# Introducción a la computación científica con Python

## ¿Qué es Python?

[Python](http://www.python.org/) es un lenguaje de programación de alto nivel, moderno, de propósito general y orientado a objetos.

Características generales de Python:

* **Lenguaje simple y limpio:** el código realizado en Python es fácil de leer e intuitivo, utiliza una sintaxis fácil de aprender y escala bien con el tamaño de los proyectos.
* **Lenguaje expresivo:** Suele requerir menos líneas de código que otros lenguajes de programación, lo que implica cometer menos errores y facilita su mantenimiento.

## Instalación y uso de Python

### Versiones de Python

#### ¿Python 2 o Python 3?

Hay dos versiones **actuales** de Python: Python 2 y Python 3.

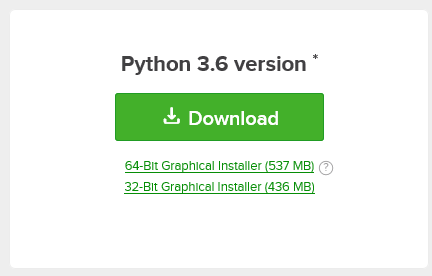
Python 3 es la versión sucesora de Python 2 pero no es completamente compatible con esta versión. Por ser la más reciente, en esta asignatura utilizaremos **Python 3**.

### Instalación

En esta asignatura se va a utilizar Windows como Sistema Operativo. Para dicho Sistema Operativo es recomendable utilizar el paquete [Anaconda](http://continuum.io/downloads). En el enlace se debe descargar la versión de Anaconda basada en Python 3 y para Windows como Sistema Operativo (también funciona de la misma forma con MacOS y Linux).

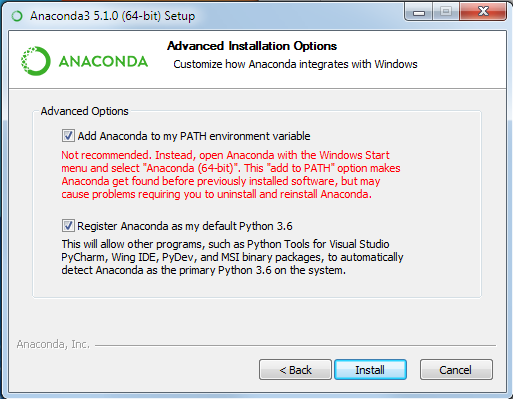
#### Descargar Anaconda

Debes descargar la versión de Python 3 (a la izquierda). Haz click en Download.



#### Instalar Anaconda

Para instalar Anaconda simplemente debes seguir todos los pasos, **pero debes marcar una opción que viene sin marcar por defecto**. Para poder utilizar los Jupyter Notebooks (entorno de programación que vamos a usar en este curso) con mayor facilidad, es conveniente **indicar a Anaconda que queremos añadirlo a la variable PATH.** Para ello, en el paso de “Advanced Installation Options” debemos marcar la primera opción (que viene desmarcada por defecto) como vemos a continuación.



Opcionalmente nos permitirá instalar Microsoft Visual Studio Code. Para esta asignatura, no es necesario instalar dicho programa.

A continuación vamos a ver las diferentes formas ejecutar código Python, entre las que está el uso de Jupyter Notebook, la forma en la que trabajaremos en esta asignatura.

### 

### Entornos para ejecutar código Python

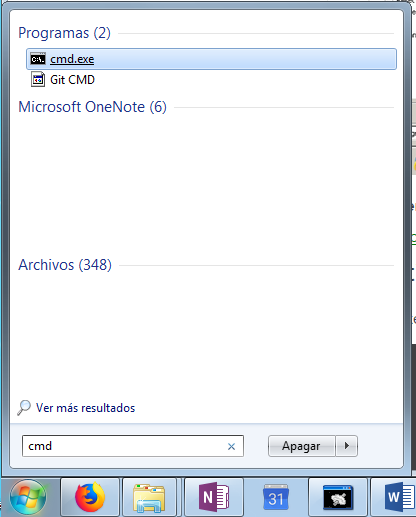
Python no solo es un lenguaje de programación sino que también se refiere a la implementación estándar del intérprete (técnicamente conocido como [CPython](http://en.wikipedia.org/wiki/CPython)) que es el encargado de ejecutar código Python en un ordenador.

Existen varios entornos diferentes a través de los cuales se puede utilizar el intérprete de Python. Cada entorno tiene sus ventajas y es adecuado para diferentes flujos de trabajo. Una fortaleza de Python es su versatilidad ya que puede ser utilizado de diferentes formas. Sin embargo, esto puede ser confuso para principiantes por lo que comenzamos con una breve descripción de los entornos de Python que son útiles para la computación científica.

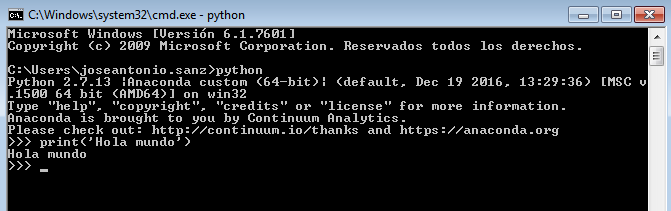
#### Consola interactiva

Hay muchas maneras de usar el lenguaje Python. Se ha mencionado que es un lenguaje **interpretado** e **interactivo**. Si ejecutamos la consola (En windows cmd.exe) y luego python, se abrirá la consola interactiva:

1. Hacemos click en Inicio
2. Escribimos cmd
3. Presionar Enter



1. Escribimos python y presionamos Enter



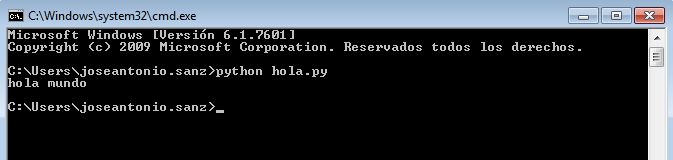
|  |
| --- |
| **En caso de error**  Si llegados a este punto recibes un error de que no se encuentra python, quiere decir que no has marcado la opción de añadir Anaconda al PATH al realizar la instalación. Puedes volver a realizar la instalación de Anaconda marcando dicha opción o puedes añadirla tú mismo. Para ello, sigue los siguientes pasos.   1. Haz clic en inicio, luego botón derecho en Equipo y haz clic en propiedades. 2. Ahora haz clic en “configuración avanzada del sistema” (en el menú de la izquierda). 3. En la pestaña “Opciones avanzadas”, haz clic en “Variables de entorno” (última opción). 4. En la variable PATH (la del usuario o del sistema, cualquiera de las dos), debes añadir las variables correspondiente a la instalación de Anaconda. Si lo has instalado para todos los usuarios serán (en Windows 7 se separan con ; mientras que en Windows 8 y posteriores aparece una línea para cada ruta):    * C:\Anaconda2\    * C:\Anaconda2\Scripts   En otro caso lo más probable es que lo hayas instalado para tu propio usuario en cuyo caso serían (cambiar nombreusuario por tu nombre de usuario):   * C:\Users\nombreusuario\Anaconda2\ * C:\Users\nombreusuario\Anaconda2\Scripts   Ahora puedes abrir una nueva consola de comandos (cmd.exe) y ejecutar Python. En caso de que siga sin funcionar debes revisar las variables de entorno (PATH) y la instalación de Anaconda. |

El símbolo >>> en la última línea indica que te encuentras en la sesión del intérprete interactiva de Python, también llamada “Python shell”. Este es diferente del símbolo de sistema normal del terminal.

En la consola interactiva se pueden escribir sentencias o pequeños bloques de código que son ejecutados inmediatamente. Por ejemplo, en la imagen anterior se puede ver el código correspondiente a un programa que muestra el texto Hola mundo por pantalla.

##### Programas

También podemos usar Python para hacer programas o scripts. Esto es, escribir nuestro código en un archivo con extensión .py y ejecutarlo con el intérprete de python. Por ejemplo, se puede escribir la instrucción mostrada en la consola interactiva (print('Hola mundo')) en un script de Python y llamarlo hola.py. A continuación, en la consola interactiva podemos ejecutar el script y el resultado será el mismo que el mostrado anteriormente.



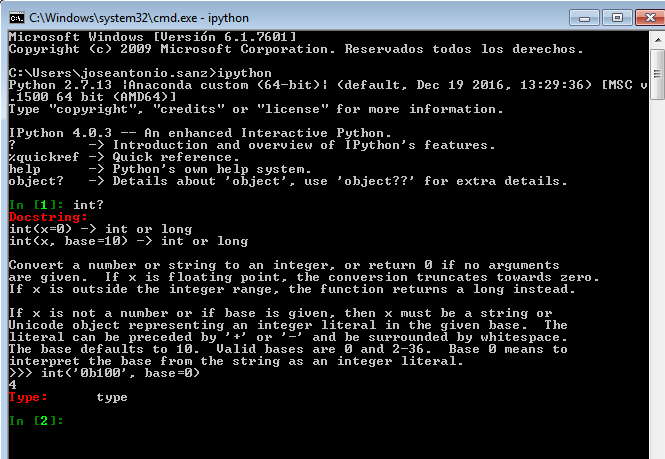
Para ejecutar scripts de Python, asegúrese de estar en la línea de comandos del sistema, el cual tendrá el símbolo > al final de la ruta, no el símbolo del intérprete de Python (que en cambio tiene >>>).

Sin embargo, este entorno es **limitado** por lo que no es la forma habitual de trabajar.

Para salir de la consola interactiva de Python se debe presiona Control+z.

#### IPython

IPython es un shell interactivo que aborda las limitaciones del intérprete estándar de Python y que es un modo de trabajo habitual. Ofrece una línea de comandos interactiva al intérprete de Python mucho más amigable que la consola interactiva. Si ejecutamos la consola (cmd.exe) y luego ipython, se abrirá la consola correspondiente a ipython:



Algunas de las características útiles de IPython son:

* Historial de comandos, que puede ser consultado con las teclas del teclado con flechas en direcciones ascendente y descendente.
* Auto-completado utilizando la tecla de la tabulación (<TAB>) u obtención rápida de ayuda de cualquier objeto (?).
* Edición de código in-line.
* Buena interacción con el sistema operativo.

#### Jupyter Notebook (IPython notebook)

Otra forma muy útil es usar los [IPython notebook](http://ipython.org/notebook.html), que es un entorno Web para Python apropiado para realizar computación interactiva. Aunque los *notebooks* utilicen un navegador web como interfaz gráfica éstos se ejecutan localmente (en el mismo ordenador que ejecuta el navegador web). Este es el entorno de desarrollo que utilizaremos en las prácticas de la asignatura.

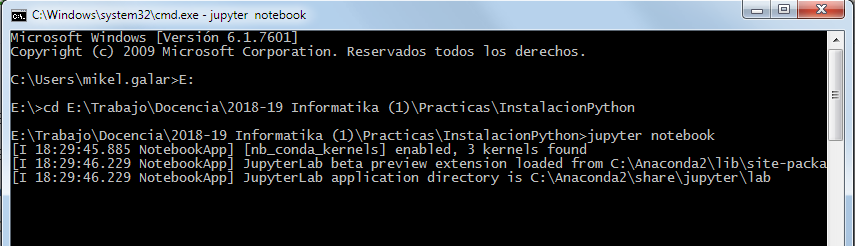
Si bien nació como parte del proyecto IPython, el mismo entorno visual se puede conectar a *"kernels"* de distintos lenguajes. Se puede usar Jupyter con Python, Julia, R, Octave y decenas de lenguajes más.

Podemos crear y editar "celdas" de código Python que podéis editar y volver a ejecutar, se pueden intercalar celdas de texto, fórmulas matemáticas, y hacer que gráficos se muestren incrustados en la misma pantalla. Estos archivos se guardan con extensión *.ipynb*, que pueden exportarse a diversos formatos estáticos como HTML o como código Python puro. (.py)

Los notebooks son muy útiles para la **"programación exploratoria"**, muy frecuente en ciencia e ingeniería

Para ejecutar IPython Notebook, si ejecutamos la consola (cmd.exe) y luego jupyter notebook, se abrirá el navegador por defecto del ordenador con el entorno IPython notebook. Lo más conveniente es ejecutar la consola (cmd.exe), acceder a la carpeta donde esté almacenado el *notebook* a ejecutar (para moverse por las carpeta desde la consola se utiliza el comando cd y el nombre de la carpeta a la que acceder: cd nombreCarpeta. Para ir a una carpeta en un nivel superior se utiliza el comando cd .. Para cambiar de unidad se utiliza F: siendo F la letra de la unidad) y ejecutar el comando jupyter notebook.

Por ejemplo, en la siguiente consola primero nos movemos a la unidad E: y después con cd a la carpeta donde tenemos los notebooks.



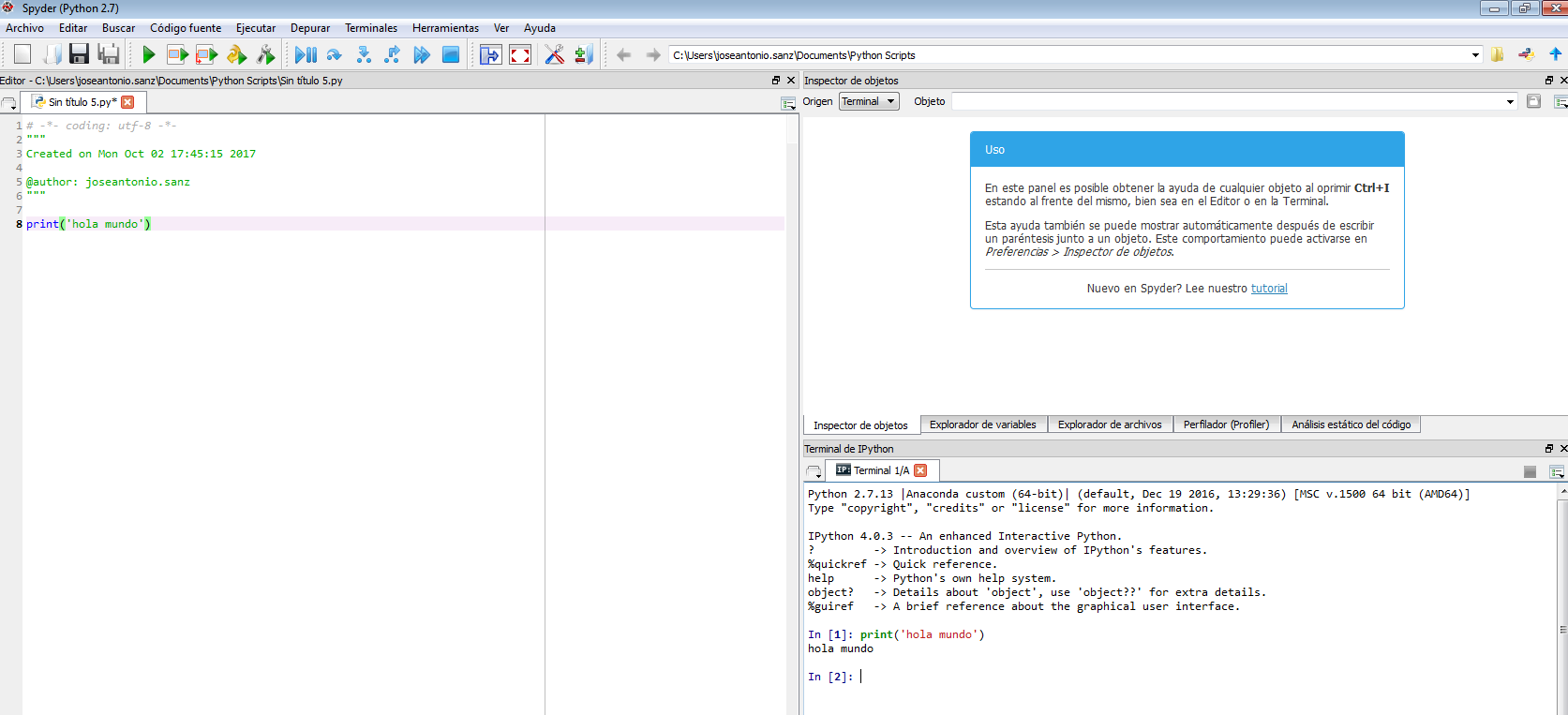
### Otros entornos de desarrollo para Python

Python no exige un editor específico y hay muchos modos y maneras de programar. En esta asignatura vamos a utilizar Jupyter Notebook, pero hay otros entornos de desarrollo muy útiles que conviene conocer.

#### Spyder

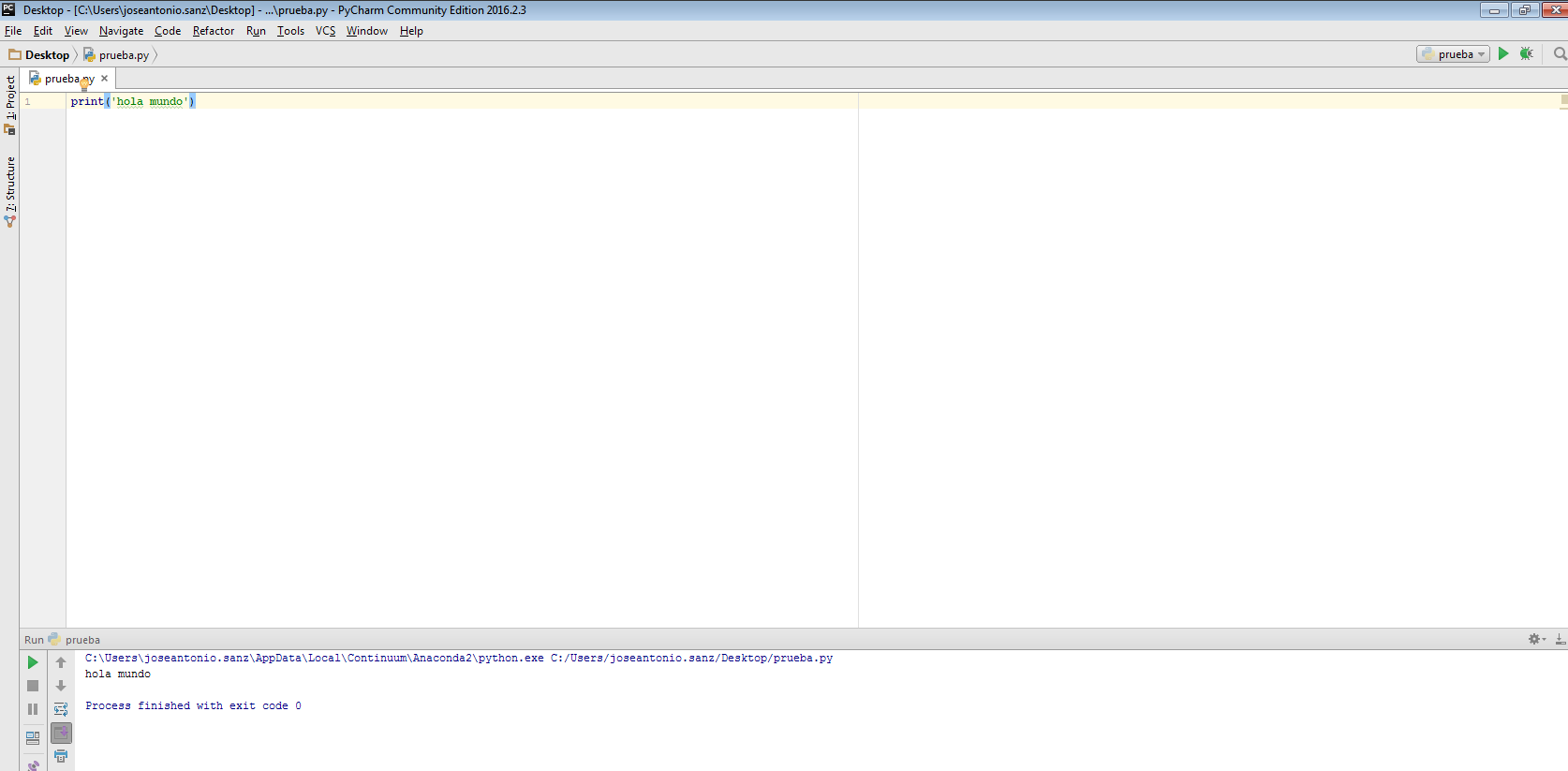
[Spyder](http://code.google.com/p/spyderlib/) es un entorno de desarrollo integrado (IDE) tipo MATLAB para computación científica con Python. Tiene muchas de las ventajas de los IDEs tradicionales como por ejemplo que la edición, ejecución y corrección de errores (debug) de código se realizan en el mismo entorno. Además, se pueden organizar los diferentes códigos en proyectos para mejorar la organización de los mismos.

*Spyder* viene con la distribución Anaconda, por lo que no es necesario instalarlo de forma adicional.



#### PyCharm

Otro buen editor es **PyCharm** edición *community*. Es otro ejemplo de IDE tipo MatLab y reúne las ventajas mencionadas para *Spyder*. Sin embargo, este IDE no viene integrado en el entorno Anaconda por lo que se debe descargar e instalar posteriormente. En enlace de descarga es el siguiente: https://www.jetbrains.com/pycharm/



### Comenzando a trabajar con Python

Ya conoces todo lo necesario para empezar a trabajar. Ahora ya puedes descargar el notebook (.ipynb) de la primera práctica y empezar a programar. Descárgalo a una carpeta en tu USB, posteriormente abre una consola (cmd.exe) y muévete a la unidad de tu USB. Ahora utilizando el comando cd muévete a la carpeta con el notebook y ejecuta “jupyter notebook”.