

# OCP - OpenClosed Principle

O OCP afirma que as classes devem estar "*abertas para extensão, mas fechadas para modificação*".

## Benefícios:

- **Facilidade de Extensão:** Novas funcionalidades podem ser adicionadas sem alterar o código existente.
- **Redução de Bugs:** Minimiza o risco de introduzir erros em funcionalidades já implementadas.

**Exemplo de Má Prática:** Modificar uma classe diretamente para adicionar um novo comportamento.

## OCP - Mau Exemplo

**Problema:** Para adicionar um novo tipo de produto, como "Alimentos", precisaríamos modificar a classe `CalculadoraDeDesconto`, violando o OCP.

```
public class CalculadoraDeDesconto
{
    public double CalcularDesconto(string tipoProduto, double valor)
    {
        if (tipoProduto == "Eletronico")
            return valor * 0.1;
        else if (tipoProduto == "Roupas")
            return valor * 0.2;
        /* modificacao:
        else if (tipoProduto == "Alimentos")
            return valor * 0.05
        */
        else
            return 0
    }
}
```

## OCP - Bom Exemplo

Use o *polimorfismo* para para estender o comportamento sem modificar a classe base:

- **Solução:** Agora, novos tipos de desconto podem ser adicionados criando novas classes, sem modificar a classe `CalculadoraDeDesconto` original.

```
public abstract class Desconto
{
    public abstract double Calcular(double valor);
}

public class DescontoEletronico : Desconto
{
    public override double Calcular(double valor) => valor * 0.1;
}

public class DescontoRoupas : Desconto
{
    public override double Calcular(double valor) => valor * 0.2;
}

public class DescontoAlimentos : Desconto
{
    public override double Calcular(double valor) => valor * 0.05;
}
```