

# ★ Python 3 para impacientes ★

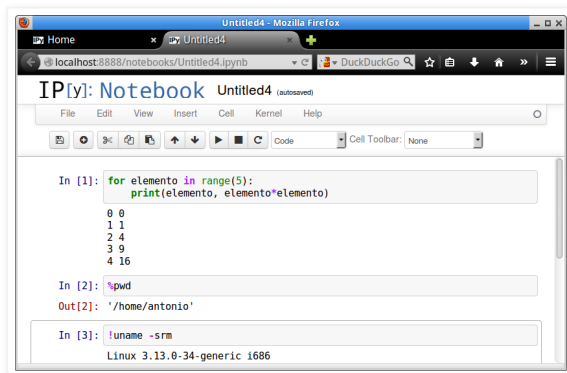


"Simple es mejor que complejo" (Tim Peters)

|        |         |         |         |            |       |
|--------|---------|---------|---------|------------|-------|
| Python | IPython | EasyGUI | Tkinter | JupyterLab | Numpy |
|--------|---------|---------|---------|------------|-------|

domingo, 17 de agosto de 2014

## Jupyter Notebook, la aplicación web



**Jupyter Notebook** es la herramienta que permite crear o utilizar cuadernos de trabajo en un navegador web. En un cuaderno de trabajo es posible combinar la ejecución de código con textos, cálculos matemáticos, representaciones gráficas y elementos multimedia.

Para facilitar el intercambio de información entre usuarios los cuadernos podrán convertirse a otros formatos extendidos tales como HTML, PDF, etc. Además, podrán compartirse mediante el visor [NBViewer](#).

### Iniciar Jupyter Notebook

Para iniciar Notebook desde la línea de comandos:

**\$ jupyter notebook**

A continuación, se pondrá en funcionamiento un servidor web y se ejecutará el navegador web que tengamos instalado por defecto (podrá ser Firefox, Opera o Chrome) y dentro de éste se abrirá la página del panel de control o **Dashboard** de Jupyter Notebook.

En la página del panel de Notebook se listará el contenido del directorio de trabajo actual y se mostrará en la parte superior un menú con las opciones siguientes: **"Notebooks"**, **"Running"** y **"Clusters"**.

- La pestaña **"Notebooks"** la utilizaremos para crear, abrir, importar o borrar cuadernos de trabajo, que podrán organizarse en distintos directorios del sistema.
- **"Running"** para detener los cuadernos en ejecución, comprobar su estado o para iniciar nuevas instancias de los mismos.
- Y la pestaña **"Clusters"** se utilizará con trabajos de computación paralela atendiendo a los perfiles disponibles en nuestra instalación IPython.

### Crear o abrir un cuaderno de trabajo

Para crear un cuaderno de trabajo haremos clic en el botón **"New Notebook"** y para abrir uno existente accederemos al directorio donde esté almacenado y haremos clic sobre el nombre del archivo.

A continuación, se creará un cuaderno nuevo, o bien, se cargará la información del cuaderno que hayamos seleccionado previamente.

Si el cuaderno es nuevo la aplicación le asignará un nombre predeterminado como **"Untitled1.ipynb"** y lo guardará en el directorio actual. Dicho nombre podrá cambiarse por otro haciendo clic sobre él en el mismo encabezado del cuaderno o en la opción **"Rename..."** del menú **"File"**.

Buscar



Python para impacientes

[Python](#)  
[IPython](#)  
[EasyGUI](#)  
[Tkinter](#)  
[JupyterLab](#)  
[Numpy](#)

Anexos

[Guía urgente de MySQL](#)  
[Guía rápida de SQLite3](#)

Entradas + populares

[Dar color a las salidas en la consola](#)

En Python para dar color a las salidas en la consola (o en la terminal de texto) existen varias posibilidades. Hay un método basado ...

[Instalación de Python, paso a paso](#)

Instalación de Python 3.6 A finales de 2016 se produjo el lanzamiento de Python 3.6 . El propósito de esta entrada es mostrar, pas...

[Variables de control en Tkinter](#)

Variables de control Las variables de control son objetos especiales que se asocian a los widgets para almacenar sus valore...

[Añadir, consultar, modificar y suprimir elementos en Numpy](#)

Acceder a los elementos de un array. [], [].  
... Acceder a un elemento de un array. Para acceder a un elemento se utiliz...

[Cálculo con arrays Numpy](#)

Numpy ofrece todo lo necesario para obtener un buen rendimiento cuando se trata de hacer cálculos con arrays. Por como está concebido...

[Operaciones con fechas y horas. Calendarios](#)

Los módulos datetime y calendar amplían las posibilidades del módulo time que provee funciones para manipular expresiones de ti...

[Tkinter: interfaces gráficas en Python](#)

Introducción Con Python hay muchas posibilidades para programar una interfaz gráfica de usuario ( GUI ) pero Tkinter es fácil d...

[Convertir, copiar, ordenar, unir y dividir arrays Numpy](#)

Esta entrada trata sobre algunos métodos que se utilizan en Numpy para convertir listas en arrays y viceversa; para copiar arrays d...

[Tkinter: Tipos de ventanas](#)

Ventanas de aplicación y de diálogos En la entrada anterior tratamos los distintos gestores de geometría que se utilizan para di...

[El módulo random](#)

El módulo random de la librería estándar de Python incluye un conjunto de funciones

En la parte superior del cuaderno, después del encabezado, se sitúan la **Barra de menú** y la **Barra de herramientas** con las opciones de uso más frecuente.

Y justamente debajo tendremos la primera celda o área de entrada (etiquetada con el término "In") que es el lugar donde empezaremos a escribir nuestro cuaderno de trabajo.

### Introducir información en una celda

¿Qué tipo de información podemos introducir en una celda?

Básicamente, se podrá introducir lo mismo que en una sesión interactiva con la [consola IPython](#) (código Python, [funciones mágicas](#) IPython, comandos del sistema, etc.); además de otro tipo de información usando el lenguaje de marcado ligero [Markdown](#) (como textos con formato, imágenes y archivos multimedia); gráficos creados a partir de datos; textos de encabezamientos de distintos tamaños y comandos para convertir la información a otros formatos.

Es necesario diferenciar entre el **modo Edición** y el **modo Comando**. El modo Edición se activa al hacer clic o **[Enter]** sobre una celda seleccionada (La celda seleccionada es la que está rodeada por el marco rectangular). Una celda activa mostrará un cursor vertical "|" para indicar el lugar de edición.

El modo Comando se activa cuando hacemos clic fuera del área de la celda que estemos editando o al presionar la tecla **[Esc]**.

En una celda se podrán escribir una o más líneas. Cuando se esté editando se avanzará a la siguiente línea con la tecla **[Enter]** y cuando existan varias líneas podremos retroceder a la línea anterior con la tecla **[Flecha Arriba]** o avanzar a la línea siguiente con **[Flecha Abajo]**.

### Ejecutar una celda

Para ejecutar la celda activa pulsaremos **[Ctrl+Enter]**. Si además deseamos que se inserte una celda nueva después de la ejecutada utilizaremos **[Alt+Enter]** o si queremos seleccionar la siguiente celda después de la ejecución presionaremos **[Mayúsculas+Enter]**.

### Guardar un cuaderno

Los cuadernos de Notebook se guardan automáticamente.

Sin embargo, si queremos guardar el trabajo en un punto determinado y continuar trabajando contando con la posibilidad de revertir a la situación anterior, guardaremos con la opción **"Save and Checkpoint"** y regresaremos al punto inicial con la opción **"Revert to Checkpoint"**. Estas opciones se encuentran en el menú **"File"**.

### Copiar un cuaderno

También, tendremos la posibilidad de trabajar con una copia exacta del cuaderno actual si elegimos la opción **"Make a Copy..."** del menú **"File"**. Es un modo rápido de obtener distintas versiones de un mismo cuaderno.

### Cerrar un cuaderno y salir

Para terminar la sesión de trabajo seleccionaremos la opción **"Close and Halt"** del menú **"File"**.

En el menú **"Help"** la opción **"User Interface Tour"** explica brevemente haciendo un tour el funcionamiento del interfaz de usuario y en **"Keyboard Shortcuts"** consultaremos las teclas de acceso rápido de Notebook.

[Ir al índice del tutorial de IPython](#)

Publicado por Pherkad en [12:10](#)



Etiquetas: [IPython](#), [Jupyter](#)

[Entrada más reciente](#)

[Inicio](#)

[Entrada antigua](#)

que permiten obtener de distintos modos números a...

Archivo

agosto 2014 (15) ▼

python.org



pypi.org



Sitios

- [ActivePython](#)
- [Anaconda](#)
- [Bpython](#)
- [Django](#)
- [Flask](#)
- [Ipython](#)
- [IronPython](#)
- [Matplotlib](#)
- [MicroPython](#)
- [Numpy](#)
- [Pandas](#)
- [Pillow](#)
- [PortablePython](#)
- [PyBrain](#)
- [PyCharm](#)
- [PyDev](#)
- [PyGame](#)
- [Pypi](#)
- [PyPy](#)
- [Pyramid](#)
- [Python.org](#)
- [PyTorch](#)
- [SciPy.org](#)
- [Spyder](#)
- [Tensorflow](#)
- [TurboGears](#)

