

**Fazendo uso da linguagem JAVA, implemente os seguintes exercícios:**

**Exercício 1**

- Implemente uma classe *Produto*
  - atributos:
    - código: código do produto (private int)
    - descrição: descrição do produto (private String)
    - valor de compra (private double)
    - custo: custo para armazenar e/ou beneficiar o produto (private double)
    - margem de lucro: margem de lucro desejada, utilizado para calcular o preço de venda do produto, (private double)
    - quantidade em estoque: quantidade de unidades do produto em estoque (private int)
  - métodos:
    - void compra(int quant){qtd\_estoque = qtd\_estoque + quant;}
    - void venda(int quant){qtd\_estoque = qtd\_estoque - quant;}
    - double calculaPrecoVenda( ) {return vl\_compra + custo + margem\_lucro\*(vl\_compra + custo) }
    - crie métodos getAtributo e setAtributo para os atributos: valor de compra, custo e margem de lucro
    - crie um construtor para inicializar o valor dos atributos: código e descrição
    - crie um construtor para inicializar o valor dos atributos: código, descrição, valor de compra, custo e margem de lucro
    - crie um construtor para inicializar o valor dos atributos: código, descrição, valor de compra, custo, margem de lucro e quantidade em estoque

**Exercício 2**

- Crie a classe "RevendaComArray" a qual deverá conter:
  - Um atributo "produtos" (do tipo array de Produto), privado (utilize a classe Produto criada no exercício anterior);
  - Um atributo privado para servir de índice.
  - Um construtor que recebe como argumento a quantidade de produtos e inicializa o atributo "produtos" com um array de produtos com o tamanho especificado;
  - Um método "inserirProduto" que recebe como parâmetro um Produto e o insere no array;
  - Um método "comprar" que recebe como parâmetro o código de um Produto e quantidade de unidades deste produto que foi adquirida pela revenda, percorre o array procurando o Produto e incrementa a quantidade em estoque deste produto, caso ele não exista, o método deve imprimir uma mensagem de erro;
  - Um método "vender" que recebe como parâmetro o código de um Produto e quantidade de unidades deste produto que foi vendida pela revenda, percorre o array procurando o Produto e decrementa a quantidade em estoque deste produto, caso ele não exista, o método deve imprimir uma mensagem de erro;
  - Um método "consultaPrecoVenda" que recebe como parâmetro o código de um Produto, percorre o array procurando o Produto e imprime o valor de venda deste produto, caso ele não exista, o método deve imprimir uma mensagem de erro;
  - Um método "listaPrecos" que imprime o código, a descrição e o valor de venda de todos os produtos;
- Elabore um programa que cria uma Revenda e oferece um menu com operações para criar produto, comprar, vender, consultar preço, listar preço além de consultar e alterar os atributos valor de compra, custo e margem de lucro em um determinado produto;

**Crie um projeto no GITHUB, copie o link do projeto em um arquivo TXT e o submeta no SIPPA, em A01.**

**Prazo de entrega: 6/12/2020**

**Atenção: ao usar o GITHUB, você deve fazer os uploads do seu projeto via comandos push. Não submeta seu trabalho ao GITHUB usando a interface web do site. Trabalhos cujo upload foi feito desta forma serão invalidados.**