



# UD02: Introducción a Python y Instalación



RA¿?

Juan Sevillano Hernández  
MP: Introducción a Python



# ÍNDICE

1. ¿Qué es Python?
2. Características Principales de Python
  3. Fue Creado Por
  4. ¿Para Qué Sirve?
5. Empresas Que Usa Python
6. Instalación de Python
7. IDEs
8. Instalación y Configuración de Visual Studio Code



# 1. ¿Qué es Python?



## 1. ¿Qué es Python?

**Python es un lenguaje de programación interpretado**, lo que significa que **no se necesita compilar el código antes de ejecutarlo**. Esto **lo hace más rápido y fácil de usar** que otros lenguajes de programación.

**Es gratuito y de código abierto**, ampliamente utilizado por su facilidad de aprendizaje, versatilidad y comunidad activa.

Además, **Python es un lenguaje de alto nivel**, lo que significa que se enfoca más en la legibilidad y facilidad de uso que en la eficiencia de la máquina.

```
package main

import "fmt"

func main() {
    fmt.Printf("hello, world")
}
```

Lenguaje de alto nivel que  
entiende el programador



```
0101010111101110001101
0100010100010101001010
0101010010101010000101
0011010001010100011110
0110010100101010101001
1110001101010010010001
```

Lenguaje de máquina que  
entiende el procesador



open source

# 1. ¿Qué es Python?

## Resumiendo:



Python es un lenguaje de programación de propósito general, interpretado, multiplataforma, multiparadigma y de código abierto, que es potente, flexible, ampliamente utilizado, compatible y fácil de aprender. Es un lenguaje de alto nivel, para procesar todo tipo de estructuras de datos tanto numéricas como de texto.

# 1. ¿Qué es Python?



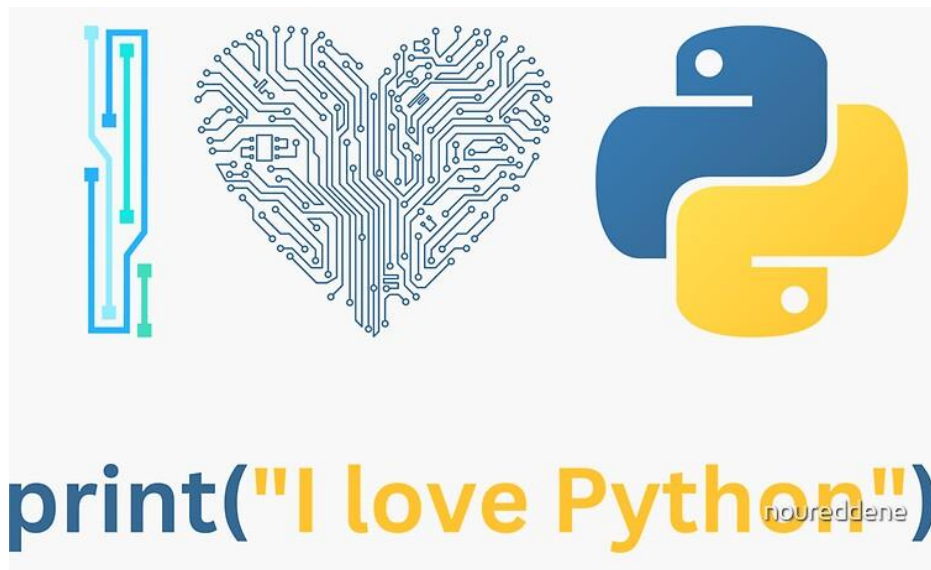


# 1. ¿Qué es Python?

## ¿Por qué Python?

- Se trata de **un lenguaje fácil de aprender, con una sintaxis muy sencilla que se asemeja bastante al pseudocódigo**. En otras palabras, **poco código hace mucho**.
- Su uso **no está ligado a un sector concreto**. Por ejemplo el lenguaje R es útil para análisis de datos, pero no puede ser usado para desarrollo web. **Python vale para todo**.
- **Tiene una comunidad enorme, además de gran cantidad de librerías para hacer prácticamente cualquier cosa**, literalmente.
- **Es un lenguaje multiplataforma**, por lo que el mismo código es compatible en cualquier plataforma (Windows, macOS, Linux) sin hacer nada.
- Por lo general **se puede hacer desarrollos en Python más rápidamente que en otros lenguajes**, acortando la duración de los proyectos.

## 1. ¿Qué es Python?



# 1. ¿Qué es Python?

## Versiones de Python:

Las versiones de Python se identifican por tres números **X.Y.Z**, en la que:

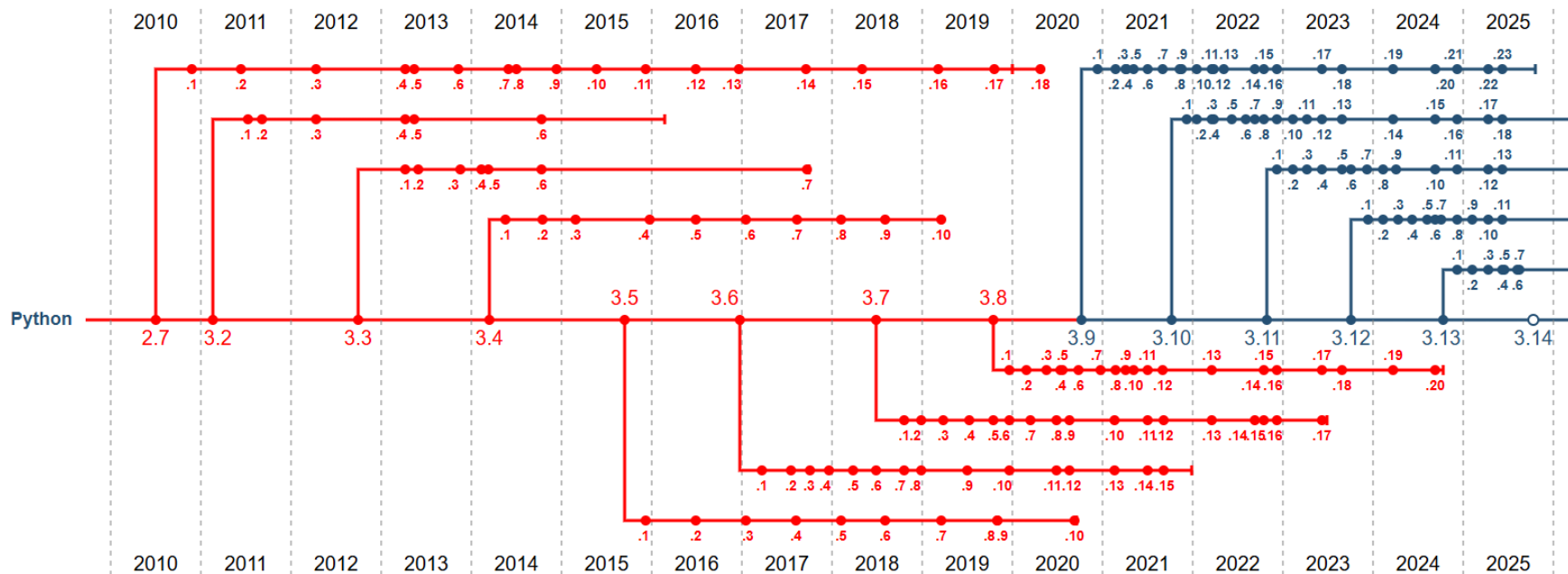
- **X corresponde a las grandes versiones de Python** (1, 2 y 3), **incompatibles entre sí**.
- **Y** corresponde a **versiones importantes en las que se introducen novedades en el lenguaje pero manteniendo la compatibilidad** (salvo excepciones).
- **Z** **corresponde a versiones menores que se publican durante el período de mantenimiento, en las que sólo se corrigen errores durante el primer año y fallos de seguridad en los cuatro restantes**. La publicación de las versiones .Z es más irregular, porque el descubrimiento de fallos de seguridad puede provocar la necesidad de publicación de una nueva versión en cualquier momento, pero hay un calendario previsto de versiones .Z.

1 . 3 . 5

X . Y . Z

# 1. ¿Qué es Python?

Versiones publicadas de Python





## 2. Características Principales de Python

## 2. Características Principales de Python

Python tiene varias características que lo hacen único y popular entre los programadores:

- **Legibilidad del código:** Utiliza una **sintaxis clara y simple que hace que el código sea fácil de leer y entender**. Esto es especialmente útil para los programadores que trabajan en equipo o que necesitan mantener el código a lo largo del tiempo.
- **Flexibilidad:** Python **es un lenguaje de programación multipropósito que puede ser utilizado en muchos campos diferentes**, lo que lo hace muy versátil.
- **Gran cantidad de bibliotecas:** Cuenta con una gran cantidad de bibliotecas, **lo que hace que sea fácil y rápido desarrollar aplicaciones y programas**. Estas bibliotecas cubren una amplia variedad de áreas, desde la manipulación de datos hasta el aprendizaje automático.
- **Interpretado:** Como se mencionó anteriormente, Python es un lenguaje de programación interpretado. **Esto significa que el código no necesita ser compilado antes de ser ejecutado, lo que ahorra tiempo y hace que el proceso de desarrollo sea más rápido.**
- **Multiplataforma:** **Se ejecuta en una amplia variedad de sistemas operativos**, incluyendo Windows, Mac OS y Linux. Esto lo hace muy útil para aquellos que necesitan desarrollar programas que se ejecuten en diferentes plataformas.

## 2. Características Principales de Python

### ¿Cuáles son las características de Python?



#### Propósito General:

Se puede crear todo tipo de programas.



#### Multiplataforma:

Se ejecuta en diferentes sistemas operativos o plataformas.



#### Multiparadigma:

Combina propiedades de diferentes paradigmas de programación: Orientada a objetos, imperativa y funcional.



#### Interpretado:

Interpreta el código del programador, lo traduce y lo ejecuta a la vez.



#### Orientado a Objetos:

Crea programas con componentes reutilizables.



#### Tipado Dinámico:

Cuando una variable puede tomar valores de distinto tipo o transformarse.



**Sintaxis Clara:** Sintaxis muy visual con márgenes.



## 2. Características Principales de Python





### 3. Fue Creado Por

### 3. Fue Creado Por



Guido van Rossum

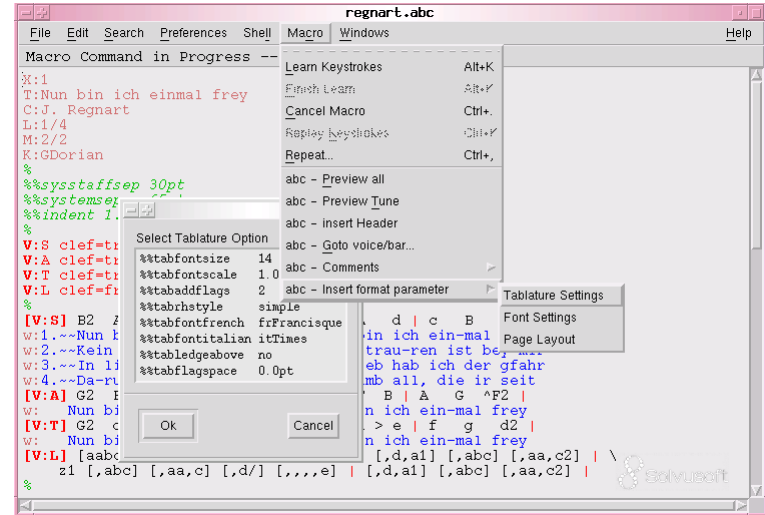
Python es un lenguaje de **programación creado por Guido van Rossum a principios de los años 90** cuyo **nombre está inspirado en el grupo de cómicos ingleses “Monty Python”**. Es un lenguaje **similar a Perl, pero con una sintaxis muy limpia** y que favorece un código legible.

En las navidades de 1989 Van Rossum, **mientras trabajaba en un centro de investigación holandés (CWI), decidió empezar un nuevo proyecto como pasatiempo personal. Pensó en darle continuidad a ABC**, un lenguaje de programación que se **desarrolló en el mismo centro en el que estaba trabajando**.



### 3. Fue Creado Por

- **Año de creación:** Finales de la década de 1980, con la primera versión en 1991
- **Objetivo:** Crear un lenguaje de programación sencillo, potente, extensible y fácil de aprender y usar.
- **Inspiración:** El lenguaje ABC.



### ¿Cómo se originó Python?

El lenguaje Python fue creado a principios de los 90 por Guido Van Rossum. Es gestionado por la Python Software Foundation, sociedad sin fines de lucro para difundir y apoyar el lenguaje. Se desarrolla bajo licencia de Open Source aprobada por OSI, por lo que se puede usar y distribuir libremente.



## 4. ¿Para Qué Sirve?

## 4. ¿Para Qué Sirve?

Python es un lenguaje de programación muy versátil que se puede utilizar para crear una amplia variedad de aplicaciones. Algunas de las cosas que se pueden crear con Python incluyen:

**1. Aplicaciones de escritorio:** Se pueden crear aplicaciones de escritorio utilizando el framework Qt o la biblioteca Tkinter.

**2. Aplicaciones web:** Se pueden crear aplicaciones web utilizando frameworks como Django, Flask y Pyramid.

3. Juegos: Se pueden crear juegos utilizando bibliotecas como Pygame y Panda3D.

**4. Análisis de datos:** Python es muy popular en el campo del análisis de datos debido a sus bibliotecas, como Pandas, Numpy y Matplotlib.

**5. Machine Learning:** Python es muy popular en el campo del aprendizaje automático y la inteligencia artificial debido a su biblioteca Scikit-Learn y TensorFlow.

**6. Automatización de tareas:** Python es muy útil para automatizar tareas repetitivas o para realizar operaciones en varios archivos o carpetas.

**7. Redes y seguridad:** Python se puede utilizar para crear herramientas de hacking ético, herramientas de seguridad y scripts para administración de redes.

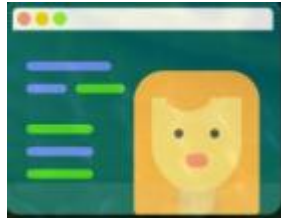


## 4. ¿Para Qué Sirve?



- Kivy
- Tkinter

**Aplicaciones  
de Escritorio y  
Móviles**



**Aplicaciones Web**

- Flask
- Django



**Videojuegos**

- Pygame
- PyOpenGL



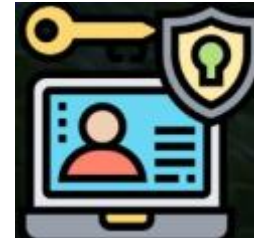
**Inteligencia  
Artificial**

- Tensorflow
- Keras
- PyTorch



**Análisis de Datos**

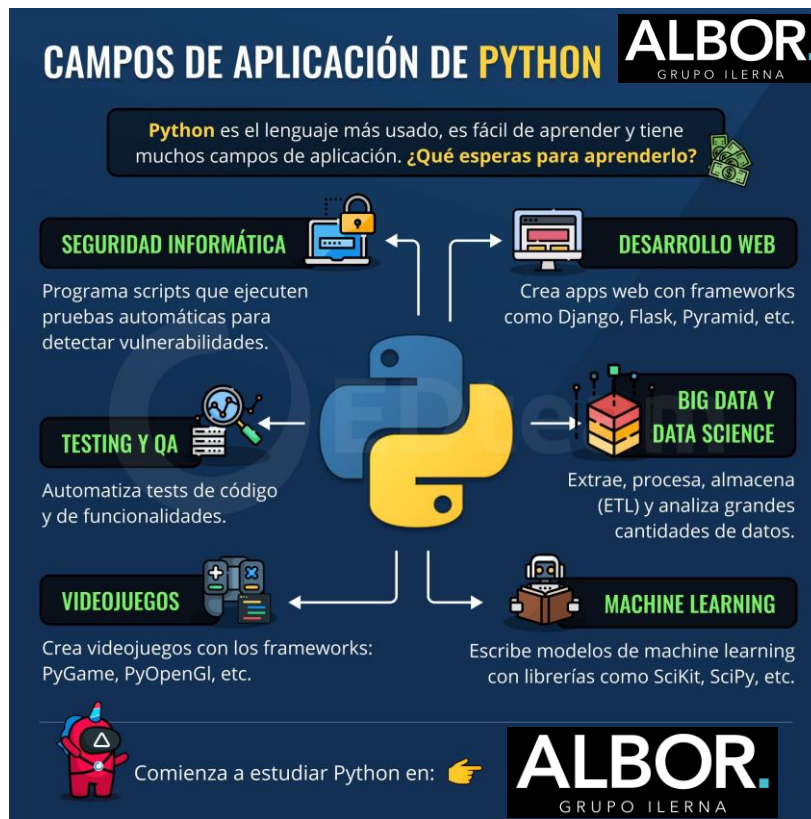
- Numpy
- Pandat
- Marplotlib



**Redes y Seguridad**

- Scapy
- dpkt
- Scrapy

## 4. ¿Para Qué Sirve?





## 5. Empresas Que Usa Python

## 5. Empresas Que Usa Python



## 5. Empresas Que Usa Python



**SOY el REY con la PYTHON**



## 6. Instalación de Python.



## 6. Instalación de Python



Instalando...

## 6. Instalación de Python

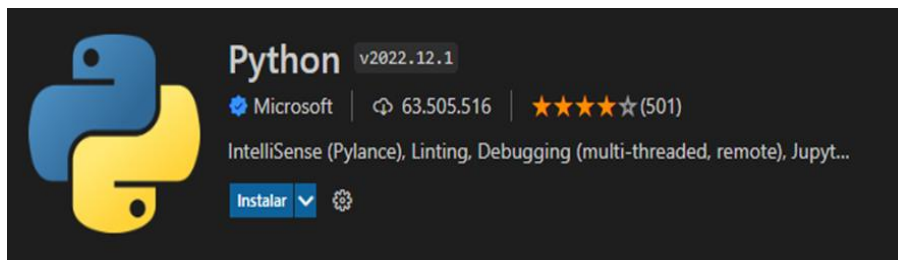


## 6. Instalación de Python

### Extensiones de Visual Studio Code para trabajar con Python:

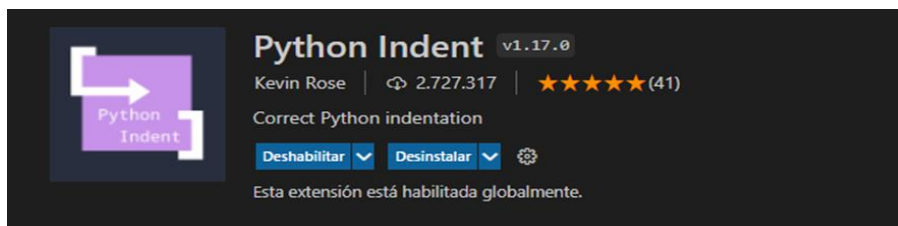
#### 1. Python

Como es obvio la primera de la lista debía de ser esta extensión, la cual **nos permite el uso de Python en VSCode**.



#### 2. Python Indent

La mejor extensión para **indentar nuestro código Python**.



## 6. Instalación de Python

### 3. Better Comments

Con esta extensión podremos mejorar los comentarios de nuestro código, los cuales los **podemos categorizar en: Alertas, Consultas o Destacados entre otros.**

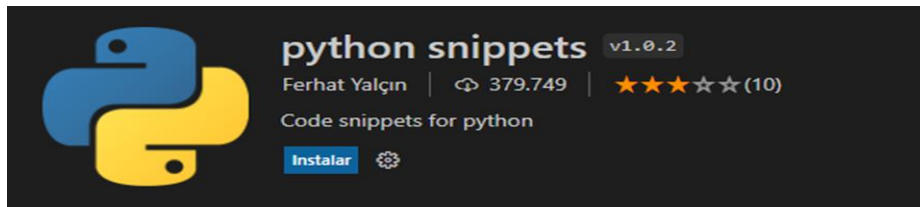
```
2  export class MyClass {
3
4      /**
5       * MyMethod
6       * * Important information is highlighted
7       * ! Deprecated method, do not use
8       * ? Should this method be exposed in the public API?
9       * TODO: refactor this method so that it conforms to the API
10      * @param myParam The parameter for this method
11      */
12     public MyMethod(myParam: any): void {
13         let myVar: number = 123;
14
15         /** This is highlighted
16         if (myVar > 0) {
17             throw new TypeError(); ///! This is an alert
18         }
19
20         ///? This is a query
21         let x = 1;
22
23         //// this.lineOfCode() == commentedOut;
24
25         //TODO: Create some test cases
26     }
27 }
```

## 6. Instalación de Python

### 4. Python Snippets

Es una extensión muy recomendada para principiantes ya que contiene fragmentos de código para listas, cadenas, diccionarios, tuplas, clases, etc. con al menos un ejemplo de cada uno de ellos.

De esta forma nos evitamos el tener que escribir una y otra vez el mismo código



### 5. Python Test Explorer

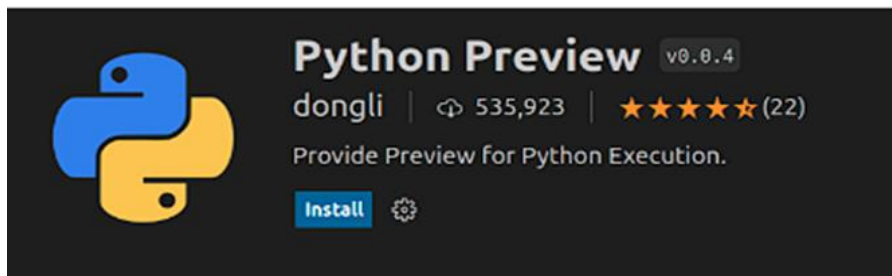
Esta extensión nos permite ejecutar pruebas de Python Unittest , Pytest o Testplan con la interfaz de usuario de Test Explorer



## 6. Instalación de Python

### 6. Python Preview

Nos permite una vista previa de depuración para Python.



### 7. Python Type Hint

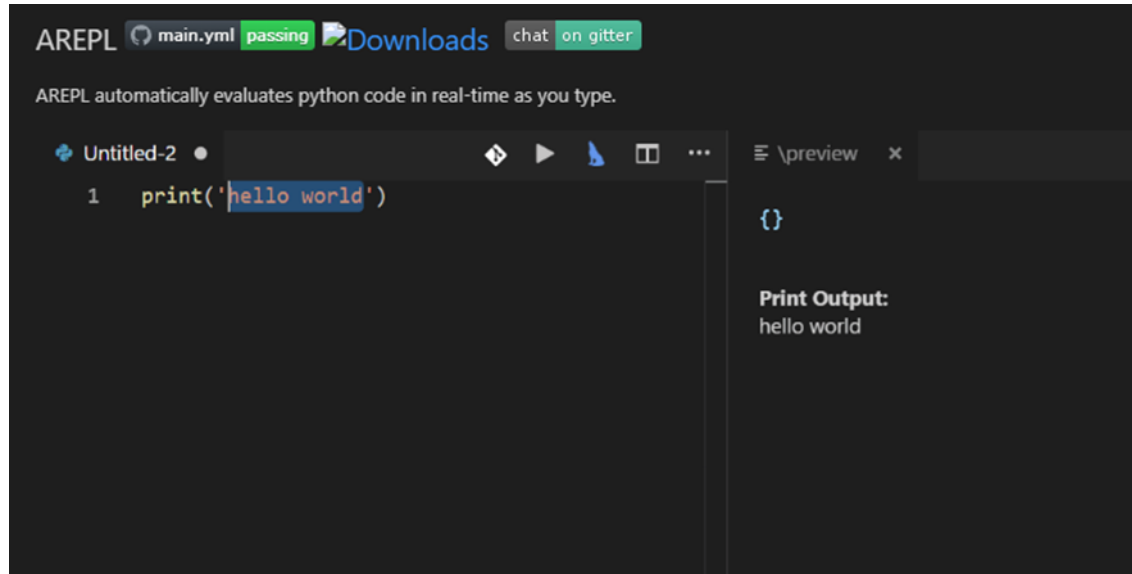
Esta extensión nos ofrece opciones de autocompletado para nuestro código.



## 6. Instalación de Python

### 8. AREPL for Python

Esta extensión me parece realmente útil, nos ofrece una vista previa en tiempo real de la ejecución de nuestro código.



## 6. Instalación de Python

### 9. AutoDocstring

Perfecta para generar documentación para las funciones de Python.



### 10. Djaneiro

Una extensión para programadores Django, con plantillas, modelos, vistas, campos y formularios.







## 7. IDEs

## 7. IDEs

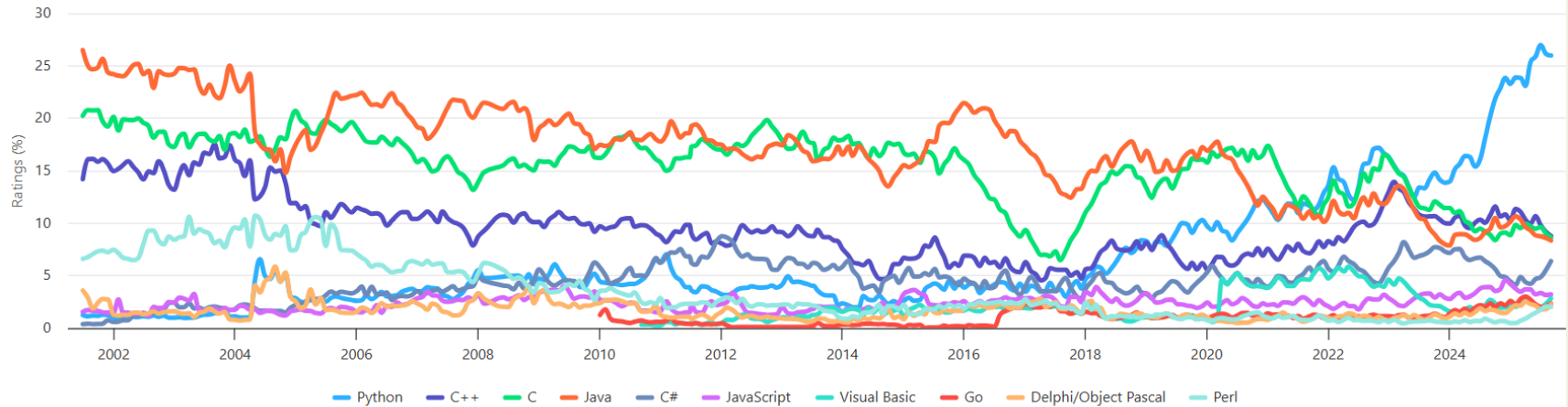














**Y antes de terminar con todo esto  
que estoy contando.... el por qué  
hemos decidido que la optativa  
sea Introducción a Python**

## TIOBE Programming Community Index

Source: [www.tiobe.com](http://www.tiobe.com)



Sep 2025	Sep 2024	Change	Programming Language		Ratings	Change
1	1			Python	25.98%	+5.81%
2	2			C++	8.80%	-1.94%
3	4	▲		C	8.65%	-0.24%
4	3	▼		Java	8.35%	-1.09%
5	5			C#	6.38%	+0.30%
6	6			JavaScript	3.22%	-0.70%
7	7			Visual Basic	2.84%	+0.14%
8	8			Go	2.32%	-0.03%
9	11	▲		Delphi/Object Pascal	2.26%	+0.49%
10	27	▲▲		Perl	2.03%	+1.33%





# ¿Alguna Pregunta?

---

