



Prática 3: Conversor de Unidades

Manipulação de tipos de dados matemáticos e arredondamento
Curso de Programação em Snap! e Pseudocódigo

Os Tipos de Dados e a Precisão

Os computadores são excelentes calculadoras, mas ao realizar divisões ou multiplicar por frações, costumam gerar números com muitas casas decimais (ex: 15.6666667). Para que os nossos programas sejam profissionais e fáceis de ler, devemos aprender a **arredondar** esses valores.

Objetivos desta prática

- Manipular tipos de dados numéricos (inteiros e decimais/reais).
- Entender o conceito de *constante de conversão* (fatores multiplicadores).
- Aplicar a função de **arredondamento** para melhorar a interface de utilizador.
- Aninhar operações matemáticas complexas.

Passo a Passo: Snap! vs Pseudocódigo

1. A Constante de Conversão

Para passar de uma unidade para outra, multiplicamos por um fator. Por exemplo, para Euros a Dólares (1.08):

- **No Snap!:** Usamos o bloco de multiplicação: `(euros * 1.08)`.
- **Em Pseudocódigo:** `dolares <- euros * 1.08`

2. Manipulação: O Arredondamento

Para evitar números como 10.799999:

- **No Snap!:** Na categoria **Operadores**, usamos o bloco `arredondar ()`. Deves introduzir toda a operação matemática dentro deste bloco.
- **Em Pseudocódigo:** Usamos a função interna `ARREDONDAR()` ou `ARRED()`.
- *Exemplo:* `dolares_limpos <- ARREDONDAR(dolares)`

3. Formatação da saída

O utilizador precisa de entender o dado final combinando texto e variáveis:

- **No Snap!:** Usamos `juntar [0 valor é] [dolares_limpos]`.
- **Em Pseudocódigo:** Escrever "0 valor é ", `dolares_limpos`

O DESAFIO: O Calculador de Viagens (Km a Milhas)

Cria uma ferramenta para um turista. O teu programa deve converter Quilómetros em Milhas e arredondar o resultado para que não tenha decimais. (*Fórmula: 1 km = 0.62 milhas*).

PARTE A: No Snap!

1. Pergunta a distância em **Quilómetros** e guarda-a numa variável `km`.
2. Cria uma variável `milhas`.
3. Fixa `milhas` ao resultado de **arredondar** a multiplicação de `km` por 0.62.
4. Mostra uma mensagem final juntando texto e o resultado arredondado.

PARTE B: Em Pseudocódigo Escreve o algoritmo correspondente no teu caderno:


```
Algoritmo ConversorViagens
  Definir km, milhas Como Real
  Escrever "Quantos quilómetros vais percorrer?"
  Ler km
  // Aplicamos a fórmula e arredondamos ao inteiro mais próximo
  milhas <- ARREDONDAR(km * 0.62)
  Escrever "O seu destino está a cerca de ", milhas, " milhas de distância."
FinAlgoritmo
```

Sabias que...?

Na maioria das linguagens de programação como Python, C ou Java, o arredondamento é uma função nativa. Em Python usa-se a função `round()`:

```
km = float(input("Quilómetros a percorrer: "))
milhas_arredondadas = round(km * 0.62)
print(f"O seu destino está a {milhas_arredondadas} milhas.")
```

Este documento é publicado sob licença

 Creative Commons Atribuição 4.0 Internacional (CC BY 4.0)