

Orientação a Objetos e Java

Sérgio Soares
sergio@dsc.upe.br

Redefinição de Métodos e Ligação Dinâmica

Continuação do exercício

- Modifique a classe Banco para que seja possível render juros de uma poupança ...

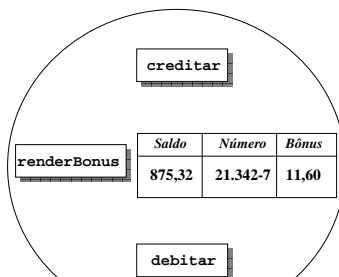
Subtipos: Verificação Dinâmica com instanceof

```
...
Conta c = this.procurar("567-8");
if (c instanceof Poupanca)
    ((Poupanca) c).renderJuros(0.01);
else
    throw new RuntimeException(
        "Este número não é de uma poupança!");
...
```

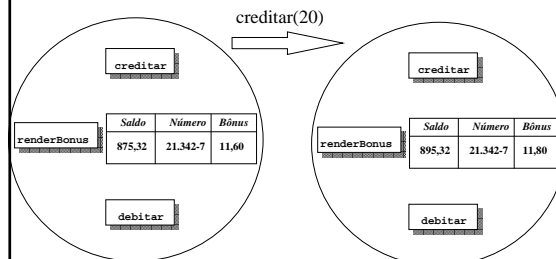
Verificação Dinâmica de Tipos

- **Casts e instanceof:**
 - ((Tipo) variável)
 - variável **instanceof** Tipo
 - O tipo de *variável* deve ser supertipo de Tipo
 - Casts geram exceções quando **instanceof** retorna **false**
 - Casts são essenciais para verificação estática de tipos (compilação)

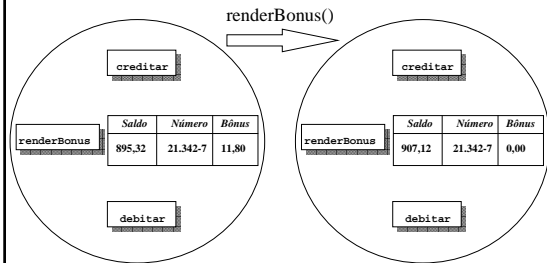
Objeto Conta Especial



Estados de uma Conta Especial



Estados de uma Conta Especial



Contas Especiais: Assinatura

```
public class ContaEspecial extends Conta {  
    public ContaEspecial(String numero) {}  
    public void renderBonus() {}  
    public double getBonus() {}  
    public void creditar(double valor) {}  
}
```

Contas Especiais: Descrição

```
public class ContaEspecial extends Conta {  
  
    private double bonus;  
  
    public ContaEspecial(String numero) {  
        super (numero);  
        bonus = 0.0;  
    }  
}
```

```
public void creditar(double valor) {  
    super.creditar(valor);  
    bonus = bonus + (valor * 0.01);  
}  
  
public void renderBonus() {  
    super.creditar(this.bonus);  
    bonus = 0;  
}  
  
public double getBonus() {  
    return this.bonus;  
}
```

Redefinição de Métodos

- Invariância: tipos dos argumentos e resultados da *redefinição* tem que ser iguais aos tipos da *definição*
- Semântica e Visibilidade dos métodos redefinidos deve ser preservada
- Só é possível acessar a definição dos métodos da superclasse imediata (via **super**)

Usando Contas Especiais

```
...  
ContaEspecial contaes;  
contaes = new ContaEspecial("21.342-7");  
contaes.creditar(200.00);  
contaes.debitar(100.00);  
contaes.renderBonus();  
System.out.print(contaes.getSaldo());  
...
```

Ligações Dinâmicas

```
...
Conta conta;
conta = new ContaEspecial("21.342-7");
((Conta)conta).creditar(200.00);
conta.debitar(100.00);
if (conta instanceof ContaEspecial) {
    ((ContaEspecial) conta).renderBonus();
}
System.out.print(conta.getSaldo());
...
```

Ligações Dinâmicas

- Dois métodos com o mesmo nome e tipo:
 - definição e redefinição, qual usar?
- O código é escolhido dinamicamente (em tempo de execução), não estaticamente (em tempo de compilação)
- Escolha é feita com base na classe do objeto associado à variável destino do método

Exercício

- Modifique a classe **Banco** para que seja possível computar o bônus de uma conta especial. Foi necessário redefinir algum método de **Banco**? Justifique a sua resposta.

Exercícios

- Altere a classe **Conta** para possibilitar a emissão de extratos.
- O que é necessário alterar em **Banco** para trabalhar com as novas contas? Faça as modificações necessárias.

Exercícios

- Modifique o sistema de controle de estoque requisitado como exercício para poder também armazenar produtos perecíveis; isto é, produtos que têm um prazo de validade.
- Modifique o sistema de controle de estoque para manter um histórico de vendas de determinados produtos.