



Práctica 3: Conversor de Unidades

Manipulación de tipos de datos matemáticos y redondeo
Curso de Programación en Snap! y Pseudocódigo

Los Tipos de Datos y la Precisión

Los ordenadores son excelentes calculadoras, pero al realizar divisiones o multiplicar por fracciones, suelen generar números con muchos decimales (ej: 15.6666667). Para que nuestros programas sean profesionales y fáciles de leer, debemos aprender a **redondear** esos valores.

Objetivos de esta práctica

- Manipular tipos de datos numéricos (enteros y decimales/reales).
- Entender el concepto de *constante de conversión* (factores multiplicadores).
- Aplicar la función de **redondeo** para mejorar la interfaz de usuario.
- Anidar operaciones matemáticas complejas.

Paso a Paso: Snap! vs Pseudocódigo

1. La Constante de Conversión

Para pasar de una unidad a otra, multiplicamos por un factor. Por ejemplo, para Euros a Dólares (1.08):

- **En Snap!:** Usamos el bloque de multiplicación: (`euros * 1.08`).
- **En Pseudocódigo:** `dolares <- euros * 1.08`

2. Manipulación: El Redondeo

Para evitar números como 10.799999:

- **En Snap!:** En la categoría **Operadores**, usamos el bloque `redondear ()`. Debes introducir toda la operación matemática dentro de este bloque.
- **En Pseudocódigo:** Usamos la función interna `REDONDEAR()` o `REDON()`.
- *Ejemplo:* `dolares_limpios <- REDONDEAR(dolares)`

3. Formateo de la salida

El usuario necesita entender el dato final combinando texto y variables:

- **En Snap!:** Usamos `unir [El valor es] [dolares_limpios]`.
- **En Pseudocódigo:** Escribir "El valor es ", dolares_limpios

EL RETO: El Calculador de Viajes (Km a Millas)

Crea una herramienta para un turista. Tu programa debe convertir Kilómetros a Millas y redondear el resultado para que no tenga decimales. (Fórmula: $1 \text{ km} = 0.62 \text{ millas}$).

PARTE A: En Snap!

1. Pregunta la distancia en **Kilómetros** y guárdala en una variable `km`.
2. Crea una variable `millas`.
3. Fija `millas` al resultado de **redondear** la multiplicación de `km` por 0.62.
4. Muestra un mensaje final uniendo texto y el resultado redondeado.

PARTE B: En Pseudocódigo Escribe el algoritmo correspondiente en tu cuaderno:


```
Algoritmo ConversorViajes
  Definir km, millas Como Real
  Escribir "¿Cuántos kilómetros vas a recorrer?"
  Leer km
  // Aplicamos la fórmula y redondeamos al entero más cercano
  millas <- REDONDEAR(km * 0.62)
  Escribir "Su destino está a unas ", millas, " millas de distancia."
FinAlgoritmo
```

¿Sabías que...?

En la mayoría de lenguajes de programación como Python, C o Java, el redondeo es una función nativa. En Python se usa la función `round()`:

```
km = float(input("Kilómetros a recorrer: "))
millas_redondeadas = round(km * 0.62)
print(f"Su destino está a {millas_redondeadas} millas.")
```

Este documento se publica bajo licencia

 Creative Commons Atribución 4.0 Internacional (CC BY 4.0)