

Proyecto 1 - Python

QUÉ ES UN OBJETO ?

Un **objeto** es una entidad que tiene un **estado** (atributos o propiedades) y un **comportamiento** (métodos o funciones). Además, a los **objetos** en programación les asignamos **nombres** para poder referirnos a ellos y manipularlos.





Proyecto 1 - Python

EN PYTHON “CASI TODO” ES UN OBJETO

Esto incluye números, cadenas de texto, listas, tuplas, diccionarios, funciones, métodos, clases, operaciones aritméticas, etc.

Qué cosas NO son objetos en Python ?

Palabras clave: `if`, `for`, `while`, `def`, `class`, etc. Son parte de la sintaxis del lenguaje.

Sentencias de control de flujo: `if`, `for`, `while`, `with`

Comentarios: Sirven para informar a los programadores.

Proyecto 1 - Python

ESPACIOS DE NOMBRES

(**NameSpaces**): Son un sistema a través del cual Python asigna un nombre único a cada objeto declarado o definido.

Built-In (incorporado): El de **Python**. Variables, funciones, métodos, constantes, etc.

Ejms: `print()`, `input()`, `__name__`

Global: Declarados o definidos a nivel de **Módulos**.

Ejms: Variables, funciones, métodos, constantes, etc.

Local: Declarados o definidos a nivel de **función**.

Ejms: Variables, funciones, métodos, constantes, etc.

Proyecto 1 - Python

ESPACIOS DE NOMBRES

(**Namespaces**): Son un sistema a través del cual Python asigna un nombre único a cada objeto declarado o definido.

Built-In (incorporado): El de **Python**. Variables, funciones, métodos, constantes, etc.

Ejms: `print()`, `input()`, `__name__`

modulo_1

Global: Lo declarado o definido dentro de él

`from modulo_2 ...`
`from modulo_4 ...`

modulo_2

Global: Lo declarado o definido dentro de él

`from modulo_3 ...`

modulo_3

Global: Lo declarado o definido dentro de él

`from modulo_4 ...`

modulo_4

Global: Lo declarado o definido dentro de él

`....`