



**INSTITUTO  
FEDERAL**  
**PARANÁ**  
Campus Telêmaco  
Borba



**MINISTÉRIO DA  
EDUCAÇÃO**

---

## Tecnologia em Análise e Desenvolvimento de Sistemas

**Disciplina:** Programação Orientada a Objetos

Prof. Me. Gregory Vinícius Conor Figueiredo

# Projeto Final

**Descrição:** Este projeto final tem o intuito de construir um diagrama de classes abordando os conceitos vistos na disciplina e realizar sua implementação em Java. Para facilitar a compreensão a atividade será proposta em partes.

**PARTE 1:** Construa um diagrama que represente o seguinte conjunto de classes e seus relacionamentos:

- **Produto:** Classe que contém dois atributos: preço e quantidade. Possui métodos para encapsulamento de seus atributos (*getters e setters*) e um método *imprimir()*, que mostra os valores destes atributos.
- **Perecível:** Subclasse de Produto. Possui, além dos atributos herdados, o atributo data\_validade. Possui métodos para encapsulamento deste atributo.
- **Pedido:** Classe associada à classe Produto. Contém um único atributo: produtos, que será definido como um *array* de produtos (objetos da classe Produto). Possui três métodos:
  - *iniciar()*, que deve simplesmente iniciar o vetor de produtos.
  - *adicionarProdutos(\$produto)*, que deve receber como parâmetro um objeto da classe Produto e adicioná-lo ao *array* de produtos.
  - *calcularValor()*: Deve varrer o vetor de produtos e retornar o a soma total de seus preços.

**PARTE 1.1:** Realize a implementação do diagrama que desenvolveu na PARTE 1 utilizando a linguagem Java.

**PARTE 2:** Acrescente ao diagrama construído na PARTE 1:

- **TipoPagamento:** Interface que declara dois métodos: *getDiasFaturamento()* e *getPorcentagemFinanceira()*.
- **CartaoDeCredito:** Classe que implementa a interface TipoPagamento. Seu método *getDiasFaturamento* deve retornar o valor 1 e seu método *getPorcentagemFinanceira* deve retornar o valor 0.1.
- **Boleto:** Outra classe que implementa a interface TipoPagamento. Seu método *getDiasFaturamento* deve retornar o valor 3 e seu método *getPorcentagemFinanceira* deve retornar o valor 0.05.
- **Pagamento:** Classe associada a interface TipoPagamento. Possui dois atributos: pagamento, atributo responsável por realizar a associação com a interface

TipoPagamento; e valor. A classe deve possuir métodos para encapsular seus atributos.

**PARTE 2.1:** Implemente o diagrama de classes criado na PARTE 2 em Java. *[Obs.: o método `setPagamento`, da classe `Pagamento`, deve receber como parâmetro um objeto de `TipoPagamento` (Cartão de Crédito ou Boletto). Não deve ser realizada nenhuma implementação específica com relação a isso na classe, mas isso deve ser levado em consideração no momento de criar os objetos, quando for testar]*

**PARTE 3:** Complete o diagrama com as seguintes classes:

- **Cliente:** Classe que possui um único atributo: id. Possui métodos para encapsulamento do atributo.
- **Compra:** Classe associada às classes Pedido (definida na PARTE 1) e Pagamento (definida da PARTE 2) por meio de relacionamentos de composição, onde pedido e pagamento fazem parte da compra. Possui os atributos: ped (objeto de Pedido), valorTotal e pag (objeto de Pagamento). Possui métodos para encapsulamento de seus atributos.
- **FichaCliente:** Classe que está associada às classes Cliente e Compra por meio de composições, onde o cliente e a compra são partes da ficha. Esta classe possui dois atributos: cli (que será um objeto da classe Cliente) e compras (um array de objetos da classe Compra). Possui um método construtor, que inicializa os atributos e instancia os objetos; e métodos para encapsular os atributos.

**PARTE 3.1:** Implemente o restante do diagrama, desenvolvido na PARTE 3, em Java.

**PARTE 4 [DESAFIO]:** Crie uma classe de teste e nela teste a aplicação desenvolvida pelas instruções das Partes 1, 2 e 3. Seu desafio é criar um objeto de ficha de cliente com ao menos uma Compra. Para fazer isso você precisará instanciar e manipular vários objetos, de praticamente todas as classes.

**Obs.:** Você pode adicionar funcionalidades às classes conforme julgar necessário para realizar os testes.