

DATO DE COLOR :

¿Por qué se llama "Python"? Guido van Rossum, el creador, quería un nombre corto, único y un poco misterioso. En ese momento, él había estado leyendo los Guiones publicados de Flying Circus de Monty Python, un popular programa de la BBC, era un programa de comedia de los años 1970. (Supuestamente obtuvimos el término "spam" de una parodia en ese programa.) Le gustó tanto el programa que decidió nombrar su nuevo lenguaje de programación en homenaje al mismo. Esto demuestra que los programadores tienen sentido del humor. si bien este es ácido y es peculiar y, a veces, está más allá del entendimiento de los simples mortales, existe.

## 🛡️ TIPOS DE DATOS EN PYTHON 🛡️

En Python, los tipos de datos son esenciales para organizar y manipular información. Dominarlos es crucial para escribir código eficiente y preciso. En esta guía, exploraremos los tipos de datos básicos y avanzados, sus características y ejemplos prácticos para ayudarte a dominar este tema fundamental.

---

### Tipos de datos básicos:

#### Numéricos:

- Enteros (int): Números sin decimales, como 1, 2, 3. Ejemplo: edad = 25
- Punto flotante (float): Números con decimales, como 3.14, 1.5. Ejemplo: precio = 12.99
- Complejos (complex): Números con parte real e imaginaria, como 1+2j. Ejemplo: z = 1 + 2j
- Cadenas (str): Secuencias de caracteres, como "Hola mundo", "Python 3". Ejemplo: nombre = "Juan Pérez"
- Booleanos (bool): Valores True o False que representan condiciones. Ejemplo: esta\_activo = True

### Tipos de datos avanzados:

- Listas (list): Colecciones ordenadas y mutables de cualquier tipo de dato. Ejemplo: lista\_nombres = ["Ana", "Pedro", "María"]
- Tuplas (tuple): Colecciones ordenadas e inmutables de cualquier tipo de dato. Ejemplo: coordenadas = (10, 20, 30)
- Diccionarios (dict): Colecciones no ordenadas que mapean claves a valores. Ejemplo: diccionario\_edades = {"Ana": 25, "Pedro": 30}
- Conjuntos (set): Colecciones no ordenadas y sin duplicados de cualquier tipo de dato. Ejemplo: conjunto\_colores = {"azul", "verde", "rojo"}
- Todos los tipos de datos son almacenados en python como objetos. Un objeto es una entidad que contiene datos y métodos. los datos de un objeto se almacenan en atributos, mientras que los metodos son las acciones que el objeto puede realizar

```
edad = 25 # Un valor entero para la edad
print(type(edad)) # permite ver el tipo de variable
```

## Flotante

```
precio = 12.99 # Un valor flotante para el precio
print(type(precio)) # permite ver el tipo de variable
```

## Cadena

```
nombre: str = "Ana Pérez"
#nombre: str los : y el str es un "hint" para decir de que
tipo de datos es la variable pero solo es una "PISTA" no es obligatorio, ya que las
variables en python son dinamicas.
cadena = "volvemos a empezar, tene cuidado que " # Una cadena de texto
print(type(cadena)) # permite ver el tipo de variable
print (cadena * 3) # multiplicar la cadena por 3
```

---

## Booleanos

### Booleano como interruptor para controlar el flujo

```
esta_activo = True # Un valor booleano para indicar si está activo
print(type(esta_activo))
```

---

## Ejemplos de errores:

### ERRORES

- Error: usar un entero para almacenar un precio con decimales

```
precio_entero = 12.99 # Esto puede generar resultados inexactos
```

### Error: usar una cadena para realizar cálculos matemáticos

```
suma_cadenas = "1" + "2" # Esto genera un error de tipo

# Error: usar un booleano para almacenar una lista de nombres

lista_booleana = True, False, True
Esto genera un error de tipo
lista_booleana = true # false porque va en Mayuscula
```

---

## Manejo de cadenas

```
miGrupoFavorito: str = "Guns & Roses"
print(miGrupoFavorito)
print(type(miGrupoFavorito))

# Union/concatenacion de cadenas con el simbolo + al ser string une cadenas de texto
(no suma)
```

```
print("Mi grupo favorito es: " + miGrupoFavorito)

miGrupoFavorito: str = "El Chaqueño Palavecino"
comentario = "Hace musica folclorica"
print(miGrupoFavorito + " " + comentario)
#Otra manera de concatenar
print ("Mi grupo favorito es: ", miGrupoFavorito )
#Otra manera de concatenar
print(f"Mi grupo favorito es: {miGrupoFavorito} {comentario}")
```

## Manejo de cadenas - sumas (concatenación vs suma - Sobre carga de operadores)

```
numero1 = "1"
numero2 = "2"
print (numero1+numero2)

numero1 = 1
numero2 = 2
print (numero1 + numero2)
```

---

## Conversion de tipos de datos

```
numero1 = "1"
numero2 = "2"
print (int(numero1)+int(numero2))
```

## Tipos de datos Bool (boolean)

```
miVariableTrue = True
miVariableFalse = False
print (miVariableTrue)

miVariable = 3 > 3
print (miVariable)
```

---

En Python, if y else son dos palabras clave (que veremos más adelante) que se usan para crear sentencias condicionales. Estas permiten ejecutar diferentes bloques de código dependiendo de si una condición se cumple o no.

Por ejemplo, Imaginen que queremos saber si un número es mayor que 10.

```
numero = 15
if numero > 10:
    print("El número es mayor que 10")
else:
    print("El número no es mayor que 10")
#! nota: importante el tabulado en el if y else
```

## Procesar entrada de datos del usuario

### FUNCION INPUT

```
# Funcion input para procesar la entrada de datos del usuario
resultado = input("Escribe un número:")
print (resultado)
```

### Convertir los datos

#### Ya que la funcion INPUT entrega un string

```
numero1 = input("Escribe un número:")
numero2 = input("Escribe otro número:")
print (int(numero1) + int(numero2))
```

#### Otra Manera de Hacerlo

```
numero1:int = int(input("Escribe un número:"))
numero2:int = int(input("Escribe otro número:"))
print (numero1 + numero2)
```

---

## Actividades

### Ejercicio 1:

Objetivo: Pedir al usuario dos números, convertirlos a enteros y mostrar la suma.

### Ejercicio 2:

Pedirle al usuario que ponga como estuvo su día (del 1 al 10 ) y luego mostrar el texto "Mi día estuvo de:" y el valor ingresado

### Ejercicio 3:

Se solicita incluir la siguiente información acerca de un libro :

Título, autor, número de páginas, año de publicación.

Se pide que el usuario ingrese la información solicitada de la siguiente manera:.

- Ingrese el Título del libro :
- Ingrese el Autor :
- Ingrese el Numero de Páginas :
- Ingrese el Año de publicación :
- Imprimir : " El Libro <nombre del libro> fue escrito por <autor> y tiene <numero de paginas> paginas, y fue publicado en el año <año de publicacion>"

---

[VOLVER](#)