

# Lista 1 - Herança

1- Crie uma classe `Carta` com os atributos `naipe` e `valor` . Implemente um método `mostrarInformacoes()` que exiba no console o naipe e o valor da carta.

Em seguida, crie uma subclasse `CartaEspecial` que herda de `Carta` e adicione um atributo adicional `efeitoEspecial` . Implemente o método `mostrarInformacoes()` na subclasse para exibir o naipe, o valor e o efeito especial da carta.

No final, crie uma instância de cada classe ( `Carta` e `CartaEspecial` ) e exiba suas informações no console.

2- Crie uma classe `Programador` com os seguintes atributos:

- `nome` : Nome do programador.
- `salarioBase` : Salário base do programador.

Implemente os seguintes métodos na classe base:

- `calcularSalario()` : Calcula o salário base do programador.
- `mostrarInfo()` : Exibe as informações do programador, incluindo o nome e o salário base.

Crie duas subclasses `Pleno` e `Senior` que herdam de `Programador` e possuem atributos e métodos específicos:

- **Pleno:**

Atributo : `numeroProjetosConcluidos` : Número de projetos concluídos.

Método: `calcularSalario()` que calcula o salário total como o salário base mais 5% de comissão por projeto concluído.

- **Senior:**

Atributo: `numeroProjetosConcluidos` (número): Número de projetos concluídos.

Método: `calcularSalario()` que calcula o salário total como o salário base mais 7% de comissão por projeto concluído.

Exiba as seguintes informações para cada programador:

- Nome

- Salário base
- Número de projetos concluídos
- Total a receber (salário base mais comissão)

3 - Crie uma classe base `Funcionario` com os seguintes atributos:

- `nome` : Nome do funcionário.
- `salarioBase` : Salário base do funcionário.

Implemente os seguintes métodos na classe base:

- `calcularSalario()` : Retorna o salário base do funcionário.
- `mostrarInfo()` : Exibe as informações do funcionário, incluindo o nome e o salário base.

Crie duas subclasses, `Junior` e `Senior`, que herdam de `Funcionario` e têm atributos e métodos específicos:

- **Junior:**

Atributo : `avaliacaoDesempenho` : Avaliação de desempenho do funcionário ("Bom", "Regular" ou "Insatisfatório").

Método: `calcularSalario()` que calcula o salário total com base no salário base e aplica um bônus de acordo com a avaliação de desempenho:

- "Bom": 10% do salário base.
- "Regular": 5% do salário base.
- "Insatisfatório": Sem bônus.

- **Senior:**

Atributo : `numeroProjetosConcluidos` (número): Número de projetos concluídos.

Método: `calcularSalario()` que calcula o salário total com base no salário base e aplica um bônus de acordo com o número de projetos concluídos:

- Mais de 10 projetos: 20% do salário base.
- Entre 5 e 10 projetos: 10% do salário base.
- Menos de 5 projetos: 5% do salário base.

Exiba as seguintes informações para cada funcionário:

- Nome
- Salário base
- Avaliação de desempenho ou número de projetos concluídos
- Total a receber (salário base mais bônus)