

# Ejercicio – Operadores, Conversión de Tipos y Funciones Básicas

February 16, 2024

## 1 Ejercicio – Operadores, Conversión de Tipos y Funciones Básicas

Presenta: Juliho Castillo Colmenares

<https://github.com/julihocc/ebac-python-backend>

### 1.1 Instrucciones

Crea un Jupyter Notebook para el ejercicio y escribe un programa pidiéndole al usuario que seleccione una operación (suma, resta, multiplicación, división y potencia) e ingrese una lista, un set y un diccionario con 5 números. Recuerda que puedes usar la función `type` para saber el tipo de entrada (lista, set, diccionario) y utilizar operadores para validar si la entrada tiene una estructura correcta. Posteriormente imprime los valores de entrada, la operación solicitada y el resultado de la operación. En caso de que la entrada sea inválida, muestra un mensaje especificando el usuario, por ejemplo “entrada inválida” o “tamaño inválido en la entrada”.

```
[ ]: from functools import reduce

def reducirCincoElementos(estructura, operacion):
    tipo = type(estructura)
    if tipo in [list, set, dict]:
        if len(estructura) == 5:
            if type(estructura) == dict:
                estructura = estructura.values()
            if operacion in ["suma", "resta", "multiplicacion", "division", ↵
↵"potencia"]:
```

```
            if operacion == "suma":
                return reduce(lambda x, y: x + y, estructura)
            elif operacion == "resta":
                return reduce(lambda x, y: x - y, estructura)
            elif operacion == "multiplicacion":
                return reduce(lambda x, y: x * y, estructura)
            elif operacion == "division":
                return reduce(lambda x, y: x / y, estructura)
            elif operacion == "potencia":
                return reduce(lambda x, y: x ** y, estructura)
        else:
```

```

        raise ValueError("Operacion no valida")
    else:
        raise ValueError("Longitud de la estructura no valida")
else:
    raise ValueError("Tipo de estructura no valida")

```

```
[ ]: reducirCincoElementos([1, 2, 3, 4, 5], "suma")
```

```
[ ]: 15
```

```
[ ]: reducirCincoElementos({1,2,3,4,5}, "suma")
```

```
[ ]: 15
```

```
[ ]: reducirCincoElementos({1:1, 2:2, 3:3, 4:4, 5:5}, "suma")
```

```
[ ]: 15
```

```
[ ]: try:
    reducirCincoElementos((1,2,3,4,5), "suma")
except ValueError as e:
    print(e)

```

Tipo de estructura no valida

```
[ ]: try:
    reducirCincoElementos([1,2,3,4], "suma")
except ValueError as e:
    print(e)

```

Longitud de la estructura no valida

```
[ ]: try:
    reducirCincoElementos([1,2,3,4,5], "interseccion")
except ValueError as e:
    print(e)

```

Operacion no valida