Python – programming language for data analysis

Installation and Usage

Download and Install

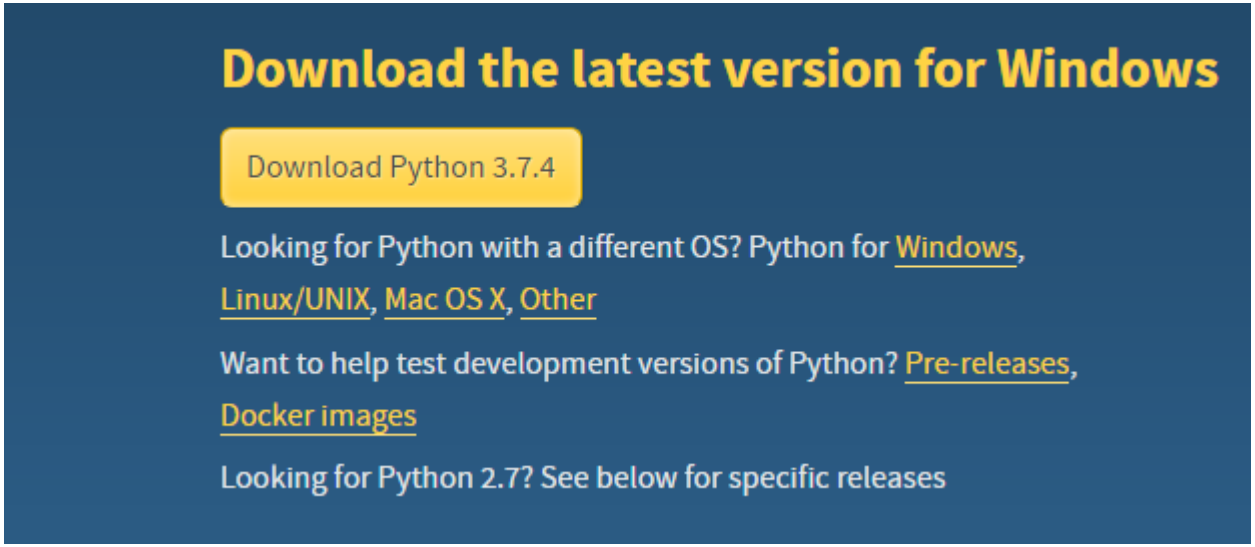
Python

- Free and Open Source programming language.
- Simple syntax easy to understand.
- Can work on many platforms.

- 2 versions:
  - Python 2.0 (October 2000): The final 2.x version 2.6 & 2.7.
  - Python 3.0 was released in 2008: Less regard for backwards compatibility.

How to install Python 3(1)

- Download and install the latest version of Python.
- https://www.python.org/downloads/



Select the Download Button

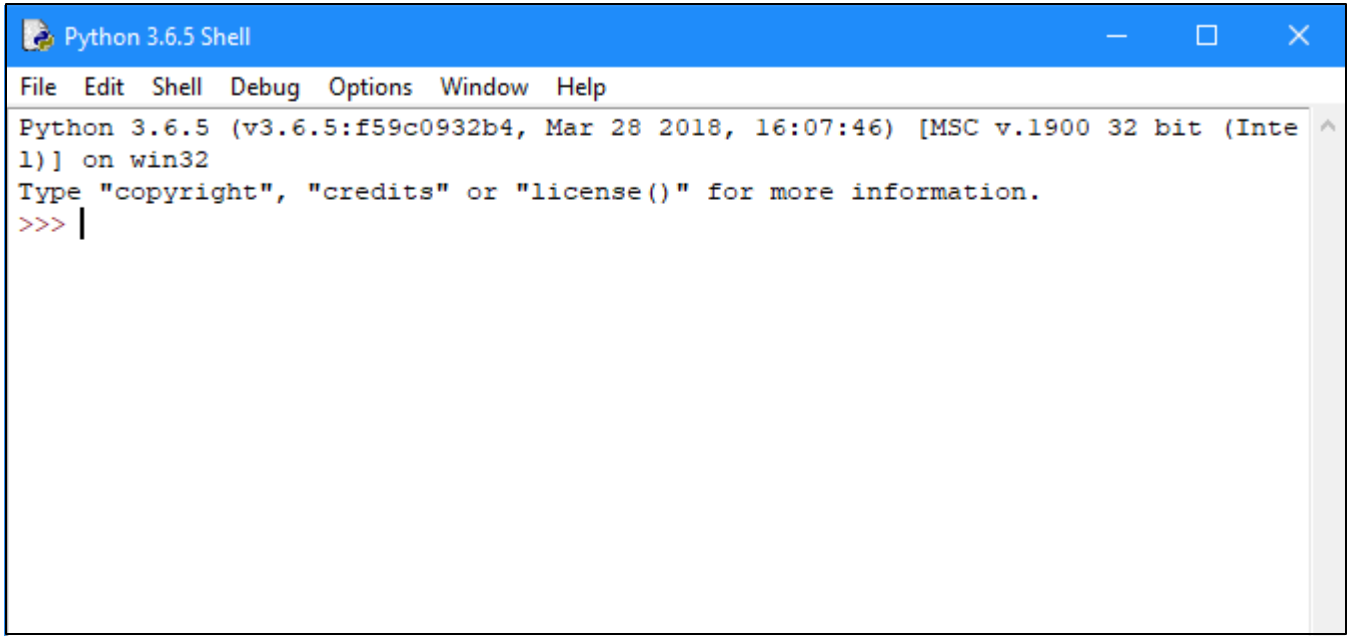
How to install Python 3(2)

- Downolad and execute . exe file.
- 2 ways to run a Python program:
    - Source file .py
    - Inetractive interpreter prompt
- From Start Menu, open IDLE as follows:
  Start → All Programs → Python 3.7 → IDLE (Python GUI)

Python IDLE

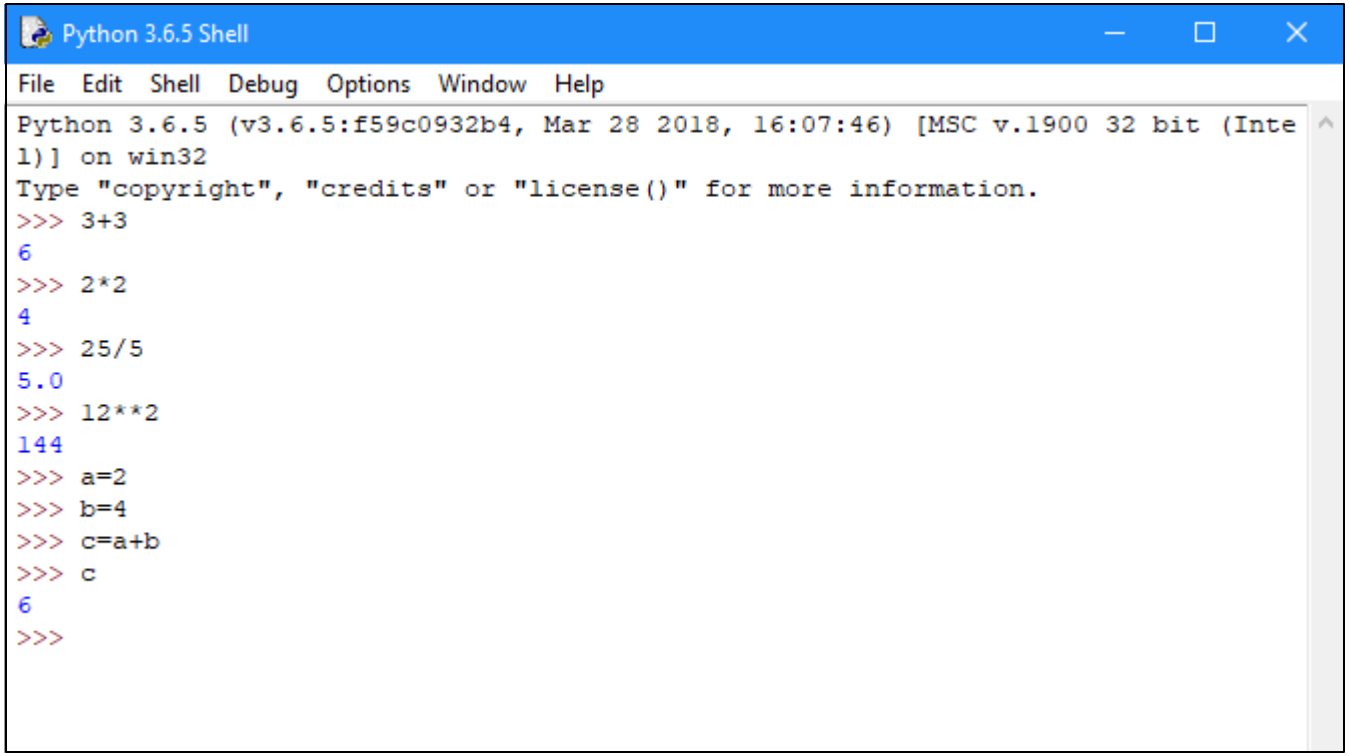
Python IDLE

- "**I**ntegrated **D**eve**L**opment **E**nvironment" for Python.
- A software package that lets us write Python commands and edit and run Python programs.
- Helps write python programs easily and create source files.



Basic Examples

- Python Shell commands



Python Modules

Python modules

- Contain a set of functions, variables of all types (arrays, dictionaries, objects etc).
- Use installed libraries by importing them at the beginning of python code.
- We can import as many libraries as we like.
- Use a module, by using the **import** statement.
- Create an alias when we import a module, by using the **as** keyword.
- Import only specific functions from a module, by using the **from** keyword.

Installing python modules with pip

- pip installer program, is included by default with the Python installer.
- pip works on Unix/Linux, mac OS, and Windows.
- Usage → pip install module_name, using the windows command line.

Pandas module

- *pandas* is an open source library.
- Easy-to-use data structures and data analysis tools.
- Installation → pip install pandas.
- DataFrame object for data manipulation.
- Reading and writing data.
- Reshape data sets, slice, index, and subset of large data sets.
- Example file: WorkingWithPandas.rar

Statistics module

- Functions for calculating mathematical statistics of numeric data.
- `mean()` →Arithmetic mean (average) of data.
- `median()` →Middle value of data.
- `mode()` →Most common value of discrete data.
- `pstdev()` →Population standard deviation of data.
- `pvariance()` →Population variance of data.
- `stdev()` →Sample standard deviation of data.
- `variance()` → Sample variance of data.

Matplotlib module

- Plotting library for Python.
- Generate plots, histograms, bar charts, scatterplots e.t.c.
- Installation → pip install matplotlib.
- Example file: WorkingWithMatplotlib.rar

NumPy module

- Linear algebra and random number capabilities.
- Array creation, printing arrays, arithmetic operations.
- Index, Slice and Iterate.
- Mathematical functions (e.g. mean, median, maximum, std).

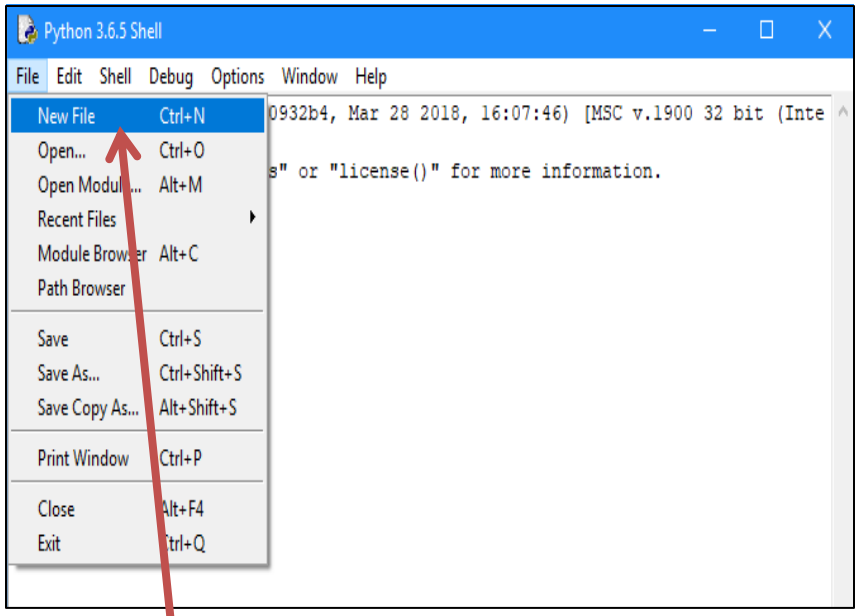
scikit-learn module

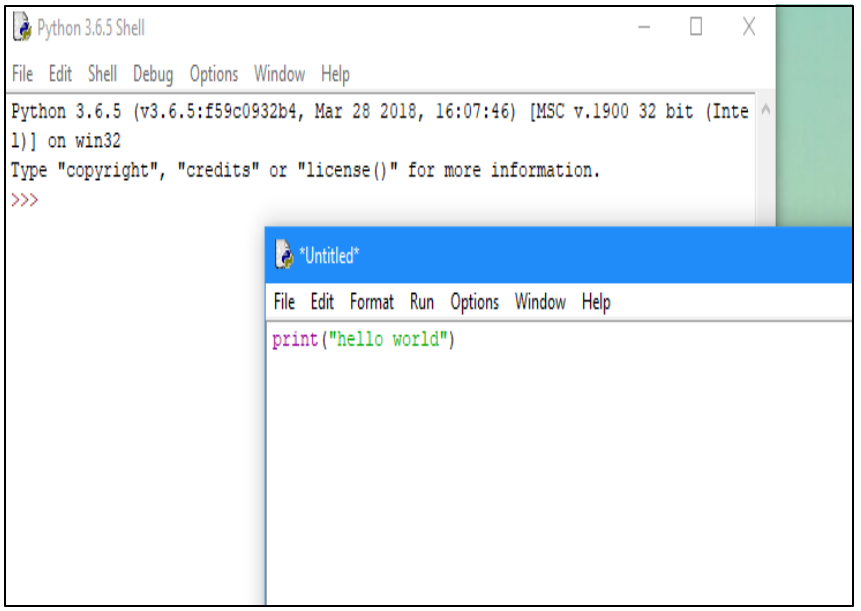
- Tools for data mining and data analysis.
- Machine learning in Python.
- Classification (e.g. Naïve Bayes, Decision Trees).
- Regression(e.g. Random Forest).
- Clustering(e.g. k-means).
- Installation → pip install scikit-learn.
- Example:

```
from sklearn import datasets, linear_model
from sklearn.metrics import mean_squared_error,r2_score
```

Using Python

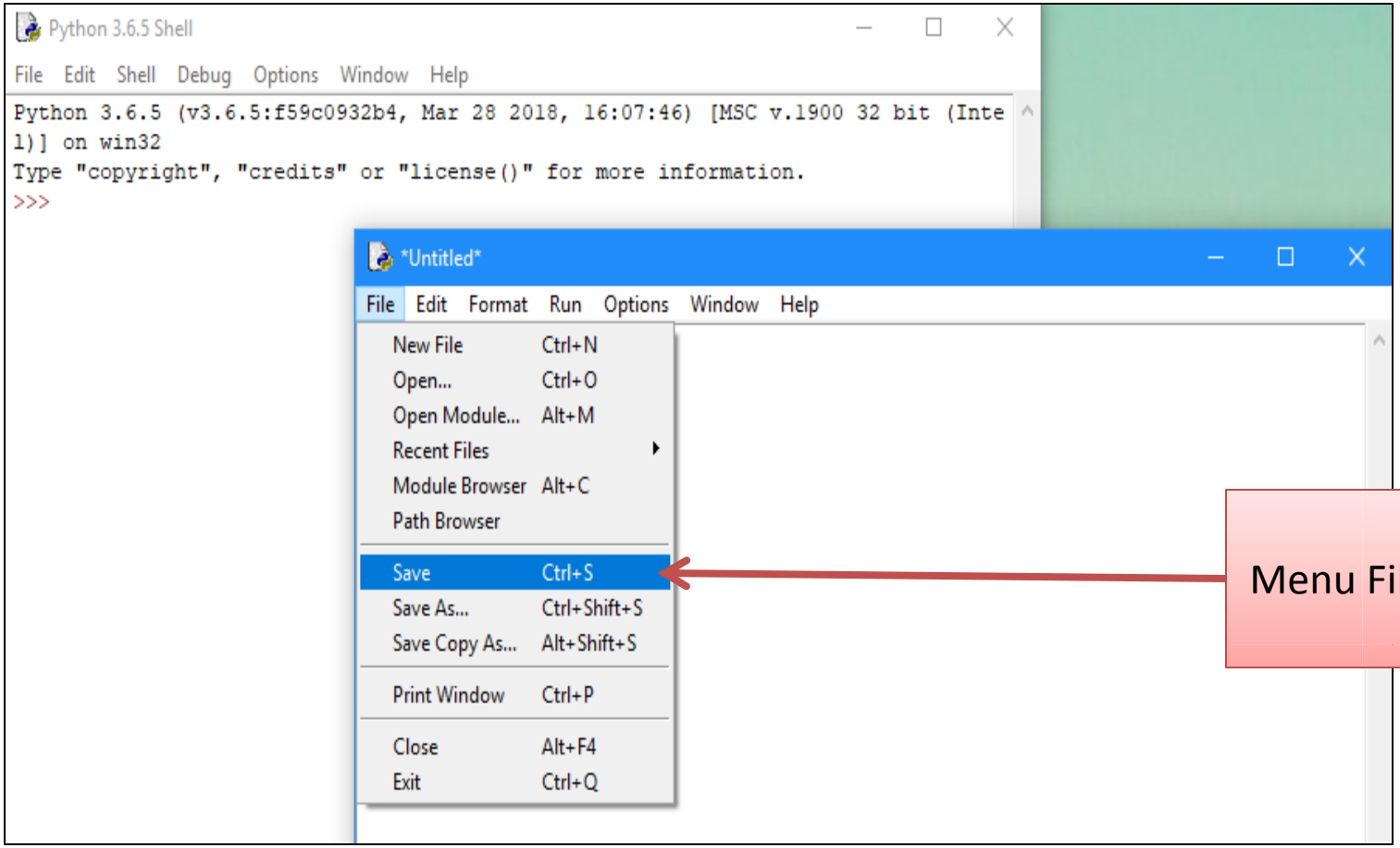
Create a Python program



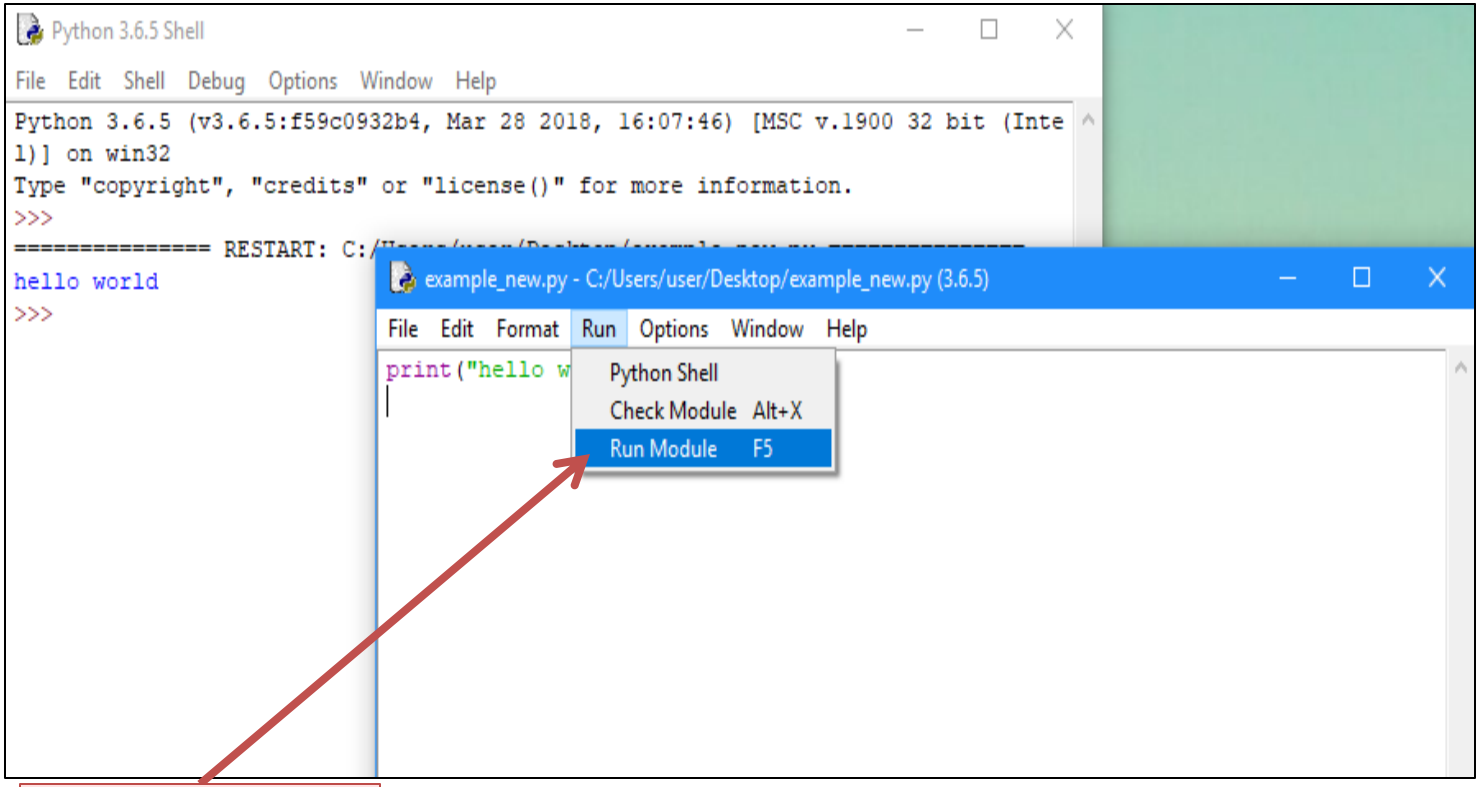


Menu
File>New File

Save Python file



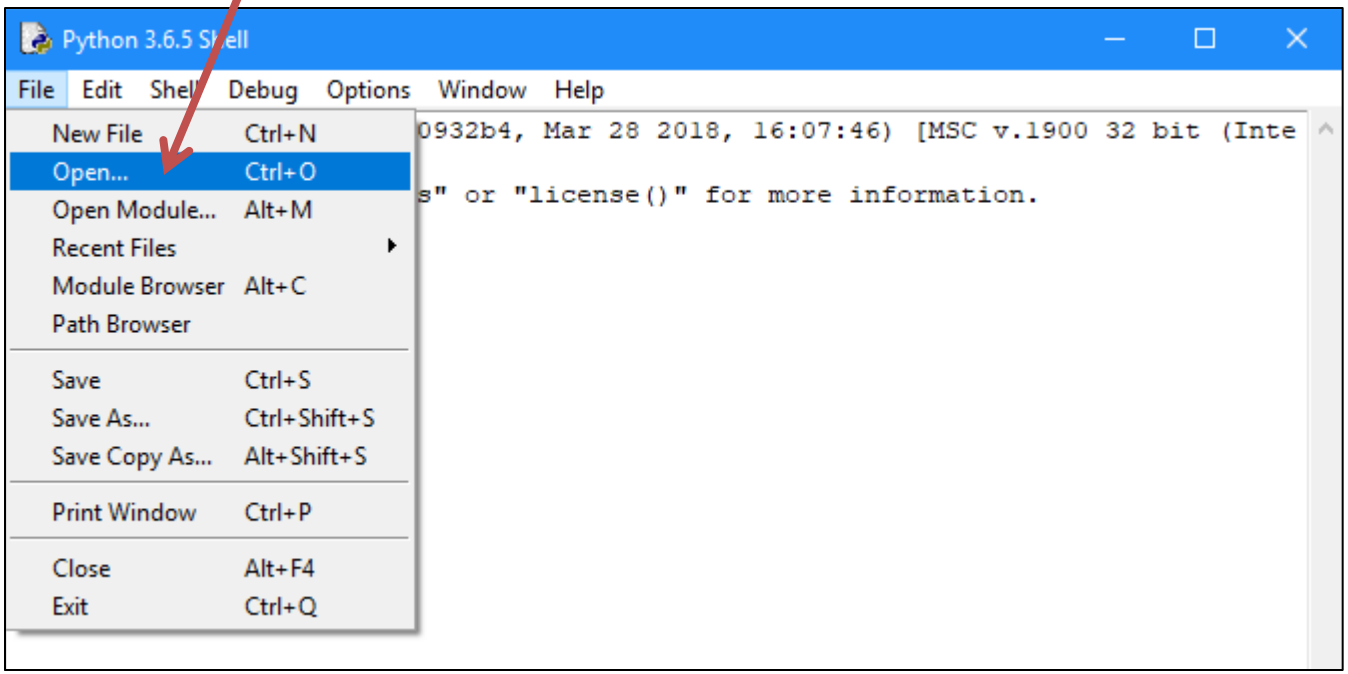
Execute a Python program



Menu Run>Run Module

Open a Python file

Menu File>Open…



Example code (import modules)

```
import pandas as pd
from statistics import *
import csv
import matplotlib.pyplot as plt
```

Example code (statistics)

```
#Read data from file
data=pd.read_csv("demo.csv", header=0)

#Compute arithmetic mean of Income
print('Mean of Income:
    '+"{:.2f}".format(mean(data['Income'])))

#Compute the standard deviation of Food Expenditure
print('Standard deviation of food expenditure:
    '+"{:.2f}".format(stdev(data['FoodExpenditure'])))
```

Example code (create plots)

```
#Plot showing the relation between Food Expenditure and Income
plt.scatter(data.Income, data.FoodExpenditure)
plt.xlabel('Income')
plt.ylabel('Food Expenditure')
plt.show()

#Add line-in the same plot-of mean value of Food Expenditure
fig, ax = plt.subplots()
data_line = ax.scatter(data.Income, data.FoodExpenditure)
fig.savefig('my_figure1.png')
plt.xlabel('Income')
plt.ylabel('Food Expenditure')
plt.axhline(mean(data['FoodExpenditure']), color='r', linestyle='--')
plt.show()

#save plot as image
fig.savefig('my_figure2.png')
```

Jupyter Notebook

Jupyter Notebook

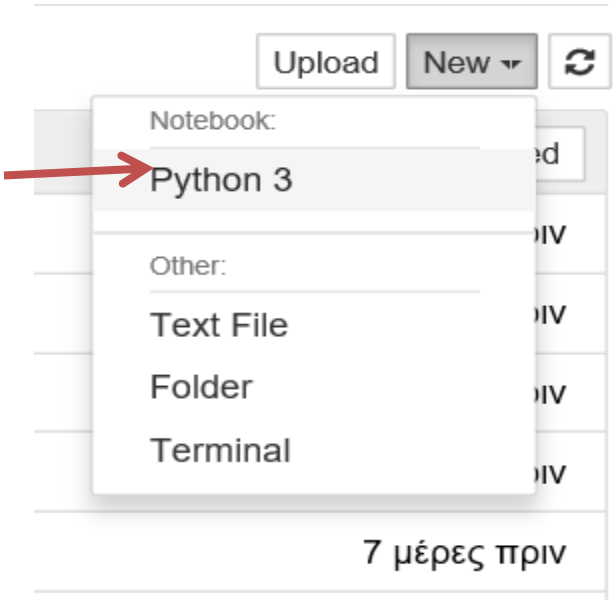
- **Jupyter notebook** is a web application that allows the user to write code.
- Interactively develop and present data.
- Installation → Install Anaconda (Python distribution for data science).
- Anaconda is pre-loaded with all the most popular python libraries and tools.
- Download the latest version of Anaconda for Python 3 from:

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

Create a Notebook

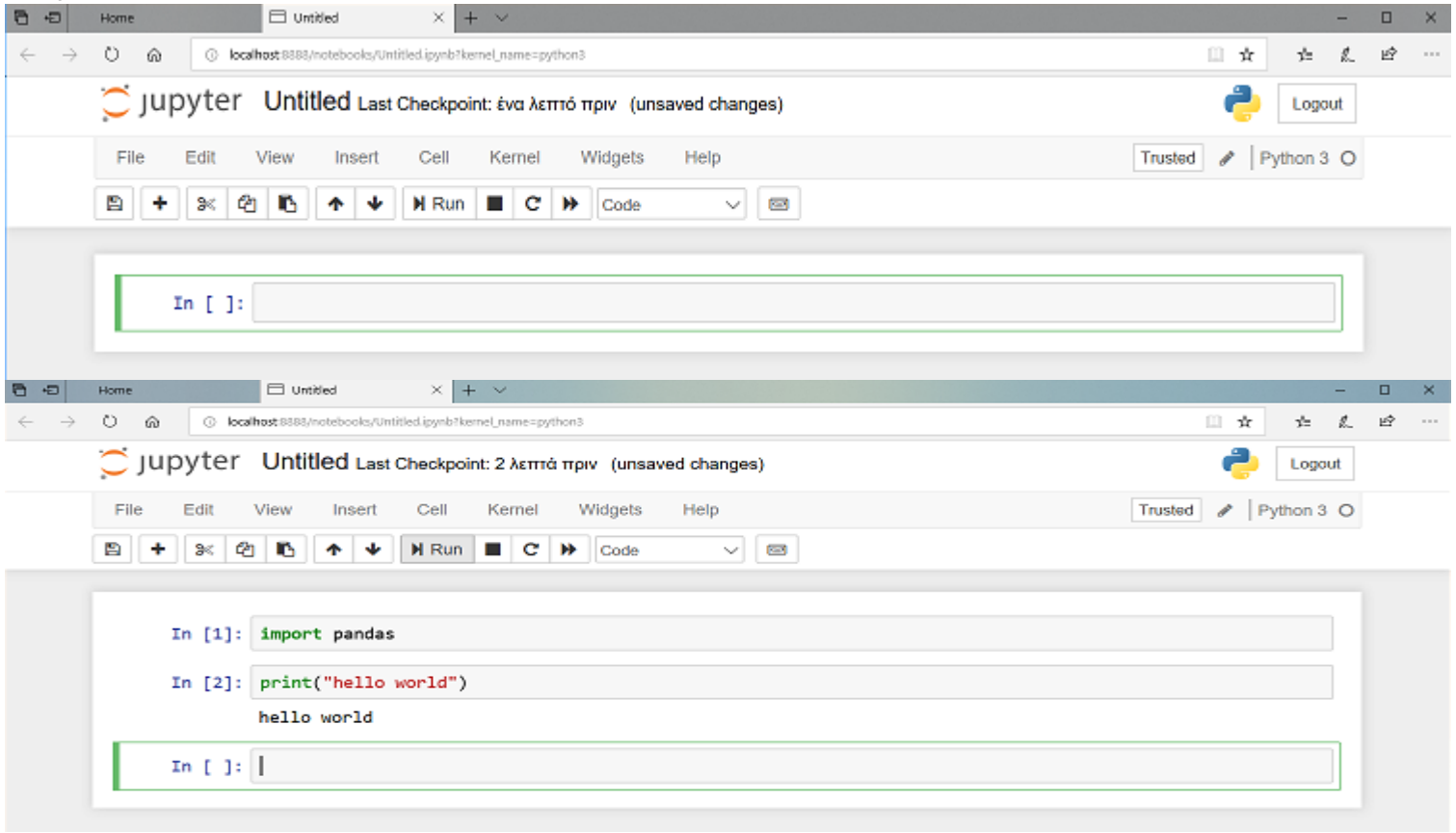
- Open Anaconda prompt from start menu on Windows and type jupyter notebook.
- OR run Jupyter via the shortcut Anaconda adds to the start menu.
- Will open a new tab in the default web browser.

click the "New" drop-down button and select "Python 3"

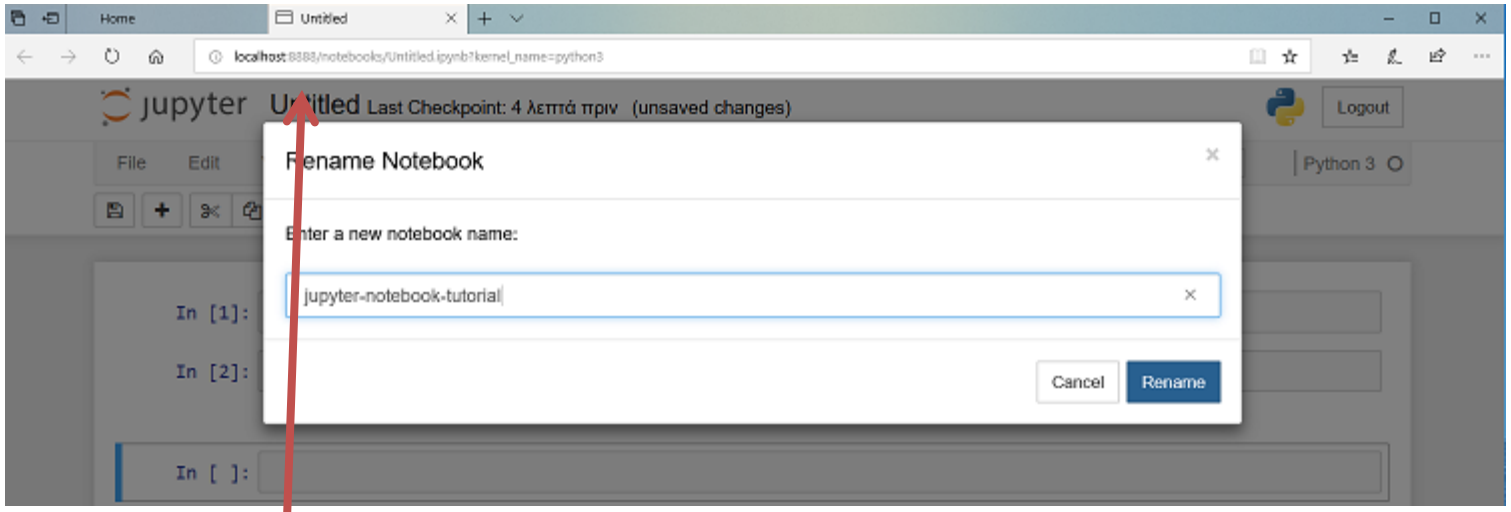


The Notebook Interface

- A **code cell** is were we write Python code to be executed and displays its output below.



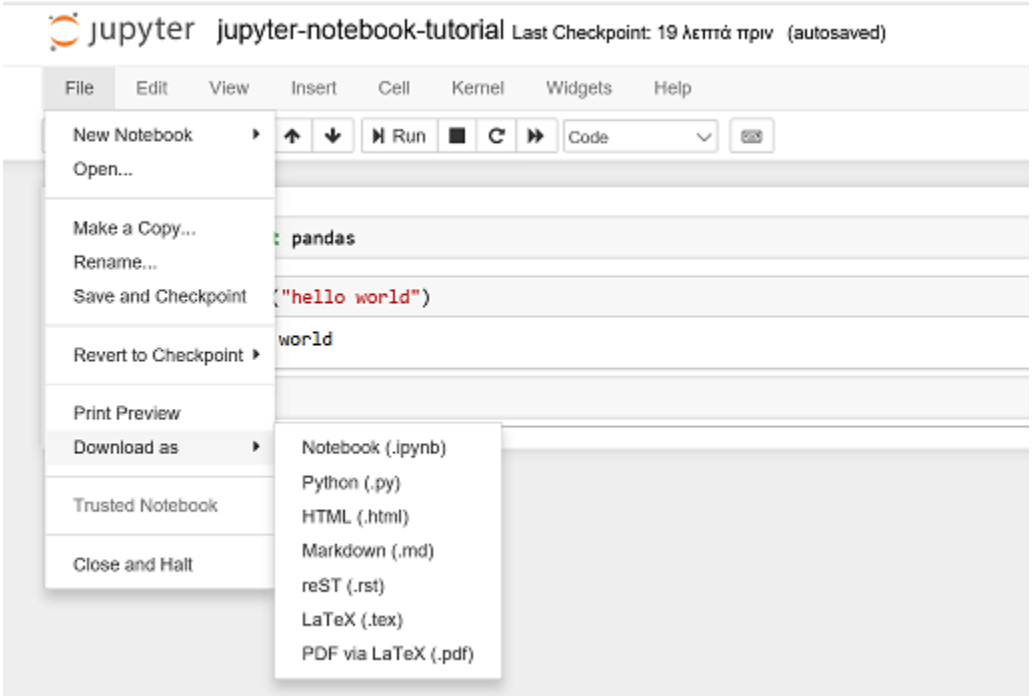
Name a notebook



Click Untitled to rename a notebook.

Export a notebook

- From the menu select "File > Download As."



Useful Links

- A byte of Python (Ελληνικά) http://dide.flo.sch.gr/Plinet/Meetings/Meeting23/A_Byte_of_Python-el.pdf
- Python

  https://www.python.org/downloads/
- Anaconda

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