



Practica 03 - Tipos de datos primitivos



Instrucciones

Entregar los ejercicios en uno o varios archivos.py

Parte 1



Ejercicio 3.1 — Evaluar tipos en una expresión compleja

Enunciado:

Evalúa la siguiente expresión y determina qué tipo de dato obtiene Python:

```
resultado = (10 +2.5) *3 /2
```

```
# Escribe tu código aquí
```

Salida esperada:

```
Resultado:18.75  
Tipo: <class'float'>
```

Ejercicio 3.2 — Identificar tipo tras conversión de texto a número

Enunciado:

Convierte la cadena `"45.67"` a un número y verifica su tipo.

```
# Escribe tu código aquí
```

Salida esperada:

```
Valor : 45.67  
Tipo: <class'float'>
```

Ejercicio 3.3 — Determinar tipos en una expresión lógica

Enunciado:

Evalúa la expresión `5 > 3 and 2 < 8` y muestra su tipo.

```
# Escribe tu código aquí
```

Salida esperada:

```
Resultado:True  
Tipo: <class'bool'>
```

Parte 2

Ejercicio 3.4

Enunciado: Dada la lista de notas de un grupo, encuentra la nota máxima, mínima y el promedio.

```
notas = [8.5,7.2,9.0,6.8,10,7.9]
```

Escribe aquí tu solución

Salida esperada:

Máxima: 10

Mínima: 6.8

Promedio: 8.23

Ejercicio 3.5

Enunciado: Ordena las notas de menor a mayor y luego invierte el orden.

```
notas = [8.5,7.2,9.0,6.8,10,7.9]
```

Escribe aquí tu solución

Salida esperada:

Orden ascendente:[6.8, 7.2, 7.9, 8.5, 9.0, 10]

Orden descendente:[10, 9.0, 8.5, 7.9, 7.2, 6.8]

Ejercicio 3.6

Enunciado: Crea una lista de productos y precios. Agrega un nuevo producto y elimina otro.

```
productos = ["leche","pan","queso","café"]
```

```
precios = [1.5,1.0,3.2,2.8]
```

Escribe aquí tu solución

Salida esperada:

Inventario actualizado:
['leche','pan','café','miel']
[1.5, 1.0, 2.8, 4.0]

Ejercicio 3.7

Enunciado: Dada una lista de precios, encuentra el precio más alto y el más bajo, y calcula la diferencia.

```
precios = [23.5,19.9,45.0,32.2,27.5]  
# Escribe aquí tu solución
```

Salida esperada:

```
Máximo: 45.0  
Mínimo: 19.9  
Diferencia: 25.1
```

Ejercicio 3.8

Enunciado: Combina las listas de nombres y edades para formar una lista única de pares.

```
nombres = ["Ana","Luis","Sofía"]  
edades = [20,22,19]  
# Escribe aquí tu solución
```

Salida esperada:

```
[["Ana", 20],["Luis", 22],["Sofía", 19]]
```