

# Conversión de Temperaturas

¿Qué pide el enunciado?

Convertir una temperatura dada en grados Celsius a dos nuevas unidades: **Fahrenheit** y **Kelvin**.

¿Qué tipo de datos y operadores utilicé?

- **Tipo de dato:** `float` (números decimales).
  - **Operadores utilizados:**
    - Multiplicación `*`
    - División `/`
    - Suma `+`
- 

¿Qué validaciones implementé o podría haber implementado?

- Validar que el usuario ingrese un **número** y no un **texto**.
- Validar que la temperatura no sea **menor al cero absoluto** (-273.15 °C).

```
# Convertidor de temperaturas

# Solicitar al usuario que ingrese la temperatura en grados Celsius
celsius = float(input("Ingrese la temperatura en grados Celsius: "))

# Realizar las conversiones
fahrenheit = (celsius * 9/5) + 32
kelvin = celsius + 273.15

# Mostrar los resultados
print(f"\nLa temperatura en Fahrenheit es: {fahrenheit:.2f} °F")
print(f"La temperatura en Kelvin es: {kelvin:.2f} K")
```

# Desafío 4: Verificar múltiplos de varios números

## ¿Qué pide el enunciado?

Solicitar un número al usuario y verificar si es múltiplo de **2, 3, 5, 7, 9, 10 y 11**, mostrando un mensaje por cada caso.

---

## ¿Qué tipo de datos y operadores utilicé?

- **Tipo de dato:** `int` (número entero).
  - **Operador utilizado:**
    - Módulo `%` para saber si un número es múltiplo de otro.
- 

## ¿Qué validaciones implementé o podría haber implementado?

- Verificar que el ingreso sea un **número entero válido**.
  - Podríamos evitar el ingreso de **números negativos** si el programa así lo requiere.
- 

```
# Verificar múltiplos

# Solicitar al usuario que ingrese un número
numero = int(input("\nIngrese un número entero: "))

# Lista de números a verificar
numeros_a_verificar = [2, 3, 5, 7, 9, 10, 11]

# Verificar y mostrar los resultados
print()
for divisor in numeros_a_verificar:
    if numero % divisor == 0:
        print(f"El número {numero} es múltiplo de {divisor}.")
    else:
        print(f"El número {numero} NO es múltiplo de {divisor}.")
```

## 📖 Webgrafía

- [Documentación oficial de Python](#)  
Guía completa del lenguaje Python, incluyendo funciones como `input()`, `float()`, operadores, tipos de datos, etc.
- [W3Schools - Python Input and Output](#)  
Explicación sencilla sobre cómo usar `input()` y convertir datos en Python.

## Enlace a repositorio de Github

- <https://github.com/GracielaBerio/programacion1>