

# Prática 2 Introdução à linguagem Java

## Objetivos

- Familiarização com a linguagem de programação Java
- Resolução de exercícios simples usando Java

## Tópicos

- Tipos primitivos
- Declaração de variáveis
- Entrada/saída de dados
- Ciclos (*for*, *while*, *do .. while*)

**Nota:** Este guião assume que já criou um projeto e um pacote (*package*) para esta aula, conforme indicado no guião 1.

### Exercício 2.1

Escreva um programa que lê um valor real que representa uma distância em quilómetros, converte-o para milhas e imprime o resultado. A fórmula de conversão é a seguinte:  $\text{Milhas} = \text{Quilómetros} / 1.609$ .

### Exercício 2.2

Escreva um programa que lê um valor real que especifica temperatura em graus Celsius, converte-o para graus Fahrenheit e imprime o resultado. A fórmula de conversão de graus Celsius (C) para graus Fahrenheit (F) é a seguinte:  $F = 1.8 * C + 32$ .

### Exercício 2.3

Escreva um programa que calcule a energia necessária para aquecer a água desde uma temperatura inicial até uma temperatura final. O programa deve ler a quantidade de água M (em quilogramas), e as temperaturas inicial e final da água (em graus Celsius). A fórmula para calcular a energia Q (em Joules) é dada por:

$$Q = M * (\text{finalTemperature} - \text{initialTemperature}) * 4184.$$

### Exercício 2.4

Um fundo de investimento fornece uma taxa de juros mensal fixa, que acumula com o saldo anterior do investimento (juros rendem juros). Escreva um programa em Java que peça ao utilizador o montante investido e a taxa de juro mensal. Indique o valor total ao final de 3 meses. (Por exemplo, para um investimento de 5000 euros e uma taxa de 1% o montante ao fim de 3 meses será de 5151.505 euros).

## Exercício 2.5

Pretende-se calcular a velocidade média de transporte do ponto A para o ponto B sendo a viagem feita em duas etapas, cujo comprimento pode ser diferente. Escreva um programa que solicite a inserção da velocidade ( $v_1$ ) e distância ( $d_1$ ) percorrida no primeiro trajeto, bem como a velocidade ( $v_2$ ) e a distância ( $d_2$ ) percorrida no segundo trajeto. Com base nestes valores calcule e imprime a velocidade média final. *Nota: deve validar que os valores inseridos são válidos (positivos).*

## Exercício 2.6

Escreva um programa que dado um tempo em segundos lido do teclado, mostre na consola o tempo com o formato hh:mm:ss. *Sugestão: para calcular o resto da divisão inteira existe o operador %.*

## Exercício 2.7

Escreva um programa em Java que represente dois pontos,  $p_1$  e  $p_2$ , pedindo que o utilizador introduza as suas coordenadas reais  $x$  e  $y$ . Calcule e imprima a distância entre os pontos. *Sugestão: para calcular a raiz quadrada use a função `Math.sqrt`.*

## Exercício 2.8

Dado um triângulo retângulo de catetos  $A$  e  $B$  e hipotenusa  $C$ , escreva um programa que leia o valor dos catetos e determine o valor da hipotenusa, bem como o valor do ângulo (em graus) entre o lado  $A$  e a hipotenusa. *Nota: deve validar que os valores inseridos são válidos (positivos).*

## Exercício 2.9

Escreva um programa que imprima uma contagem decrescente (até 0) a partir de um valor positivo  $N$ , introduzido pelo utilizador através do teclado. Escreva o output numa única linha, separando os números por vírgulas. *Extra: introduza uma quebra de linha sempre que escreve um número múltiplo de 10.*

## Exercício 2.10

Escreva um programa que leia um conjunto de números reais do teclado, terminado por um valor igual ao primeiro que foi introduzido. No fim, indique o valor máximo, o valor mínimo, a média e o total de números lidos.

## Exercício 2.11 (extra)

Crie funções (métodos estáticos) para introdução de dados pelo utilizador, com validação. Por exemplo, crie uma função que peça ao utilizador um valor inteiro dentro de um intervalo e que valide o valor introduzido. Adapte o código dos exercícios anteriores para usar estas funções.

Durante as semanas seguintes complete com novas funções e organize-as numa classe (p.ex. `utils.UserInput`).