



# UPskill - JAVA

Linguagem de Programação Java

## EXERCÍCIO CRÉDITOS BANCÁRIOS

Uma instituição bancária regista informação sobre créditos bancários para poder calcular o valor a receber por cada um desses créditos bancários e os respetivos juros até ao final do contrato. Para auxiliar à informatização deste processo, pretende-se que implemente em JAVA um projeto com as classes necessárias para representar esses créditos bancários.

Para simplificação, este trabalho prático assume algumas particularidades que não se verificam nos créditos bancários praticados pelas instituições bancárias, nomeadamente, não são considerados seguros, comissões e outras taxas que as instituições bancárias poderiam cobrar.

Um crédito bancário consiste num empréstimo de um determinado montante monetário a um cliente. O valor total a pagar pelo cliente por esse empréstimo, corresponde a esse montante acrescido de juros aplicados a uma taxa de juro anual fixa (que em alguns créditos bancários corresponde à soma de um “spread” e de uma taxa Euribor).

Mensalmente o cliente amortiza parte do financiamento e paga os juros sobre o capital em dívida no momento. Assuma que o montante do crédito bancário é amortizado com o mesmo valor durante toda a vigência do financiamento. Os juros a pagar mensalmente pelo cliente, recaem sobre o capital em dívida no mês anterior. O capital em dívida num determinado mês é resultado da diferença do capital em dívida no mês anterior menos a amortização de capital.

Qualquer crédito bancário é caracterizado pelo nome do cliente (para simplificação, somente um titular), profissão, montante (em euros) e prazo de financiamento (número de meses).

Um crédito bancário pode ser de dois tipos: crédito à habitação ou crédito ao consumo. O crédito ao consumo pode ainda ser um crédito automóvel ou um crédito à educação.

De seguida, são apresentadas algumas particularidades para cada um destes tipos de crédito bancário.

Crédito à Habitação: caracterizado por um “spread” acordado com o cliente e por uma taxa Euribor a 12 meses que é comum a todos os clientes (a soma do “spread” e da taxa Euribor a 12 meses corresponde à taxa de juro anual). Assuma o valor da taxa Euribor a 12 meses igual a 0,1%<sup>1</sup>.

Crédito ao Automóvel: caracterizado por uma taxa de juro anual de 6%<sup>1</sup> que é comum a todos os clientes e por um desconto a obter sobre o montante total a pagar em cada mês do financiamento se o prazo de financiamento for inferior ou igual a 24<sup>1</sup> meses. Considere, para tal, uma taxa de desconto de 1%<sup>1</sup>.

---

<sup>1</sup> Estes valores podem ser atualizados.

Crédito à Educação: caracterizado por uma taxa de juro anual de 2%<sup>1</sup> que é comum a todos os clientes e por um período de carência (número de meses em que o cliente só paga juros sobre o montante total do empréstimo) acordado entre a instituição bancária e o cliente.

Todo o código produzido deve ter sempre em consideração os principais princípios da programação orientada por objetos: abstração, encapsulamento, herança e polimorfismo.

As classes criadas (com exceção da classe principal) devem obedecer ao seguinte conjunto de especificações:

- implementação de construtores (pelo menos o construtor completo e o construtor sem parâmetros);
- implementação de métodos que sejam relevantes para aceder e modificar o valor dos atributos;
- reescrita do método *toString*;
- inclusão de comentários *Javadoc*.

Adicionalmente devem ser implementados, nas respetivas classes, os métodos `calcularMontanteARreceberPorCadaCredito()` e `calcularMontanteTotalJuros()` para o cálculo do montante e dos juros a receber até ao final de cada crédito bancário realizado pela instituição bancária. Devem também ser criados testes unitários para os cálculos referidos anteriormente, apenas para os créditos à habitação e para os créditos à educação.

Na classe principal, o código a implementar deve preencher os seguintes requisitos:

- a) Criação do seguinte conjunto de instâncias:
  - 2 créditos à habitação;
  - 2 créditos automóvel;
  - 2 créditos à educação;
- b) Criação de um contentor do tipo *array* e armazenamento no mesmo das instâncias criadas;
- c) Criação de listagens separadas, sobre o contentor, para:
  - obter o nome do cliente e o valor que a instituição bancária irá receber até ao final de cada contrato de crédito ao consumo realizado;
  - obter o nome do cliente e o valor dos juros (para além do montante do empréstimo) que o cliente terá de pagar até ao final de cada contrato de crédito bancário realizado;
- d) Apresentação das quantidades de instâncias de créditos à habitação e de créditos ao consumo criadas (em separado), sem percorrer o contentor;

## Linguagem de Programação Java

Herança e Polimorfismo

- e) Cálculo e apresentação do valor total e dos respetivos juros que a instituição bancária irá receber por todos os créditos bancários realizados, percorrendo apenas uma vez o contentor.

### Nota

Para ajudar à compreensão dos cálculos que são necessários realizar para obter o montante a receber pela instituição bancária até ao final do contrato relativo a um crédito bancário, é disponibilizada uma folha de cálculo com os cálculos associados a um empréstimo bancário do tipo “Crédito à Habitação”, com um montante de 120000 euros, um “spread” de 1%, uma taxa Euribor a 12 meses de 0.1% e um prazo de financiamento de 240 meses (20 anos) e um empréstimo bancário do tipo “Crédito à Educação”, com um montante de 18000 euros, uma taxa de juro anual de 2%, um prazo de financiamento de 60 meses (5 anos) e um período de carência de 24 meses.