



PОО com Java - 03



Herança

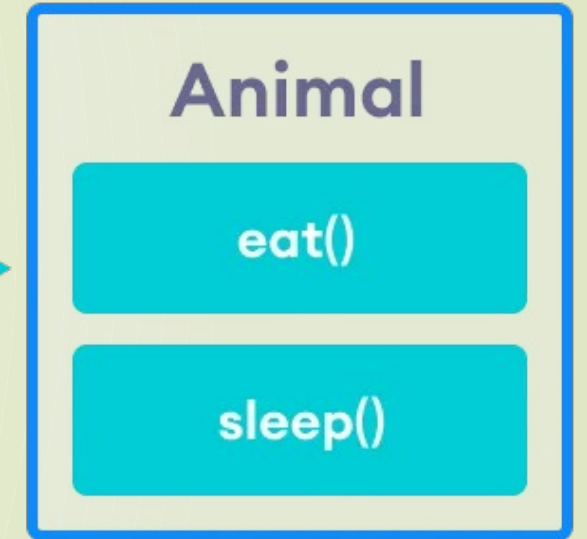


- ❑ Mecanismo que remete à herança biológica.
- ❑ Uma das principais características da OOP.
- ❑ Ocorre quando um objeto adquire propriedades e comportamentos de um objeto pai.
- ❑ Promove o aproveitamento de código e a manutenibilidade.
- ❑ A ideia da herança é permitir novas classes que estendem os métodos e atributos, criando novos ou alterando-os.

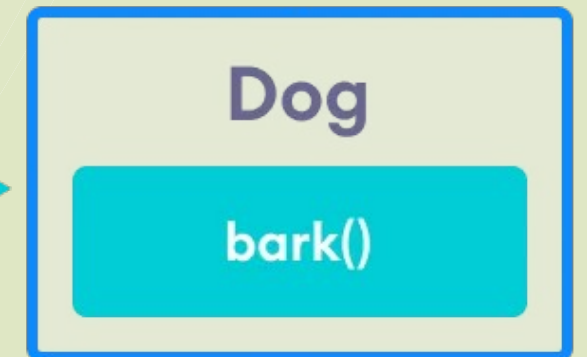
Herança

- ❑ Classe
Modelo genérico
- ❑ Superclasse
Classe de origem
- ❑ Subclasse
Classe que herda

**Base
Class**



**Derived
Class**



Herança

□ Classe

□ Superclasse

□ Subclasse

Base
Class

Animal

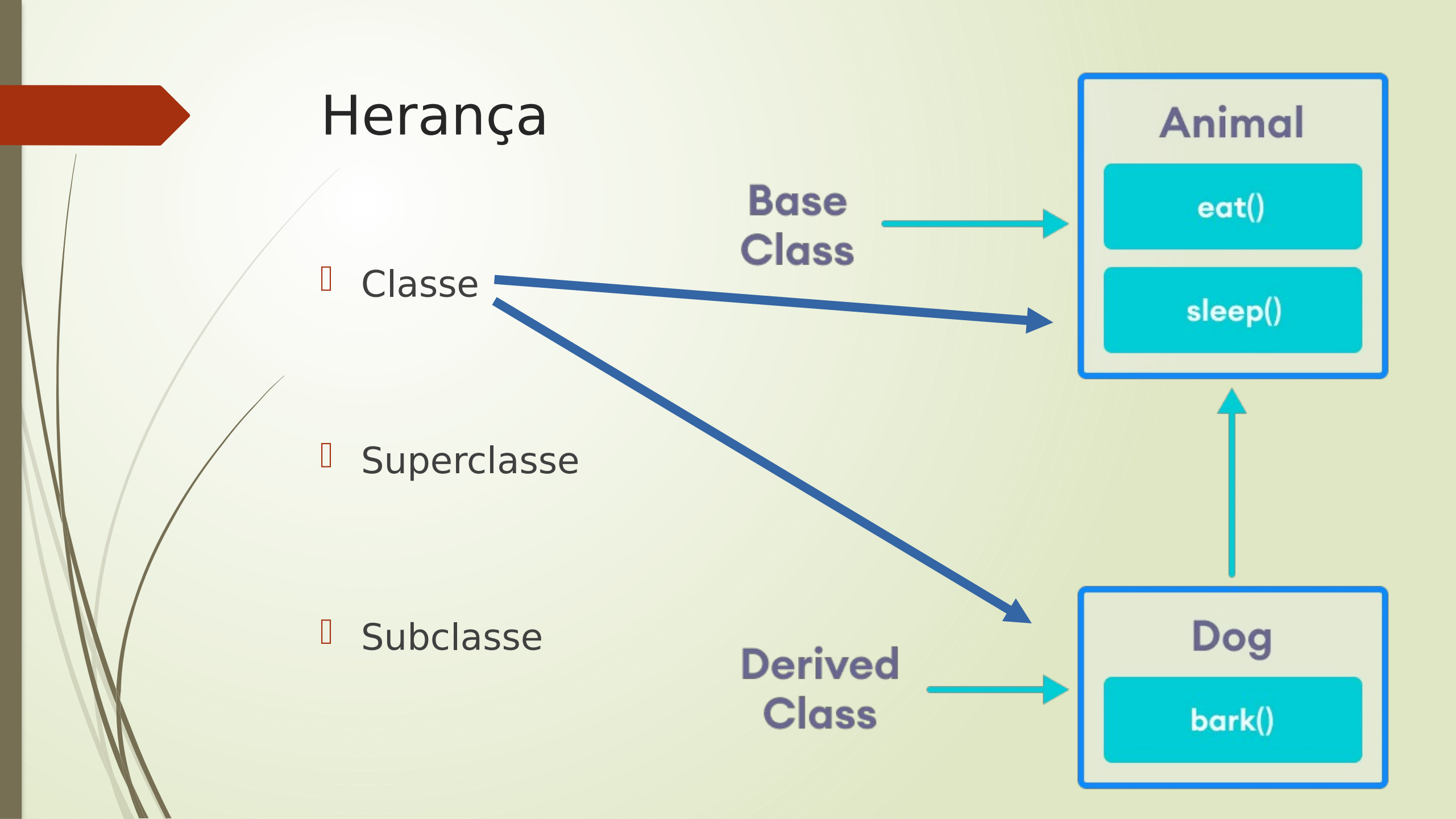
eat()

sleep()

Derived
Class

Dog

bark()



Herança

- Classe

- Superclasse

- Subclasse

**Base
Class**

Animal

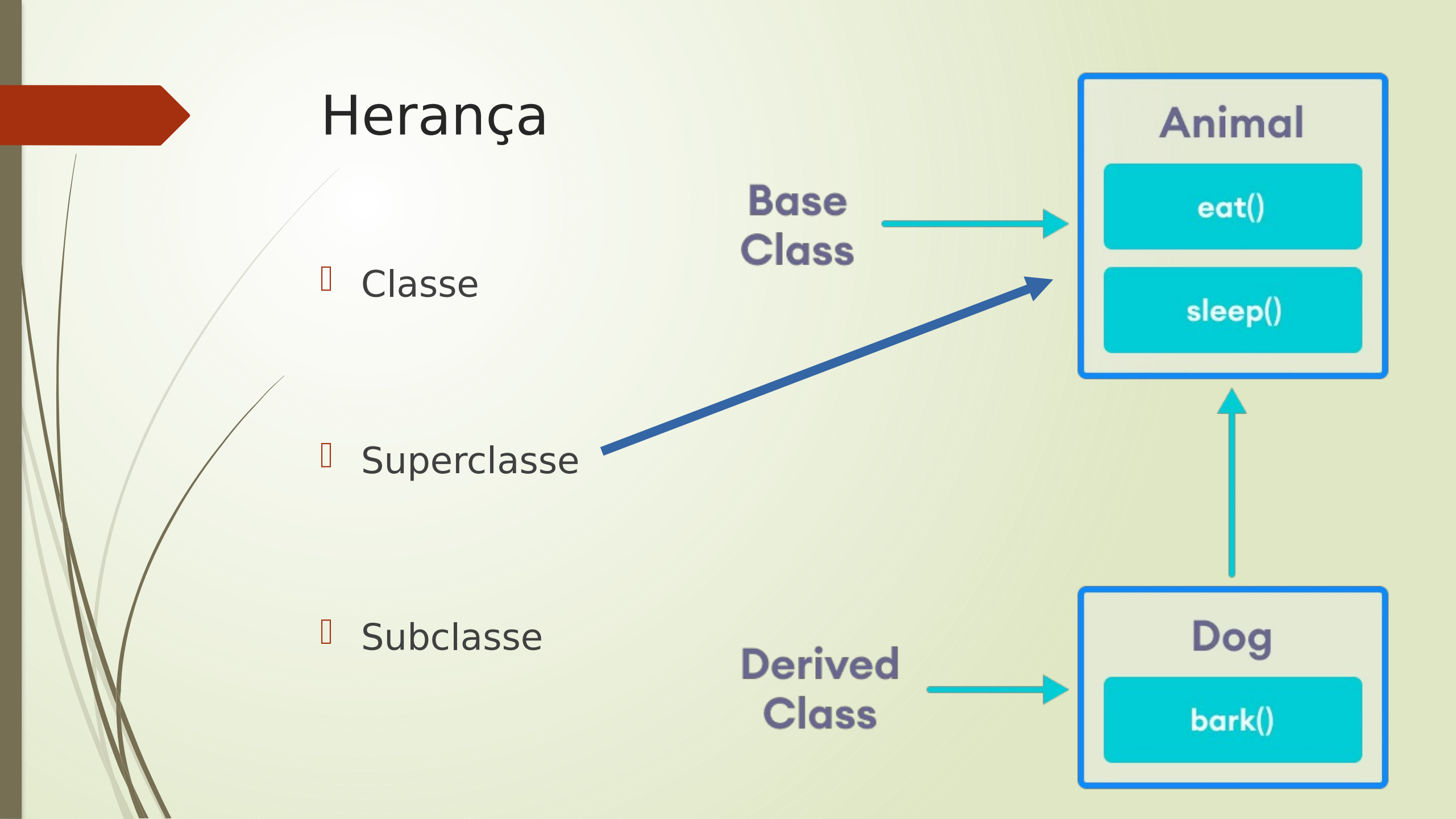
eat()

sleep()

**Derived
Class**

Dog

bark()



Herança

□ Classe

□ Superclasse

□ Subclasse

**Base
Class**

Animal

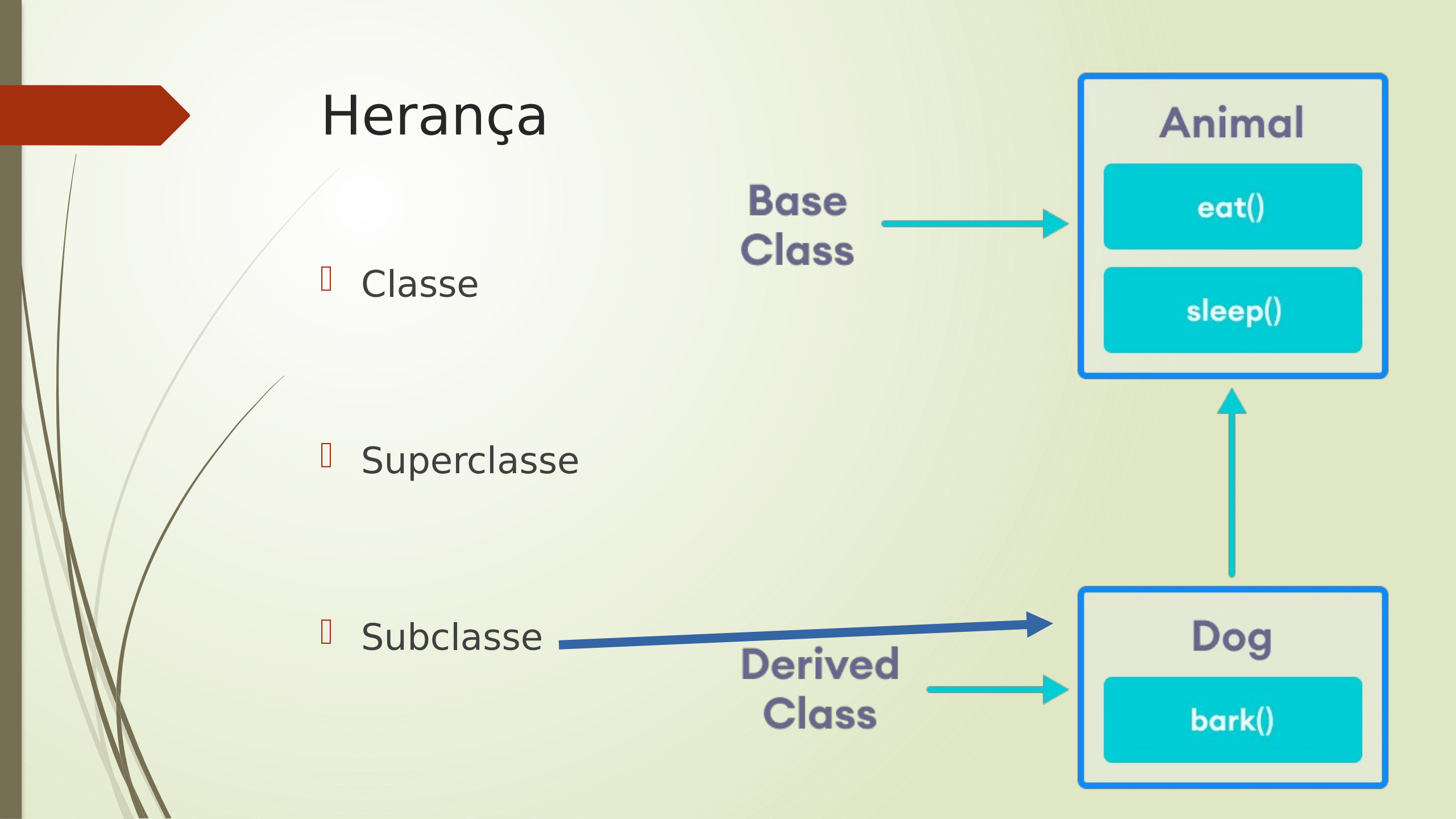
eat()

sleep()

**Derived
Class**

Dog

bark()





Herança

Superclasse

```
class Animal {  
    public Boolean vivo = true;  
    public void Comer(String alimento) {  
        System.out.println("Comendo: " + alimento);  
    }  
}
```



Herança

Superclasse

```
class Cachorro extends Animal {  
    public String nome = "Rex";  
    public void Latir() {  
        System.out.println("Cachorro latindo...");  
    }  
}
```



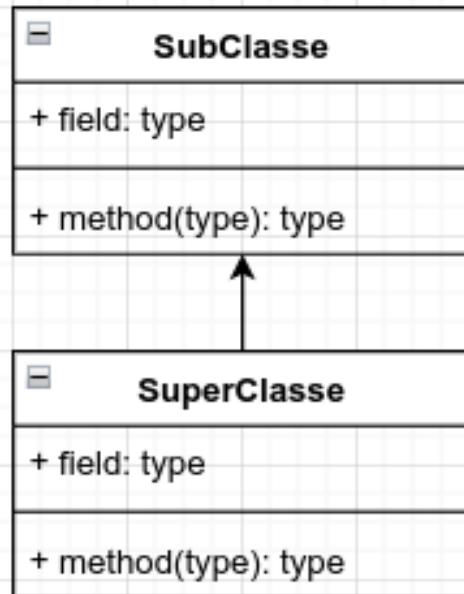

Herança

Superclasse

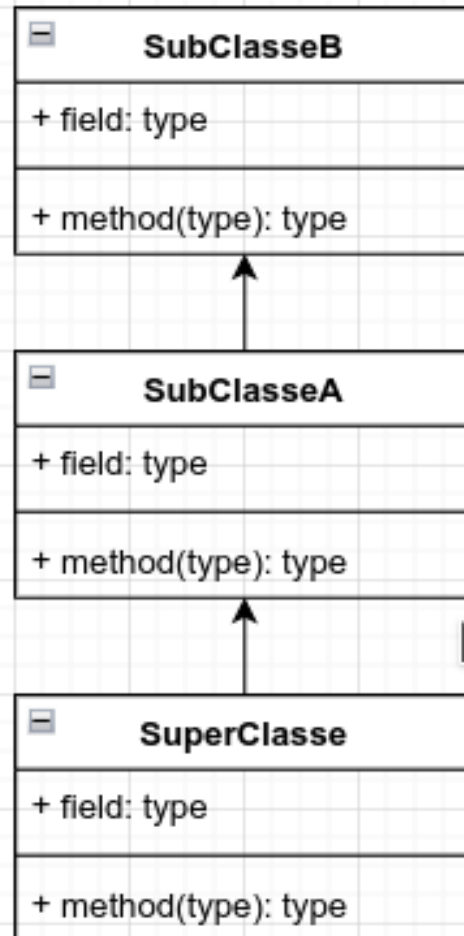
```
class Heranca {  
    public static void main(String[] args) {  
        Animal animalGenerico = new Animal();  
        animalGenerico.Comer("batata");  
        System.out.println("Está vivo? " + animalGenerico.vivo);  
  
        Cachorro cachorro = new Cachorro();  
        System.out.println("Nome do doguinho: " + cachorro.nome);  
        System.out.println("Doguinho vive? " + cachorro.vivo);  
        cachorro.Latir();  
        cachorro.Comer("Ração"); } }
```

Tipos de Herança

Única



Multinível



Hierárquico

