

Clase 2

Programar en Python

Instalación

Instalación

En esta clase veremos una guía paso a paso de instalación de Python. Si nuestro sistema de trabajo es alguna distribución de Linux o MAC, probablemente ya tengamos preinstalada alguna versión de Python, no obstante, veremos cómo actualizar a la versión deseada.

Si en cambio trabajamos en Windows, tendremos que realizar la instalación desde cero.

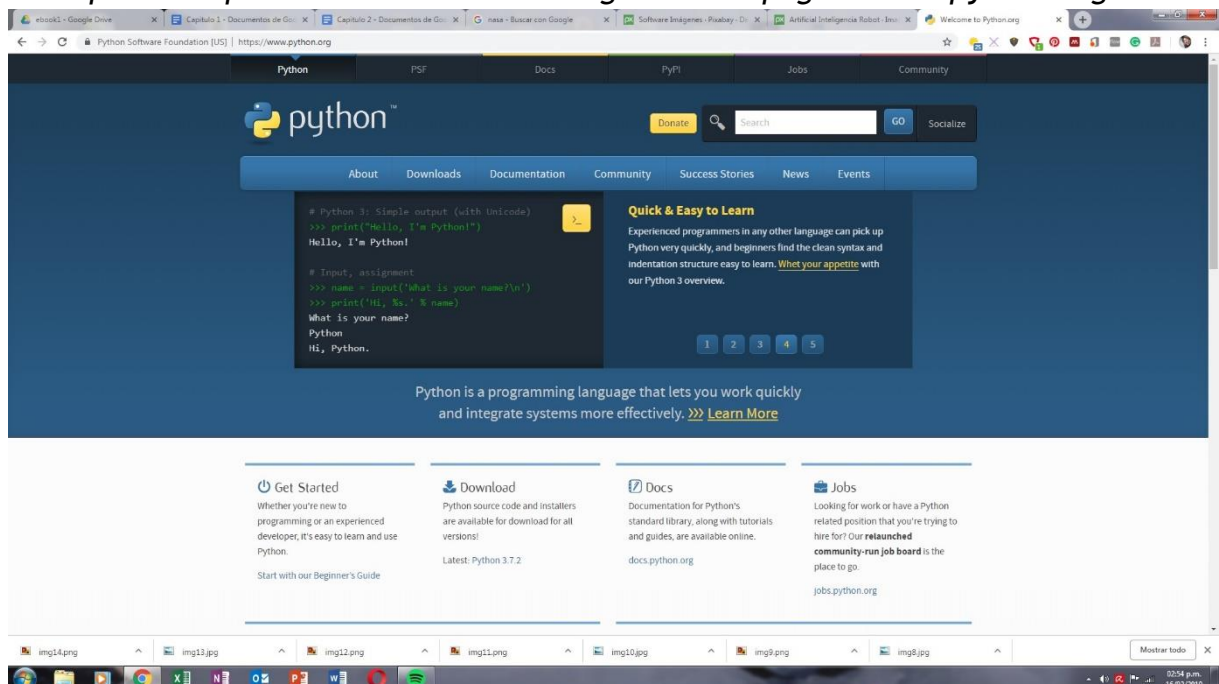
Una duda que puede surgir al querer instalar Python es ¿qué versión me conviene utilizar? Para aclarar un poco el panorama, veamos el siguiente video:

<https://youtu.be/ZznSGjAutQU>

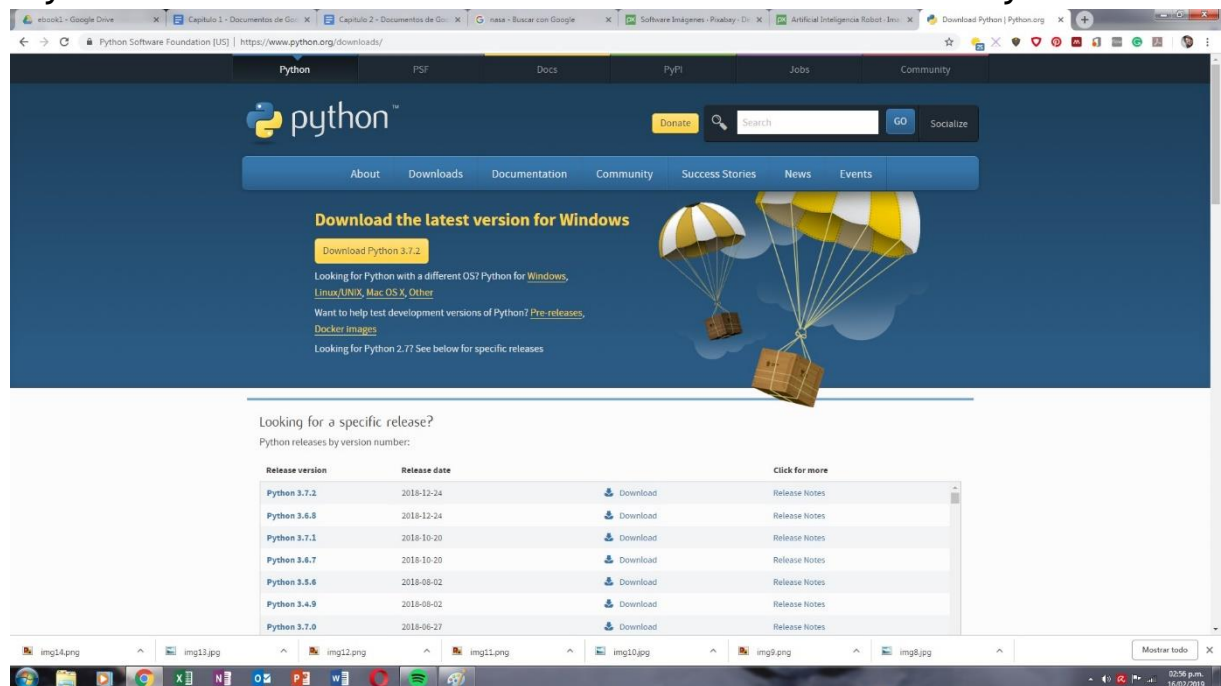
Instalar Python en Windows

A continuación se explicará cómo realizar la instalación de Python desde cero en Windows. Desde elegir la versión adecuada hasta comprobar que la instalación se realizó correctamente, pasando por todas las opciones necesarias para que el intérprete funcione sin inconvenientes.

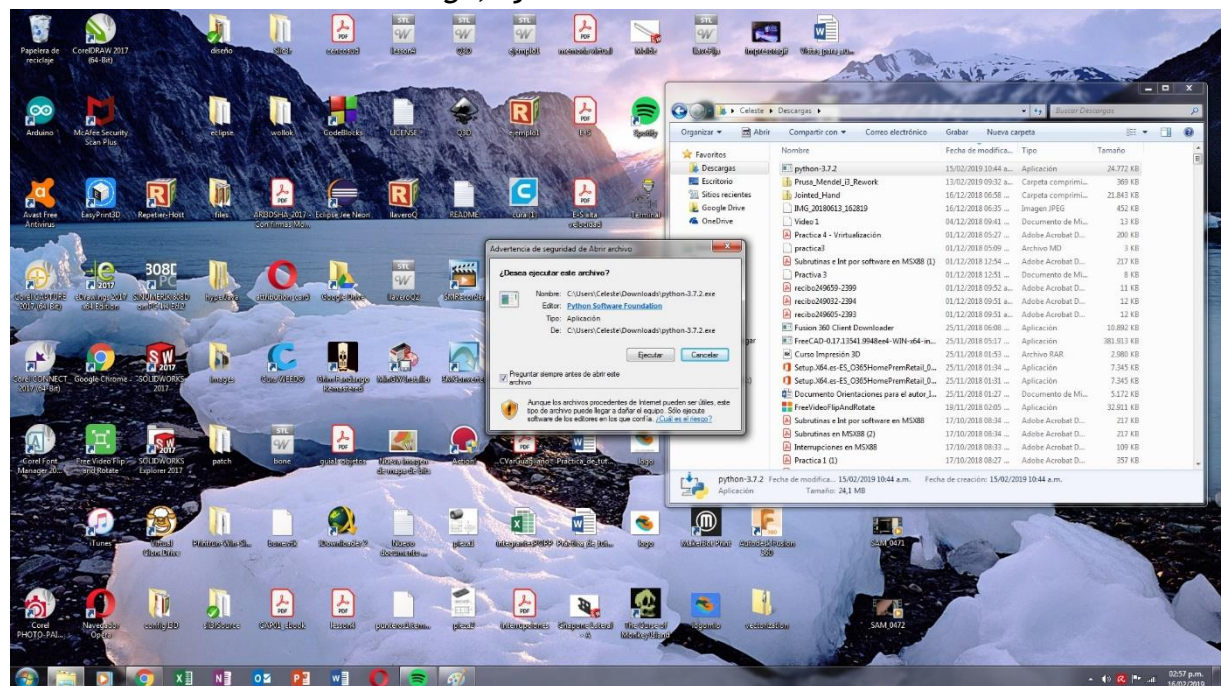
1. Lo primero que deberemos hacer es ingresar a la página www.python.org.



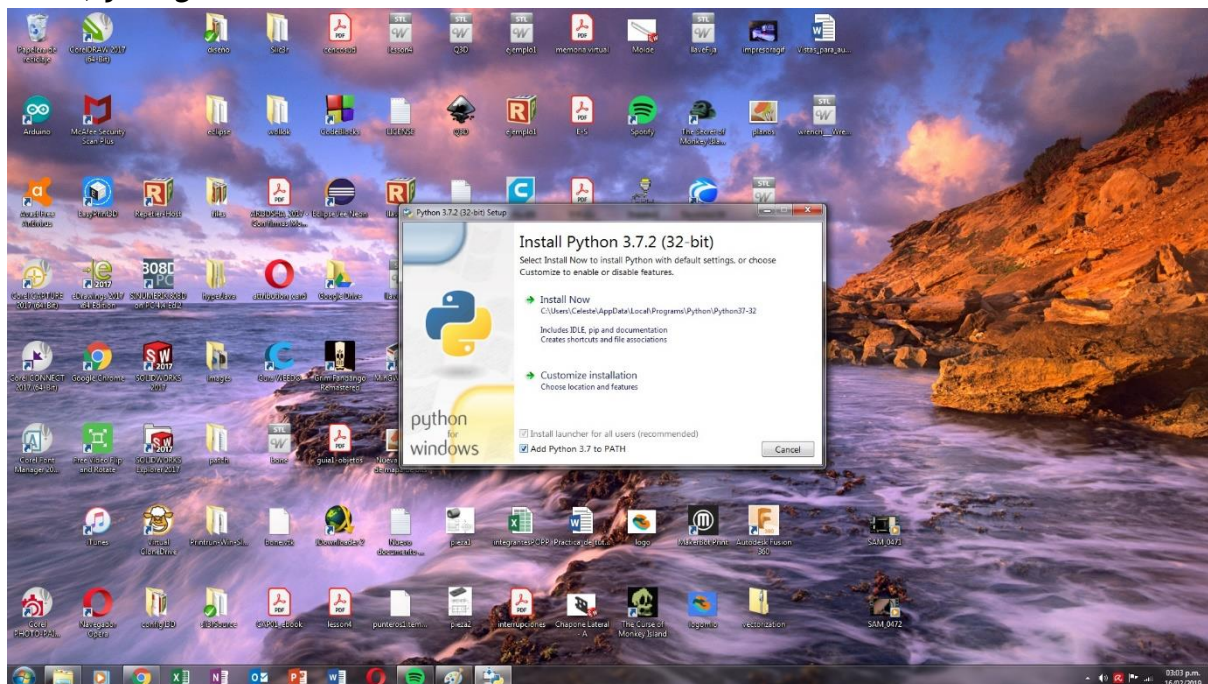
2. Desde la página podremos acceder a la descarga del instalador de la versión que hayamos decidido utilizar. Para ello hacemos clic sobre “download Python 3.x.x”



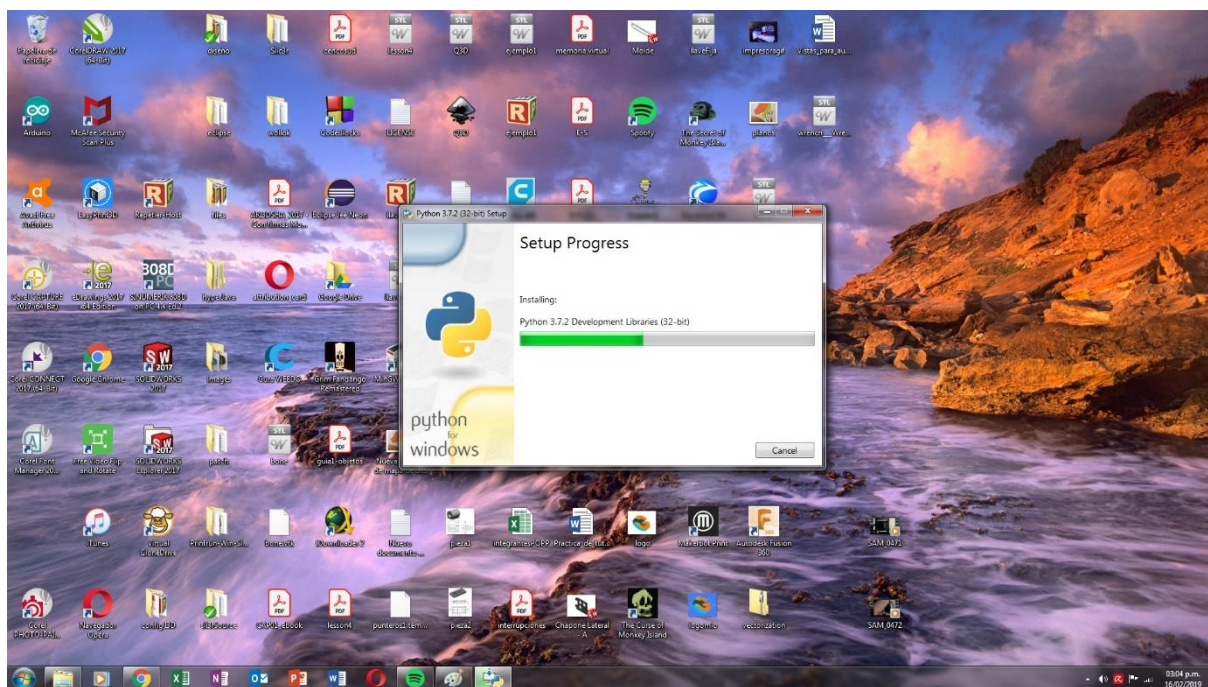
3. Una vez realizada la descarga, ejecutamos el instalador



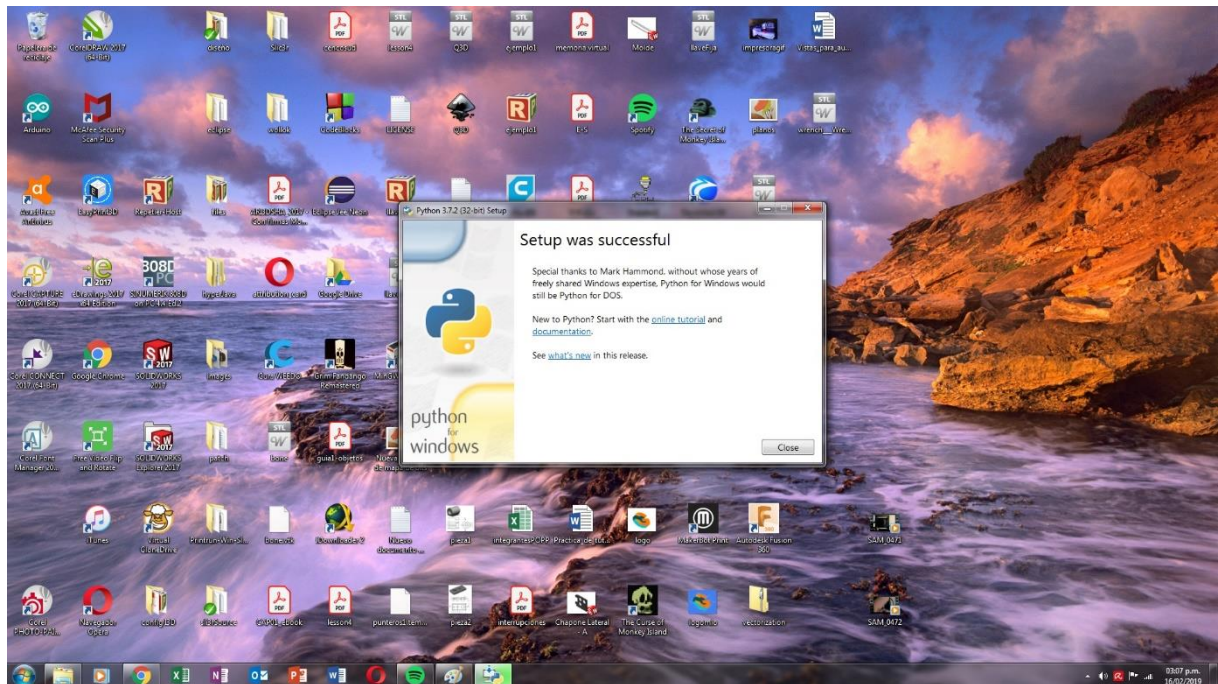
4. Al comenzar el proceso de instalación, tildamos la opción “Add Python 3.x.x to PATH”, y luego clickeamos en el botón “Install Now”



5. Veremos el avance de la instalación en todo momento, esto demora varios minutos en Windows.



6. Al finalizar el proceso de instalación, aparecerá un cartel informando el éxito de la misma y ya estaremos en condiciones de utilizar Python en nuestro sistema Windows.



Testeo de Python en Windows

Para comenzar a utilizar Python, ingresando desde el acceso directo en el menú inicio o bien desde el símbolo del sistema tecleando Python3 la pantalla del intérprete se verá muy similar a la terminal de Linux o a la vieja pantalla de DOS.



Una vez en esta pantalla, podemos comenzar a experimentar con códigos sencillos.

Algunos ejemplos sencillos para ver el correcto funcionamiento de Python se basan en realizar algunos cálculos, declarar variables y mostrar datos por pantalla, lo que se visualiza con los símbolos '>>>' previos al texto es lo que se ingresa por teclado y lo que se simboliza en un renglón sin estar acompañado de dichos símbolos son los resultados obtenidos luego de presionar enter:

```
>>>5+3
8
>>>print('Hola Python')
Hola Python
>>>a=3
>>>b=12
>>>a+b
15
>>>a='Hello'
>>>print(a)
Hello
```

Instalar Python en Linux

Como se dijo con anterioridad, Python viene preinstalado tanto en Linux como en MAC OS. No obstante, dependiendo de la versión de Python que queramos utilizar y de la versión de Linux o MAC que estemos utilizando, se deberá realizar la actualización o no de la versión instalada.

En este caso, utilizamos como sistema base Ubuntu 18.04,

1. Revisemos la versión de Python que viene instalada con nuestro sistema operativo. Para ello, debemos tipear desde la terminal de Linux:

```
python3 --version
```

Si Python3 se encuentra ya instalado en el sistema, se verá la información de la version, caso contrario el sistema informará que no encontró dicho paquete.

2. Para instalar Python 3 en Linux, desde la Terminal escribimos:

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

```
celeste@celeste-HP-Pavilion-11-x360-PC: ~
Archivo Editar Ver Buscar Terminal Ayuda
celeste@celeste-HP-Pavilion-11-x360-PC:~$ sudo apt-get install python3.6
[sudo] password for celeste: █
```

se pedirá la clave del usuario para proceder con la instalación

3. Linux informará sobre el paquete encontrado y deberemos confirmar la instalación.

```
celeste@celeste-HP-Pavilion-11-x360-PC: ~
Archivo Editar Ver Buscar Terminal Ayuda
celeste@celeste-HP-Pavilion-11-x360-PC:~$ sudo apt-get install python3.6
[sudo] password for celeste:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libpython3.6-minimal libpython3.6-stdlib python3.6-minimal
Paquetes sugeridos:
  python3.6-venv python3.6-doc binfmt-support
Se instalarán los siguientes paquetes NUEVOS:
  libpython3.6-minimal libpython3.6-stdlib python3.6 python3.6-minimal
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
Se necesita descargar 4.473 kB de archivos.
Se utilizarán 23,0 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] █
```

S y enter o simplemente enter para confirmar la instalación.

4. El proceso de instalación en Linux dura algunos segundos y se muestra todo el proceso por la Terminal.

```
Archivo Editar Ver Buscar Terminal Ayuda
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libpython3.6-minimal libpython3.6-stdlib python3.6-minimal
Paquetes sugeridos:
  python3.6-venv python3.6-doc binfmt-support
Se instalarán los siguientes paquetes NUEVOS:
  libpython3.6-minimal libpython3.6-stdlib python3.6 python3.6-minimal
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 6 no actualizados.
Se necesita descargar 4.473 kB de archivos.
Se utilizarán 23,0 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://ppa.launchpad.net/jonathonf/python-3.6/ubuntu xenial/main amd64 libpython3.6-minimal amd64 3.6.7-1-16.04.york3 [577 kB]
Des:2 http://ppa.launchpad.net/jonathonf/python-3.6/ubuntu xenial/main amd64 python3.6-minimal amd64 3.6.7-1-16.04.york3 [1.687 kB]
Des:3 http://ppa.launchpad.net/jonathonf/python-3.6/ubuntu xenial/main amd64 libpython3.6-stdlib amd64 3.6.7-1-16.04.york3 [1.969 kB]
Des:4 http://ppa.launchpad.net/jonathonf/python-3.6/ubuntu xenial/main amd64 python3.6 amd64 3.6.7-1-16.04.york3 [241 kB]
Descargados 4.473 kB en 21s (212 kB/s)
Seleccionando el paquete libpython3.6-minimal:amd64 previamente no seleccionado.
(Leyendo la base de datos ... 180460 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../libpython3.6-minimal_3.6.7-1-16.04.york3_amd64.deb ...
Desempaquetando libpython3.6-minimal:amd64 (3.6.7-1-16.04.york3) ...
Seleccionando el paquete python3.6-minimal previamente no seleccionado.
Preparando para desempaquetar .../python3.6-minimal_3.6.7-1-16.04.york3_amd64.deb ...
Desempaquetando python3.6-minimal (3.6.7-1-16.04.york3) ...
Seleccionando el paquete libpython3.6-stdlib:amd64 previamente no seleccionado.
Preparando para desempaquetar .../libpython3.6-stdlib_3.6.7-1-16.04.york3_amd64.deb ...
Desempaquetando libpython3.6-stdlib:amd64 (3.6.7-1-16.04.york3) ...
Seleccionando el paquete python3.6 previamente no seleccionado.
Preparando para desempaquetar .../python3.6_3.6.7-1-16.04.york3_amd64.deb ...
Desempaquetando python3.6 (3.6.7-1-16.04.york3) ...
Procesando disparadores para man-db (2.7.5-1) ...
Procesando disparadores para gnome-menus (3.13.3-6ubuntu3.1) ...
Procesando disparadores para desktop-file-utils (0.22-1ubuntu5.2) ...
Procesando disparadores para bamfdaemon (0.5.3-bzr0+16.04.20180209-0ubuntu1) ...
Rebuilding /usr/share/applications/bamf-2.index...
Procesando disparadores para mime-support (3.59ubuntu1) ...
Configurando libpython3.6-minimal:amd64 (3.6.7-1-16.04.york3) ...
Configurando python3.6-minimal (3.6.7-1-16.04.york3) ...
Configurando libpython3.6-stdlib:amd64 (3.6.7-1-16.04.york3) ...
Configurando python3.6 (3.6.7-1-16.04.york3) ...
celeste@celeste-HP-Pavilion-11-x360-PC:~$
```

Al finalizar el proceso se vuelve a ver el cursor en la Terminal de Linux

Testeo de Python en Linux

Ahora que ya se completó la instalación de Python 3 en el sistema, se pueden realizar algunos tests.

Para comenzar a utilizar Python, ingresando a la Terminal de Linux, escribiremos:

Python3

y de esta forma iniciaremos el intérprete de Python.

De la misma forma que en Windows, se pueden realizar algunos tests básicos tales como algunos cálculos, declaración de variables, mostrar texto por pantalla:


```
celeste@celeste-HP-Pavilion-11-x360-PC: ~
Archivo Editar Ver Buscar Terminal Ayuda
celeste@celeste-HP-Pavilion-11-x360-PC:~$ python3.6
Python 3.6.7 (default, Oct 25 2018, 09:16:13)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 3+4
7
>>> a=4
>>> b=25
>>> a+b
29
>>> print("Hola Python")
Hola Python
>>> █
```

Testeo básico de Python en Linux

Instalar en Mac

En Mac al igual que en Linux, Python viene preinstalado, por lo que de la misma forma que en Linux, se deberá comprobar la versión y proceder a la instalación de la versión deseada en el caso que no sea la que ya se encuentra en el sistema.

Para realizar la instalación, los pasos son muy similares a la instalación que vimos en Linux, ya que el entorno de trabajo es muy similar.

1. Ingresar a la Terminal de Mac. Para ello, escribir “Terminal” en Finder o Spotlight
2. Comprobar la versión de Python escribiendo en la terminal:

`python3 --version`

Si no se encuentra la versión 3 instalada, procederemos a la instalación en el siguiente paso.

¿Qué ocurre si desinstalamos la versión previa de Python?

Bajo ningún concepto desinstalaremos la versión de Python 2 preinstalada en el sistema, tanto en Mac como en Linux. Si bien es probable que no la utilicemos nunca, si se encontraba preinstalada es probable que el sistema operativo la necesita para su normal funcionamiento. Varias versiones de Python pueden coexistir perfectamente en el mismo sistema, así que dejaremos Python 2.x y abriremos la página de descargas de Python para proceder a instalar Python 3.

3. Ingresar a la página www.python.org y realizar la descarga de la versión de Python deseada.

4. Realizar la instalación y comprobar nuevamente la versión de Python instalada para asegurar el éxito de la operación.

Un detalle no menor y que se ha mencionado en el paso a paso de instalación para Linux, es que varias versiones de Python pueden coexistir en el sistema, por lo tanto, si la versión 2 ya se encontraba instalada en Mac, ahora se encontrarán tanto la 2 como la 3. Para utilizar el intérprete con la versión 2 de Python, desde la terminal se escribirá:

`python`

ya que por default la versión de Python que se busca es la 2.

Si en cambio se quiere utilizar el intérprete con la versión 3. entonces desde la terminal se escribirá:

`python3`

y de esta forma, se estará utilizando la versión que se acaba de instalar según el paso a paso anterior.

Intérpretes de Python

Como se ha mencionado con anterioridad, hay varios intérpretes de Python que se desarrollaron de diferentes maneras, en diferentes lenguajes y con diferentes propósitos. A continuación se mencionan los más destacables.

Cpython



Es la implementación oficial y más ampliamente utilizada del lenguaje de programación Python. Cuando instalamos python, estamos también instalando esta implementación del intérprete. Es decir que tanto desde windows como linux y Mac, cuando testeamos desde la consola las diferentes pruebas de código que realizamos hasta el momento, sin saberlo hemos utilizado Cpython.

Está escrita en C, como podemos suponer por su nombre.

Además de CPython, hay otras implementaciones con calidad para producción: Jython, escrita en Java; IronPython, escrita para el Common Language Runtime y PyPy, escrita en un subconjunto del propio lenguaje Python.

Anaconda



Anaconda es una distribución libre y abierta de los lenguajes Python y R, que se utiliza en ciencia de datos, y machine learning. Anaconda se utiliza principalmente para procesamiento de grandes volúmenes de información, análisis predictivo y cómputos científicos.

Las diferentes versiones de los paquetes se administran mediante el sistema de administración del paquete conda, el cual lo hace bastante sencillo de instalar, correr, y actualizar software de ciencia de datos y machine learning tales como Scikit-team, TensorFlow y SciPy.3

La distribución Anaconda es utilizada por 6 millones de usuarios e incluye más de 250 paquetes de ciencia de datos válidos para Windows, Linux y MacOS.



pypy

Es una implementación de Python escrita en el propio lenguaje Python, esto permite realizar ciertas modificaciones sobre el propio lenguaje y da lugar a los desarrolladores a realizar mejoras y cambios sustanciales sobre el lenguaje. Al estar implementado en un lenguaje de alto nivel PyPy es más flexible y permite mayor experimentación que CPython.

PyPy tiene por objeto proporcionar una traducción común y un framework conceptual para la producción de implementaciones de lenguajes dinámicos, haciendo hincapié en una separación limpia entre la especificación del lenguaje y los aspectos de implementación. Intenta además proporcionar una implementación compatible, flexible y rápida del Lenguaje Python utilizando el mencionado framework para desarrollar nuevas características avanzadas sin tener que codificar detalles a bajo nivel.

Lo que aprendimos en este módulo

En esta clase aprendimos a instalar Python en las tres familias principales de sistemas operativos del mercado.

Vimos algunos ejemplos básicos de programación en python utilizando el intérprete oficial Cpython incluido en la distribución que instalamos.

Realizamos un recorrido por los principales intérpretes y en qué casos es conveniente utilizar cada uno.

