

## Classe Produto e Calculo de Produto

---

Criar um projeto -> Aula05

Criar um pacote chamado entity.

Criar uma classe completa de Produto contendo os atributos (código, nome, preço, quantidade, tipo, ativo, desconto), os construtores vazio e cheio, toString, e getters e setters.

```
package entity;

public class Produto {

    private Integer codigo;
    private String nome;
    private Double preco;
    private Integer quantidade;
    private String tipo;
    private Boolean ativo;
    private transient Double desconto=0.;
```

O atributo "ativo" é do tipo Boolean pois ele estamos informando se o produto está ativo ou não, representa a condição dele. O atributo "desconto" é tipado transient pois não será gravado em banco, ele será calculado e dessa forma, precisa ser iniciado com o valor 0.

```
public Produto() {

}

public Produto(Integer codigo, String nome, Double preco,
               Integer quantidade, String tipo, Boolean ativo) {
    super();
    this.codigo = codigo;
    this.nome = nome;
    this.preco = preco;
    this.quantidade = quantidade;
    this.tipo = tipo;
    this.ativo = ativo;
}
```

```
@Override
public String toString() {
    return "Produto [codigo=" + codigo + ", nome=" + nome +
", preco=" + preco + ", quantidade=" + quantidade + ", tipo=" +
tipo + ", ativo=" + ativo + "]";
}

public Integer getCodigo() {
    return codigo;
}
public void setCodigo(Integer codigo) {
    this.codigo = codigo;
}
public String getNome() {
    return nome;
}
public void setNome(String nome) {
    this.nome = nome;
}
public Double getPreco() {
    return preco;
}
public void setPreco(Double preco) {
    this.preco = preco;
}
public Integer getQuantidade() {
    return quantidade;
}
public void setQuantidade(Integer quantidade) {
    this.quantidade = quantidade;
}
public String getTipo() {
    return tipo;
}
public void setTipo(String tipo) {
    this.tipo = tipo;
}
public Boolean isAtivo() {
    return ativo;
}
public void setAtivo(Boolean ativo) {
    this.ativo = ativo;
}
}
```

```
public Boolean getAtivo() {
    return ativo;
}
public Double getDesconto() {
    return desconto;
}
public void setDesconto(Double desconto) {
    this.desconto = desconto;
}
}
```

Criar um pacote chamado control, depois uma classe chamada CalcularProduto para fazermos algum tipo de cálculo relacionado a classe Produto.

```
package control;

import entity.Produto;

public class CalcularProduto {

    public static void calcularDescontoProduto(Produto p) {
        if (p.getTipo().equalsIgnoreCase("celular")) {
            p.setDesconto((p.getPreco() * 10) / 100);
        } else if (p.getTipo().equalsIgnoreCase("computador")) {
            p.setDesconto((p.getPreco() * 20) / 100);
        } else {
            p.setDesconto(0.);
        }
    }
}
```

Criamos um método "static" chamado calcularDescontoProduto, como o nome já diz, para calcular o desconto de um produto. Um método estático pode ser chamado direto pelo nome, sem precisar chamar a classe primeiro para depois chamá-lo.

Criamos uma condição, se o produto for do tipo celular, o método "equalsIgnoreCase" é utilizado para qualquer que seja o tipo da letra (maiúscula ou minúscula), será reconhecido pelo sistema e efetuará o cálculo do desconto.

No setDesconto passar a formula do cálculo do desconto dessa condição.

A outra condição é se o produto for do tipo computador, será efetuado um outro tipo de cálculo.

E se não for nenhuma das duas condições, não haverá desconto.

```
public static void main(String[] args) {  
  
    Produto p = new  
    Produto(10, "arroz", 4., 10, "alimento", true);  
    CalcularProduto.calcularDescontoProduto(p);  
  
    System.out.println("Produto: " + p.getNome());  
    System.out.println("Total :"  
    ( (p.getPreco() - p.getDesconto()) *  
p.getQuantidade()));  
  
    Produto p2 = new  
    Produto(12, "Hiphone", 2000., 1, "celular", true);  
    CalcularProduto.calcularDescontoProduto(p2);  
    System.out.println(p2.getNome());  
    System.out.println("Total :"  
    ( (p2.getPreco() - p2.getDesconto()) *  
p2.getQuantidade() ) );  
    }  
}
```

No teste main, criamos um objeto de produto com os dados. Chamamos a classe de calculo e o método para ser calculado se houver algum tipo de desconto. Pedimos para imprimir no console o nome do produto e o preço final (já com desconto).

Se o método calcularDescontoProduto não fosse estático, o main teria que ser escrito de uma outra forma. Chamaríamos a classe, criaríamos um objeto pra ela. Depois abriríamos espaço de memória no método para trabalharmos com ele.

```
public static void main(String[] args) {  
  
    Produto p = new  
    Produto(10, "arroz", 4., 10, "alimento", true);  
    CalcularProduto cp = new CalcularProduto();
```

```
    cp.calcularDescontoProduto(p);  
}
```

