

Aula 7: Herança, reescrita e polimorfismo

Professor(a): Virgínia Fernandes Mota
virginia@teiacoltec.org

ALGORITMOS E ESTRUTURAS DE DADOS - DESENVOLVIMENTO DE SISTEMAS



Exercícios

- 1. Vamos criar uma classe Conta, que possua um saldo, os métodos para pegar saldo, depositar e sacar.
- 2. Adicione um método na classe Conta, que atualiza essa conta de acordo com uma taxa percentual fornecida.

```
1 class Conta {  
2     protected double saldo;  
3  
4     // outros metodos aqui tambem ...  
5  
6     public void atualiza(double taxa) {  
7         this.saldo += this.saldo * taxa;  
8     }  
9 }
```

Exercícios

- 3. Crie duas subclasses da classe Conta: ContaCorrente e ContaPoupanca. Ambas terão o método atualiza reescrito: A ContaCorrente deve atualizar-se com o dobro da taxa e a ContaPoupanca deve atualizar-se com o triplo da taxa. Além disso, a ContaCorrente deve reescrever o método deposita, a fim de retirar uma taxa bancária de dez centavos de cada depósito.
- 4. Crie uma classe com método main e instancie essas classes, atualize-as e veja o resultado.

Exercícios

```
public class ContaPoupanca extends Conta {  
    public void atualiza(double taxa) {  
        this.saldo += this.saldo * taxa * 3;  
    }  
}
```

```
public class ContaCorrente extends Conta {  
    public void atualiza(double taxa) {  
        this.saldo += this.saldo * taxa * 2;  
    }  
  
    public void deposita(double valor) {  
        this.saldo += valor - 0.10;  
    }  
}
```

Exercícios

```
public class TestaContas {  
    public static void main(String[] args) {  
        Conta c = new Conta();  
        ContaCorrente cc = new ContaCorrente();  
        ContaPoupanca cp = new ContaPoupanca();  
  
        c.deposita(1000);  
        cc.deposita(1000);  
        cp.deposita(1000);  
  
        c.atualiza(0.01);  
        cc.atualiza(0.01);  
        cp.atualiza(0.01);  
  
        System.out.println(c.getSaldo());  
        System.out.println(cc.getSaldo());  
        System.out.println(cp.getSaldo());  
    }  
}
```

Exercícios

- 5. Vamos criar uma classe que seja responsável por fazer a atualização de todas as contas bancárias e gerar um relatório com o saldo anterior e saldo novo de cada uma das contas.

Exercícios

```
public class AtualizadorDeContas {  
    private double saldoTotal = 0;  
    private double selic;  
  
    public AtualizadorDeContas(double selic) {  
        this.selic = selic;  
    }  
  
    public void roda(Conta c) {  
        // aqui voce imprime o saldo anterior, atualiza a conta,  
        // e depois imprime o saldo final  
        // lembrando de somar o saldo final ao atributo saldoTotal  
    }  
  
    // outros metodos, colocar o getter para saldoTotal!  
}
```

- 6. Crie uma classe Banco que possui um array de Conta. Repare que num array de Conta você pode colocar tanto ContaCorrente quanto ContaPoupanca. Crie um método public void adiciona(Conta c), um método public Conta pegaConta(int x) e outro public int pegaTotalDeContas(), muito similar a relação anterior de Empresa-Funcionario. Faça com que seu método main crie diversas contas, insira-as no Banco e depois, com um for, percorra todas as contas do Banco para passá-las como argumento para o AtualizadorDeContas.