

Prática 4 Classes

Objetivos

- Introdução às classes
- Decomposição de complexidade através de métodos/funções

Tópicos

- Classes e objetos
- Métodos e atributos

Exercício 4.1

Implemente classes que permitam modelar as seguintes formas geométricas:

- Círculo, caracterizado pelo seu *raio*;
- Triângulo, caracterizado pela dimensão dos seus lados (*lado1*, *lado2*, *lado3*);
- Retângulo, caracterizado por *comprimento* e *altura*.

Garanta ainda as seguintes especificações:

- a) crie classes que representem cada uma das figuras geométricas, implementando construtores e métodos adequados para cada classe.
- b) adicione todos os métodos especiais importantes (`toString()`, `equals()`, `get...()`, `set...()`, ...);
- c) nos construtores e métodos modificadores (`set...`), verifique pré-condições adequadas: raio e lados têm de ser valores positivos e lados do triângulo têm de satisfazer a desigualdade triangular;
- d) implemente um método para calcular a área de cada tipo de figura (para triângulo, ver fórmula de Heron);
- e) implemente um método para calcular o perímetro de cada tipo de figura;
- f) implemente um programa que teste todas as classes e métodos criados; o programa deve criar um conjunto de figuras, especificadas pelo utilizador através de um menu, listá-las (o método `toString()` deve mostrar o tipo e características da figura), e comparar os pares de figuras do mesmo tipo (método `equals()`).

Exercício 4.2

Analise o código fornecido no ficheiro `CashRegisterDemo.java`. Complete a classe *CashRegister* de forma a guardar um vetor de produtos e apresentar no ecrã o conteúdo no seguinte formato:

Product	Price	Quantity	Total
Book	9.99	3	29.97
Pen	1.99	10	19.90
Headphones	29.99	2	59.98
Notebook	19.99	5	99.95
Phone case	5.99	1	5.99

Total value: 215.79

Exercício 4.3

Analise o código no ficheiro SimpleCarDemo.java. Este programa regista informação sobre carros (classe *Car*) e regista as viagens efetuadas por cada carro, acumulando a quilometragem total do carro. Cada carro é representado pelos seguintes atributos: marca, modelo, ano de matrícula, quilometragem atual.

Complete a classe *Car* e o método *listCars*, de modo a obter o seguinte resultado no ecrã:

Carros registados:

Renault Megane Sport Tourer, 2015, kms: 35356

Toyota Camry, 2010, kms: 32456

Mercedes Vito, 2008, kms: 273891

Carro 1 viajou 773 quilómetros.

Carro 1 viajou 374 quilómetros.

Carro 1 viajou 585 quilómetros.

Carro 0 viajou 875 quilómetros.

Carro 1 viajou 692 quilómetros.

Carro 2 viajou 681 quilómetros.

Carro 2 viajou 224 quilómetros.

Carro 0 viajou 100 quilómetros.

Carro 2 viajou 95 quilómetros.

Carro 2 viajou 516 quilómetros.

Carros registados:

Renault Megane Sport Tourer, 2015, kms: 36331

Toyota Camry, 2010, kms: 34880

Mercedes Vito, 2008, kms: 275407

Exercício 4.4

Estenda o programa anterior, usando agora como base o código no ficheiro CarDemo.java.

O programa começa por criar um vetor para registar até 10 carros, sendo os carros registados sequencialmente desde a posição 0 do vetor. Os dados de cada carro devem ser introduzidos pelo utilizador, numa única linha e de acordo com o formato indicado abaixo, até ser introduzida uma linha em branco:

Toyota Camry 2010 74500

Os dados de entrada (marca, modelo, ano, quilometragem atual) devem ser verificados de acordo com as seguintes regras: marca é composta por uma única palavra, modelo é composto por uma ou mais palavras, ano é um número inteiro positivo composto por 4 algarismos, quilometragem é um número inteiro positivo. *Sugestão: crie uma função para esta verificação; pode, por exemplo, usar expressões regulares.*

Depois de registados os carros, o programa apresenta uma listagem dos carros e depois pede ao utilizador para registar viagens associadas a um dos carros existentes usando como formato de entrada “carro:distância”. No final, deve ser apresentada novamente a listagem dos carros.

Implemente os métodos no código fornecido, tentando reproduzir o seguinte resultado:

Insira dados do carro (marca modelo ano quilómetros): Renault Megane Sport Tourer 2015 35356

Insira dados do carro (marca modelo ano quilómetros): Toyota 2010 32456

Dados mal formatados. Tente novamente.

Insira dados do carro (marca modelo ano quilómetros): Toyota Camry 2010 32456

Insira dados do carro (marca modelo ano quilómetros):

Carros registados:

Renault Megane Sport Tourer, 2015, kms: 35356

Toyota Camry, 2010, kms: 32456

Registe uma viagem no formato "carro:distância": -1:4sdf

Dados mal formatados. Tente novamente.

Registe uma viagem no formato "carro:distância": asdc

Dados mal formatados. Tente novamente.

Registe uma viagem no formato "carro:distância": 1:673

Registe uma viagem no formato "carro:distância": 1:234

Registe uma viagem no formato "carro:distância": 0:96

Registe uma viagem no formato "carro:distância": 0:478

Registe uma viagem no formato "carro:distância": 1:123

Registe uma viagem no formato "carro:distância":

Carros registados:

Renault Megane Sport Tourer, 2015, kms: 35930

Toyota Camry, 2010, kms: 33486