
ANÁLISIS DE DATOS ESPACIALES CON PROGRAMACIÓN

UNIDAD II

INTRODUCCIÓN A LA PROGRAMACIÓN Y LENGUAJE PYTHON

Tema - Introducción a Python

Fecha - 18/Octubre/2019

Docentes

Mtro. Pablo López Ramírez

E. en G. Juan Carlos Hernández Orta



DESARROLLO DE LA UNIDAD

Introducción a la Programación y Lenguaje Python

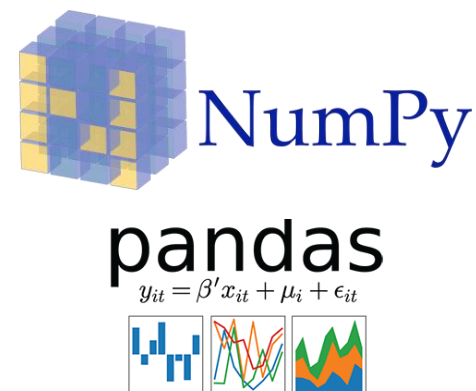
Introducción a
Python



Recursos de
Programación



Librerías de
Cómputo Científico



TRABAJO DE LA SESIÓN

Objetivos de la Sesión

- Poseer los conocimientos y las herramientas necesarias para instalar y comenzar a trabajar en Python sin importar el equipo de trabajo utilizado.
- Comprender las estructuras básicas y las herramientas de control de flujo disponibles en Python, y tener la capacidad de utilizarlas en un nivel introductorio.

Preguntas Introductorias

¿Qué tan familiarizados están con la programación?

¿Qué es lo que saben o han escuchado de Python?

¿Qué es lo que esperan del curso?

¿QUÉ ES PYTHON?

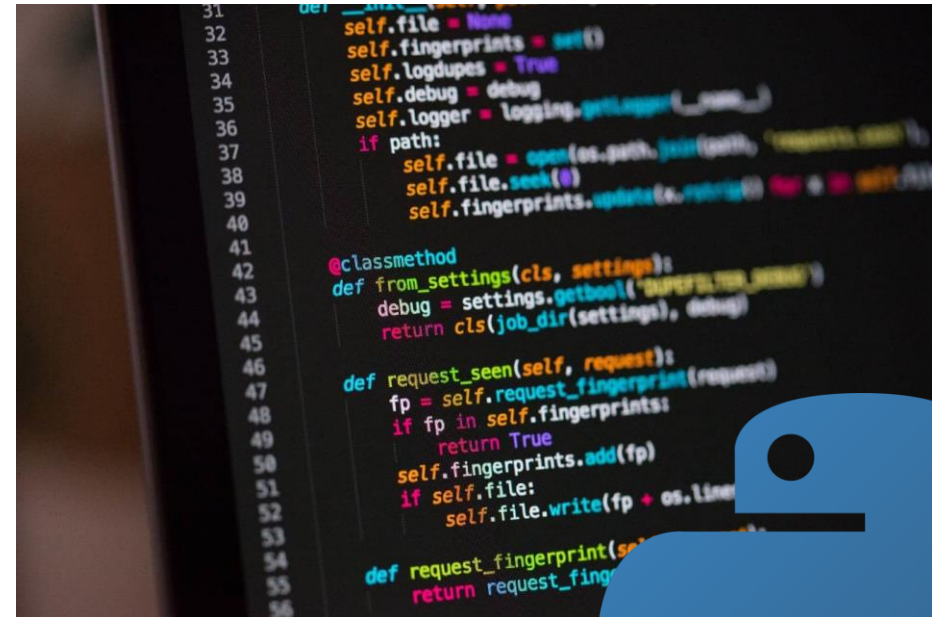
Lenguaje de Programación creado por el programador holandés Guido van Rossum en 1991, cuyo objetivo principal es ser lo más sencillo y funcional posible.

“Bello es mejor que feo; explícito es mejor que implícito; simple es mejor que complejo.” (Zen of Python, 2004)

Python busca ser un lenguaje lo más legible, sencillo y estético posible, sin sacrificar sus capacidades de programación.

Se encuentra diseñado para ser expandido por cualquier colaborador, a través de **Librerías**, gracias a su accesibilidad como **Código Abierto**.

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



R VS PYTHON



R

- Especializado en el análisis estadístico y matemático.
- Gran número de librerías, casi todas especializadas en alguna técnica estadística.
- Facilita enormemente la visualización estética de resultados (Gráficas, Mapas, Tablas, etc.)
- El enlace de los resultados de un análisis espacial más directo, pues ya se encuentran en un formato fácil de traducir.



Python

- Diseñado para el desarrollo e implementación de código en general.
- Múltiples librerías de diferentes ramas, cada una con objetivos diferentes.
- Facilita el enlace del código con otro tipo de aplicaciones y métodos (Aplicaciones Web, Inteligencia Artificial, etc.).
- El procesamiento de datos espaciales y la ejecución de algoritmos es sencillo y directo, sin gran complejidad.

¿Cuál utilizar?

El que mejor se ajuste a las necesidades del proyecto, a los conocimientos de los involucrados y...

El que te facilite el trabajo y lo vuelva más interesante.

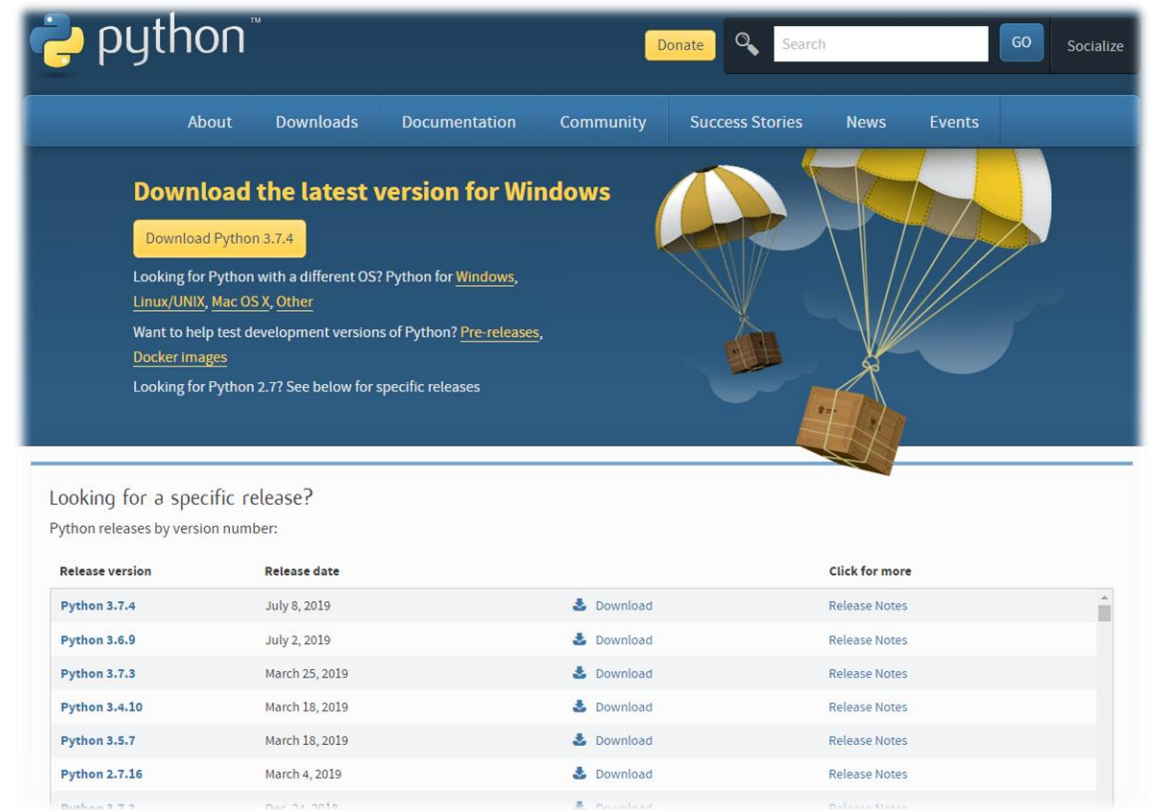
INSTALACIÓN DE PYTHON

Para el caso de **Windows** y **Macintosh**, la versión más reciente y estable puede ser descargada a través de la página oficial:

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

En el caso de **Ubuntu**, se trata de uno de los lenguajes precargados al Sistema Operativo; basta con ejecutar el siguiente comando en la *Línea de Comandos* para verificar la versión instalada:

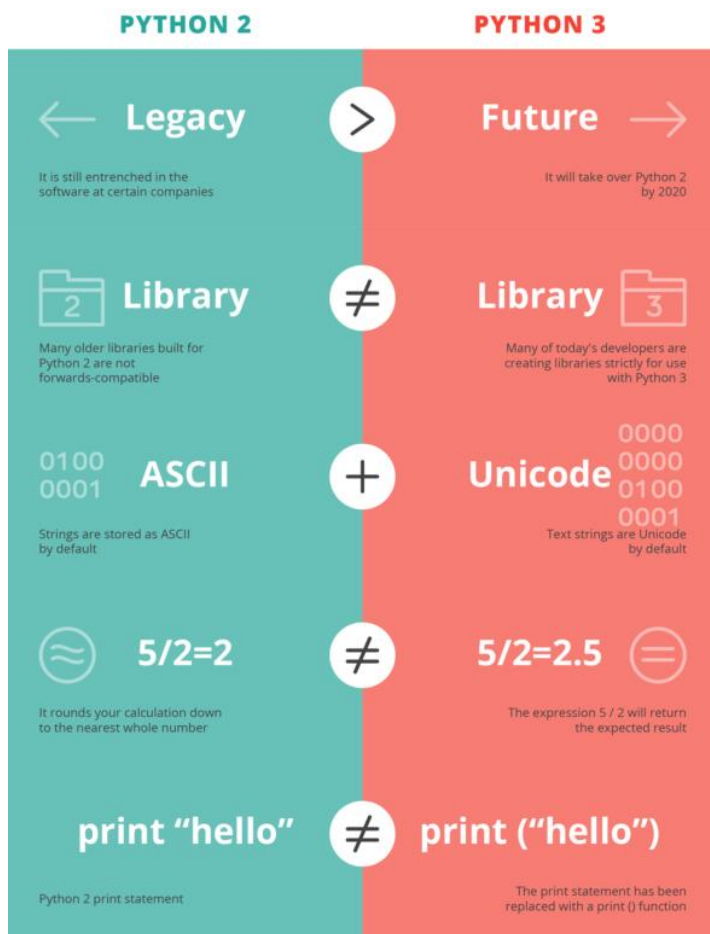
```
python -V
```



The screenshot shows the Python.org website. The header includes the Python logo, a search bar, and links for Donate, GO, and Socialize. The main navigation bar contains links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a large banner with the text "Download the latest version for Windows" and a button "Download Python 3.7.4". Below this, there are links for "Python for Windows, Linux/UNIX, Mac OS X, Other", "Pre-releases", "Docker images", and "Looking for Python 2.7? See below for specific releases". The banner also features an illustration of two parachutes carrying boxes. Below the banner, there is a section titled "Looking for a specific release?" with the text "Python releases by version number:". This section contains a table with the following data:

Release version	Release date		Click for more
Python 3.7.4	July 8, 2019	Download	Release Notes
Python 3.6.9	July 2, 2019	Download	Release Notes
Python 3.7.3	March 25, 2019	Download	Release Notes
Python 3.4.10	March 18, 2019	Download	Release Notes
Python 3.5.7	March 18, 2019	Download	Release Notes
Python 2.7.16	March 4, 2019	Download	Release Notes
Python 3.7.2	Dec 24, 2018	Download	Release Notes

PYTHON 2.x VS PYTHON 3.x



Múltiples librerías complementarias fueron desarrolladas cuando Python aún se encontraba en su versión 2.x.

Al momento de actualizarse a su versión 3.x, el lenguaje realizó múltiples cambios de enorme repercusión, pues provocó que no existiera compatibilidad con su versión anterior y, por ende, volvió parcialmente inutilizables cualquier librería o código no actualizado.

¡Importante!

- Asegúrate de conocer en qué versión de Python trabajarás, o cuál utiliza tu lugar de trabajo.
- Al buscar referencias y guías en línea, verifica que correspondan a la versión que utilizas; existen librerías que ya no funcionan en Python 3.x
- Python 2.x dejará de recibir soporte oficial en 2020

PAQUETERÍAS DE PYTHON

Existen cerca de 200,000 paqueterías complementarias para Python, escritas por diferentes autores alrededor del mundo, todas con el objetivo de facilitar alguna tarea/algoritmo con el lenguaje. Para instalarlas y poder utilizarlas, se recurre a:

Python Package Index (PyPI)

Sistema de gestión de paquetes oficial del lenguaje; se encuentra precargado al momento de instalar Python.

Permite instalar cualquier paquetería disponible, a través de una sola línea de código:

```
pip install nombre_paquetería
```



<https://pypi.org/>

Conda

Sistema de gestión desarrollado por terceros, que pretende solucionar conflictos derivados del hecho de que no todos los paquetes de Python son compatibles entre sí, y que *PyPI* no es capaz de identificar.

```
conda install nombres_paqueterías
```



<https://www.anaconda.com/>

INSTALACIÓN DE CONDA

Su versión completa, llamada **Anaconda**, incluye todos los paquetes posibles gestionados por el sistema; es más pesado, pero instala en la computadora todo lo que conoce:

<https://www.anaconda.com/distribution/>

Su versión más versátil, llamada **Miniconda**, no incluye más que lo mínimo necesario para funcionar, permitiendo instalar los paquetes conforme se vayan utilizando:

<https://docs.conda.io/en/latest/miniconda.html>

```
Anaconda Prompt (Miniconda3)
C:\Users\geo>conda activate base

(base) C:\Users\geo>conda install numpy
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: C:\Users\geo\Miniconda3

added / updated specs:
- numpy

The following packages will be downloaded:
```

package	build	
blas-1.0	mkl	6 KB
ca-certificates-2019.8.28	0	165 KB
certifi-2019.9.11	py37_0	155 KB
conda-4.7.12	py37_0	3.0 MB
icc_rt-2019.0.0	h0cc432a_1	6.0 MB
intel-openmp-2019.4	245	1.4 MB
mkl-2019.4	245	99.2 MB
mkl_fft-1.0.12	py37h14836fe_0	118 KB
mkl_random-1.0.2	py37h343c172_0	272 KB
numpy-1.16.4	py37h19fb1c0_0	48 KB
numpy-base-1.16.4	py37hc3f5095_0	3.3 MB



'ENVIRONMENTS' DE PYTHON

Comúnmente, es de gran utilidad *aislar* toda la paquetería relacionada con un proyecto del resto, en caso de que puedan existir conflictos entre las dependencias de uno y otro. Crear un **Environment** que auxilie a este aislamiento ayuda controlar el ambiente de programación del proyecto, permitiendo que éste se ejecute cuanto se necesite.

En Windows Cmd:

Crear environment

```
python -m venv nombre
```

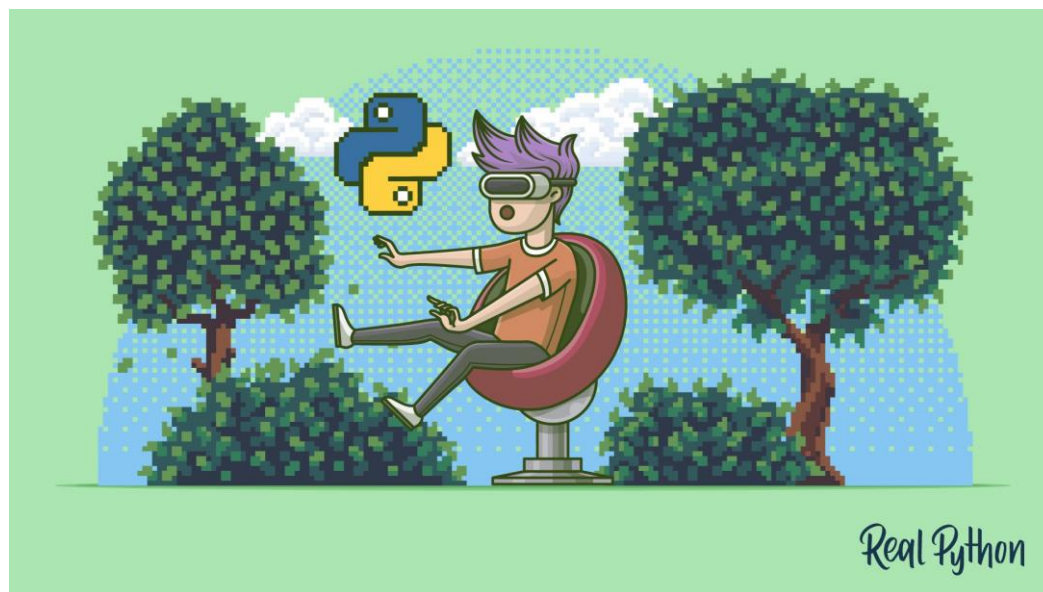
* Éste se crea en el mismo directorio donde se encuentre actualmente la Línea de Comandos

Activarlo

```
nombre\Scripts\activate
```

Desactivarlo

```
deactivate
```



A través de Conda:

Crear environment

```
conda create -n nombre
```

* Se crea en un directorio común para todos los *Environments* existentes en la computadora

Activarlo

```
conda activate nombre
```

Desactivarlo

```
conda deactivate
```

RECURSOS DE PROGRAMACIÓN – INTÉRPRETES INTERACTIVOS

Jupyter Notebook

Aplicación interactiva basada en web, en la cual es posible escribir y ejecutar secciones de código en espacios llamados *Celdas*, explicarlo a través de texto editable y visualizar y descargar los resultados en diversos formatos (HTML, PDF, etc.)

`conda install jupyter`



Ambos dependen del paquete **IPython**, el cual es el precursor y la base para crear una consola interactiva

IP[y]:
IPython

Spyder

Programa ejecutable especializado en trabajar código de Python, haciéndolo más legible y permitiendo visualizar los resultados del mismo en un ambiente controlado. Funciona de forma similar a como *Rstudio* lo hace para el lenguaje R.

`conda install spyder`



¡Objetivo Cumplido!

Poseer los conocimientos y las herramientas necesarias para instalar y comenzar a trabajar en Python sin importar el equipo de trabajo utilizado.



Siguiente objetivo:

Comprender las estructuras básicas y las herramientas de control de flujo disponibles en Python, y tener la capacidad de utilizarlas en un nivel introductorio.



Para ello, es necesario utilizar lo aprendido para trabajar con un *Jupyter Notebook* (.ipyb)

SIGUIENTE PASO, DESCARGAR EL ARCHIVO *CLASE_1.IPYB* Y ABRIRLO A TRAVÉS DE JUPYTER NOTEBOOK...