

## TALLER – FUNDAMENTOS DE PYTHON

### SECCIÓN 1: VARIABLES

1. Intercambio sin variable auxiliar Intercambia los valores de dos variables a y b sin usar una variable extra

```
num1 = int(input("Ingresa un numero: "))
num2 = int(input("Ingresa un numero: "))
```

```
print(f"Antes de Intercambiar")
print(f"Numero 1: {num1}")
print(f"Numero 2: {num2}")
num1, num2 = num2, num1
```

```
print(f"Despues de Intercambiar")
print(f"Numero 1: {num1}")
print(f"Numero 2: {num2}")
```

2. Suma de cuadrados Declara dos variables  $a = 5$ ,  $b = 3$ , y almacena el resultado de  $a^2 + b^2$  en una tercera variable c.

```
a = 5
b = 3
c = (a ** 2) + (b ** 2)
print(f"El resultado de la operacion es: {c}")
```

3. Conversión de tipos Convierte un número entero en string, luego a float y finalmente de nuevo a int. Muestra los resultados.

```
numero = int(input("Ingresa un numero: "))
print(f"Numero original: {numero} {type(numero)}")
numString = str(numero)
print(f"Numero en String: {numString} {type(numString)}")
flotante = float(numString)
print(f"Numero en Flotante: {flotante} {type(flotante)}")
entero = int(flotante)
print(f"Numero en Entero: {entero} {type(entero)}")
```

4. Redondeo y formato Redondea el número 17.8567 a dos decimales y convierte el resultado en string para imprimirlo como: "Resultado: 17.86".

```
numero = 17.8567
redondeo = round(numero, 2)
print(f"Numero original: {numero}")
print(f"Numero redondeado a dos decimales: {redondeo}")
```

5. Concatenación de variables Declara dos variables nombre y edad, y construye un mensaje como: "Hola, mi nombre es Juan y tengo 25 años."

```
nombre = input("Ingresa tu nombre: ")
edad = int(input("Ingresa tu Edad: "))
print(f"Hola!, mi nombre es: {nombre} y tengo {edad} años")
```

6. Cálculo de edad actual Usando anio\_actual y anio\_nacimiento, calcula y muestra la edad actual.

```
ANO_ACTUAL = 2025
ano_nacimiento = int(input("Ingresa el año de nacimiento: "))
edad_actual = ANO_ACTUAL - ano_nacimiento
print(f"Usted tiene: {edad_actual} años")
```

7. Cuenta regresiva con variables Simula una cuenta regresiva desde 10 hasta 0 usando una sola variable.

```
print("¡Comienza la cuenta regresiva!")
for contador in range(10, -1, -1):
    print(contador)
print("¡Despegue!")
```

8. Valor absoluto sin usar abs() Implementa una lógica para obtener el valor absoluto de un número entero sin usar abs().

```
numeros = [26, -5, 0, -78]
```

```
for num in numeros:
```

```
    valor_absoluto_actual = num if num >= 0 else -num
```

```
    print(f"El valor absoluto de {num} es {valor_absoluto_actual}")
```

9. Incremento/decremento según paridad Si una variable n es par, suma 1. Si es impar, réstale 1. Imprime el nuevo valor.

```
numero = int(input("Ingresa tu numero: "))
```

```
if (numero % 2 == 0):
```

```
    print(f"Numero original: {numero}")
```

```
    par = numero + 1
```

```
    print(f"Numero sumado: {par}")
```

```
else:
```

```
    print(f"Numero original: {numero}")
```

```
    impar = numero - 1
```

```
    print(f"Numero restado: {impar}")
```

10. Máximo entre tres números Dadas tres variables a, b y c, encuentra y muestra cuál es la mayor sin usar max().

```
a = float(input("Ingresa el primer número: "))
```

```
b = float(input("Ingresa el segundo número: "))
```

```
c = float(input("Ingresa el tercer número: "))
```

```
if a > b:
```

```
    if a > c:
```

```
        mayor = a
```

```
    else:
```

```
        mayor = c
```

```
else:
```

```
    if b > c:
```

```
        mayor = b
```

```
    else:
```

```
        mayor = c
```

```
print(f"El número mayor es: {mayor}")
```

## SECCIÓN 2: OPERADORES RELACIONALES

- 11. Verificación de rango** Verifica si un número entero  $x$  está entre 10 y 100 (inclusive). Imprime el resultado booleano

```
x = 50
en_rango = 10 <= x <= 100
print("¿Está en rango?:", en_rango)
```

- 12. Comparación de strings (ignorar mayúsculas)** Dadas dos cadenas  $a$  y  $b$ , comprueba si son iguales ignorando mayúsculas/minúsculas.

```
a = "Diego"
b = "DIEGO"
son_iguales = a.lower() == b.lower()
print("¿Son iguales ignorando mayúsculas?:", son_iguales)
```

- 13. Igualdad entre tres variables** Verifica si tres variables distintas tienen el mismo valor.

```
x1 = 10
x2 = 10
x3 = 10
iguales = x1 == x2 == x3
print(" ¿Son iguales las tres variables?:", iguales)
```

- 14. Presencia en lista** Declara una lista de números y verifica si un número  $x$  está contenido en ella.

```
numeros = [1, 3, 5, 7, 9]
x = 5
esta_en_lista = x in numeros
print(" ¿Está en la lista?:", esta_en_lista)
```

- 15. Divisibilidad por 3 y 5** Dado un número  $n$ , determina si es divisible simultáneamente por 3 y 5.

```
n = 15
es_divisible= (n % 3 == 0) and (n % 5 == 0)
print("Es divisible por 3 y 5 ", es_divisible)
```

16. Números en orden estricto Verifica si tres variables a, b, c están en orden ascendente estricto ( $a < b < c$ )

```
a = 5
b = 10
c = 15
orden_estricto = a < b < c
print(f"{a}, {b}, {c} están en orden estricto? {orden_estricto}")
```

17. Comparación doble Dado un número x, y dos límites a y b, verifica si x está estrictamente entre a y b.

```
x = 7
limite_a = 5
limite_b = 10
esta_entre = limite_a < x < limite_b
print(f"{x} está entre {limite_a} y {limite_b}: {esta_entre}")
```

18. Relación proporcional Verifica si una variable a es exactamente el doble de otra variable b.

```
a = 10
b = 5
doble = a == 2 * b
print(f"{a} es el doble de {b}: {doble}")
```

19. Cambio de signo si negativo Si una variable x es negativa, conviértela en positiva. Si ya es positiva, déjala igual.

```
x = 42
tipo = type(x).__name__
print(f" El tipo de {x} es: {tipo}")
```

20. Detección de tipo Dada una variable x, imprime si es un entero (int), flotante (float) o cadena (str).

```
x_int = 42
x_float = 3.14
x_str = "hola"
print(f" {x_int} es {type(x_int).__name__}")
print(f" {x_float} es {type(x_float).__name__}")
print(f" {x_str} es {type(x_str).__name__}")
```

### Sección 3: Operadores Lógicos (21–30)

21. ¿Puede votar? Verifica si una persona puede votar. Debe tener al menos 18 años y tener documento de identidad (True o False).

```
edad = 20
tiene_documento = True
puede_votar = edad >= 18 and tiene_documento
print(f"Puede votar: (edad: {edad}, documento: {tiene_documento}) = {puede_votar}")
```

22. Control de acceso lógico Una persona puede entrar si: o Tiene pase VIP, o o Paga entrada y no está ebria. Evalúa esto con tres variables booleanas.

```
pase_vip = False
paga_entrada = True
esta_ebria = False
puede_entrar = pase_vip or (paga_entrada and not esta_ebria)
print("22. ¿Puede entrar?:", puede_entrar)
```

23. Validación XOR Dadas dos condiciones booleanas cond1 y cond2, verifica si exactamente una es verdadera (operación XOR lógica).

```
cond1 = True
cond2 = False
xor_logico = (cond1 or cond2) and not (cond1 and cond2)
print("23. XOR lógico:", xor_logico)
```

24. Validación compuesta múltiple Dado un número n, verifica si es mayor que 0 y (múltiplo de 2 o de 3)

```
n = 6
es_valido = n > 0 and (n % 2 == 0 or n % 3 == 0)
print(" Es positivo y múltiplo de 2 o 3 :", es_valido)
```

25. Condición contradictoria Escribe una condición lógica que nunca se cumple (una contradicción), como  $x > 5$  and  $x < 1$ .

```
x = 3
contradiccion = x > 5 and x < 1
print(" Contradicción (x={x} > 5 and x < 1): {contradiccion}")
```

26. Negación lógica Dada una condición  $x > 10$ , reescríbela usando not para que tenga el mismo efecto lógico

```
x = 8
# Condición original:  $x > 10$ 
# Negación equivalente:  $\text{not } (x > 10)$  que es lo mismo que  $x \leq 10$ 
condicion_original = x > 10
condicion_negada = not (x > 10)
condicion_equivalente = x <= 10
print(f" x={x} > 10: {condicion_original}")
print(f" not (x > 10): {condicion_negada}")
print(f" x <= 10: {condicion_equivalente}")
```

27. Aprobación de estudiante Verifica si un estudiante aprueba:  $\text{nota} \geq 3.0$  y  $\text{asistencia} \geq 80$ .

```
nota = 3.5
asistencia = 85
aprueba = nota >= 3.0 and asistencia >= 80
print(" ¿El estudiante aprueba?:", aprueba)
```

28. Simulación de alarma Si la temperatura es menor a 0 o mayor a 38 y la humedad es mayor a 80%, se activa una alarma.

```
temperatura = -5
humedad = 85
alarma = (temperatura < 0 or temperatura > 38) and humedad > 80
print(f" Alarma (temp: {temperatura}°C, humedad: {humedad}%) = {alarma}")
```

29. Contraseña segura Verifica si una contraseña cumple con: o Longitud mayor a 8 caracteres o Contiene al menos un número (Pista: puedes usar `any(char.isdigit() for char in passw)`)

```
password = "mipassword123"
longitud_ok = len(password) > 8
tiene_numero = any(char.isdigit() for char in password)
es_segura = longitud_ok and tiene_numero
print(f" ¿Contraseña segura? '{password}' = {es_segura}")
print(f" Longitud > 8: {longitud_ok}")
print(f" Tiene número: {tiene_numero}")
```

30. Doble negación lógica Evalúa una expresión como: "No es falso que no tenga acceso" usando operadores lógicos. Traduce eso a Python.

```
tiene_acceso = False
resultado = not (not tiene_acceso)
print(" Resultado de doble negación:", resultado)
```