

Instalando Python 3 y bibliotecas auxiliares para el manejo de datos

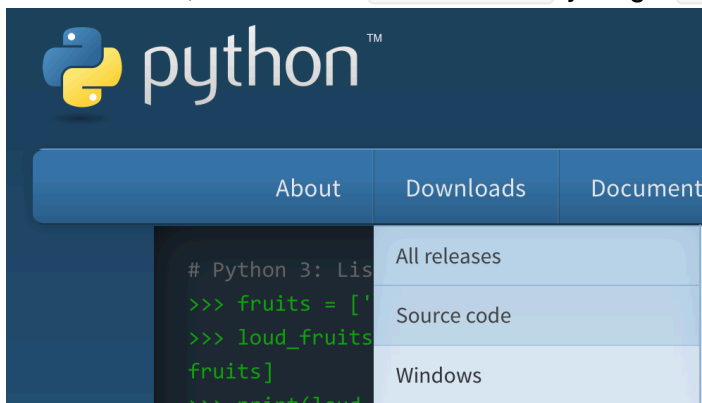
Instalación de Python básico

la última version de Python, hasta septiembre de 2016 es la 3.5.2

Instalación para Windows

NOTA: Anaconda es una distribución que incluye varios módulos para Python, tiene un IDE y un instalador de paquetes. Aquí sin embargo estaremos utilizando la versión original básica de Python directamente del sitio de los desarrolladores. Considero que es mejor instalar sólo lo que necesitarás y cuando lo necesitarás, lo cual sólo es posible con esta versión básica.

1. Conectarse a <https://www.python.org>
2. En los menús, seleccionar `Downloads` y luego `Windows`



3. En la lista, escoger `Latest Python 3 Release - Python 3.5.2`

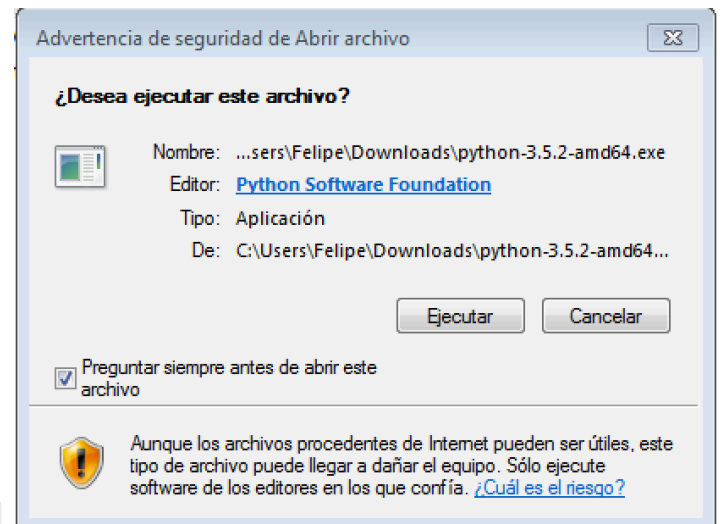
Python Releases for Windows

- [Latest Python 2 Release - Python 2.7.12](#)
- [Latest Python 3 Release - Python 3.5.2](#)

4. Bajar hasta la sección `Files` y ahí escoger `Windows x86-64 executable installer`

Files

Version	Operating System
Gzipped source tarball	Source release
XZ compressed source tarball	Source release
Mac OS X 32-bit i386/PPC installer	Mac OS X
Mac OS X 64-bit/32-bit installer	Mac OS X
Windows help file	Windows
Windows x86-64 embeddable zip file	Windows
Windows x86-64 executable installer	Windows



5. En la ventana que aparece, escoger `Ejecutar`
6. En la siguiente ventana **Hacer click en Add python.exe to path** y luego `Install now`
7. Si lo pide, seleccionar `Permitir que el programa realice cambios en el equipo`
8. Seleccionar `Close`

Con estos pasos se instalará el Python 3.5 (64 bits) (versión actual a 15 de septiembre de 2016, **no se recomienda todavía la versión 3.6 por ser inestable**).

El programa ejecutable que deja en el menú de Windows (`Python 3.5`) es **la consola** de Python.

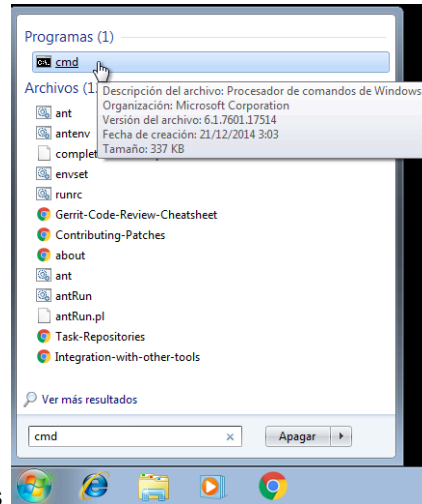
También se instalará en la carpeta Python 3.5 el programa **IDLE** que es un ambiente de programación muy sencillo, así como la documentación de Python y sus módulos básicos.

Nota: El Python así instalado ya se puede ejecutar y se puede programar con un editor o un Ambiente Integrado de Desarrollo (IDE), como por ejemplo PyCharm (recomendado), Komodo, Geany, etc. Nosotros utilizaremos el ambiente de libretas **Jupyter**, por ser muy flexible y útil para exposiciones.

Instalación de Jupyter

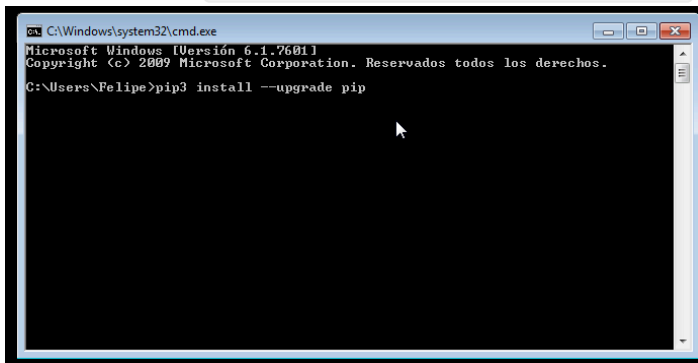
Instalación para Windows

NOTA: Algunas versiones de Internet Explorer no soportan al Jupyter, por lo que se recomienda bajar e instalar otro navegador como el Chrome, antes de estos pasos.



1. Abrir una terminal de Windows

2. Ejecutar en terminal: `pip3 install --upgrade pip` (ignorar los mensajes de error, si es que



salen)

3. Ejecutar en terminal: `pip3 install jupyter`

4. Ejecutar `pip3 install --upgrade jupyter`

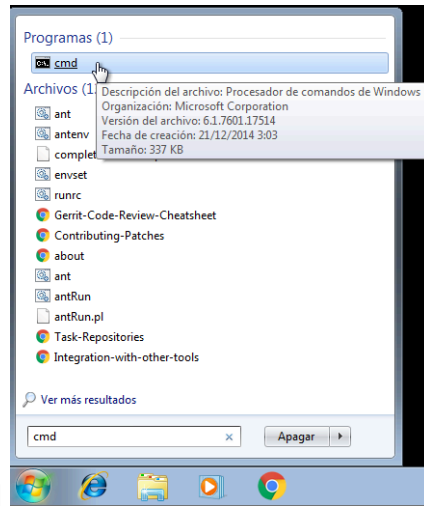
5. Ejecutar en terminal: `jupyter notebook` (de ser necesario, seleccionar `Habilitar configuración de intranet`)

Esto abrirá una ventana del internet explorer o el navegador que se tenga predeterminado.

Instalación de módulos auxiliares

Instalación para Windows

NOTA: Este manejo de terminal, aunque te pueda parecer un poco "rústico" por no usar ventanas, es la forma estándar de proceder en todos los sistemas operativos, por ello la prefiero a otros instaladores que no se sabe lo que sucede exactamente en el fondo.



1. Abrir una terminal
2. `pip3 install matplotlib`
3. `pip3 install html5lib`
4. `pip3 install bs4`
5. `pip3 install quandl <---` esto también instala pandas!
6. Conectarse con el navegador a <http://www.lfd.uci.edu/~gohlke/pythonlibs/> y descargar las últimas versiones (NO RC) de numpy+mkl scikit-learn y scipy. Por ejemplo, para Numpy el archivo a descargar es el de la 6a línea, pues es el último de los que dicen "numpy-1.11.1+mkl-..." en su nombre, los que siguen tienen "rc1" que significa *release candidate* uno, o sea, candidato a liberarse, primera etapa. En otras palabras, los de versión 1.11.2 aún no son estables.

NumPy, a fundamental package needed for scientific computing with Python.

Numpy+MKL is linked to the Intel® Math Kernel Library and includes required DLLs in the numpy.core directory.

[numpy-1.11.1+mkl-cp27-cp27m-win32.whl](#)
[numpy-1.11.1+mkl-cp27-cp27m-win_amd64.whl](#)
[numpy-1.11.1+mkl-cp34-cp34m-win32.whl](#)
[numpy-1.11.1+mkl-cp34-cp34m-win_amd64.whl](#)
[numpy-1.11.1+mkl-cp35-cp35m-win32.whl](#)
[numpy-1.11.1+mkl-cp35-cp35m-win_amd64.whl](#) ←
[numpy-1.11.2rc1+mkl-cp27-cp27m-win32.whl](#)
[numpy-1.11.2rc1+mkl-cp27-cp27m-win_amd64.whl](#)
[numpy-1.11.2rc1+mkl-cp34-cp34m-win32.whl](#)
[numpy-1.11.2rc1+mkl-cp34-cp34m-win_amd64.whl](#)
[numpy-1.11.2rc1+mkl-cp35-cp35m-win32.whl](#)
[numpy-1.11.2rc1+mkl-cp35-cp35m-win_amd64.whl](#)

7. Luego instalarlos con pip3:
 1. `pip3 install numpy-1.11.1+mkl-cp35-cp35m-win_amd64.whl`
 2. `pip3 install scipy-0.18.0-cp35-cp35m-win_amd64.whl`
 3. `pip3 install scikit_learn-0.17.1-cp35-cp35m-win_amd64.whl`
 4. `pip3 install lxml-3.6.4-cp35-cp35m-win_amd64.whl`
8. `pip3 install sklearn-pandas`

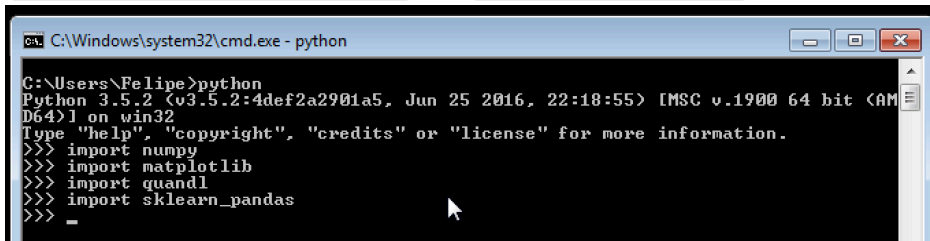
9. `pip3 install openpyxl`
10. `pip3 install xlrd`

Numpy, scipy y sklearn contienen rutinas en C que deben ser compiladas apropiadamente, como esto es más complicado de lo que se necesita aquí, bajamos los binarios del sitio de Gohlke que, aunque no oficial, es el recomendado para ello.

Verificación del funcionamiento

1. Abrir una terminal
2. Teclear python, esto debe mostrar la versión instalada y el "prompt" >>>, indicando que la consola de Python está lista para recibir instrucciones. En general no utilizaremos la consola de Python, pero aquí es útil para ver que los módulos fueron instalados.
3. En el prompt de python teclear:

1. `import numpy`
2. `import matplotlib`
3. `import quandl`
4. `import sklearn_pandas` (o `import sklearn`)



5. `import openpyxl`
6. `import xlrd`

Si no marca ningún error, es que todo estuvo bien instalado.