# -----------------------------------------------------------------------------

# 📌 1️⃣ TIPOS NUMÉRICOS EN PYTHON

# -----------------------------------------------------------------------------

# 🔹 Enteros (int)

# Los enteros son números sin decimales, pueden ser positivos o negativos.

numero\_entero = 42

numero\_negativo = -10

print("Número entero:", numero\_entero)

print("Número negativo:", numero\_negativo)

print("Tipo de dato:", type(numero\_entero)) # <class 'int'>

# 🔹 Decimales (float)

# Los números flotantes son aquellos que tienen decimales.

numero\_decimal = 3.14159

temperatura = -12.5

print("Número decimal:", numero\_decimal)

print("Temperatura:", temperatura)

print("Tipo de dato:", type(numero\_decimal)) # <class 'float'>

# 🔹 Números complejos (complex)

# Los números complejos tienen una parte real y una imaginaria (se usa "j").

num\_complejo = 2 + 3j

print("Número complejo:", num\_complejo)

print("Parte real:", num\_complejo.real, "| Parte imaginaria:", num\_complejo.imag)

print("Tipo de dato:", type(num\_complejo)) # <class 'complex'>

# 🔹 Operaciones matemáticas básicas con números

suma = 10 + 5

resta = 10 - 3

multiplicacion = 4 \* 2

division = 10 / 3 # Siempre devuelve un float

division\_entera = 10 // 3 # División sin decimales

modulo = 10 % 3 # Resto de la división

potencia = 2 \*\* 3 # Exponente (2 elevado a 3)

print("\nOperaciones matemáticas:")

print("Suma:", suma, "| Resta:", resta, "| Multiplicación:", multiplicacion)

print("División:", division, "| División entera:", division\_entera)

print("Módulo:", modulo, "| Potencia:", potencia)

# 📌 EJERCICIO PRÁCTICO 1️⃣:

# 📌 Crea dos números enteros y muestra la suma y la resta de ellos.

# 📌 Declara un número flotante y muestra su valor dividido entre 2.

# -----------------------------------------------------------------------------

# 📌 2️⃣ BOOLEANOS (bool)

# -----------------------------------------------------------------------------

# Un booleano solo puede tener dos valores: True (verdadero) o False (falso)

es\_python\_facil = True

es\_python\_dificil = False

print("\nBooleanos:")

print("Es Python fácil?", es\_python\_facil)

print("Es Python difícil?", es\_python\_dificil)

# 🔹 Comparaciones que devuelven valores booleanos

mayor\_que = 10 > 5 # True porque 10 es mayor que 5

menor\_que = 10 < 5 # False porque 10 no es menor que 5

igualdad = 10 == 10 # True porque ambos son iguales

diferente = 10 != 5 # True porque 10 y 5 son distintos

print("10 es mayor que 5:", mayor\_que)

print("10 es menor que 5:", menor\_que)

print("10 es igual a 10:", igualdad)

print("10 es distinto de 5:", diferente)

# 🔹 Operaciones lógicas con booleanos

and\_logico = True and False # False porque ambos deben ser True

or\_logico = True or False # True porque al menos uno es True

not\_logico = not True # False porque invierte el valor

print("\nOperaciones lógicas:")

print("True AND False:", and\_logico)

print("True OR False:", or\_logico)

print("NOT True:", not\_logico)

# 📌 EJERCICIO PRÁCTICO 2️⃣:

# 📌 Declara una variable booleana que sea True si 15 es mayor que 8.

# 📌 Escribe una expresión que use "and" y otra que use "or" y muestra el resultado.

# -----------------------------------------------------------------------------

# 📌 3️⃣ CADENAS DE TEXTO (str)

# -----------------------------------------------------------------------------

# 🔹 Una cadena de texto es una secuencia de caracteres delimitados por comillas

texto\_simple = "Hola, mundo"

texto\_doble = 'Python es genial'

print("\nCadenas de texto:")

print("Texto simple:", texto\_simple)

print("Texto doble:", texto\_doble)

# 🔹 Concatenación de cadenas (unir textos)

mensaje = "Python" + " es " + "increíble"

print("Concatenación de cadenas:", mensaje)

# 🔹 Repetición de cadenas

repetida = "Python! " \* 3

print("Repetición de cadenas:", repetida)

# 🔹 Acceder a caracteres de una cadena usando índices

palabra = "Python"

primera\_letra = palabra[0] # Primer carácter (P)

ultima\_letra = palabra[-1] # Último carácter (n)

print("\nAcceso a caracteres:")

print("Primera letra:", primera\_letra)

print("Última letra:", ultima\_letra)

# 🔹 Slicing (extraer partes de una cadena)

subcadena = palabra[0:3] # Obtiene 'Pyt'

print("Subcadena:", subcadena)

# 🔹 Métodos útiles para cadenas

mayusculas = palabra.upper() # Convierte en mayúsculas

minusculas = palabra.lower() # Convierte en minúsculas

longitud = len(palabra) # Devuelve la cantidad de caracteres

print("\nMétodos de cadenas:")

print("Mayúsculas:", mayusculas)

print("Minúsculas:", minusculas)

print("Longitud de la palabra:", longitud)

# 📌 EJERCICIO PRÁCTICO 3️⃣:

# 📌 Crea una variable con tu nombre y apellidos y muestra su longitud.

# 📌 Concatena tu nombre con una frase de presentación.

# 📌 Convierte tu nombre en mayúsculas y minúsculas.

# 📌 Extrae los primeros 3 caracteres de tu nombre usando slicing.

# -----------------------------------------------------------------------------

# 📌 FIN DEL SCRIPT 🎯

# -----------------------------------------------------------------------------