

★ Python 3 para impacientes ★



"Simple es mejor que complejo" (Tim Peters)

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

miércoles, 6 de agosto de 2014

Configuración de IPython



Argumentos del intérprete

Hay opciones y archivos que pueden añadirse como argumentos al iniciar IPython:

\$ ipython [opciones] archivos

Si la ejecución se hace sin opciones se ejecutarán una serie de archivos según una determinada secuencia y se leerán las configuraciones establecidas en archivos de configuración como `ipython_config.py`.

Para mostrar ayuda sobre los usos de IPython:

\$ ipython --help-all | --help

Para mostrar ayuda detallada sobre algún uso o componente concreto:

\$ ipython help opcion

Ejemplos:

\$ ipython help profile

\$ ipython help locate

Para consultar con comodidad la ayuda:

\$ ipython --help | less

Directorio IPython

Por defecto, IPython almacena sus archivos de configuración, historial de comandos y extensiones en el directorio:

```
~/.ipython/
%USERPROFILE%\ipython\
```

Como ese directorio puede ser otro si deseamos conocer en un momento dado cuál es:

\$ ipython locate

Variable de entorno `IPYTHONDIR`

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 valores...

[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

Si se establece esta variable con la ruta de un directorio, IPython la utilizará como ruta para los datos de usuarios.

La siguiente opción también se puede utilizar para sobrescribir el directorio predeterminado IPython:

\$ ipython --ipython-dir=<path>

Perfiles

La mayor parte de IPython puede ser ajustado a través de sus archivos de configuración. El intérprete IPython puede gestionar varios perfiles con una configuración y un historial de las sesiones de trabajo diferentes. Por defecto, si no se especifica ningún perfil se utilizará el predeterminado (profile_default) aunque podremos crear en cualquier momento un nuevo perfil desde el terminal.

Para crear un nuevo perfil:

\$ ipython profile create nombreperfil

Para iniciar con el nuevo perfil:

\$ ipython --profile=nombreperfil

Aunque los perfiles normalmente se almacenan en el directorio de IPython un perfil se puede mantener en el directorio donde estemos trabajando, por ejemplo, para distribuirlo con un proyecto.

Para localizar el directorio de un perfil en el sistema de archivos:

\$ ipython locate profile nombreperfil

Ejemplo:

\$ ipython locate profile default

Para listar todos los perfiles existentes:

\$ ipython profile list

Archivos de configuración

Los archivos de configuración se encuentran en la carpeta de un perfil.

Ejemplo de contenido de un archivo "ipython_config.py":

```
c = get_config()

c.TerminalIPythonApp.display_banner = True
c.InteractiveShellApp.log_level = 20
c.InteractiveShellApp.extensions = [
    'myextension'
]
c.InteractiveShellApp.exec_lines = [
    'import numpy',
    'import scipy'
]
c.InteractiveShellApp.exec_files = [
    'mycode.py',
    'fancy.ipynb'
]
c.InteractiveShell.autoindent = True
c.InteractiveShell.colors = 'LightBG'
c.InteractiveShell.confirm_exit = False
c.InteractiveShell.deep_reload = True
c.InteractiveShell.editor = 'nano'
c.InteractiveShell.xmode = 'Context'

c.PromptManager.in_template = 'In [{}]: '
c.PromptManager.in2_template = ' .D.: '
c.PromptManager.out_template = 'Out[{}]: '
c.PromptManager.justify = True

c.PrefilterManager.multi_line_specials = True

c.AliasManager.user_aliases = [
```

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](#)

```
('la', 'ls -al')  
]
```

Cambiar la configuración con la función %config

También se puede cambiar la configuración de IPython desde el propio intérprete con la [función mágica](#) %config:

```
: %config IPCompleter.greedy = True
```

Ficheros de inicio

Para ejecutar código al principio de cada sesión IPython podemos agregar scripts Python (.py) o IPython (.ipy) en el directorio "**startup**" (ejemplo: profile_default/startup/) del perfil de un usuario:

Ejemplo (inicio.ipynb):

```
%cd /home/usuario/proyectos  
!ls
```

(El script "inicio.ipynb" iniciará IPython en el directorio de trabajo que se indica en la primera línea y después listará su contenido.)

Los scripts de la carpeta "**startup**" se ejecutarán antes que cualquier otro código o scripts y lo harán siguiendo el orden alfabético de sus nombres.

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

Publicado por Pherkad en [4:29](#)



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

[Entrada más reciente](#)

[Inicio](#)

[Entrada antigua](#)