

Extra - 06

Orientação a Objetos - DCC025

Prof. Edmar Welington Oliveira
oliveira.edmar@ufjf.edu.br

Universidade Federal de Juiz de Fora - UFJF
Departamento de Ciência da Computação - DCC

Exercício 01

■ Exercício 01

- Dada a classe ItemOrcamento que representa um item de um orçamento

```
public class ItemOrcamento {  
    private String historico; // historico do item  
    private float valor; // valor do item  
    public ItemOrcamento(String historico, float valor) {  
        this.historico = historico;  
        this.valor = valor;  
    }  
    public String getHistorico() {  
        return historico;  
    }  
    public float getValor() {  
        return valor;  
    }  
}
```

Exercício 01

■ Continuação

- Escreva uma classe herdeira de `ItemOrçamento` denominada `ItemOrçamentoComplexo` que mantenha um vetor com subitens de orçamento que podem ser da classe `ItemOrçamento` ou da classe `ItemOrçamentoComplexo`. A classe implementa os seguintes métodos:

Construtor	Além dos parâmetros da superclasse, recebe como parâmetro o vetor com os subitens de orçamento.
getValor	Sobrescreve o método da superclasse, retornando a soma de valores de todos os subitens de orçamento.
encontraltem	Recebe como parâmetro o histórico de um subitem (String) e retorna o objeto correspondente ao subitem que possui este histórico, se existir. Se não existir retorna null.

Exercício 02

Representa o total de despesas de um mês

```
public class DespesaMes {  
    private int mes; // mes da despesa  
    private float valor; // valor da despesa  
    public DespesaMes(int mes, float valor) {  
        this.mes = mes;  
        this.valor = valor;  
    }  
    public int getMes() {  
        return mes;  
    }  
    public float getValor() {  
        return valor;  
    }  
}
```

Representa o total de despesas de um dia

```
public class DespesaDia extends DespesaMes {  
    private int dia; // dia da despesa  
    public DespesaDia(int dia, int mes, float valor) {  
        super(mes, valor);  
        this.dia = dia;  
    }  
    public int getDia() {  
        return dia;  
    }  
}
```

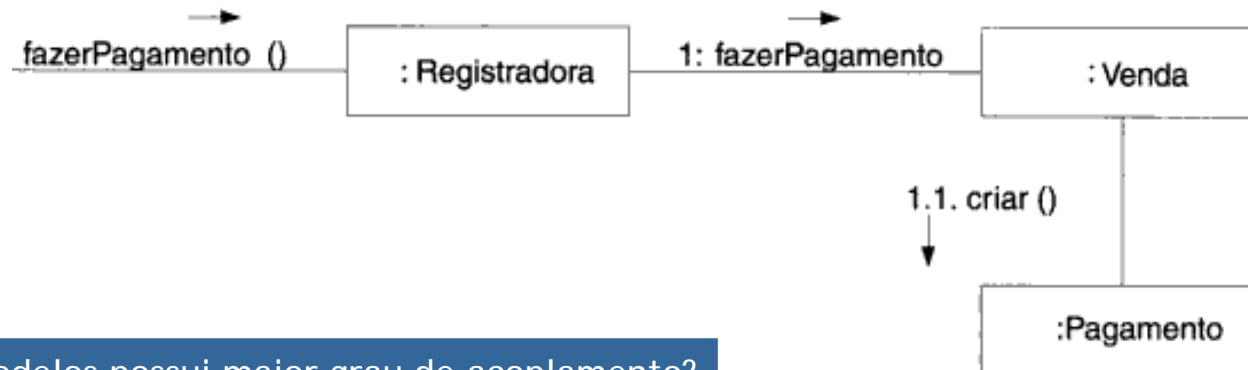
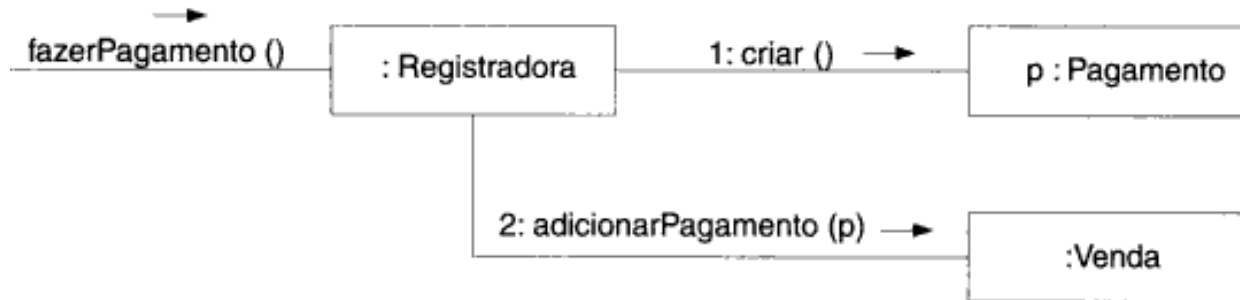
Exercício 02

■ Exercício 02

- Escreva uma classe que representa todas as despesas de um indivíduo. Ela mantém um vetor onde podem ser registradas tanto despesas de um dia (DespesaDia), quanto despesas de um mês (DespesaMes). A classe implementa os seguintes métodos:

<i>Construtor</i>	Recebe como parâmetro o CPF e um vetor com as despesas de um indivíduo e as guarda.
<i>getCPF</i>	Retorna o CPF do indivíduo.
<i>totalizaMes</i>	Recebe um parâmetro com um mês (int) e retorna um objeto da classe <code>DespesaMes</code> onde estará registrada a soma de todas as despesas que o indivíduo fez naquele mês.

Exercício 03



Qual dos modelos possui maior grau de acoplamento?

Extra - 06

Orientação a Objetos - DCC025

Prof. Edmar Welington Oliveira
oliveira.edmar@ufjf.edu.br

Universidade Federal de Juiz de Fora - UFJF
Departamento de Ciência da Computação - DCC