

Sistema

FIRJAN




INFORMA. FORMA. TRANSFORMA.

Programação Desktop


Fabício Curvello Gomes

Sistema


FIRJAN



INFORMA. FORMA. TRANSFORMA.



Herança e Polimorfismo



Herança


É a possibilidade de se criar classes derivadas de outras já existentes.


As classes filhas contêm todos os atributos e métodos da super classe.

Para referenciar super classe usa-se a palavra reservada ***extends***

Exemplo:

```
class SuperClasse {}
class SubClasse extends SuperClasse {}
```


3




A Classe ***Object***



A classe ***Object*** é a superclasse de todas as classes do Java e é definida em *java.lang*.

Esta herança é implícita, não precisa ser especificada no código.

Toda classe herda os métodos definidos pela classe ***Object***.

O método ***toString*** é chamado sempre que uma conversão para String for necessária.



4





INFORMA. FORMA. TRANSFORMA.

Criando Objeto a Partir de SubClasse

Um objeto criado a partir de uma subclasse possuirá todos os atributos tanto da subclasse quanto da superclasse.

Exemplo: Ver exercício **12_Herança** (Próximo Slide)


5



INFORMA. FORMA. TRANSFORMA.

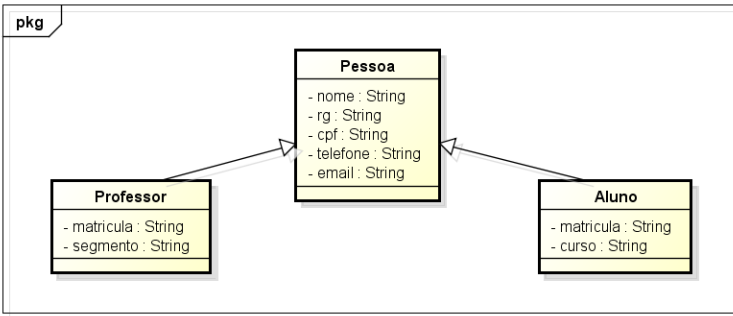
Exercício **12_Heranca**

Criar as classes com herança no pacote *model*, e no pacote *controller*, criar a Classe Teste, que dará funcionalidade ao sistema, implementando objetos com herança.

pkg

```

classDiagram
    class Pessoa {
        - nome : String
        - rg : String
        - cpf : String
        - telefone : String
        - email : String
    }
    class Professor {
        - matricula : String
        - segmento : String
    }
    class Aluno {
        - matricula : String
        - curso : String
    }
    Pessoa <|-- Professor
    Pessoa <|-- Aluno
        
```



powered by astah®

Descrição da tarefa no arquivo [PD - TI - 07.1 - Código Exercício 12 Heranca.pdf](#)

6

Polimorfismo

Polimorfismo é a capacidade de decidir em tempo de execução qual método deve ser chamado.

Para que o polimorfismo seja possível são necessários:

Uma hierarquia de classes onde existam métodos sobrescritos.

- A possibilidade de se criar um objeto de uma subclasse e colocá-lo em uma referência de uma superclasse.

Só é possível utilizar polimorfismo na existência de herança.



7

Sistema
FIRJAN



INFORMA, FORMA, TRANSFORMA.

Sobrescrita de Métodos

Consiste em definir, em alguma subclasse, um método com a mesma assinatura (mesmo nome e mesmos argumentos) e mesmo tipo de retorno de uma superclasse.



Se for necessário chamar o método sobrescrito da superclasse pode ser usada a palavra super:

`super.metodo() ;`

A partir da versão 5 sugere-se a utilização da anotação **@Override** antes do método sobrescrito.



8




INFORMA. FORMA. TRANSFORMA.



Método *toString*

Método da Super Classe Object.

Se for utilizado por uma classe JAVA é necessária a utilização da notação `@Override`.

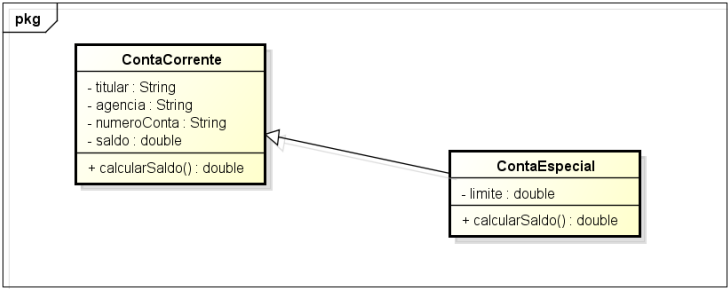
Utilizado para retornar um valor em formato String para apresentação.


9



INFORMA. FORMA. TRANSFORMA.

Exercício *13_Polimorfismo*

Criar as classes com herança no pacote *model*, e no pacote *controller*, criar a Classe Teste, que dará funcionalidade ao sistema.




```



classDiagram
    class ContaCorrente {
        - titular : String
        - agencia : String
        - numeroConta : String
        - saldo : double
        + calcularSaldo() : double
    }
    class ContaEspecial {
        - limite : double
        + calcularSaldo() : double
    }
    ContaCorrente <|-- ContaEspecial
            
```

powered by astah

Descrição da tarefa no arquivo [PD - TI - 07.2 - Código Exercício 13_Polimorfismo.pdf](#)



10

A 3D white figure stands next to a large blue question mark, symbolizing doubt or a question. The figure is in a thinking pose, with one hand on its chin. The question mark is large and blue, casting a shadow on the ground. The background is a light blue gradient.




INFORMA, FORMA, TRANSFORMA.


Bibliografia



Java Como Programar 8ª Edição
Paul Deitel e Harvey Deitel
Ed. Pearson



Java 7 Ensino Didático
Sérgio Furgeri
Ed. Érica



Fundamentos de Computação e Orientação a Objetos Usando Java
Francisco A. C. Pinheiro
Ed. LTC

13