



**TECNOLOGIA E INOVAÇÃO
EM PROL DA INDÚSTRIA**



Curso Técnico em Informática

Herança

Prof: Diego Corrêa

Conceito

- É o mecanismo pelo qual pode-se definir uma nova classe de objetos a partir de uma classe já existente
- Esta nova classe poderá aproveitar o comportamento e possíveis atributos da classe estendida

Conceito

- A classe sendo refinada é chamada de superclasse ou classe base, enquanto que a versão refinada da classe é chamada uma subclasse ou classe derivada

Conceito

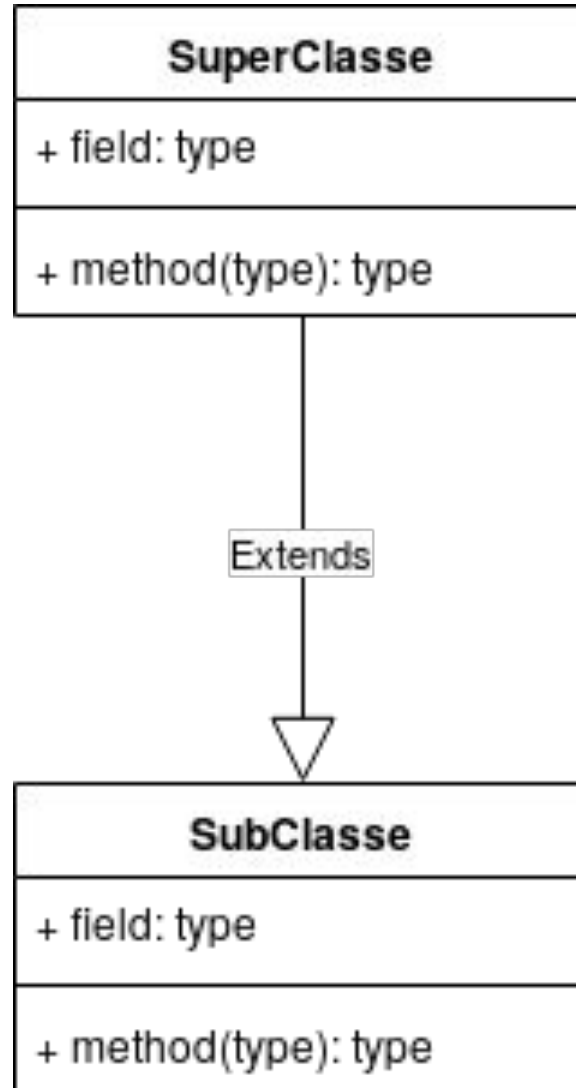
A partir de um grupo de classes, identificar características que são comum a todas:

- Carro -> Automóvel
- Moto -> Automóvel

- iPhone 4S -> iPhone
- iPhone 5 -> iPhone

- Morcego -> Mamífero
- Humano -> Mamífero

UML



Herança

- Herança múltipla, em orientação a objetos, é o conceito de herança de duas ou mais classes
- Ela é implementada nas linguagens de programação C++ e em Python, por exemplo
- A linguagem Java possui apenas herança simples (uma classe possui no máximo uma classe pai), mas permite que uma classe implemente várias interfaces

Herança vs Associações

Código

```
Public Class A {
```

```
    // Atributos
```

```
    // Construtores
```

```
    // Métodos
```

```
}
```

```
Public Class B extends A {
```

```
    // Declarações da Classe B
```

```
}
```

Código

Para utilizar os métodos ou construtores da SuperClasse/ClassePai é necessário fazer uma chamada:

```
B(int a, int b, int c){  
    super(a, b, c); // Construtor Pai  
}
```

```
super.metodo();//Chamando método da classe  
pai
```

Código

É possível sobrescrever os métodos da SuperClasse/ClassePai, usando o decorador **@override**

@override

```
public void metodo() {
```

```
    // Faz algo diferente da classe pai
```

```
}
```

Código

Mesmo durante a sobreposição é possível invocar o método pai

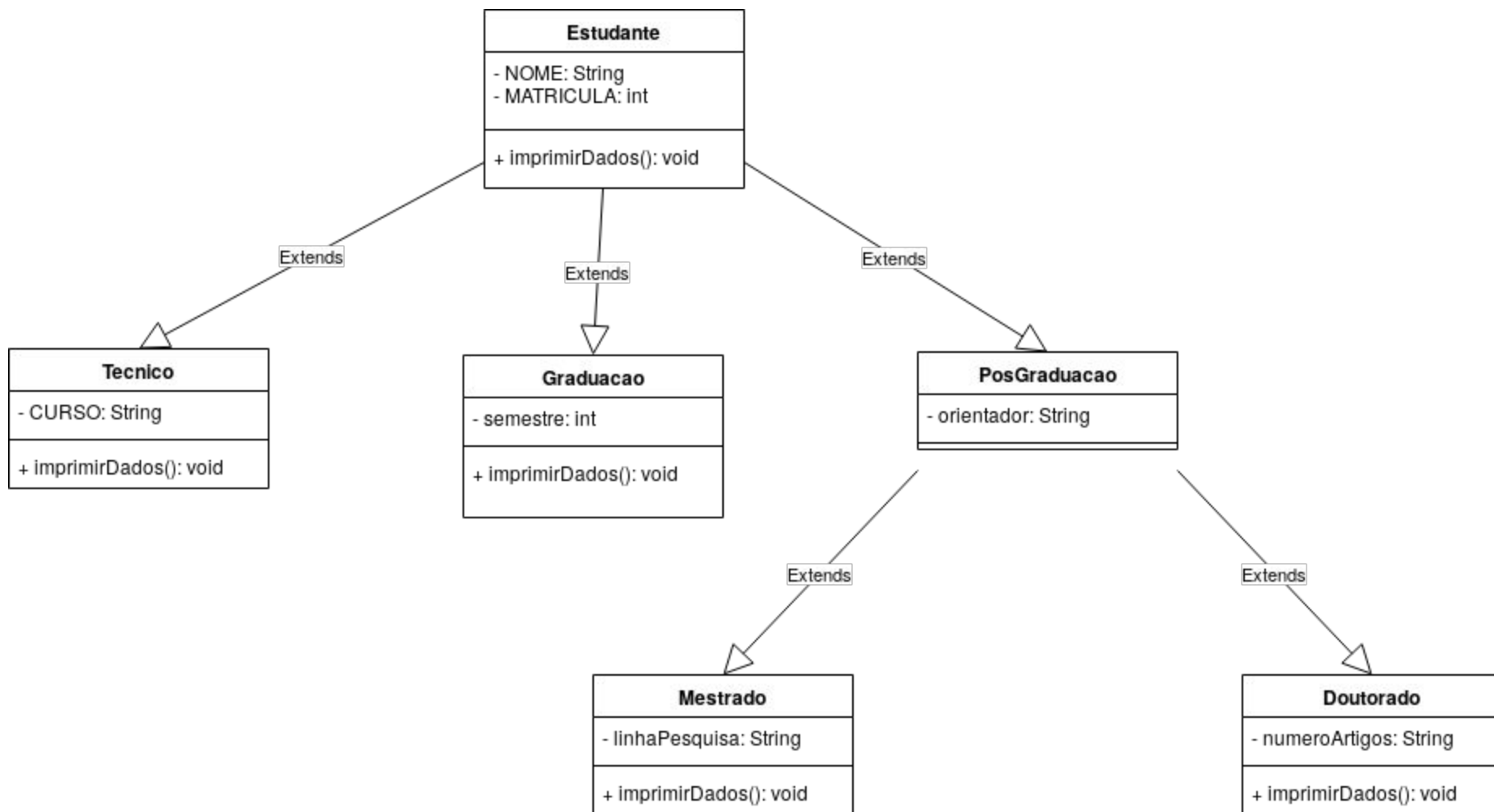
@override

```
public void metodo() {  
    super.metodo(); // Invocando metodo pai  
    // Faz algo diferente da classe pai  
}
```

Exercicio

Criar as 6 classes do diagrama a seguir e inserir dados de cada um dos estudantes

Exercicio



Exercicio

```
public class Heranca {
    public static void main(String[] args) {
        estudante();
        tecnico();
        graduacao();
        mestrado();
        doutorado();
    }

    public static void estudante() {
        Estudante estudanteNormal = new Estudante("Estudioso do Estudo Estudado", 10020172);
        estudanteNormal.imprimirDados();
    }

    public static void tecnico() {
        Tecnico alunoTecnico = new Tecnico("Desenvolvimento de Software", "Tecnaldo Tecnilson", 20020191);
        alunoTecnico.imprimirDados();
    }

    public static void graduacao() {
        Estudante estudanteGraduacao = new Graduacao(1, "Estudioso do Estudo Estudado", 10020172);
        estudanteGraduacao.imprimirDados();
    }

    public static void mestrado() {
        Mestrado alunoPos = new Mestrado("Importante Pesquisa", "Orientinaldo que Orienta", "Mestrando que vai mestrar", 20020191);
        alunoPos.imprimirDados();
    }

    public static void doutorado() {
        Doutorado alunoPos = new Doutorado(11, "Orientinaldo que Orienta", "Doutorando que vai Doutorar", 20020191);
        alunoPos.imprimirDados();
    }
}
```