

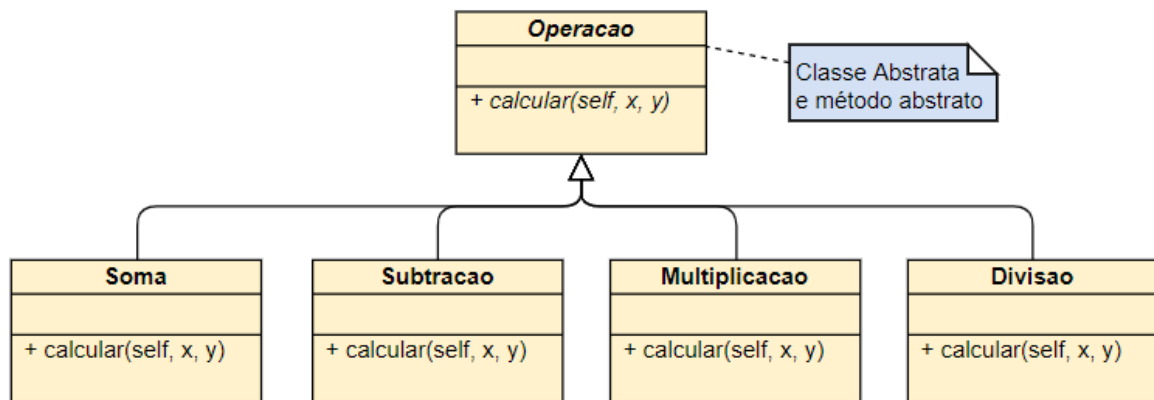
## Exercício 01

Implemente o diagrama abaixo utilizando polimorfismo para calcular as operações matemáticas.

A classe Operacao deve ser abstrata. O método calcular da classe Operação também deve ser abstrato.

As classes herdeiras devem implementar o método calcular. Esse método deve realizar a operação correspondente e retornar o resultado.

Observe que as classes **não possuem atributos, portanto não precisam ter construtores**.



Utilize o trecho de programa para testar suas classes

```
# Programa Principal

soma = Soma()
sub = Subtracao()
div = Divisao()
mult = Multiplicacao()

print(soma.calcular(10, 5))    # 15
print(sub.calcular(10, 5))    # 5
print(div.calcular(10, 5))    # 2.0
print(mult.calcular(10, 5))    # 50
```

---

## Exercício 02

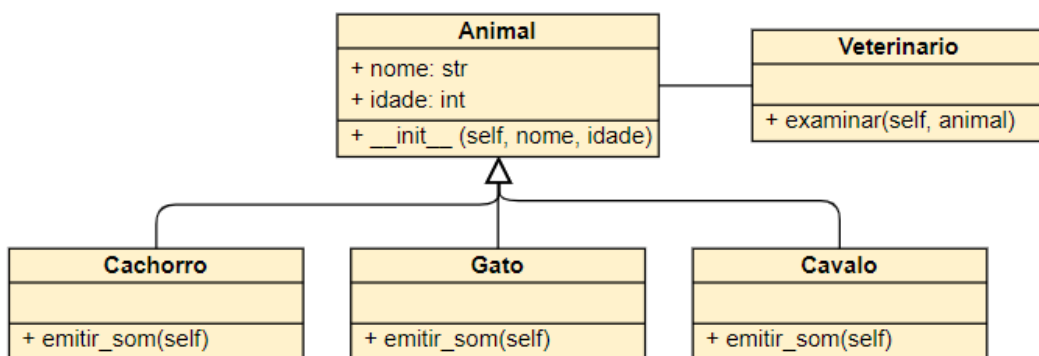
Crie uma hierarquia de classes conforme o diagrama abaixo.

Os atributos em comum devem ficar na classe Animal.

Os métodos `emitir_som` devem imprimir uma mensagem simulando a emissão do som do animal correspondente.

A classe `Veterinario` deve conter o método `examinar()`, que recebe como entrada um objeto que representa um animal e, quando o animal for examinado, esse animal deve emitir o seu som.

No programa principal, crie objetos dos 3 tipos de animais e execute o método que emite o som de cada um.



Veja um exemplo de programa principal

```
dog = Cachorro("Rex", 3)
horse = Cavalo("Horse", 6)
cat = Gato("Tina", 1)

dog.emitir_som()           # exibe o som do cachorro
horse.emitir_som()         # exibe o som do cavalo
cat.emitir_som()           # exibe o som do gato

vet = Veterinario()
vet.examinar(dog)          # exibe o som do cachorro
vet.examinar(horse)        # exibe o som do cavalo
vet.examinar(cat)          # exibe o som do gato
```

---

### Exercício 03

Crie uma hierarquia de classes para representar os diferentes tipos de funcionários de um escritório que tem os seguintes cargos: gerente, assistente e vendedor.

Escreva uma superclasse abstrata chamada **Funcionario** que define o método abstrato **calcular\_salario()**.

Essa classe também deve ter os seguintes atributos: **nome**, **matricula** e **salario\_base**.

A classe Funcionario deverá ser herdada pelas outras classes que são: **Gerente**, **Assistente** e **Vendedor**.

Em cada classe-filha deve-se sobrescrever o método calcular\_salario(). Este método deve calcular e retornar o salário de cada funcionário, da seguinte forma:

- O assistente recebe o salário\_base.
- O gerente recebe duas vezes o salário\_base.
- O vendedor recebe o salário\_base mais uma comissão de 10%.

Implemente um programa principal que cria um objeto de cada tipo (gerente, assistente e vendedor) e os armazena em uma lista

Percorra essa lista e imprima o salário de cada funcionário.

---

### Exercício 04

Um banco trabalha com três tipos de contas:

- conta corrente
- conta corrente especial
- conta poupança

Em todas as contas é necessário armazenar o número da conta, o nome do correntista e o saldo.

Para a conta corrente especial é necessário armazenar também o valor do limite.

As operações possíveis são: depósito, saque e impressão do saldo. Essas operações devem ser definidas numa classe abstrata denominada Conta.

A operação de depósito e saldo são iguais para os três tipos de conta.

A operação de saque só é diferente na conta com limite, pois esta admite que o saldo fique negativo até o limite estabelecido.

Nas demais contas o saque não pode ser realizado se não houver saldo suficiente.

Implemente a hierarquia de classes definida acima.