

# Manual Educativo y Técnico de Python

---

**Nivel:** Básico – Intermedio

**Autor:** Lorenzo Pérez Carbonell

**Objetivo:** Aprender Python desde los fundamentos hasta la programación estructurada con un enfoque práctico y comprensible.

---

## 1. Introducción

Python es un lenguaje de programación interpretado, de alto nivel y muy fácil de leer.

Se utiliza en áreas como desarrollo web, ciencia de datos, automatización, inteligencia artificial y más.

### Ventajas:

- Sintaxis sencilla y cercana al lenguaje humano.
  - Gran comunidad y millas de librerías.
  - Multiplataforma (Windows, macOS, Linux).
- 

## 2. Fundamentos del Lenguaje

### 2.1 Comentarios

Los comentarios sirven para documentar el código, mejorar su legibilidad y facilitar el mantenimiento.

```
# Esto es un comentario de una sola línea

"""
Esto es un comentario
de varias líneas.
Ideal para documentación.
"""
```

---

### 2.2 Variables y Reglas de Nombres

- Se crean al asignar un valor: `nombre = "Ana"`

- No se declaran tipos explícitos.
- Hijo **sensible a mayúsculas y minúsculas** ( `Edad`  $\neq$  `edad` ).

```
nombre = "Lucas"
edad = 25
es_activo = True
```

Convención: las constantes se escriben en MAYÚSCULAS:

```
PI = 3.1416
```

---

## 3. Tipos de datos en Python

Tipo	Descripción	Ejemplo	Resultado
<code>int</code>	Entero	<code>x = 10</code>	10
<code>float</code>	Decimal	<code>x = 3.14</code>	3.14
<code>str</code>	Texto	<code>x = "Hola"</code>	Hola
<code>bool</code>	Booleano	<code>x = True</code>	Verdadero
<code>None</code>	Sin valor	<code>x = None</code>	Ninguno

---

### Ejemplo:

```
nombre = "Sofía"
edad = 21
print(f"{nombre} tiene {edad} años.")
```

### Salida:

Sofía tiene 21 años.

---



## 4. Operadores y Expresiones

### 4.1 Aritméticos

Operador	Descripción	Ejemplo	Resultado
+	Suma	3 + 2	5
-	Resto	5 - 1	4
*	Multiplicación	4 * 2	8
/	División	5 / 2	2.5
//	División entera	5 // 2	2
%	Módulo	5 % 2	1
**	Potencia	2 ** 3	8

## 4.2 Comparación

Operador	Significado	Ejemplo	Resultado
==	Igual	5 == 5	Verdadero
!=	Distinto	3 != 2	Verdadero
>	Mayor	4 > 2	Verdadero
<	Menor	2 < 5	Verdadero
>=	Mayor o igual	5 >= 5	Verdadero
<=	Menor o igual	4 <= 5	Verdadero

## 4.3 Lógicos

Operador	Ejemplo	Resultado
and	True and False	FALSO
or	True or False	Verdadero
not	not True	FALSO

## Ejemplo:

```
x = 10
y = 5
```

```
if x > y and y > 0:
    print("Condición cumplida.")
```

**Salida:**

Condición cumplida.

---

## 5. Estructuras de control

### 5.1 Condicionales

```
nota = 85

if nota >= 90:
    print("Excelente")
elif nota >= 70:
    print("Aprobado")
else:
    print("Reprobado")
```

**Salida:**

Aprobado

---

### 5.2 Operador Ternario

```
edad = 20
mensaje = "Mayor" if edad >= 18 else "Menor"
print(mensaje)
```

**Salida:**

Mayor

---

### 5.3 `match` (Python 3.10+)

```
opcion = "guardar"
```

```
match opcion:
    case "abrir":
        print("Abriendo archivo...")
    case "guardar":
        print("Guardando cambios...")
    case _:
        print("Opción no válida.")
```

**Salida:**

Guardando cambios...

---

## 6. Bucles

### 6.1 for

```
for i in range(3):
    print(f"Iteración {i}")
```

**Salida:**

Iteración 0  
Iteración 1  
Iteración 2

---

### 6.2 while

```
i = 0
while i < 3:
    print(i)
    i += 1
```

**Salida:**

0  
1  
2

---

## 6,3 break años continúe

```
for num in range(5):  
    if num == 3:  
        break  
    print(num)
```

**Salida:**

```
0  
1  
2
```

---

## 7. Estructuras de Datos

### 7.1 Listas

```
frutas = ["manzana", "pera", "uva"]  
frutas.append("kiwi")  
print(frutas[1])
```

**Salida:**

```
pera
```

---

### 7.2 Tuplas

```
coordenadas = (10, 20)  
print(coordenadas[0])
```

---

### 7.3 Diccionarios

```
persona = {"nombre": "Ana", "edad": 30}  
print(persona["nombre"])
```

**Salida:**

## 7.4 Conjuntos ( `set` )

```
numeros = {1, 2, 3, 3}
print(numeros)
```

**Salida:**

```
{1, 2, 3}
```

---

## 8. Funciones

```
def sumar(a, b):
    """Devuelve la suma de dos números"""
    return a + b

resultado = sumar(4, 3)
print(resultado)
```

**Salida:**

```
7
```

---

## 9. Módulos e Importaciones

```
import math
print(math.sqrt(25))
```

**Salida:**

```
5.0
```

---

## 10. Manejo de errores

```
try:
    numero = int(input("Ingresa un número: "))
    print(10 / numero)
except ZeroDivisionError:
    print("Error: no puedes dividir entre cero.")
except ValueError:
    print("Error: debes ingresar un número.")
finally:
    print("Fin del programa.")
```

---

## 11. Archivos

### Escribir y leer archivos:

```
# Escribir
with open("notas.txt", "w") as archivo:
    archivo.write("Hola, mundo")

# Leer
with open("notas.txt", "r") as archivo:
    print(archivo.read())
```

---

## 12. Fechas y Tiempo

```
from datetime import datetime
ahora = datetime.now()
print(ahora.strftime("%d/%m/%Y %H:%M:%S"))
```

### Salida:

15/10/2025 12:30:00

---

## 13. Ejercicio práctico

### Problemas:



Crea un programa que pida al usuario su nombre y edad, y diga si puede votar (edad  $\geq$  18).

## Código:

```
nombre = input("Ingresa tu nombre: ")
edad = int(input("Ingresa tu edad: "))

if edad >= 18:
    print(f"{nombre}, puedes votar.")
else:
    print(f"{nombre}, aún no puedes votar.")
```

## Explicación:

- Se capturan dos entradas del usuario.
  - Se convierte la edad a entera.
  - Se evalúa con una estructura condicional.
- 

## 14. Conclusión

Ha aprendido los **fundamentos esenciales de Python**, su estructura, lógica y manipulación de datos. Este conocimiento te permitirá avanzar hacia temas como **P00, automatización, bases de datos y análisis de datos**.