

# Curso Python 2025 Día 1: Introducción al lenguaje y entorno



# Sobre este curso:

- ✓ Se recomienda tener conocimientos básicos de programación
- ✓ Entenderemos como es un entorno de trabajo en Python
- ✓ Aprenderemos a manejar elementos básicos del lenguaje
- ✓ Nos centraremos en la librería estándar

```
@lru_cache
def fib(n):
    if n <= 1:
        return n
    else:
        return fib(n-2) + fib(n-1)
```



# Contenidos:

## Entorno de trabajo

- PEP8
- venv
- Jupyter

## Variables

- Strings
- Numéricas
- Secuencias

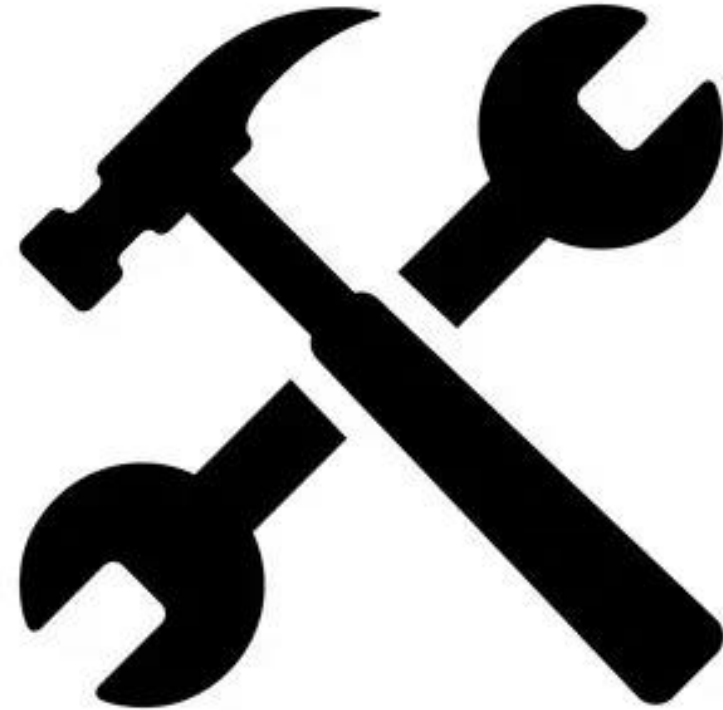
## Estructuras

- Condicionales
- Bucles
- Excepciones\*



# Hoy:

- ✓ Introducción al lenguaje
- ✓ Creación del entorno
- ✓ venv y pip
- ✓ PEP8



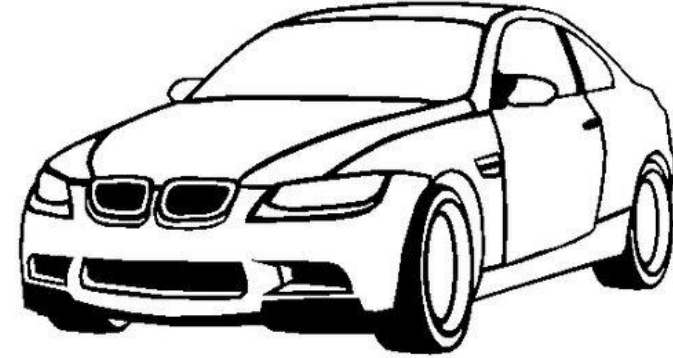
# Introducción al lenguaje

- ¿Qué es Python?
- ¿Quién lo crea?
- ¿Por qué lo utilizaríamos en nuestros proyectos?



# ¿Qué?

- Lenguaje de programación multipropósito
- Es interpretado (se ejecuta según se lee)
- Todo son objetos con atributos y funcionalidades



## Atributos

- Color
- Marca
- Posición
- Velocidad

## Funcionalidades

- Arrancar
- Encender luces
- Acelerar
- Frenar

# ¿Quién?

- Diseña Python poniendo el foco en la legibilidad
- Propone la PEP8
- “El código se lee más de lo que se escribe”- Guido van Rossum



# ¿Por qué?

- Legibilidad
- Disponibilidad y popularidad
- Librerías de ML
- Plasticidad e integración





# Creación del entorno

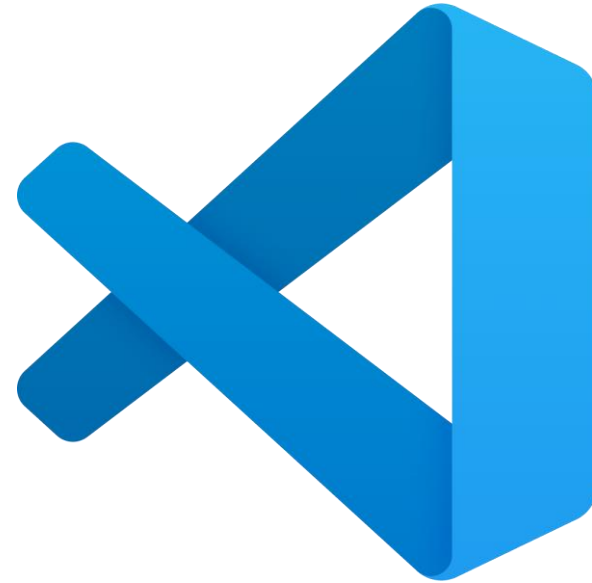
Herramientas:



# Entorno de trabajo: Vscode

Extensiones:

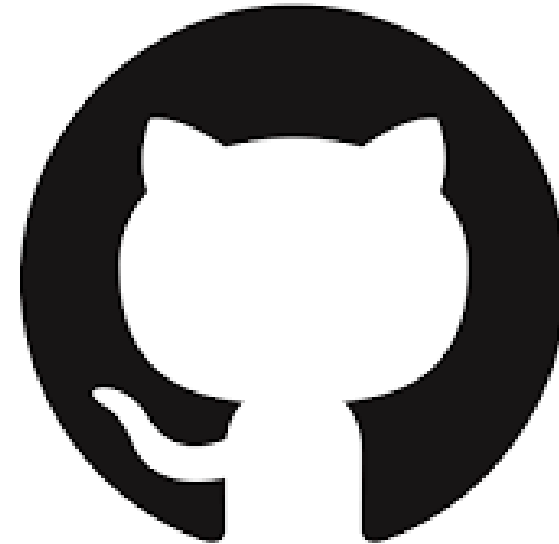
- vscode-icons
- Jupyter (Extensión Pack (4))
- Python



# Entorno de trabajo: GitHub

Archivos y cambios:

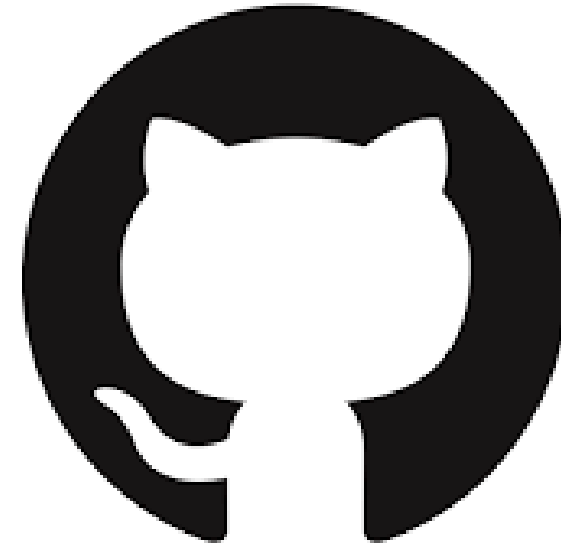
- Plataforma de alojamiento de repositorios
- Se sirve de Git para el control de versiones



# Entorno de trabajo: Git y Github

Archivos estructura de un proyecto:

- README.md
- .gitignore
- requirements.txt (pip)
- Otros



# Entorno de trabajo: Jupyter

Tipos de celdas:

- Markdown
- Code



# Entorno de trabajo: Jupyter Markdown

Ejemplos:

```
1 # Title 1
2 ## Title 2
3
4 - Item 1
5 - Item 2
6   - Subitem 1
7   - Subitem 2
8
9 ```python
10 print('Hello world')
11 ```
12 No line
13 jump
14
15 Line\
16 jump
17
```

Title 1

Title 2

- Item 1
- Item 2
  - Subitem 1
  - Subitem 2

```
print('Hello world')
```

No line jump

Line  
jump

```
1 $e^i - 1 = 0$
```

$$e^i - 1 = 0$$

# Entorno de trabajo: venv

Virtual enviroment:

- Modular
- Copia de nuestro Python global
- Ahí instalaríamos librerías\*



# Entorno de trabajo: venv

## Comandos:

- `python -m venv .venv`
- `.venv\Scripts\activate` (Windows)
- `source .venv/bin/activate` (Linux y Mac)





# Entorno de trabajo: venv y pip

Una vez tengamos activo nuestro venv:

- `pip install <module>`
- `pip install -r requirements.txt`



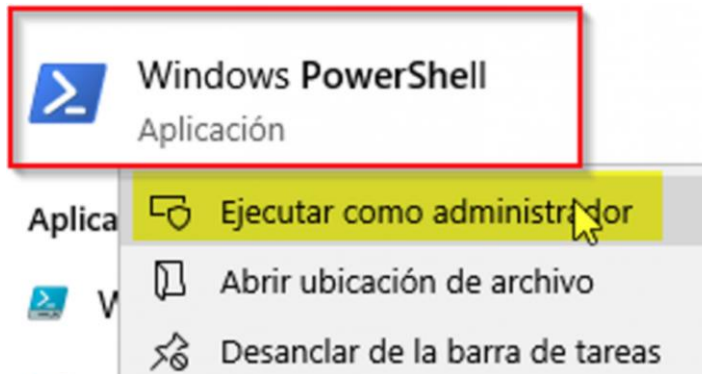
# Problema venv: Windows

```
Windows PowerShell
PS C:\Users\Arturo\Documents> & '.\First Script.ps1'
& : No se puede cargar el archivo C:\Users\Arturo\Documents\First Script.ps1 porque la ejecución de scripts está
deshabilitada en este sistema. Para obtener más información, consulta el tema about_Execution_Policies en
https://go.microsoft.com/fwlink/?LinkID=135170.
En línea: 1 Carácter: 3
+ & '.\First Script.ps1'
+ ~~~~~
+ CategoryInfo          : SecurityError: (:) [], PSSecurityException
+ FullyQualifiedErrorId : UnauthorizedAccess
PS C:\Users\Arturo\Documents>
```

Get-ExecutionPolicy -List

```
Windows PowerShell
PS C:\Users\Arturo\Documents> Get-ExecutionPolicy -List

Scope ExecutionPolicy
-----
MachinePolicy Undefined
UserPolicy Undefined
Process Undefined
CurrentUser Undefined
LocalMachine Undefined
```



Set-ExecutionPolicy RemoteSigned -Scope CurrentUser

```
Windows PowerShell
PS C:\Users\Arturo\Documents> Set-ExecutionPolicy RemoteSigned -Scope CurrentUser
```

# PEP8

- Guía de estilo para la implementación de Python en C (C-Python)
- snake\_case
- Uso de “\_” para variables internas/reservadas/...

snake_case	PascalCase camelCase
bank_account	BankAccount bankAccount
__add__ __lt__ _internal_var	... ... ...

# PEP8: ¿Indiscutible?

“A Foolish Consistency is the Hobgoblin of Little Minds”

- Hay que ser inteligente
- Mira ejemplos y ¡pregunta!
- Juzga lo que crees que es más conveniente
- No rompas compatibilidad de versiones anteriores



# PEP8: ¿Indiscutible?

Pero, si no tienes un buen motivo para romper las reglas:

- PEP8



# Git (Opcional)

Comandos:

- `git clone <remote-repository-url>`
- `git add <filename>`
- `git commit`
- `git push`

