



# Python para Ciencia de Datos

Isabel Maniega





# Contenido

## Introducción a Python

- Instalación y configuración del entorno

- Fundamentos del lenguaje (sintaxis, variables, estructuras de control).

- Introducción a la manipulación de archivos (CSV, JSON) Análisis.

## Procesamiento de Datos

- Bibliotecas Clave

- Gestión, limpieza y transformación de datos (filtrado, manejo de valores nulos).

- Importación de datos desde fuentes externas (archivos, bases de datos).

- Análisis de datos geoespaciales y meteorológicos relevantes en emergencias.

## Automatización de Procesos y Alertas

- Automatización.

- Scripts para el procesamiento de alertas y notificaciones.

- Integración con APIs de Emergencia y Servicios Externos.

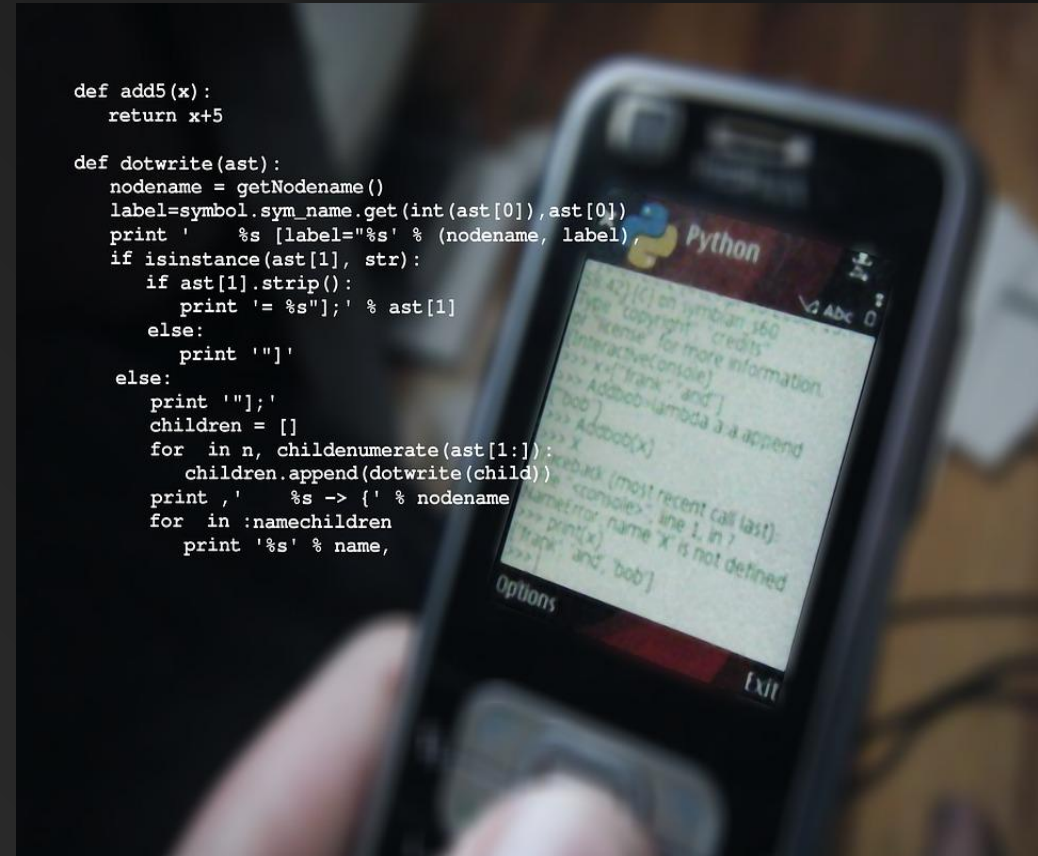
- Creación de dashboard interactivos

## Ejercicios prácticos

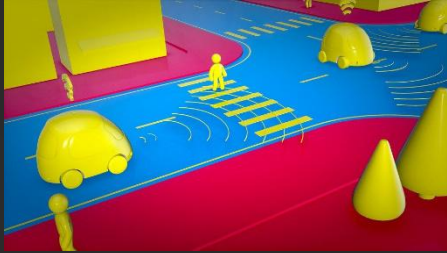


# Lenguaje de programación Python

- Python es un lenguaje de programación potente y fácil de aprender.
- Tiene estructuras de datos de alto nivel eficientes y un simple pero efectivo sistema de programación orientado a objetos.
- La elegante sintaxis de Python y su tipado dinámico, junto a su naturaleza interpretada lo convierten en un lenguaje ideal para scripting y desarrollo rápido de aplicaciones en muchas áreas, para la mayoría de plataformas.



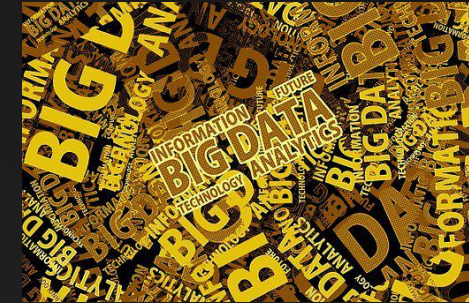
# Salidas Profesionales Python



Vehículos de  
conducción autónoma



Detección de fraude



Big Data



Sistemas de  
recomendación



Frameworks webs



Deep Learning


# Fundamentos de Python



Python es un lenguaje de programación de alto nivel, interpretado, orientado a objetos y de uso generalizado que se utiliza para la programación de propósito general.

Un **lenguaje de alto nivel** es similar al lenguaje natural ya que usa símbolos, palabras y convenciones legibles para los humanos y nos permite expresar comandos a computadoras que son mucho más complejas que la lista de instrucciones (IL) que estas entienden.

Es importante conocer que hay dos formas diferentes de transformar un programa de un lenguaje de programación de alto nivel a un lenguaje de máquina:

- **Compilación** - El programa fuente se traduce una vez obteniendo un archivo que contiene el código máquina (por ejemplo, un .exe si se trabaja en windows) y es este el archivo que se distribuye. Cada vez que se modifica el código fuente debe repetirse esta traducción. El programa que realiza la traducción se llama compilador o traductor.
  - **Interpretación** - El programa fuente se traduce a código máquina cada vez que se ejecuta el programa. El programa que realiza este tipo de transformación se denomina intérprete ya que interpreta el código cada vez que se ejecuta. El usuario final necesitará el código fuente más el intérprete para poderlo ejecutar.
- 



# Fundamentos de Python

## COMPILACIÓN

### VENTAJAS

- La ejecución del código traducido suele ser más rápida.
- Solo el programador debe tener el compilador; el usuario final puede usar el código sin él.
- El código traducido se almacena en lenguaje máquina, ya que es muy difícil de entender, es probable que tus propios inventos y trucos de programación sigan siendo un secreto.

### DESVENTAJAS

- La compilación en sí misma puede llevar mucho tiempo; es posible que no puedas ejecutar tu código inmediatamente después de cualquier modificación.
- Tienes que tener tantos compiladores como plataformas de hardware en las que deseas que se ejecute tu código.

## INTERPRETACIÓN

- Puedes ejecutar el código en cuanto lo completes; no hay fases adicionales de traducción.
- El código se almacena utilizando el lenguaje de programación, no el de la máquina; esto significa que puede ejecutarse en computadoras que utilizan diferentes lenguajes máquina; no se compila el código por separado para cada arquitectura diferente.
- No esperes que la Interpretación incremente tu código a alta velocidad: tu código compartirá la potencia de la computadora con el intérprete, por lo que no puede ser realmente rápido.
- Tanto tú como el usuario final deben tener el intérprete para ejecutar el código.



# Fundamentos de Python

Python fue creado por **Guido van Rossum**, nacido en 1956 en Haarlem, Países Bajos. Por supuesto, Guido van Rossum no desarrolló y evolucionó todos los componentes de Python.

Aunque puede que conozcas a la pitón como una gran serpiente, el nombre del lenguaje de programación Python proviene de una vieja serie de comedia de la BBC llamada **Monty Python's Flying Circus**.

En el apogeo de su éxito, el equipo de Monty Python estaba realizando sus escenas en vivo para audiencias en todo el mundo, incluso en el Hollywood Bowl.

Dado que Monty Python es considerado uno de los dos nutrientes fundamentales para un programador (el otro es la pizza), el creador de Python nombró el lenguaje en honor al programa de televisión.



# Instalaciones necesarias

Python version 3.11.9: <https://www.python.org/downloads/>

Python version	Maintenance status	First released	End of support	Release schedule
3.14	pre-release	2025-10-01 (planned)	2030-10	PEP 745
3.13	bugfix	2024-10-07	2029-10	PEP 719
3.12	bugfix	2023-10-02	2028-10	PEP 693
3.11	security	2022-10-24	2027-10	PEP 664
3.10	security	2021-10-04	2026-10	PEP 619
3.9	security	2020-10-05	2025-10	PEP 596
3.8	end of life, last release was 3.8.20	2019-10-14	2024-10-07	PEP 569

Windows se encuentra directamente en la tienda para su instalación (mejor opción ejecutable):

<https://www.python.org/downloads/release/python-3119/>

Ubuntu viene instalado por defecto con python 3.10.12

<https://docs.python.org/es/3.11/tutorial/>

<https://docs.python.org/es/3.11/using/windows.html>



# Python

- Comprobación de instalación: `python --version`

```
isabel@isabel-SVE1512E1EW:~$ python3 --version
Python 3.8.10
```

- Intérprete de Python:

```
isabel@isabel-SVE1512E1EW:~$ python3
Python 3.8.10 (default, Mar 15 2022, 12:22:08)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```

- ¿Cómo realizar una instalación de librerías?

Existe una página con todos los repositorios de las librerías esa página es PYPI:

<https://pypi.org/>

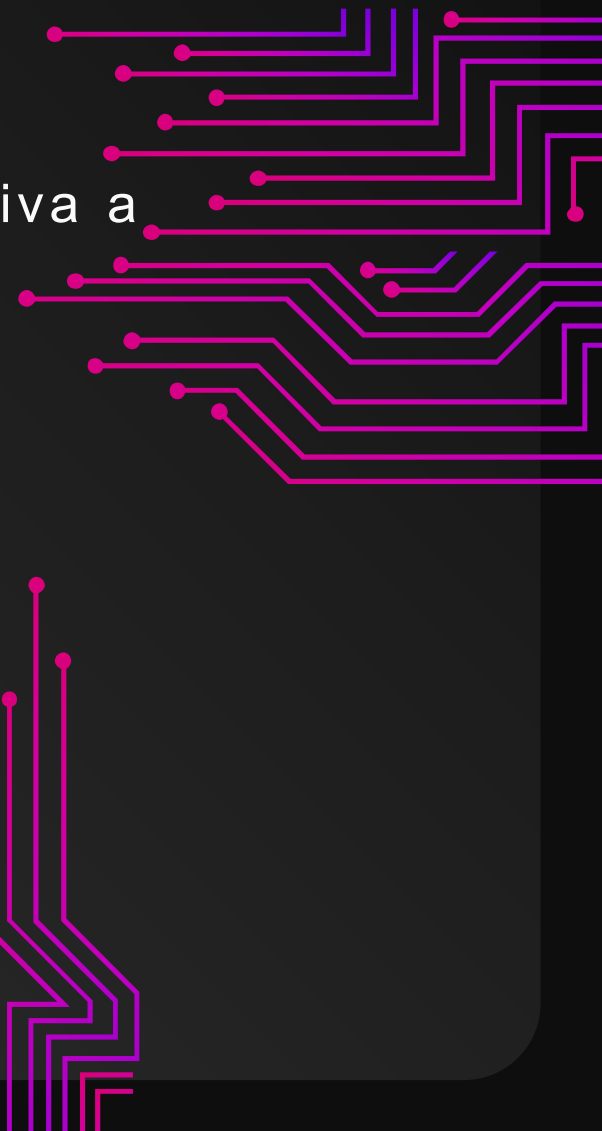
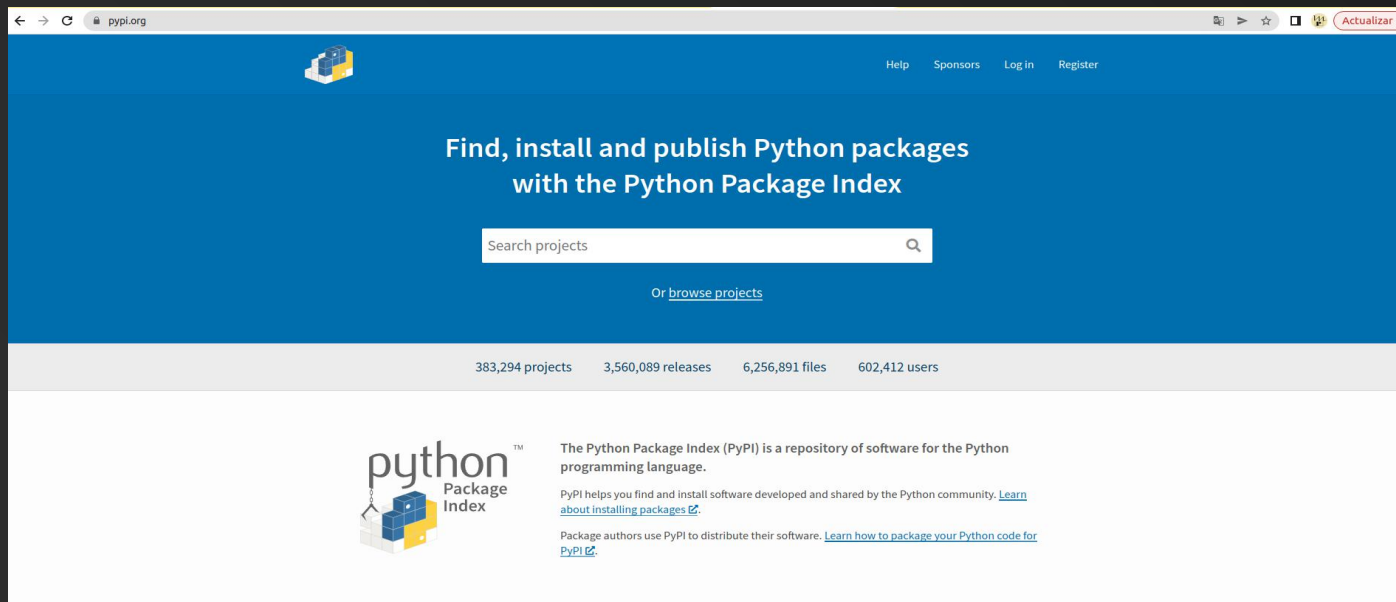
# Python

- ¿Cómo realizar una instalación de librerías ?

Existe una página con todos los repositorios de las librerías esa página es **PYPI**:

<https://pypi.org/>

En esta página podemos encontrar toda la información relativa a documentación, ejemplos, como instalar, versiones disponibles, etc.



# Python

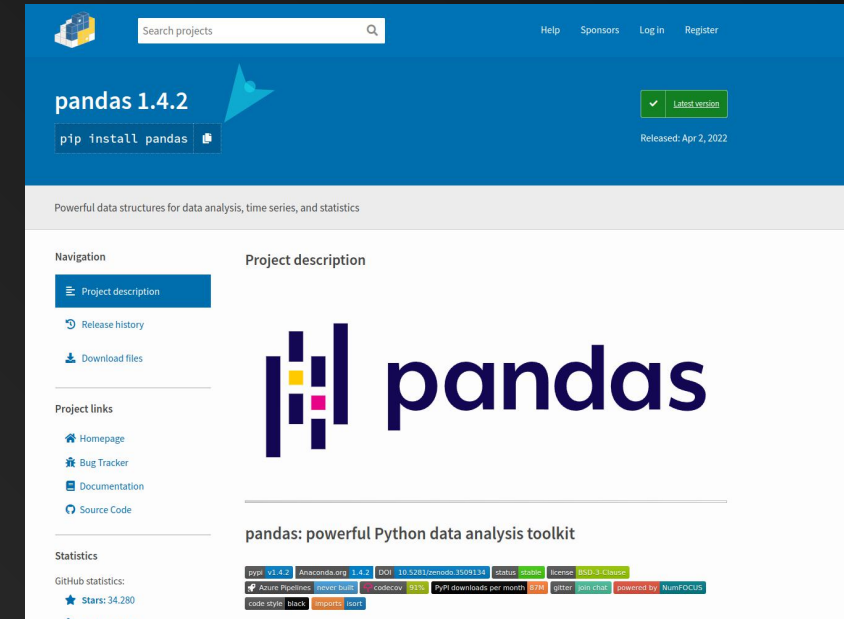
¿Cómo realizar una instalación de librerías?

1. Para la realización de la instalación iremos a esa página y buscaremos la librería a instalar, copiaremos la instrucción para su instalación tal y como se indica en la imagen:

2. Iremos a la cmd y pegaremos la instrucción y pulsaremos intro:

*En algunos sistemas operativos se pide la instalación de pip, ya que aún no está instalada*

3. Veremos que nos a realizado con éxito la instalación.



```
(env) isabel@isabel-SVE1512E1EW:~/atom$ pip install pandas
Collecting pandas
  Using cached pandas-1.4.2-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.7 MB)
Collecting pytz>=2020.1
  Using cached pytz-2022.1-py2.py3-none-any.whl (503 kB)
Collecting numpy>=1.18.5; platform_machine != "aarch64" and platform_machine != "arm64" and python_version < "3.10"
  Downloading numpy-1.22.4-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (16.9 MB)
    |████████████████████| 16.9 MB 10.5 MB/s
Collecting python-dateutil>=2.8.1
  Using cached python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
Collecting six>=1.5
  Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: pytz, numpy, six, python-dateutil, pandas
Successfully installed numpy-1.22.4 pandas-1.4.2 python-dateutil-2.8.2 pytz-2022.1 six-1.16.0
(env) isabel@isabel-SVE1512E1EW:~/atom$
```

# Instalaciones necesarias

- Jupyter notebook: <https://jupyter.org/install>



## Jupyter Notebook

Install the classic Jupyter Notebook with:

```
pip install notebook
```

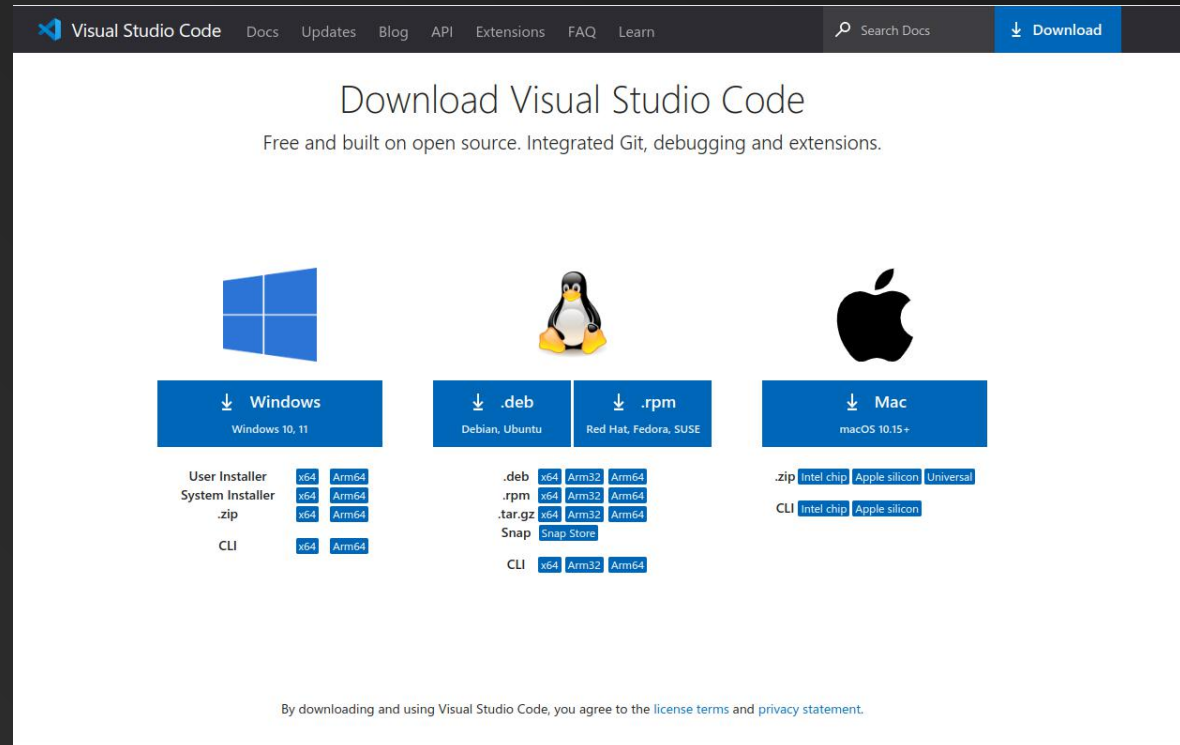
To run the notebook:

```
jupyter notebook
```



# Instalaciones necesarias

- Visual Studio Code: <https://code.visualstudio.com/download>




The screenshot shows the Visual Studio Code download page. At the top, there's a navigation bar with links for Docs, Updates, Blog, API, Extensions, FAQ, and Learn, along with a search bar and a 'Download' button. The main heading is 'Download Visual Studio Code', followed by the tagline 'Free and built on open source. Integrated Git, debugging and extensions.' Below this, there are three main sections for Windows, Linux, and Mac. Each section has a download button and a list of available installers for different architectures.

Visual Studio Code Docs Updates Blog API Extensions FAQ Learn Search Docs Download

## Download Visual Studio Code


Free and built on open source. Integrated Git, debugging and extensions.



### Windows


Windows 10, 11

User Installer	x64	Arm64
System Installer	x64	Arm64
.zip	x64	Arm64
CLI	x64	Arm64



### Linux

.deb	x64	Arm32	Arm64
.rpm	x64	Arm32	Arm64
.tar.gz	x64	Arm32	Arm64
Snap	Snap Store		
CLI	x64	Arm32	Arm64



### Mac

macOS 10.15 +

.zip	Intel chip	Apple silicon	Universal
CLI	Intel chip	Apple silicon	

By downloading and using Visual Studio Code, you agree to the [license terms](#) and [privacy statement](#).

