

## Atividade Prática 8 – Questionário Diagrama de Classes

### Instruções:

- Gere uma folha de respostas para o questionário abaixo, destacando/grife a resposta correta conforme com uma cor.
- Quando necessário, gere o diagrama na ferramenta case Visual Paradigm e anexe a imagem na questão correspondente.
- Grave como **atividade8\_SeuNome.pdf**
- Entregue a atividade via moodle.

- **Considerando os diagramas de classes UML, relacione as colunas e assinale a alternativa com a seqüência correta.**

1. Associação. 2. Multiplicidade. 3. Operações. 4. Classe abstrata.

( ) Não pode ser instanciada diretamente. ( ) Linha cheia entre duas classes. ( ) Indica quantos objetos podem preencher a propriedade. ( ) Métodos presentes em uma classe.

- 1 – 3 – 4 – 2.
- 4 – 1 – 2 – 3.
- 4 – 1 – 3 – 2.
- 3 – 1 – 2 – 4.
- 1 – 4 – 3 – 2.

- **Com relação à engenharia de software, julgue o item.**

UML é uma linguagem de programação usada apenas no desenvolvimento de sistemas para a Internet.

- Certo
- Errado

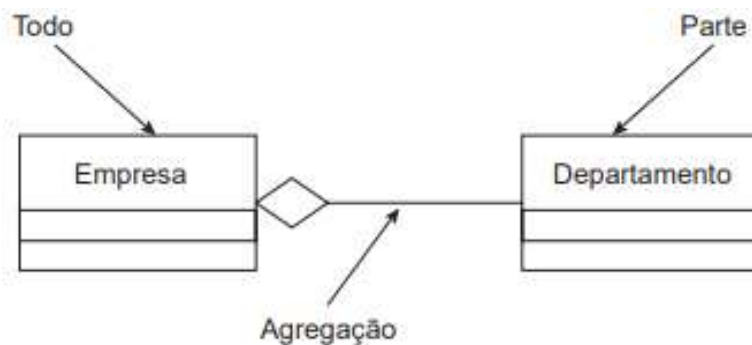
- **Acerca da linguagem de modelagem unificada (UML), assinale a alternativa correta.**

- A UML é uma linguagem de código que tem a finalidade de criar, especificamente, o modelo físico de um determinado sistema.
- Sua sintaxe foi projetada apenas para atender às linguagens-alvo mais recentes, como a JavaScript.
- Apesar de ser uma ferramenta de modelagem muito poderosa, ela não é capaz de capturar conhecimento e expressá-lo.
- A UML tem a finalidade de documentar e visualizar os artefatos que são especificados e construídos na análise e no projeto de um sistema.
- A melhor definição para a UML, de acordo com diversos analistas, é que ela é uma linguagem de programação visual.

- **Na análise orientada a objetos, o diagrama que descreve a estrutura dos objetos e seus relacionamentos, descreve a estrutura estática de um sistema,**

isto é, descreve como o sistema é estruturado, e não como ele se comporta, é o diagrama de

- classe.
  - pacotes.
  - colaboração.
  - estados.
  - atividades.
- No contexto de UML, a respeito dos conceitos de agregação e composição, assinale a alternativa correta.
    - A agregação representa uma relação de “é um” ou “é uma” entre duas classes, enquanto a composição representa uma relação de “parte-todo” entre duas classes.
    - A composição representa uma relação de “é um” ou “é uma” entre duas classes, enquanto a agregação representa uma relação de “parte-todo” entre duas classes.
    - Composição e agregação são apenas sinônimos para definir relações de “parte-todo”, não havendo diferença entre os dois conceitos.
    - Uma agregação é uma forma mais forte de composição na qual o *aggregate* tem a responsabilidade de gerenciar suas partes.
    - Uma composição é uma forma mais forte de associação na qual o *composite* tem a responsabilidade de gerenciar suas partes.
  - A UML (Unified Modeling Language) é uma linguagem de especificação de software orientado a objetos. Em uma modelagem orientada a objetos, os três relacionamentos mais importantes são as dependências, as generalizações e as associações. O analista Antônio começou a fazer um projeto usando a UML, e a primeira ação foi modelar a empresa e seus departamentos, numa forma simples de agregação “Todo/Parte”, conforme está ilustrado na Figura abaixo:



A agregação que a Figura ilustra é um tipo especial de

- dependência
- generalização
- associação
- classe
- método

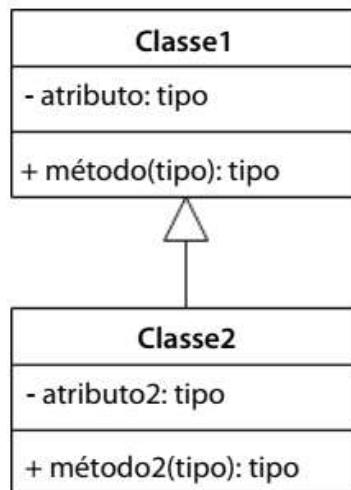
- Sobre a UML, considere o trecho abaixo.

A UML (*Unified Modeling Language*) não é uma \_\_\_\_\_ de desenvolvimento, é uma \_\_\_\_\_ para a elaboração da estrutura de projetos de software. A UML permite que desenvolvedores visualizem os produtos de seus trabalhos em \_\_\_\_\_ padronizados(as). Junto com uma notação gráfica, ela também especifica significados, isto é, semântica.

Assinale a alternativa que preencha correta e respectivamente as lacunas.

- fase / atividade / etapas
- metodologia / linguagem-padrão / diagramas
- linguagem-padrão/ metodologia / passos
- metodologia / metodologia / metodologias

- Considere o diagrama de classes representado pelas classes Classe1 e Classe2:



Assinale a alternativa que representa a relação entre as classes de acordo com o paradigma de orientação a objetos.

- Herança.
  - Refatoração.
  - Acoplamento.
  - Coesão.
- 
- No uso da linguagem UML, o nível de visibilidade de um atributo que só pode ser acessado por classes declaradas dentro do mesmo pacote da classe a que pertence o atributo é denominado
    - pacote.
    - público.
    - protegido.
    - privado.
    - domínio.
- 
- Na UML, um relacionamento é uma ligação entre itens, podendo ser representado graficamente por meio de diferentes tipos de linhas. Sobre os

**relacionamentos na UML, analise as afirmativas abaixo.**

I A associação determina que as instâncias de uma classe estão de alguma forma ligadas às instâncias da outra classe.

II A dependência entre classes indica que os objetos de uma classe dependem de serviços dos objetos de outra classe.

III A generalização demonstra que as informações de uma classe precisam ser complementadas por uma outra classe.

Dentre as afirmativas, estão corretas:

- II e III.
- **I e II.**
- I e III.
- I, II e III.

- **Considere o seguinte diagrama de classes.**

The screenshot shows a Microsoft Word document titled "atividade DC\_final\_2 - Microsoft Word". The document contains the following text:

c. I e III.  
d. I, II e III.

12. Considere o seguinte diagrama de classes.

Sobre o diagrama acima, é correto afirmar que

a. As representações gráficas das classes Pessoa e Empresa não seguem o padrão da especificação UML.  
b. Salário é um método da classe Emprego.

The UML class diagram shows three classes: **Pessoa**, **Empresa**, and **Emprego**. **Pessoa** and **Empresa** are connected by a solid association line with an open circle at the **Pessoa** end and a solid line at the **Empresa** end. The association is labeled "pessoa" at the **Pessoa** end and "empresa" at the **Empresa** end. **Emprego** is a class that inherits from **Pessoa** (indicated by a solid line with an open triangle arrowhead pointing to **Pessoa**). **Emprego** has a method named "salário" of type "Integer".

c. Um objeto da classe Pessoa pode estar associado a zero ou mais objetos da classe Empresa.  
d. A classe Emprego é uma classe de associação.  
e. Uma empresa deve ter pelo menos um funcionário.

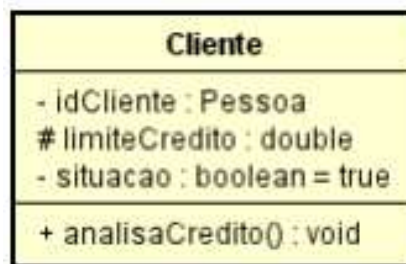
13. Na engenharia de software, os diagramas de classe são exemplos de modelo de:

a. processamento de dados.  
b. composição.  
c. arquitetura.  
d. classificação.  
e. estímulo-resposta.

**Sobre o diagrama acima, é correto afirmar que:**

- **As representações gráficas das classes Pessoa e Empresa não seguem o padrão da especificação UML.**
- Salário é um método da classe Emprego.
- Um objeto da classe Pessoa pode estar associado à zero ou mais objetos da classe Empresa.
- A classe Emprego é uma classe associativa.
- Uma empresa deve ter pelo menos um funcionário.

- Na engenharia de software, os diagramas de classe são exemplos de modelo de:
  - processamento de dados.
  - composição.
  - arquitetura.
  - classificação.
  - estímulo-resposta.
- Considere o modelo abaixo, baseado de acordo com a especificação da UML (*Unified Modeling Language*), versão 2.5.1.

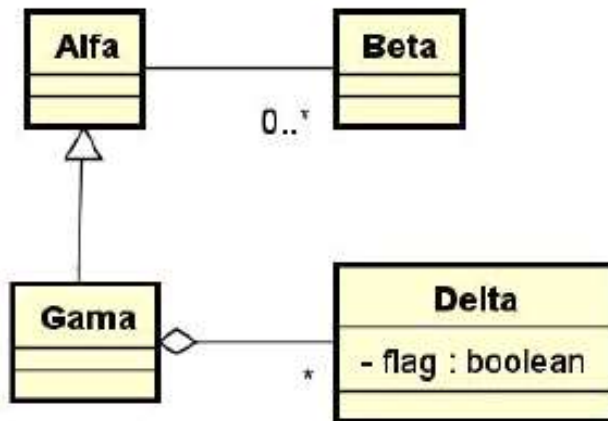


Com base nesse modelo, podemos afirmar que:

- I) o atributo "idCliente" é do tipo "Pessoa", que se refere a uma outra classe já modelada.
- II) o atributo "limiteCredito" é o único atributo que é público, ou seja, não está encapsulado como os demais.
- III) o atributo "situacao", do tipo "boolean" possui a restrição de obrigatoriedade do seu conteúdo.
- IV) o "void" do método "analisaCredito()" irá indicar que o método não possui parâmetros.

**O CORRETO está em:**

- I e III somente.
  - I, II e IV somente.
  - II e III somente.
  - I somente.
  - III somente.
- Considere o diagrama de classes fornecido abaixo.



Sobre o que está especificado neste diagrama, assinale a alternativa correta.

- Se Gama tivesse atributos, estes seriam herdados por Alfa
  - Um objeto da classe Beta pode relacionar-se ou não com várias instâncias da classe Alfa.
  - **Delta representa partes de um todo e o todo é representado por Gama.**
- **Considere as seguintes afirmações sobre *Unified Modeling Language (UML)* – Diagrama de Classes da Linguagem de Modelagem Unificada.**

I - O diagrama de classes é um diagrama comportamental.

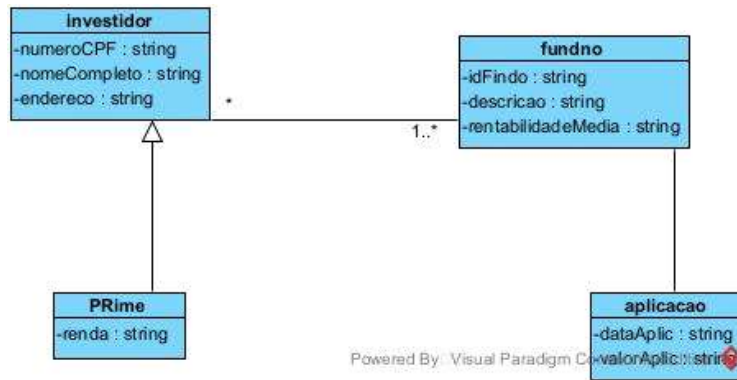
II - A multiplicidade **2..\*** estabelece que existem pelo menos dois objetos envolvidos no relacionamento, podendo haver muitos objetos envolvidos.

III- Em uma agregação, os objetos-partes não são destruídos quando o objeto-todo ao qual estão relacionados é destruído.

Quais estão corretas?

- Apenas I.
  - Apenas II.
  - Apenas I e III.
  - Apenas II e III.
  - **I, II e III.**
- **Gere a representação gráfica do diagrama de classes que possui a seguinte especificação:**

Uma classe INVESTIDOR, que possui os atributos *numeroCPF*, *nomeCompleto* e *endereço*. Essa classe é modelada como superclasse de outra, denominada PRIME, com o atributo *renda*. A classe INVESTIDOR associa-se a outra classe chamada FUNDO, que tem os atributos *idFundo*, *descrição* e *rentabilidadeMedia*. Essa associação possui as seguintes multiplicidades: 1..\* de INVESTIDOR para FUNDO, e \* de FUNDO para INVESTIDOR. Ela apresenta uma classe associativa chamada APLICAÇÃO, com os atributos *dataAplicacao* e *valorAplicado*.



A partir da explicação apresentada, pode-se inferir que: *(existem duas alternativas corretas)*

- A classe associativa **APLICAÇÃO** tem como objetivo guardar informações específicas de cada um dos FUNDOS do INVESTIDOR.
- O fundo tem no mínimo um investidor associado a ele.
- Um mesmo investidor pode aplicar em um mesmo fundo apenas uma vez.
- O objeto *prime* possui um atributo e pode aplicar em um ou vários fundos

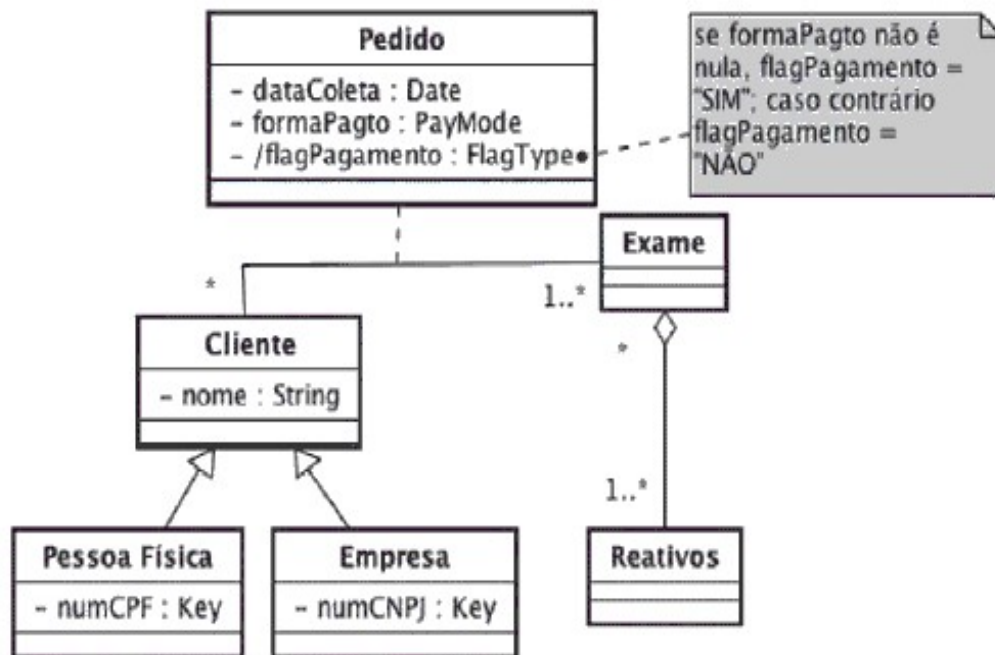
- Considere o seguinte Diagrama de Classes da UML.



O símbolo  indica

estudegratis.com.br

- Um tipo especial de relacionamento de associação, denominado composição.
  - Uma relação todo-parte onde o objeto “parte” só pode pertencer a um objeto “todo” e tem o seu tempo de vida coincidente com o dele.
  - Um tipo especial de relacionamento de associação, denominado herança.
  - Uma relação todo-parte onde um objeto “parte” pode fazer parte de vários objetos “todo”.
- Um laboratório de análises clínicas contratou uma empresa de desenvolvimento de software para informatizar o controle dos exames que seus pacientes realizam. No início do projeto, decidiu-se que a linguagem de modelagem a ser utilizada seria a UML. Após uma primeira rodada de levantamento de requisitos, o analista de dados apresentou para validação o diagrama de classes a seguir.



Inferre-se, pela leitura desse modelo, que

- Cada objeto da classe Reativos só pode ser instanciado a partir de algum método da classe Exame.
- A nota associada à classe Pedido especifica a implementação de um método de controle do construtor da classe Pedido.
- Os objetos da classe Empresa possuem dois atributos: o seu nome e o número de seu CNPJ.
- Um cliente realiza nenhum ou vários pedidos envolvendo vários exames, cada qual composto por um ou mais reativos.
- Um mesmo cliente pode realizar um mesmo exame mais de uma vez, com a diferenciação dos pedidos realizada pela data da coleta.
- Considere a relação todo-parte com um forte vínculo entre as classes Tribunal e Departamento, sendo que Departamento representa as áreas administrativas que integram um Tribunal. Na UML esta relação é tratada como:
  - Agregação. Isso significa que a parte vive mesmo sem o todo.
  - Composição. Isso significa que a parte não vive sem o todo.
  - Herança múltipla. Isso significa que a parte vive mesmo sem o todo porque existem outras partes que compõem o todo.



- Considerando o cenário a seguir, crie um Diagrama de classes:

As informações a seguir se referem a uma aplicação de controle de comanda eletrônica de uma Padaria.

O cliente usa uma comanda eletrônica durante suas compras na Padaria. A cada produto consumido, o atendente registra em sua comanda (que possui uma numeração) o produto e a quantidade.

Ao passar no caixa na saída da Padaria, A Caixa lê os gastos da comanda, finalizando a compra. Na leitura da comanda, verifica-se o valor unitário de cada produto a fim de calcular o valor total da compra

Considerando as classes, atributos e métodos identificados na tabela a seguir, crie o Diagrama de Classes correspondente.

UML 51 Exercícios - Ana Cristina Melo.pdf - Adobe Acrobat Reader DC (64-bit)

Lista de exercícios ...

Fazer login

14 / 129

125%

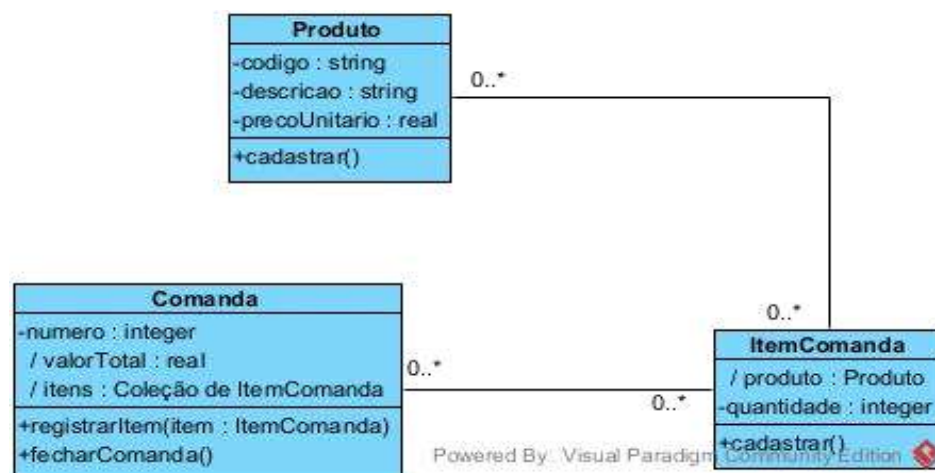
24°C  
Pouco ensolarado

16:49  
25/05/2022

EXERCÍCIO:  
Identifique as classes, atributos e métodos desse cenário.  
Represente os relacionamentos como atributos derivados.

RESOLUÇÃO:

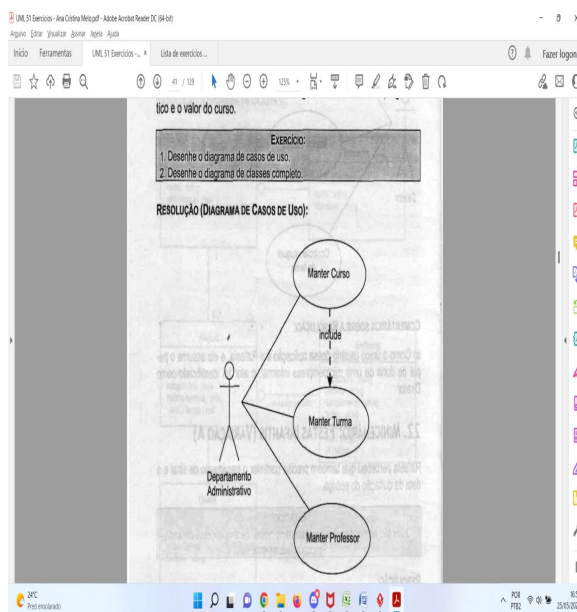
| Classe      | Atributos   | Métodos  |
|-------------|---|--|
| Produto     | codigo : string<br>descricao : string<br>precoUnitario : real           | cadastrar  |
| Comanda     | numero : integer<br>valorTotal : real<br>itens : Coleção de ItemComanda | registrarItem( item : ItemComanda )<br>fecharComanda |
| ItemComanda | /produto : Produto<br>quantidade : integer                              | cadastrar  |

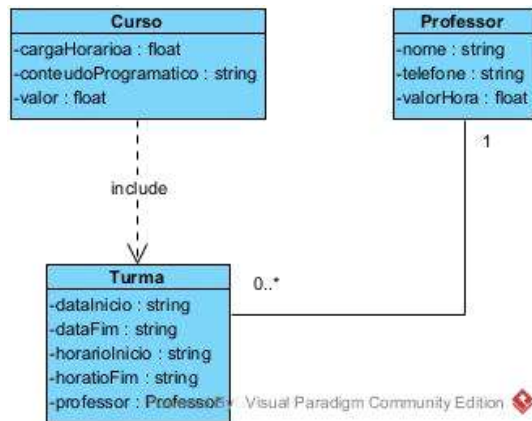


- **Considerando o cenário a seguir e seu respectivo Diagrama de Casos de Uso, crie o Diagrama de Classes:**

Uma empresa oferece diversos cursos de aperfeiçoamento profissional. O dono da empresa precisa controlar, inicialmente: as turmas dos cursos oferecidos. Mais especificamente: a data de início e término, o horário de início e término. Bem como o nome, o celular do professor e o valor da sua hora/aula do professor. Cada turma possui um único professor, mas o professor pode possuir nenhuma ou várias turmas, desde que os horários sejam compatíveis.

Para cada curso deve-se controlar: a carga horária, o conteúdo programático e valor do curso.

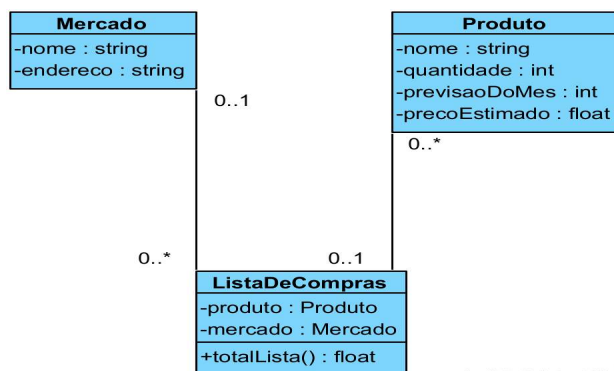




- **Considerando o cenário a seguir, crie um Diagrama de Classes:**

Carolina controla por meio de uma planilha no Excel sua lista de compras mensal, sua planilha retorna o total estimado para a lista de compras gerada.

Para isso, Carolina lança o nome do produto, a unidade de compra, a quantidade prevista para o mês e o preço estimado (atualizado todo mês). Além disso, Carolina também registra o supermercado onde a lista de compras será feita. Do supermercado, basta saber o nome e o endereço do mesmo.



- Observe as duas classes representadas na forma de tabela abaixo. Usando a ferramenta case crie uma superclasse que contenha os atributos comuns reformulando todas as classes.

UML 51 Exercícios - Ana Cristina Melo.pdf - Adobe Acrobat Reader DC (64-bit)

Arquivo Editar Visualizar Assinar Janela Ajuda

Início Ferramentas UML 51 Exercícios - ... Lista de exercícios ...

27 / 129 125%

a) Foi incluído o atributo *listaMateriaisAlocados*, que está em negrito. Além disso, foi criada a classe *Material*.

b) Uma alocação de sala poderá ter, também, uma lista de materiais alocados. Isso cria o relacionamento entre a classe *AlocacaoSala* e *Material*, representada pelo atributo *listaMateriaisAlocados*.

### 16. EXERCÍCIO: HERANÇA

Observe as duas classes adiante:

| Classe             | Atributos  | Métodos  |
|--------------------|--|--|
| <b>Funcionario</b> | matricula : integer<br>nome : string<br>dataNascimento : date<br>/endereco : Classe Endereco<br>/telefoneContato : Coleção de Telefone<br>/cargo : Classe Cargo<br>salario : real<br>dataAdmissao : date | cadaststrar<br>obteridade<br>reajustarSalario (percentual : real)<br>promover(novoCargo : Cargo) |

Encontrando Classes, Atributos e Relacionamentos • 23

24°C Pred ensolarado

POR PTB2

UML 51 Exercícios - Ana Cristina Melo.pdf - Adobe Acrobat Reader DC (64-bit)

Arquivo Editar Visualizar Assinar Janela Ajuda

Início Ferramentas UML 51 Exercícios - ... Lista de exercícios ...

28 / 129 125%

| Classe             | Atributos  | Métodos  |
|--------------------|--|--|
| <b>Funcionario</b> | matricula : integer<br>nome : string<br>dataNascimento : date<br>/endereco : Classe Endereco<br>/telefoneContato : Coleção de Telefone<br>/cargo : Classe Cargo<br>salario : real<br>dataAdmissao : date | cadaststrar<br>obteridade<br>reajustarSalario (percentual : real)<br>promover(novoCargo : Cargo) |

Encontrando Classes, Atributos e Relacionamentos • 23

| Classe         | Atributos   | Métodos                   |
|----------------|---|---------------------------|
| <b>Cliente</b> | codigo : string<br>nome : string<br>dataNascimento : date<br>/endereco : Classe Endereco<br>/telefoneContato : Coleção de Telefone<br>/profissao : Classe Profissao | cadaststrar<br>obteridade |

**EXERCÍCIO (A):**  
Crie uma superclasse que contenha os atributos comuns, reformulando todas as classes.

**RESOLUÇÃO:**

| Classe             | Atributos  | Métodos  |
|--------------------|--|--|
| <b>Funcionario</b> | matricula : integer<br>nome : string<br>dataNascimento : date<br>/endereco : Classe Endereco<br>/telefoneContato : Coleção de Telefone<br>/cargo : Classe Cargo<br>salario : real<br>dataAdmissao : date | cadaststrar<br>obteridade<br>reajustarSalario (percentual : real)<br>promover(novoCargo : Cargo) |

24°C Pred ensolarado

POR PTB2

- Com base na descrição a seguir e usando uma ferramenta case crie o diagrama de Casos de Uso e o Diagrama de Classes.

Uma empresa atua no ramo de aluguel de decoração de festas de aniversário infantil. Tal empresa possui vários temas e precisa controlar os aluguéis. Para isso, precisa de uma aplicação que permita cadastrar: o nome e o telefone do cliente, o endereço completo da festa, o tema escolhido, a data da festa, a hora do início e término da festa.

Além disso, para alguns clientes antigos, a empresa oferece descontos. Sendo assim, é

preciso saber o valor realmente cobrado num determinado aluguel.

Para cada tema, é preciso controlar: a lista de itens que compõem o tema (ex.: castelo, boneca Cinderela, bruxa, etc.), o valor do aluguel e a cor da toalha da mesa que deve ser usada no tema. Os itens que compõem o tema não possuem valores individuais, o valor do aluguel é fixo por tema.

Para cada aluguel feito, também é preciso controlar o valor do pagamento do sinal e a data da quitação do aluguel.

- **Interprete e descreva a arquitetura do sistema representado pelo Diagrama de Classes apresentado a seguir:**

