

<p>Sumário</p> <ul style="list-style-type: none"> • Interfaces <p>Documentação complementar Java:</p> <ul style="list-style-type: none"> • Interfaces

Nota: Gere o JavaDoc para o(s) projeto(s) utilizado(s) na resolução desta ficha de trabalho.

Exercício 5

Partindo da resolução dos exercícios anteriores, e tal como descrito pela Figura 1, pretende-se agora que crie o código Java necessário para:

1. Armazenar informação sobre um Hipermercado que, apesar de ser muito similar a um Supermercado, apresenta as seguintes diferenças:
 - a) Permite aos seus clientes a obtenção de crédito, sendo que a prestação mensal a suportar pelos mesmos pode ser calculada da seguinte forma:

$$< \text{taxa de juro mensal} > = (1 + < \text{taxa de juro anual} >^{\frac{1}{12}}) - 1$$

$$< \text{prestação mensal} > = \frac{(< \text{taxa de juro mensal} > \times < \text{valor do empréstimo} >)}{1 - ((1 + < \text{taxa de juro mensal} >)^{-< n^{\circ} \text{ de meses} >})}$$

- b) Oferece aos seus clientes um determinado número de pontos de fidelização, que estes podem utilizar para obtenção de brindes promocionais;
 - c) Contudo, o método de cálculo do número de pontos atribuídos pelo valor de uma determinada compra varia de acordo com a política comercial de cada Hipermercado;
2. Criar uma cadeia de hipermercados `JumbHipermarket` na qual os pontos de fidelização são calculados da seguinte forma:
 - a) De 2ª a 6ª feira, por cada 10€ de compras são atribuídos 3 (três) pontos;
 - b) Ao fim-de-semana, por cada 10€ de compras é atribuído 1 (um) ponto;
 3. Criar uma cadeia de hipermercados `ContinentHipermarket` na qual os pontos de fidelização são calculados da seguinte forma:
 - a) Até ao dia 20 (inclusive) de cada mês, por cada 5€ de compras é atribuído 1 (um) ponto;
 - b) A partir do dia 21 (inclusive) de cada mês, por cada 20€ de compras são atribuídos 3 (três) pontos.

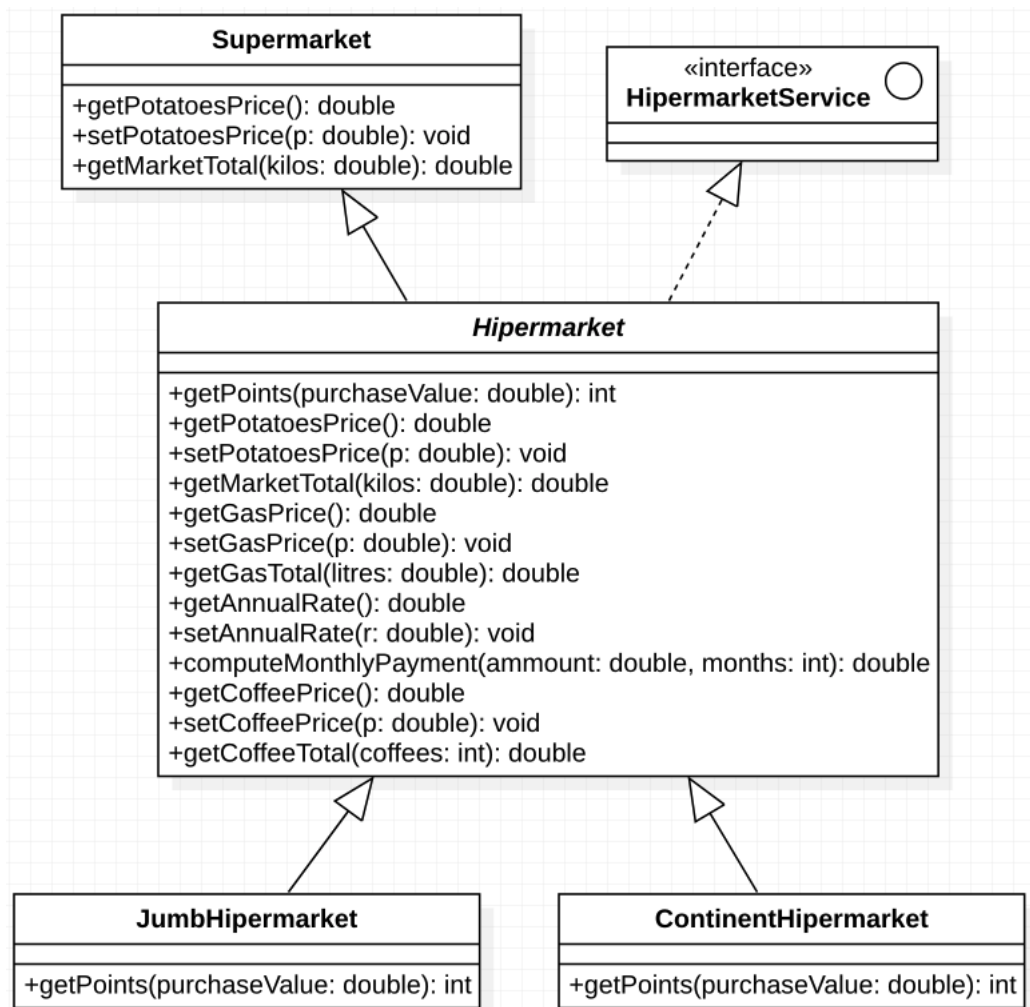


Figura 1 - Diagrama (parcial) de classes

Exercício 6

Partindo da resolução dos exercícios anteriores pretende-se agora que crie o código Java necessário para:

1. Alterar a classe `Hipermarket` para que esta permita saber se, na data actual e para uma compra de um valor dado, um Hipermercado atribui mais pontos do que outro;
2. Criar um método que, recebendo como parâmetro um conjunto de Bombas de Gasolina, devolva aquela em que a gasolina é mais barata (em caso de empate deve ser devolvida a que apareça primeiro no conjunto);
3. Criar um método que, recebendo como parâmetro um conjunto de Hipermercados, devolva aquele em que a compra de um cabaz (composto por q quilos de batatas, l litros de gasolina e c cafés) fica mais barato (em caso de empate deve ser devolvida a que apareça primeiro no conjunto).