

HERANÇA E POLIMORFISMO

Samuel Cruz



PILARES DA PROGRAMAÇÃO ORIENTADA À OBJETOS



HERANÇA



Herança é o direito de herdar, ou seja, receber características e/ou comportamentos de uma situação anterior.

“Ela herdou ... da mãe”

HERANÇA

O mesmo conceito é utilizado em POO. As classes podem ser mães de outras classes, assim suas filhas herdam todas as características (atributos) e comportamentos (métodos)

Este conceito é bastante utilizado para evitar a repetição de código


HERANÇA

Exemplo:


É possível perceber que existem características e comportamentos em comum dentre as classes citadas



Aluno
- nome
- idade
- sexo
- matr
- curso
+ fazerAniv()
+ cancelarMatr()

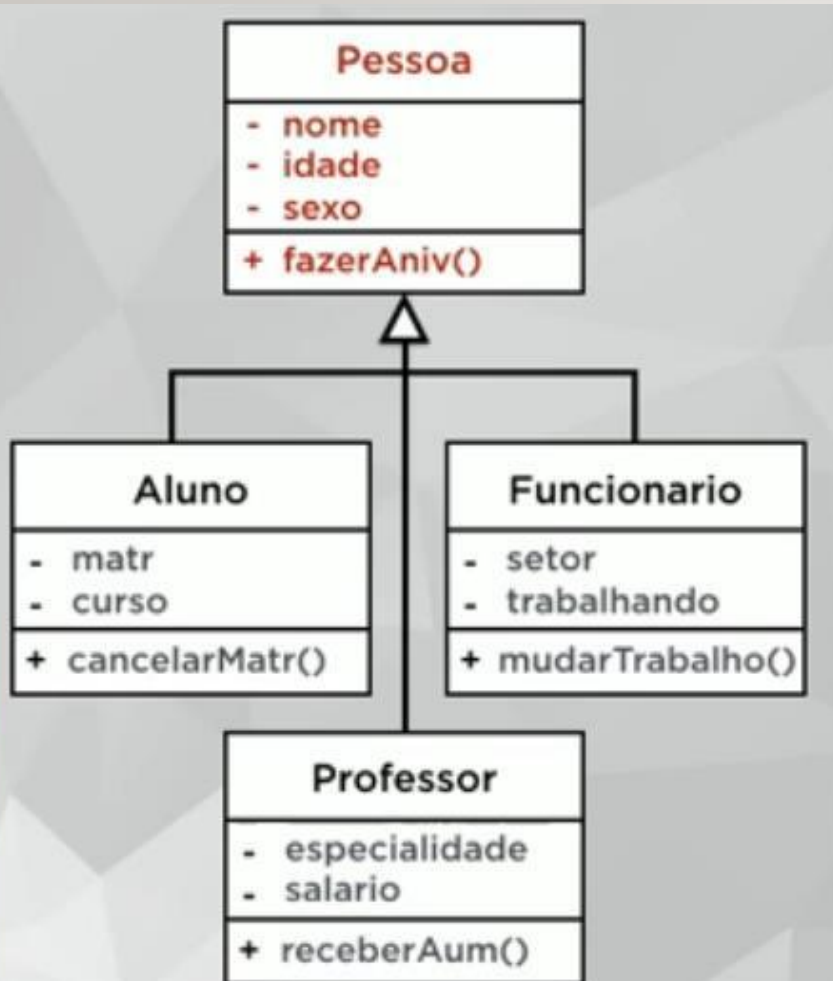


Funcionario
- nome
- idade
- sexo
- setor
- trabalhando
+ fazerAniv()
+ mudarTrabalho()



Professor
- nome
- idade
- sexo
- especialidade
- salario
+ fazerAniv()
+ receberAum()

HERANÇA



Sendo assim, é possível criar uma classe genérica que contenha todos estes atributos e métodos em comum.

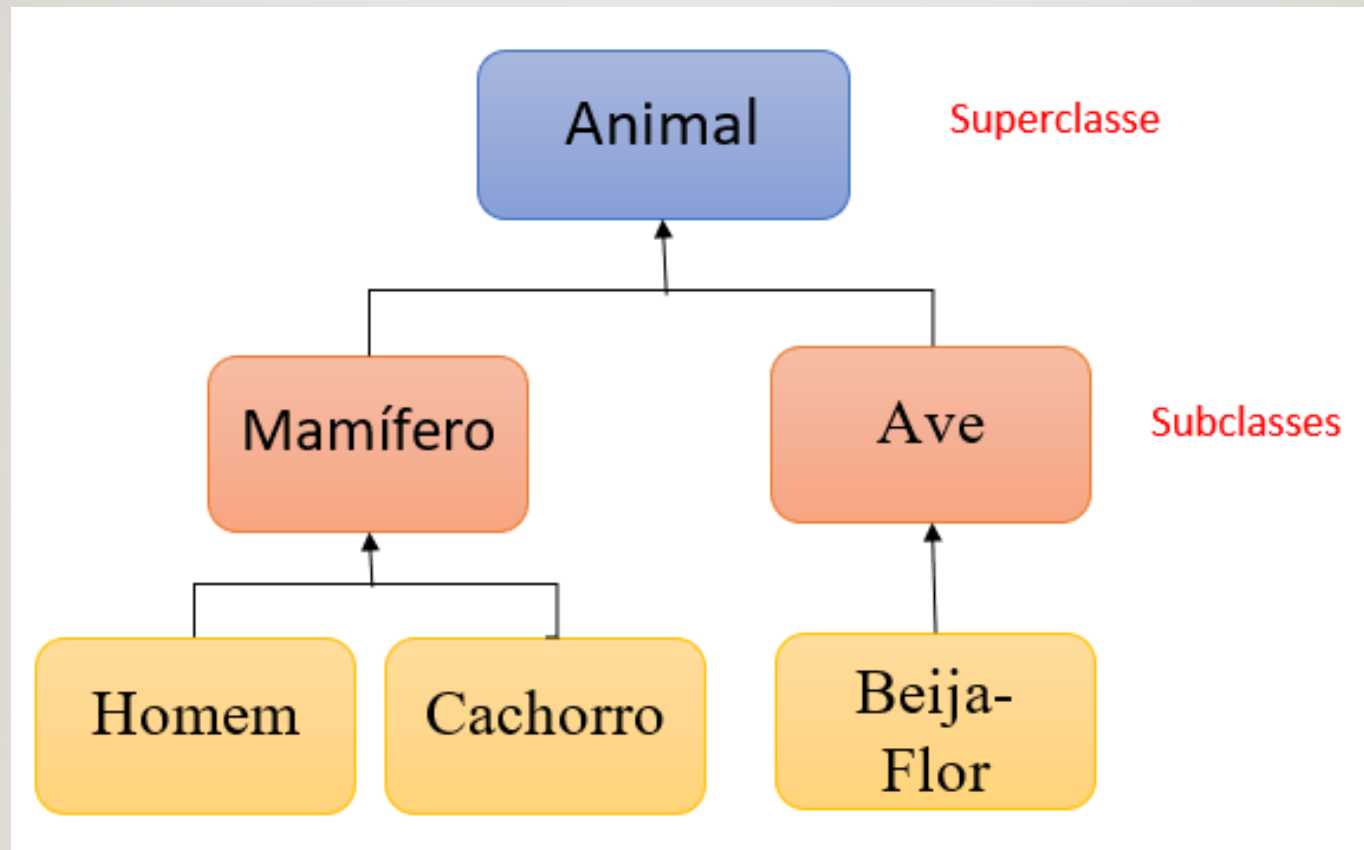
A seta representa que as classes Aluno, Funcionário e Professor são filhas da classe Pessoa. Elas estendem todos os atributos e métodos da classe mãe e adicionam mais algumas características.

HERANÇA

Para dizer que uma classe é filha de outra, em Java se usa a palavra extends

```
public class Funcionario extends Pessoa{  
  
    private String setor;  
    private boolean trabalhando;  
    // ... restante da classe  
}
```

HERANÇA



REESCRITA DE MÉTODOS

Em alguns casos, uma subclasse pode precisar de uma implementação diferente que sua classe mãe realiza. É possível que a subclasse possa reescrever o método para fazer sua própria implementação.



REESCRITA DE MÉTODOS

```
public class Funcionario extends Pessoa{  
  
    private String setor;  
    private boolean trabalhando;  
  
    @Override  
    public float recebeAumento() {  
        return super.recebeAumento() + 100;  
    }  
    // ... restante da classe  
}
```

POLIMORFISMO

Poli = muitas

Morfo = formas

Polimorfismo é a capacidade de um Objeto poder ser referenciado de várias formas.

Existem vários tipos de polimorfismo, dentre eles está o de Sobreposição e Sobrecarga.



POLIMORFISMO

Sobreposição: permite referenciar um objeto de uma subclasse como um objeto da superclasse. Também é possível sobrescrever métodos herdados.

```
Funcionario f1 = new Funcionario();  
Pessoa p1 = new Funcionario();  
Pessoa p2 = new Aluno();  
Pessoa p3 = new Professor();
```


POLIMORFISMO

Sobrecarga: Permite ter vários métodos com o mesmo nome, diferenciando-os pela sua assinatura, quantidade e tipos de parâmetros.

```
● showConfirmDialog(Component cmpnt, Object o) int
● showConfirmDialog(Component cmpnt, Object o, String string, int i) int
● showConfirmDialog(Component cmpnt, Object o, String string, int i, int il) int
● showConfirmDialog(Component cmpnt, Object o, String string, int i, int il, Icon icon) int
```

PROJETO

- Máximo de 3 pessoas – Quem faz Banco de dados na mesma equipe;
- Deve ser um sistema de cadastro
- Deve possuir no mínimo 3 CRUDs (cadastro, listagem, atualização e remoção);
- Deve persistir os dados (arquivo .txt ou Banco de Dados);
- Temas/Equipes devem ser decididas até dia 10/04;
- Deve ser apresentado junto com o projeto um documento, contendo a descrição do projeto, classes utilizadas e ação que cada classe realiza;
- Entra do projeto por e-mail até dia 13/06

OBRIGADO !!

