



python™

Aprenda a programar con Python 3

## 1. Introducción a Python y su entorno de programación

Carolina Mañoso, Ángel P. de Madrid y Miguel Romero



# Índice

◆ Origen

◆ Características

◆ Versiones

◆ Instalación de Python 3

◆ Ejecutar código Python

- Desde la shell
- Desde el IDLE
- Práctica: Mi primer programa

◆ Código en Python



# Origen

- ◆ Fue desarrollado por Guido van Rossum.
- ◆ A finales de los 80.
- ◆ En el instituto de investigación holandés Centro para las Matemáticas y la Informática (CWI, *Centrum Wiskunde & Informatica*).
- ◆ Sucesor del lenguaje ABC.
- ◆ El nombre del lenguaje proviene de la afición de su creador por los humoristas británicos *Monty Python*.



# Características

- ◆ Lenguaje de programación de alto nivel.
- ◆ Lenguaje interpretado.
- ◆ Tipado dinámico.
- ◆ Sintaxis clara y expresiva.
- ◆ Multiparadigma.
- ◆ Multiplataforma.
- ◆ Licencia de código abierto.
- ◆ Extensa biblioteca.

*"Python es un lenguaje de programación poderoso y fácil de aprender. Cuenta con estructuras de datos eficientes y de alto nivel y un enfoque simple pero efectivo a la programación orientada a objetos. La elegante sintaxis de Python y su tipado dinámico, junto con su naturaleza interpretada, hacen de éste un lenguaje ideal para scripting y desarrollo rápido de aplicaciones en diversas áreas y sobre la mayoría de las plataformas."*



# Características

- ◆ La web oficial de Python: <https://www.python.org>

The screenshot shows the official Python website. At the top is a dark blue header with the Python logo and the word "python" in white. To the right are a search bar with a magnifying glass icon, a "Search" button, a "GO" button, and a "Socialize" button. Below the header is a navigation bar with tabs: "About", "Downloads", "Documentation", "Community", "Success Stories", "News", and "Events". The main content area has two sections. On the left, there's a code snippet demonstrating Python arithmetic:

```
# Python 3: Simple arithmetic
>>> 1 / 2
0.5
>>> 2 ** 3
8
>>> 17 / 3 # classic division returns a float
5.666666666666667
>>> 17 // 3 # floor division
5
```

On the right, there's a section titled "Intuitive Interpretation" with the following text:

Calculations are simple with Python, and expression syntax is straightforward: the operators `+`, `-`, `*` and `/` work as expected; parentheses `()` can be used for grouping. [More about simple math functions in Python 3.](#)

At the bottom of the page, there's a footer with the text: "Python is a programming language that lets you work quickly and integrate systems more effectively. [» Learn More](#)".

# Características

- ◆ Es administrado por la Python Software Foundation (PSF): <https://www.python.org/psf-landing/>

The screenshot shows the homepage of the Python Software Foundation. At the top, there is a navigation bar with links for About, Sponsorship, Membership, Donations & Grants, Legal, Media, and PSF Blog. To the right of the navigation bar is a search bar with a magnifying glass icon and a 'GO' button, along with links for Socialize and Sign In. Below the navigation bar, a large yellow banner states: "The Python Software Foundation is an organization devoted to advancing open source technology related to the Python programming language." The main content area is divided into four sections: "Become a Member" (with a "Membership FAQ" button), "Donate" (with a "How to Contribute" button), "Volunteer" (with a "How to Volunteer" button), and "Sponsors" (with a "Sponsorship Possibilities" button). At the bottom, there are two more sections: "PSF Grants Program" (describing grant proposals for Python projects) and "PSF News" (listing available Python versions: 3.6.4rc1, 3.7.0a3, 3.7.0a2, and 3.6.3).

**Become a Member**  
Help the PSF promote, protect, and advance the Python programming language and community!

[Membership FAQ](#)

**Donate**  
Assist the foundation's goals with a donation. The PSF is a recognized 501(c)(3) non-profit organization.

[How to Contribute](#)

**Volunteer**  
Learn how you can help the PSF and the greater Python community!

[How to Volunteer](#)

**Sponsors**  
Without our sponsors we wouldn't be able to help the Python community grow and prosper.

[Sponsorship Possibilities](#)

**PSF Grants Program**  
The Python Software Foundation welcomes grant proposals for projects related to the development of Python, Python-related technology, and educational resources.

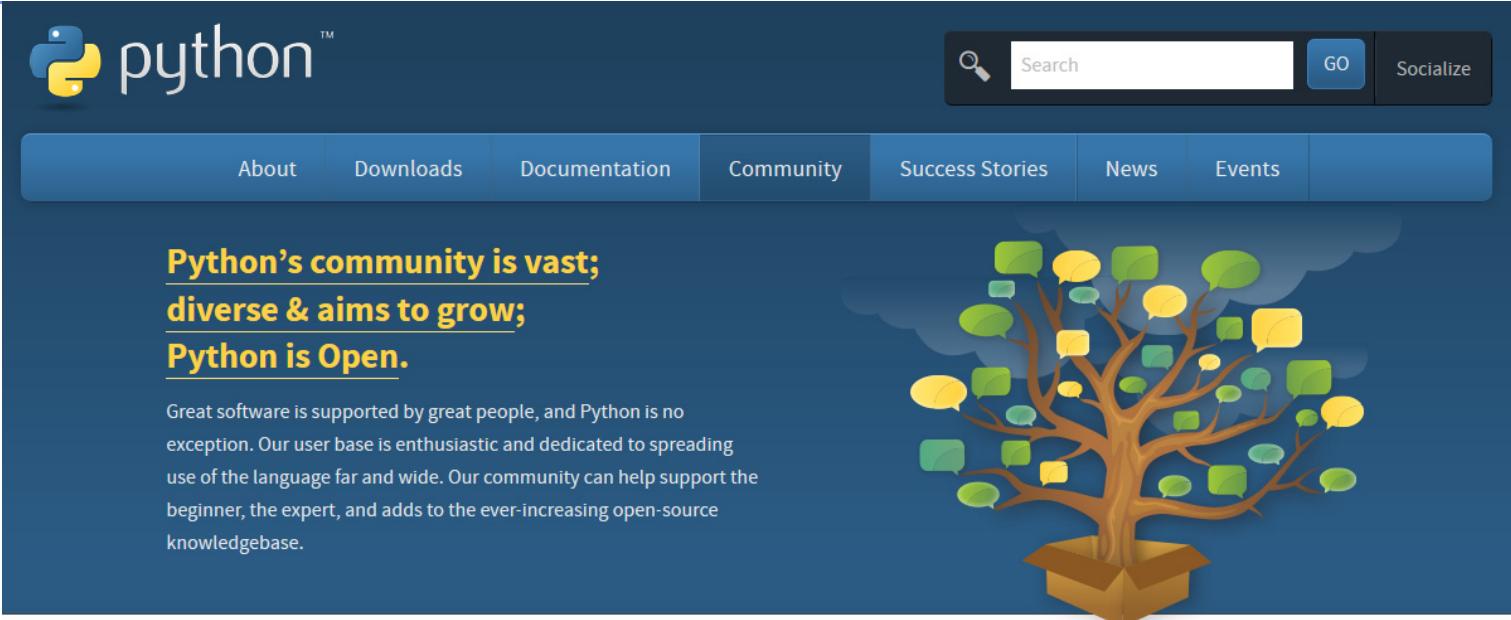
**PSF News**  
Python 3.6.4rc1 and 3.7.0a3 now available for testing  
Python 3.7.0a2 now available for testing  
Python 3.6.3 is now available



# Características

## ◆ Respaldado por una gran comunidad:

<https://www.python.org/community/>



The screenshot shows the Python.org community page. At the top, there's a navigation bar with links for About, Downloads, Documentation, Community (which is highlighted), Success Stories, News, and Events. To the right of the navigation is a search bar and a "GO" button. Below the navigation, a large heading reads: "Python's community is vast; diverse & aims to grow; Python is Open." To the right of this text is a graphic of a tree with many speech bubbles of different colors (green, yellow) growing from its branches, symbolizing communication and growth. Below the heading, there's a paragraph of text about the Python community's support and diversity. At the bottom of the page, there are four sections: "Getting Started", "Success Stories", "Python Weekly", and "Internet Relay Chat".

### Getting Started

New to the community? Here are some great places to get started:

- Python FAQs
- Attend a Conference
- Diversity Statement

### Success Stories

*My experience with the Python community has been awesome. I have met some fantastic people through*

### Python Weekly

Python Weekly is a free weekly email newsletter featuring curated news, articles, new releases, jobs, and more. Curated by Rahul Chaudhary every Thursday.

Go to [pythonweekly.com](http://pythonweekly.com) to sign up.

### Internet Relay Chat

Freenode IRC hosts several channels. Select an IRC client, register your nickname with Freenode, and you can be off and running!

### Freenode IRC General



# Versiones

- ◆ Python 3.0 fue lanzado en 2008.
- ◆ Python 2.7 fue la versión final a mediados de 2010. No habrá Python 2.8.
- ◆ Son intencionalmente incompatibles.
- ◆ Hay conversores.
- ◆ ¿Qué es nuevo en Python 3?
  - Limpieza de inconsistencias y redundancias.
  - Sintaxis más clara.
  - Eliminar errores de diseño ("print" como función,...).
  - Todas las cadenas de texto son Unicode por defecto.

<https://docs.python.org/3/whatsnew/3.0.html>

<https://wiki.python.org/moin/Python2orPython3>

**“Python 2.x es el legado, Python 3.x es el presente  
y el futuro del lenguaje.”**



# Instalación de Python 3

- ◆ Muchas distribuciones de **Linux** instalan Python de forma predeterminada.

- Para comprobar si tenemos instalado Python 3 ejecutamos desde la línea de comandos:

```
$ which python3
```

- Si está instalado aparecerá en pantalla la ruta de instalación, habitualmente en:

```
/usr/bin/python3
```

- Si no lo está, no aparecerá ningún mensaje y deberemos instalarlo ejecutando:

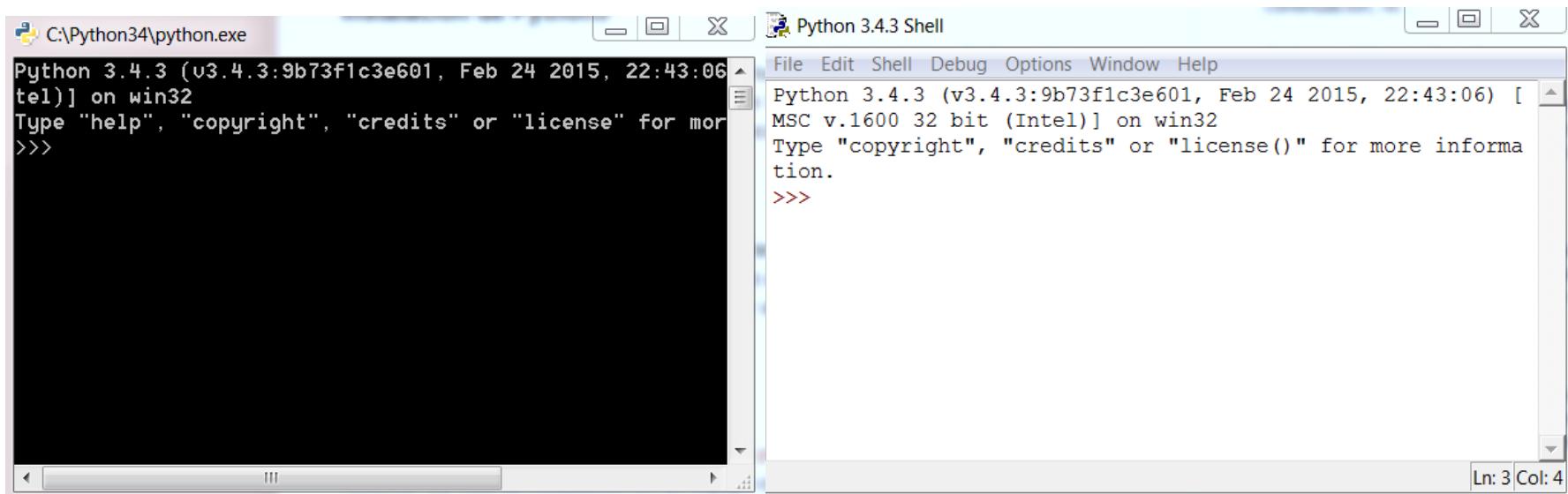
```
$ sudo apt-get install python3
```

- ◆ Para **Windows**, podemos descargar e instalar desde la página oficial la versión más reciente. La ruta de instalación habitual, aunque se puede cambiar: c:\Python37.



# Ejecutar código Python

- ◆ Existen varias formas de ejecutar código Python:
  - Desde la línea de comandos.
  - Desde un entorno de desarrollo integrado (IDE):
    - ◆ Funcionalidades como editor, comprobación de sintaxis, facilidades de depuración, ejecutar sin abandonar el editor, ...
    - ◆ IDLE por defecto, otros entornos son iPython, PyDEV, SPE, Ninja-IDE, BOA constructor, PyCharm...
- ◆ Los signos **>>>** indican donde introducir las sentencias.



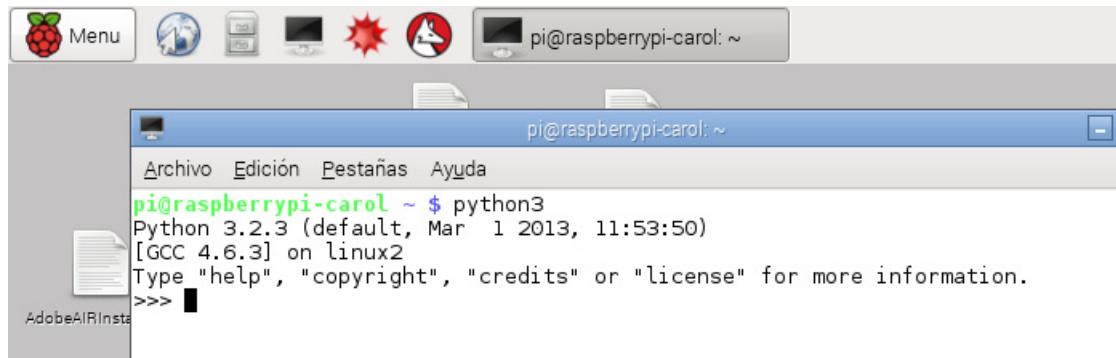
The screenshot shows two windows side-by-side. The left window is titled 'C:\Python34\python.exe' and displays the Python 3.4.3 shell prompt: 'Python 3.4.3 (v3.4.3:9b73f1c3e601, Feb 24 2015, 22:43:06) [MSC v.1600 32 bit (Intel)] on win32'. It also shows the standard Python copyright and license information. The right window is titled 'Python 3.4.3 Shell' and has a similar title bar. Both windows have a menu bar with File, Edit, Shell, Debug, Options, Window, and Help. The main area of both windows shows the same Python 3.4.3 shell environment, including the prompt 'Python 3.4.3 (v3.4.3:9b73f1c3e601, Feb 24 2015, 22:43:06) [MSC v.1600 32 bit (Intel)] on win32' and the copyright/license text. The bottom right corner of the right window indicates 'Ln: 3 Col: 4'.



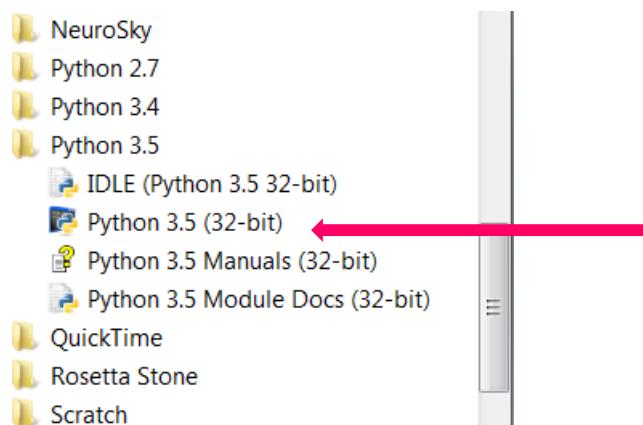
# Ejecutar código Python desde la shell

## ◆ Desde Linux:

- Escribimos la orden `python3`, para ejecutar Python. Para salir `exit()`.



## ◆ Desde Windows:

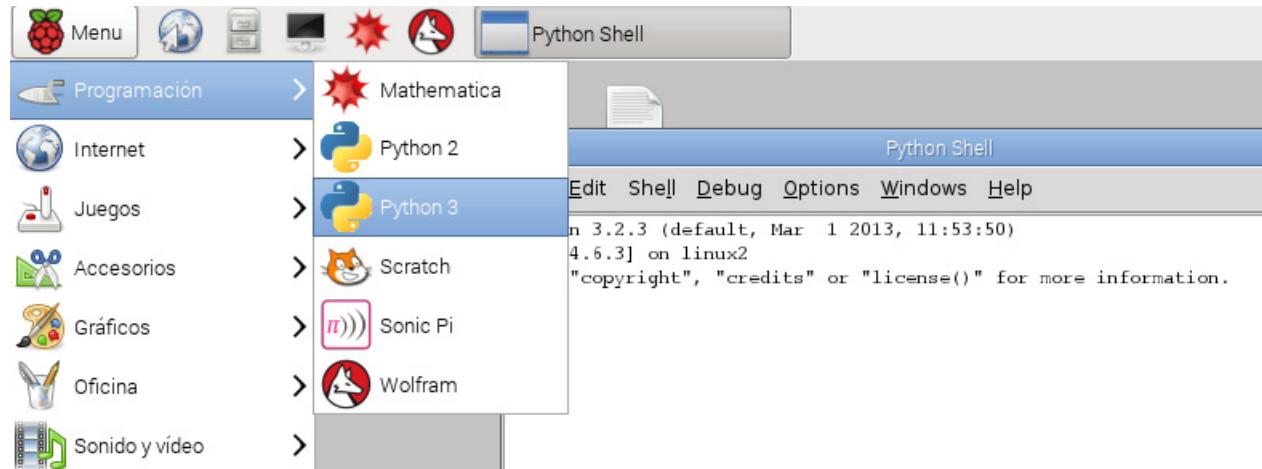


# Ejecutar código Python desde IDLE

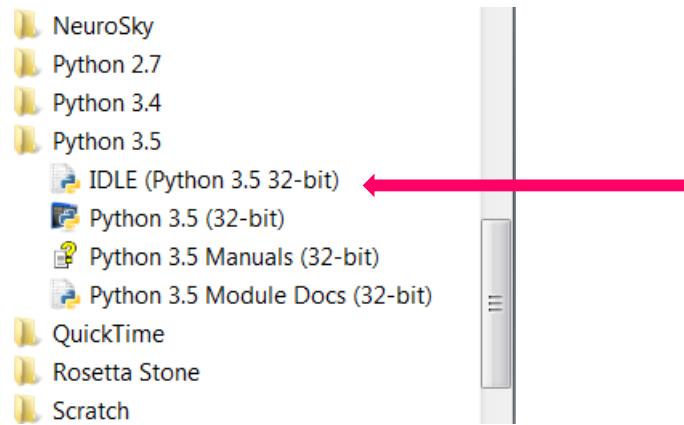
## ◆ Desde Linux:

- Escribimos la orden `idle3` o se selecciona desde el menú.

Para salir `exit()`.



## ◆ Desde Windows:



# Ejecutar código Python: mi primer programa (1/2)

## ◆ De forma interactiva (línea a línea):

```
>>> print("hola mundo")
```

## ◆ Escribir el código en una fichero con extensión .py y ejecutarlo.

### ■ Desde Windows:

- Para escribir el fichero se debe utilizar el bloc de notas o desde el editor del IDLE.
- Los archivos .py ya están asociados al intérprete, por lo que basta con hacer doble clic sobre ellos para ejecutarlos o desde el menú Run del fichero seleccionamos Run Module.

The screenshot shows two windows side-by-side. On the left is the 'primerprograma.py - C:\Python34\Scripts\primerprogra...' window of the IDLE editor. It displays the code `print("hola mundo")`. The 'Run' menu is open, and the 'Run Module F5' option is highlighted. On the right is the 'Python 3.4.3 Shell' window, which shows the output of the executed code: 

```
Python 3.4.3 (v3.4.3:9b73f1c3e601, Feb 24 2015, 22:43:06) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
hola mundo desde fichero
>>>
```



# Ejecutar código Python: mi primer programa (2/2)

## ■ Desde Linux:

- ◆ Para escribir el fichero, `namefile.py`, se debe usar un editor desde la ventana de comandos o el editor integrado del IDLE.
- ◆ Desde la ventana de comandos, navegamos al directorio donde está el fichero y ejecutamos la orden `python3 namefile.py` o desde IDLE el menú Run del fichero seleccionamos Run Module.
- ◆ Para que el sistema operativo abra el fichero con el intérprete adecuado directamente debemos añadir una primera línea de código en el archivo, llamada *shebang* (`#!`), que lo indica:

```
#!/usr/bin/python3
```

- ◆ Y debemos hacer el fichero ejecutable con:  
`chmod +x namefile.py`
- ◆ **Ojo, ya que `#!/usr/bin/python` llama al intérprete de Python 2.**



# Práctica

## ◆ Ejecute Python:

- Abra el intérprete desde la línea de comandos o desde el entorno de desarrollo integrado (IDE). Estudie sus posibilidades.
- Pruebe a ejecutar **de forma interactiva** alguna instrucción, por ejemplo:  
`>>> print("hola mundo")`
- Pruebe a escribir y ejecutar **el archivo** llamado `primerprograma_linux.py` que haga lo mismo.



# Código en Python (1/3)

- ◆ Por defecto los archivos fuente son tratados como codificados en UTF-8.
- ◆ La guía de estilo es PEP 8:
  - El código se escribe en cada línea desde el primer espacio. Cuando es necesario el sangrado se usan 4 espacios (o múltiplos).
  - El tamaño de la línea es máximo de 79 caracteres.
  - Comentarios, a ser posible, en una sola línea (#), si abarcan varias líneas, debemos comenzarlos y terminarlos con tres comillas (simples o dobles).
  - Usar espacios alrededor de operadores y luego de comas.
  - Usar líneas en blanco para separar funciones y clases.
  - ...



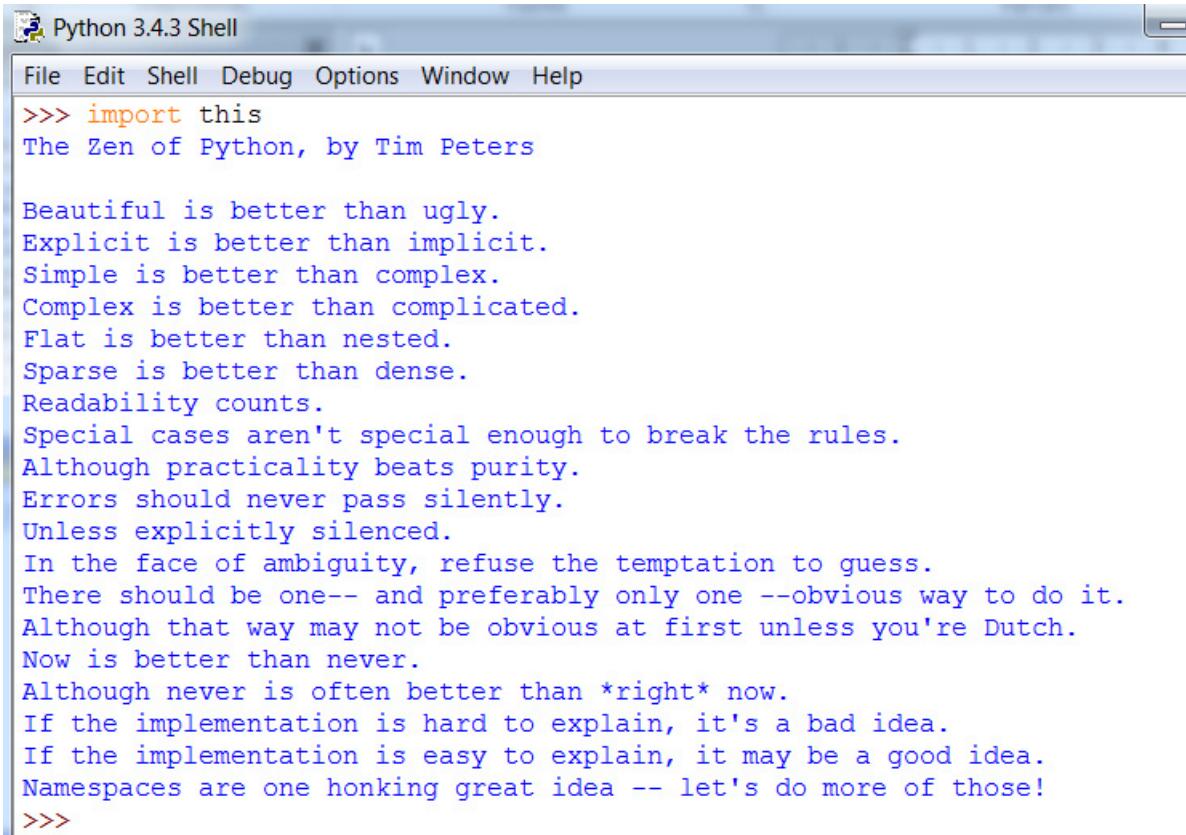
# Código en Python (2/3)

- ◆ Un programa en Python consiste en una serie de instrucciones que se ejecutan de arriba abajo.
- ◆ Utiliza variables que permiten almacenar información para usarla más tarde.
- ◆ Se puede controlar el camino de ejecución usando bucles y sentencias de control.
- ◆ Se puede reutilizar código usando funciones, que hacen los programas más fáciles de entender y mantener.
- ◆ Se puede importar módulos y usar métodos ya definidos para realizar parte del trabajo.
- ◆ Consejos de Python para programar bien:  
`>>> import this`



# Código en Python (3/3)

## ◆ Zen de Python de Tim Peters (Desarrollador de Python)



```
Python 3.4.3 Shell
File Edit Shell Debug Options Window Help
>>> import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!
>>>
```



# Aviso



*Aprenda a programar con Python 3* by C. Mañoso, A. P. de Madrid, M. Romero is licensed under a [Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional License](#).

Esta colección de transparencias se distribuye con fines meramente docentes.

Todas las marcas comerciales y nombres propios de sistemas operativos, programas, hardware, etc. que aparecen en el texto son marcas registradas propiedad de sus respectivas compañías u organizaciones.



*Aprenda a programar con Python 3* by C. Mañoso, A. P. de Madrid, M. Romero is licensed under a [Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional License](#).