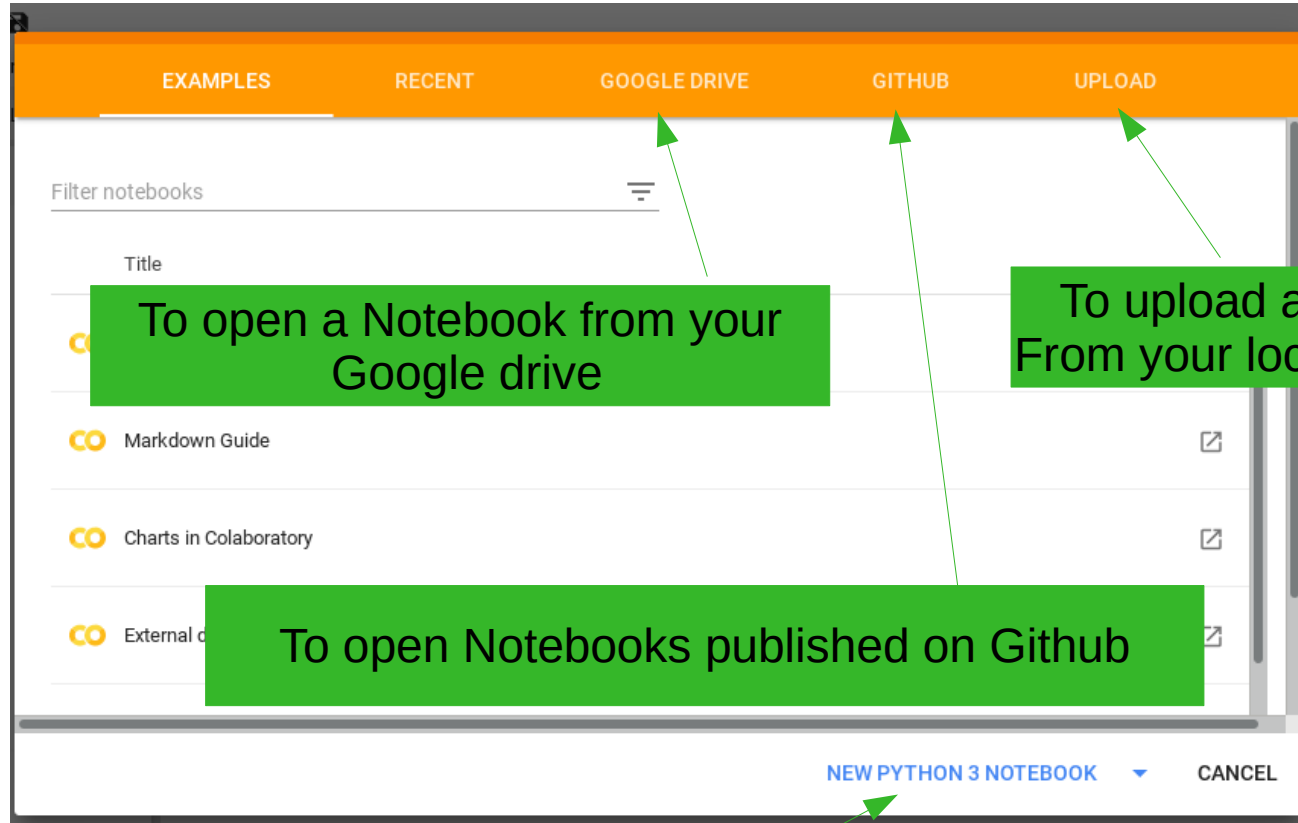
## 6- Google Colab

- Google Colaboratory is :
  - A **Jupyter notebook** environment that requires no setup to use.
  - It is a research **a free tool** for machine learning education and research
  - The Notebooks are stored in **Google Drive**
  - It supports **Python 2.7** and **Python 3.6**
  - It **doesn't support** other kernels (for now)
  - The code is executed in a **virtual machine** dedicated to a user account.
  - To use it, go to(**<https://colab.research.google.com>**)



school of ai

## 6- Google Colab



To open a Notebook from your Google drive

To upload a Notebook From your local file system

To open Notebooks published on Github

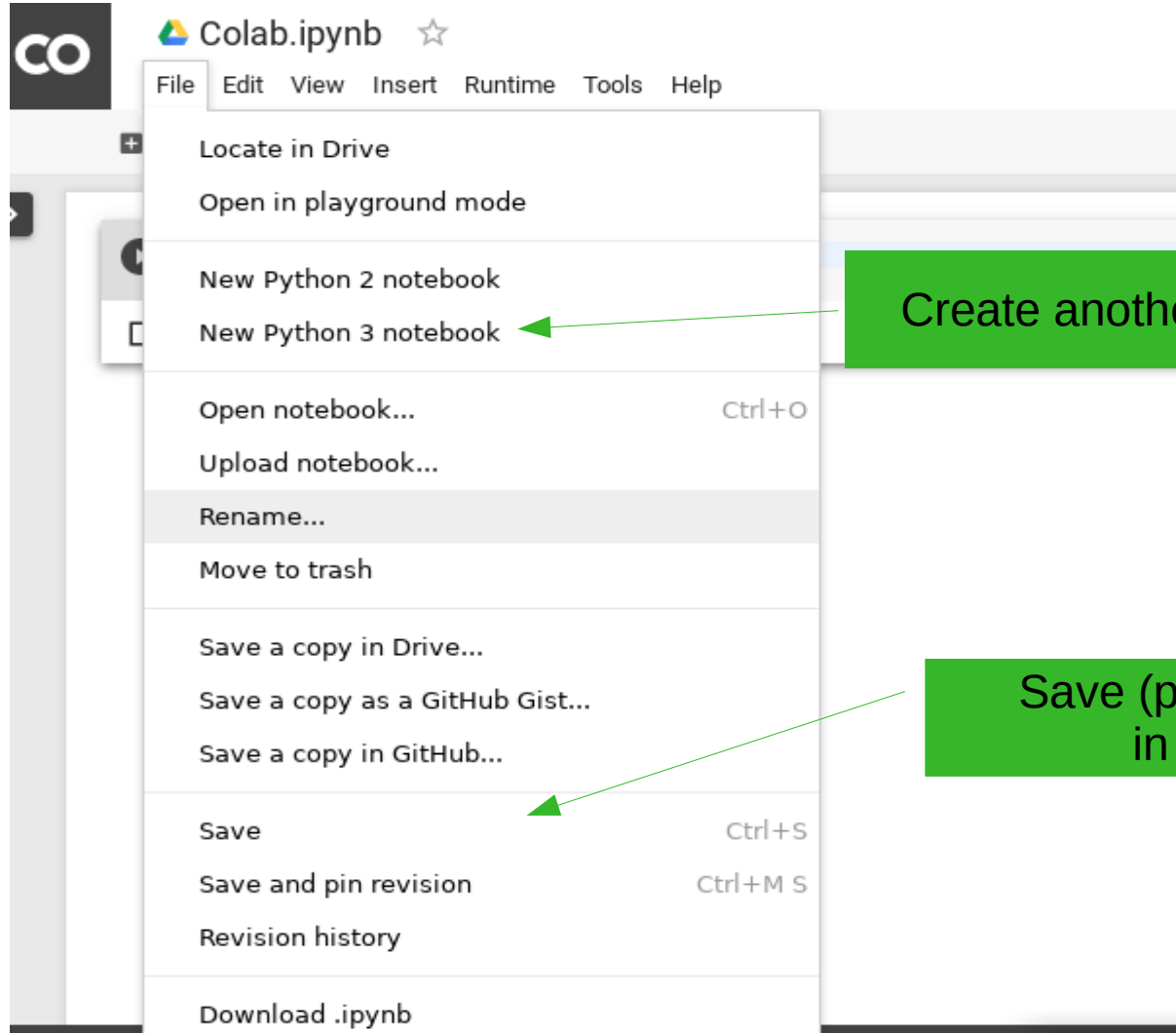
To create a new Notebook





school of ai

## 6- Google Colab





school of ai

## 6- Google Colab

The screenshot shows the Google Colab web interface. At the top, there's a 'CO' logo and the text 'Colab.ipynb' with a star icon. Below this is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. A toolbar contains buttons for '+ CODE', '+ TEXT', '↑ CELL', and '↓ CELL'. A sidebar on the left has tabs for 'Table of contents', 'Code snippets', and 'Files'. The 'Code snippets' tab is active, displaying a list of pre-installed code snippets under the heading 'Filter code snippets'. The list includes: 'Altair: Bar Plot', 'Altair: Histogram', 'Altair: Interactive Brushing', 'Altair: Interactive Scatter Plot', 'Altair: Linked Brushing', 'Altair: Linked Scatter-Plot and Histogram', and 'Altair: Scatter Plot with Rolling Mean'. Each item has a right-pointing arrow. A green arrow points from the 'Code snippets' tab to a green callout box. Another green arrow points from the 'Altair: Bar Plot' snippet to a second green callout box.

CO Colab.ipynb ☆

File Edit View Insert Runtime Tools Help

+ CODE + TEXT ↑ CELL ↓ CELL

Table of contents **Code snippets** Files X

Filter code snippets

- Altair: Bar Plot →
- Altair: Histogram →
- Altair: Interactive Brushing →
- Altair: Interactive Scatter Plot →
- Altair: Linked Brushing →
- Altair: Linked Scatter-Plot and Histogram →
- Altair: Scatter Plot with Rolling Mean →

A lot of sample codes  
ready to use

Just Click in, and a new cell  
will be created with the  
corresponding code



→
▶ Hello World!
Run the Cell

```

1 # load an example dataset
2 from vega_datasets import data
3 cars = data.cars()
4
5 # plot the dataset, referencing dataframe column names
6 import altair as alt
7 alt.Chart(cars).mark_bar().encode(
8     x='mean(Miles_per_Gallon)',
9     y='Origin',
10    color='Origin'
11 )

```

Origin	Mean of Miles_per_Gallon
Europe	~28
Japan	~31
USA	~20

[Export as SVG](#) [Export as PNG](#) [View Source](#) [Open in Vega Editor](#)

This is the results after running the Cell



school of ai

## 6- Google Colab

- Another interesting feature about Google Colab, is that we can install new **libraries** (we will talk about this later).

```
1 !pip install gym

Collecting gym
  Downloading https://files.pythonhosted.org/packages/9b/50/ed4a03d2be47ffd043b
    100% |████████████████████████████████████████| 1.5MB 7.0MB/s
Requirement already satisfied: numpy>=1.10.4 in /usr/local/lib/python3.6/dist-p
Requirement already satisfied: requests>=2.0 in /usr/local/lib/python3.6/dist-p
Requirement already satisfied: six in /usr/local/lib/python3.6/dist-packages (f
Collecting pygame>=1.2.0 (from gym)
  Downloading https://files.pythonhosted.org/packages/1c/fc/dad5eaaab68f0c21e2f
    100% |████████████████████████████████████████| 1.0MB 1.2MB/s
Requirement already satisfied: urllib3<1.23,>=1.21.1 in /usr/local/lib/python3.
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.6/d
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /usr/local/lib/python3.
Requirement already satisfied: idna<2.7,>=2.5 in /usr/local/lib/python3.6/dist-
Requirement alreadyv satisfied: future in /usr/local/lib/python3.6/dist-packages
```

- In this course, we will use Google Colab to run our Programs
- Since we program in Python 3, we will use **Python 3 Notebook**, unless we specify the other version



school of ai

# References

- Sheppard Clinton. Genetic Algorithms with Python. Clinton Sheppard, 2016.
- Python Software Foundation. Python. On-line at <https://www.python.org/>. Accessed on 22-09-2018.
- Google.Colaboratory. On-line at <https://research.google.com/colaboratory/faq.html>. Accessed on 22-09-2018.
- Jupyter. The jupyter notebook. On-line at <http://jupyter.org/>. Accessed on 22-09-2018.
- Joshi Prateek. Artificial intelligence with Python. Packt Publishing, 2017.
- Stuart J Russell and Peter Norvig. Artificial intelligence: a modern approach. Pearson Education, Inc, 2010.
- Al Sweigart. Automate the boring stuff with Python: practical programming for total beginners. No Starch Press, 2015.



# Thank you!

FOR ALL YOUR TIME