

★ Python 3 para impacientes ★



"Simple es mejor que complejo" (Tim Peters)

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

miércoles, 27 de agosto de 2014

Glosario de funciones mágicas IPython



Funciones mágicas orientadas a líneas

A

%alias

Definir un alias para un comando del sistema.

%alias_magic

Crear un alias para una línea o una celda mágica: %alias_magic [-l] [-c] name target.

%autocall

Activar/Desactivar la escritura sin paréntesis cuando se invocan funciones, métodos...

%automagic

Activar/Desactivar la escritura del carácter inicial "%" en las funciones mágicas

%autosave

Establecer el intervalo de guardado automático en Notebook (en segundos).

B

%bookmark

Gestionar el sistema de marcadores a directorios de IPython.

C

%cd

Cambiar el directorio de trabajo actual.

%clear

Borrar la pantalla de la consola.

%colors

Cambiar el esquema de colores para los avisos, sistema de información y excepciones.

%config

Configurar IPython.

%connect_info

Imprimir la información para la conexión de otros clientes con este kernel.

D

%debug

Depurar programas. %debug [--breakpoint FILE: LINE] [declaración [declaración ...]].

%dhist

Imprimir el historial de directorios visitados.

%dirs

Mostrar la pila de directorios actual (gestionada con %pushd y con %popd).

%doctest_mode

Activar/Desactivar el modo doctest para que IPython se comporte como la shell Python.

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

E**%ed**

Alias para la función mágica %edit.

%edit

Abrir un editor y ejecutar el código introducido (después de guardar y salir).

%env

Listar variables de entorno. Genera un diccionario varent = %env | print(varent["HOME"])

G**%gui**

Activar/Desactivar la integración del bucle de eventos de varios GUI para IPython.

%guieref

Mostrar una referencia básica sobre la consola gráfica (QT console).

H**%hist**

Alias para la función mágica %history.

%history

Mostrar el historial de entradas (y de salidas).

I**%install_default_config**

(Obsoleto). Usar: "ipython profile create <nombre>" para crear un nuevo perfil con las opciones predeterminadas.

%install_ext

Descargar e instalar una extensión desde una URL, por ejemplo.

%install_profiles

(Obsoleto). Usar: "ipython profile list" para ver perfiles disponibles; "ipython profile create <nombre>" para crear un nuevo perfil o "ipython -profile=<nombre>" para comenzar la sesión con el perfil indicado

K**%killbgscripts**

Matar todos los procesos en segundo plano iniciados por %%scripts y su familia.

L**%less**

Mostrar el contenido de un archivo.

%load

Cargar código en la sesión actual. Ejemplo: %load -r 7 script.py (carga línea 7).

%load_ext

Cargar una extensión IPython por su nombre de módulo.

%loadpy

Alias de la función mágica %load.

%logoff

Detener temporalmente el registro de entradas (y salidas) en un archivo tipo LOG.

%logon

Reiniciar el registro de entradas (y salidas) en un archivo tipo LOG.

%logstart

Comenzar el registro de entradas (y salidas) en un archivo tipo LOG.

%logstate

Mostrar el estado del sistema de registro de entradas (y salidas).

%logstop

Detener totalmente el registro de las entradas (y salidas) en un archivo tipo LOG.

%lsmagic

Listar las funciones mágicas disponibles en IPython.

M**%macro**

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](#)

Definir una nueva macro. Acepta rangos del historial.

%magic

Mostrar información general sobre las funciones mágicas.

%man

Mostrar información de ayuda de un comando del sistema (de las páginas "man").

%matplotlib

Activar soporte para trabajar con el módulo matplotlib.

%more

Mostrar en pantalla el contenido de un archivo.

N**%notebook**

Exportar historial para notebook o convertir cuadernos a otros formatos.

P**%page**

Mostrar información de un objeto aplicando convenciones de estilos de formatos de texto (pretty printing).

%pastebin

Subir código al servicio Github's Gist paste bin, devolviendo la URL.

%pdb

Controlar la ejecución automática del depurador interactivo pdb.

%pdef

Mostrar definición de llamada de un objeto. Ejemplo: %pdef func_suma → func_suma(a, b)

%pdoc

Mostrar cadena de documentación de un objeto.

%pfile

Mostrar información del archivo donde se define un objeto.

%pinfo

Mostrar información detallada de un objeto.

%pinfo2

Mostrar información extra de un objeto.

%popd

Establecer como directorio de trabajo el primer directorio de la pila y suprimirlo de la pila.

%pprint

Activar/Desactivar salida con convenciones de estilos de formato de texto (pretty printing).

%precision

Ajustar precisión decimal en la salida (pretty printing): Ejemplo: from math import pi; %precision 2; pi → 3,14

%profile

Mostrar perfil activo (obsoleto). Ahora usar: get_ipython().profile

%prun

Ejecutar una instrucción a través del generador de perfiles de código Python.

%psearch

Buscar objetos en el espacios de nombres utilizando el comodín "*"

%psource

Mostrar el código fuente de un objeto.

%pushd

Añadir a la pila de directorios el directorio actual y cambiar si se indica el directorio actual de trabajo

%pwd

Mostrar ruta del directorio actual de trabajo.

%pycat

Mostrar archivo resaltando con colores su sintaxis. Ejemplo %pycat script.py

%pylab

Cargar los módulos numpy/matplotlib para trabajar con datos y gráficos. %pylab [- no-import-all].

Q

%qtconsole

Abrir una nueva consola desde la consola QT actual.

%quickref

Mostrar ayuda de consulta rápida.

R**%recall**

Seleccionar un comando anterior para ejecutarlo o editarlo.

%rehashx

Actualizar la tabla de alias con todos los archivos ejecutables accesibles desde \$PATH.

%reload_ext

Actualizar una extensión IPython por su nombre de módulo.

%rep

Alias de la función mágica %recall.

%rerun

Volver a ejecutar una entrada anterior.

%reset

Restablecer espacio de nombres eliminando todos los nombres definidos por el usuario.

%reset_selective

Restablecer espacio de nombres eliminando nombres definidos por el usuario.

%run

Ejecutar un archivo desde IPython. Ejemplo: %run programa.py.

S**%save**

Guardar un conjunto de líneas o una macro en un archivo.

%sc

Ejecutar comando del sistema y capturar salida. %sc -l myfiles = ls.

%store

Persistencia ligera en variables de Python. Ejemplo %store var (guarda variable) y para usar en otra sesión: %store -r; print(var)

%sx

Ejecutar comando de la shell y capturar su salida.

%system

Ejecutar comando de la shell y capturar su salida.

T**%tb**

Mostrar información de rastreo de la última excepción producida.

%time

Medir el tiempo en ejecutarse una sentencia o expresión Python.

%timeit

Medir el tiempo en ejecutarse una sentencia o expresión Python.

U**%unalias**

Suprimir un alias de la tabla de alias.

%unload_ext

Descargar una extensión IPython por su nombre de módulo.

W**%who**

Mostrar el nombre de todas las variables.

%who_ls

Mostrar una lista ordenada con todas las variables.

%whos

Mostrar una tabla con información detallada de todas las variables.

X

%xdel

Eliminar una variable, tratando de borrarla de cualquier lugar.

%xmode

Cambiar modos para control de excepciones.

Funciones mágicas orientadas a celdas**%%!**

Ejecutar comando de la shell y capturar su salida.

%%HTML

Alias para la función %%HTML.

%%SVG

Alias para la función %%svg.

%%bash

Función mágica para escribir y ejecutar scripts bash.

%%capture

Ejecutar celda y capturar salida estándar/error: %capture [--no-stderr] [--no-stdout] [out].

%%debug

Depurar código %debug [--breakpoint FILE: LINE] [declaración [declaración ...]].

%%file

Alias de la función mágica %%writefile.

%%html

Tratar la celda como un bloque de código HTML

%%javascript

Ejecutar en la celda código Javascript.

%%latex

Tratar la celda como un bloque latex.

%%perl

Ejecutar en la celda código Perl.

%%prun

Ejecutar una declaración a través del generador de perfiles de código Python.

%%pypy

Ejecutar un script Pypy.

%%python

Ejecutar un script Python.

%%python3

Ejecutar un script Python3.

%%ruby

Ejecutar un script Ruby.

%%script

Ejecutar un script %shebang [--proc PROC] [--bg] [--err ERR] [--out OUT].

%%sh

Ejecutar un script sh.

%%svg

Tratar la celda como un literal SVG.

%%sx

Ejecutar comando de la shell y capturar su salida.

%%system

Ejecutar comando de la shell y capturar su salida.

%%time

Medir el tiempo de ejecución de una sentencia o expresión Python.

%%timeit

Medir el tiempo de ejecución de una sentencia o expresión Python.

%%writefile

Escribir el contenido de la celda a un fichero. %%writefile [-a] archivo.

[Ir al índice del tutorial de IPython](#)Publicado por Pherkad en [15:33](#)Etiquetas: [IPython](#), [Jupyter](#)[Entrada más reciente](#)[Inicio](#)[Entrada antigua](#)

2014-2020 | Alejandro Suárez Lamadrid y Antonio Suárez Jiménez, Andalucía - España
. Tema Sencillo. Con la tecnología de [Blogger](#).