

Programação Orientada a Objetos

Exercício Introdutório sobre Modelagem Orientada a Objetos

Enunciado geral: considere a enumeração “NomeMoeda” e a classe “Moeda” nos exercícios que seguem:

```
public enum NomeMoeda {  
    UmReal, Cinquenta, VinteCinco, Dez, Cinco, Um  
}
```

```
public class Moeda{  
    private NomeMoeda nome;  
  
    public Moeda(NomeMoeda nome){  
        this.nome = nome;  
    }  
  
    public NomeMoeda getNomeMoeda(){  
        return nome;  
    }  
  
    public int getValorCentavos(){  
        switch(nome){  
            case UmReal: return 100;  
            case Cinquenta: return 50;  
            case VinteCinco: return 25;  
            case Dez: return 10;  
            case Cinco: return 5;  
            case Um: return 1;  
            default: return 0;  
        }  
    }  
  
    public double getValorReais(){  
        switch(nome){  
            case UmReal: return 1.0;  
            case Cinquenta: return 0.5;  
            case VinteCinco: return 0.25;  
            case Dez: return 0.1;  
            case Cinco: return 0.05;  
            case Um: return 0.01;  
            default: return 0.0;  
        }  
    }  
  
    public String toString(){  
        return nome.toString();  
    }  
}
```

- 1) Considerando a enumeração “NomeMoeda” e a classe “Moeda” apresentadas, escreva uma classe Java que modela um cofrinho de moedas. A **interface pública** da classe “Cofrinho” deve ter os seguintes métodos:

<code>public boolean insere(Moeda moeda);</code>	Insere uma moeda no cofrinho. Como um “cofrinho” tem capacidade limitada, deve retornar true se conseguiu inserir a moeda e false caso contrário.
<code>public Moeda retira();</code>	Retira do cofrinho a última moeda inserida (se esta operação for chamada várias vezes deve ir retirando todas as moedas na ordem inversa em que foram inseridas). Deve retornar a moeda retirada ou “null” caso o cofrinho esteja vazio
<code>public int getQtidadeMoedas();</code>	Informa quantas moedas estão guardadas no cofrinho
<code>public int getQtidadeMoedasTipo(NomeMoeda nomeMoeda);</code>	Informa quantas moedas de um certo tipo estão guardadas no cofrinho
<code>public int getValorTotalCentavos();</code>	Informa o valor total armazenado no cofrinho (em centavos)
<code>public double getValorTotalReais();</code>	Informa o valor total armazenado no cofrinho (em reais)

- 2) Escreva um programa Java que cria um cofrinho e insere 10 moedas no mesmo. Em seguida o programa deve imprimir:
- Quantas moedas foram armazenadas no cofrinho.
 - Quantas moedas de um real estão armazenadas no cofrinho.
 - Quantas moedas de 50 centavos estão armazenadas no cofrinho.
 - Qual o valor total em centavos armazenado no cofrinho.
 - Qual o valor total em reais armazenado no cofrinho.
 - Qual o valor total em centavos armazenado no cofrinho após a retirada das duas últimas moedas inseridas.
- 3) **(extra)** Escreva um programa que ofereça para o usuário opções para inserir e retirar moedas. A cada operação deve informar a quantidade de moedas informadas e o valor total em reais armazenado.