

UNIP
UNIVERSIDADE PAULISTA

Unified Modeling Language

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

1

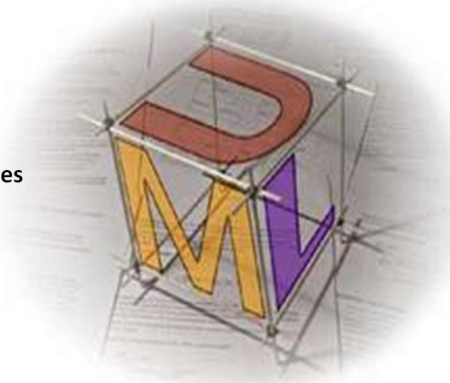
UNIP
UNIVERSIDADE PAULISTA

UML

Disciplina: **ALPOO (UML)**

Tema: **Diagrama de Classes**

Professor: **Ricardo Drudi**



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

2

UNIP
UNIVERSIDADE PAULISTA

Conceitos

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

3

UNIP
UNIVERSIDADE PAULISTA

Diagrama de Classes

- O Diagrama de Classes é a principal ferramenta de comunicação entre as equipes de Análise, Projeto e Implementação no desenvolvimento de um sistema.
- Nele são representados tanto os **objetos** reais que devem ser modelados pelo sistema quanto os objetos virtuais utilizados na solução de *software* do problema.
- É bastante comum que um Diagrama de Classes sofra alterações durante o desenvolvimento do sistema, por qualquer uma das três equipes (Análise, Projeto e Implementação).

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

4

Objetos

- É a representação de uma entidade que pode ser real ou conceitual. Pode representar alguma coisa concreta (uma cupom fiscal) ou um conceito (uma transação bancária).
- Um objeto tem definidas as seguintes características:
 - **Estado:** define uma das possíveis condições nas quais um objeto pode existir.
 - **Comportamento:** determina a maneira pela qual um objeto responde a mensagens de outros objetos, determinando cada ação que um objeto pode realizar.
 - **Identidade:** indica a unicidade de cada objeto em um sistema, mesmo se seu estado for idêntico ao de outro objeto

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

5

Classes

- Uma classe é uma abstração de um objeto, um modelo que define as características (**atributos**) e o comportamento (**métodos**) que todos os objetos daquela classe terão.
- Uma classe também pode ser entendida com um tipo de variável que pode conter diversos campos de tipos diferentes (atributos) e as funções que manipulam esses campos (métodos).
- Cada objeto é uma **instanciação** diferente de uma classe. Vários objetos podem ser da mesma classe (terem a mesma estrutura), porém um objeto só pode ser de uma única classe (ter uma única estrutura).

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

6

Diagrama de Classes

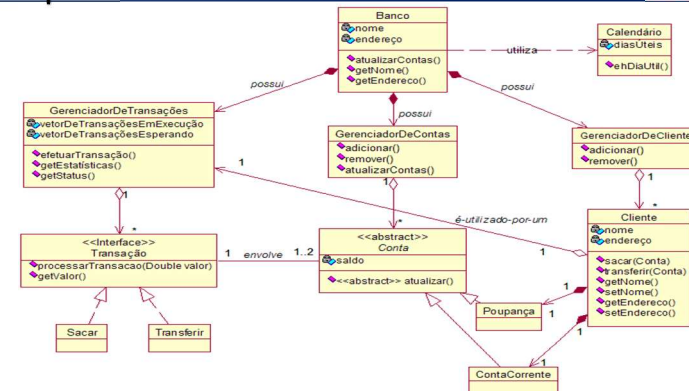
- O Diagrama de Classes é a representação gráfica das classes de um sistema, contendo seus atributos, seus métodos, e suas relações com outras classes.
- As estruturas e as relações existentes entre as classes do diagrama serão as estruturas e as relações entre os objetos do sistema.
- O objetivo final das classes é servirem de modelo para os objetos que serão instanciados durante a execução do sistema. Um programa no paradigma OO é constituído de instanciações e trocas de mensagens entre os objetos que representam as entidades reais do sistema informatizado.

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

7

Exemplo



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

8

Componentes

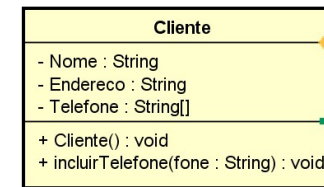
Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

9

Classes

- No Diagrama de Classes, cada classe é representada por um retângulo dividido em três áreas: o nome da classe, os atributos, os métodos.



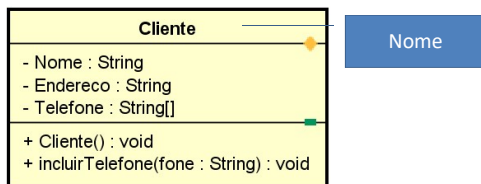
Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

10

Nome

- O nome da classe deve identificar a entidade que será modelada por aquela classe.
- Usualmente são substantivos simples, escrito no singular, e com a primeira letra maiúscula.
- O nome de uma classe deve ser único por pacote no sistema.



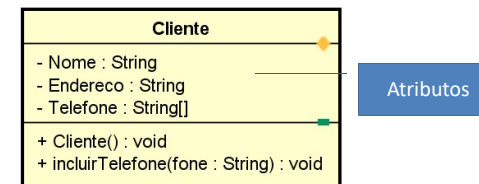
Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi

11

Atributos

- Um atributo é uma propriedade, um estado, uma característica do objeto, cujo domínio é definido pelo tipo de variável do atributo.
- Uma classe pode ter qualquer quantidade de atributos, inclusive nenhum.



Unified Modeling Language
Diagrama de Classes

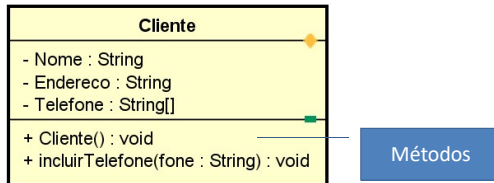
Prof. Ricardo Drudi

12

Métodos



- Um método é uma funcionalidade do objeto, uma operação que ele tem a capacidade de realizar.
- Os objetos comunicam-se entre si através de mensagens, que nada mais são do que as chamadas aos seus métodos.
- Um método tem uma assinatura, que define seus parâmetros e seu tipo de retorno.



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
13

Classes Especiais



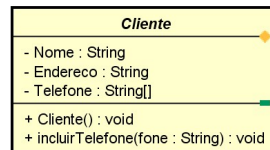
Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
14

Classe Abstrata



- Uma classe abstrata não pode ser instanciada, sendo geralmente criada para reutilização de código.
- Classes abstratas são geradas através do processo de generalização de classes concretas.
- No Diagrama de Classes são representadas pelo nome em *itálico*.



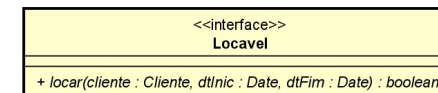
Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
15

Interfaces



- *Interfaces* são coleções de métodos abstratos, que devem ser implementados nas classes que realizam a *interface*.
- Não são classes propriamente ditas, pois não podem ter atributos nem métodos concretos.
- Sua função em um sistema OOP é padronizar a interface pública da classe, ou seja, quais métodos a classe colocará a disposição das demais classes do sistema.
- São representadas pela tag **<<interface>>** logo acima do nome.



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
16

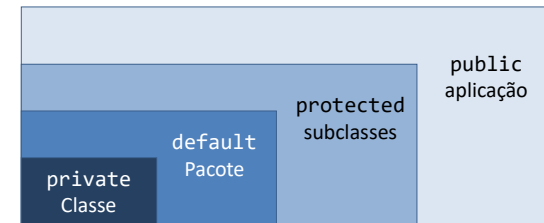
Modificadores de Visibilidade

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
17

Visibilidade

- Na POO pode ser definido o nível de acessibilidade de atributos e métodos → visibilidade
- A visibilidade de um atributo é o mesmo tanto para leitura quanto alteração do valor do atributo



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
18

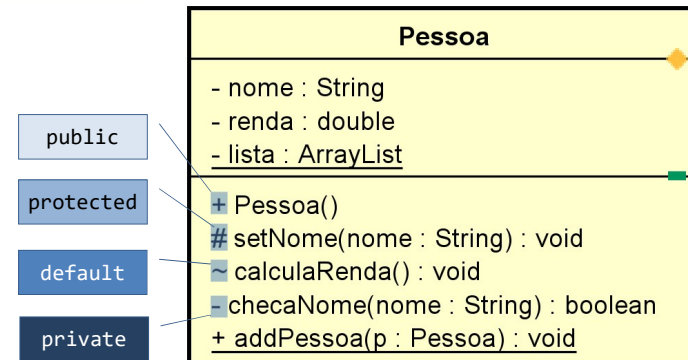
Visibilidade

Modificador	Classe	Pacote	Subclasse	Outros
public	Sim	Sim	Sim	Sim
protected	Sim	Sim	Sim	Não
default	Sim	Sim	Não	Não
private	Sim	Não	Não	Não

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
19

Visibilidade



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
20

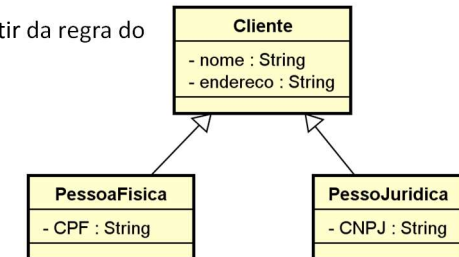
Relações

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
21

Herança

- Herança é um tipo de relação de generalização/especialização, onde uma classe (subclasse) pode derivar de uma outra classe (superclasse), herdando desta os atributos e métodos implementados
- Heranças são criadas a partir da regra do **É UM**
- No UML:



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
22

Associação

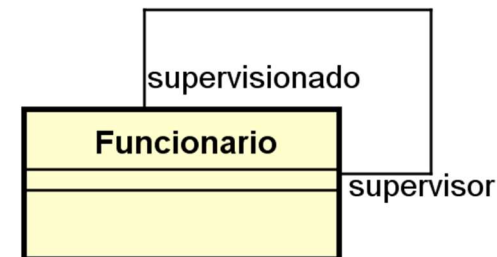
- É uma relação onde um objeto mantém um vínculo com outro objeto
- Geralmente o objeto associação é um atributo no objeto que o associa
- Associações podem ser: Unárias / Binárias / Múltiplas
- Associações podem utilizar texto para identificar o tipo de relacionamento entre as classes
- Associações têm cardinalidade e navegabilidade
- No UML:

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
23

Associação Unária

- É quando um objeto está associado a outro objeto da mesma classe



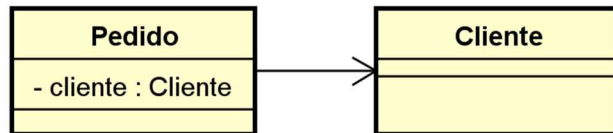
Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
24

Associação Binária



- É o tipo mais comum de associação
- Utilizada quando um objeto é um componente de outro objeto



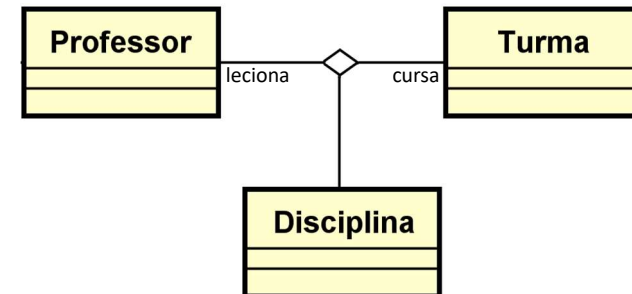
Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
25

Associação Múltipla



- Ocorre quando há mais de 2 objetos relacionados



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
26

Cardinalidade



- Determine quantos objetos de uma classe podem estar relacionados a um único objeto da outra classe

Indicadores			
1	Exatamente um	0..1	Zero ou um (opcional)
0..* ou *	Zero ou mais	1..*	Um ou mais
2..8	Faixa específica	2,4,6,8	Valores específicos



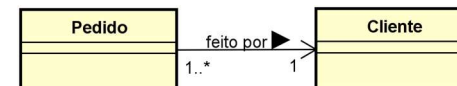
Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
27

Navegabilidade



- Define se um objeto terá conhecimento de com qual instância do outro objeto ele está associado



O Pedido sabe quem é o Cliente, porém o Cliente não identifica o Pedido.



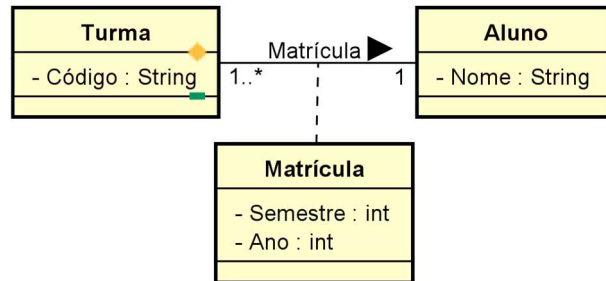
Neste caso, tanto o Cliente quanto o Endereco identificam com quem estão associados

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
28

Classe Associativa

- São classes que não representam objetos reais do modelo, mas que são necessárias para o sistema computacional

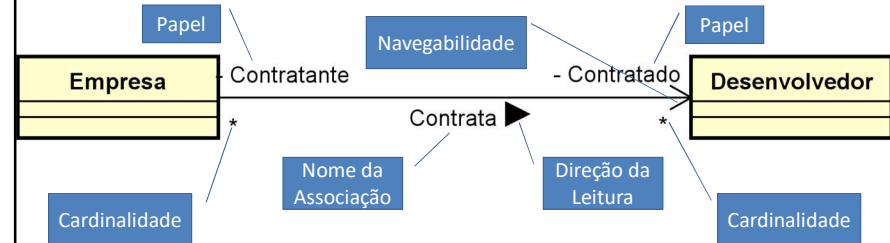


Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
29

Informação Textual

- Utilizada para auxiliar na compreensão do tipo de associação entre as classes do relacionamento.



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
30

Agregação e Composição

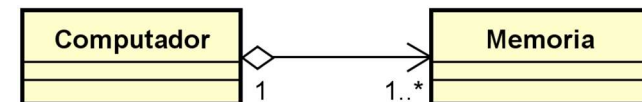
- São casos particulares de Associações, advindos de relacionamentos do tipo **todo-parte**, nos quais um objeto faz parte de outro.
- Agregação**: o objeto associado existe como entidade própria, podendo, inclusive, ser associado a outro objeto. Exemplo: Computador → Memória.
- Composição**: o objeto associado não tem existência fora da associação. Exemplo: Pedido → ItemDoPedido.
- A diferenciação entre Agregação e Composição é apenas semântica da própria UML, não havendo suporte para essa diferenciação na Orientação a Objetos.

Unified Modeling Language
Diagrama de Classes

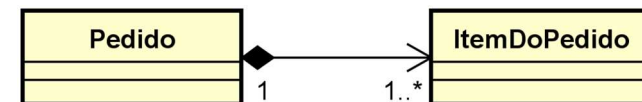
Prof. Ricardo Drudi
31

Agregação e Composição

- Agregação**: losango aberto



- Composição**: losango fechado.



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
32

Agregação e Composição



- Regra prática para saber se o relacionamento entre duas classes A e B é uma Associação simples, uma Agregação ou uma Composição:
 - 1) A é composto por B (B é parte de A)? Se não, a relação é uma Associação simples.
 - 2) Se A for excluído, B também deve ser excluído? Se sim, é Composição. Se não, é Agregação;
- Defina as relações:
 - Playlist → Música
 - Combo1 → BigMac
 - Cliente → Endereço

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
33

Dependência



- Relacionamento entre classes do tipo Cliente → Fornecedor.
- A classe Fornecedor disponibiliza uma funcionalidade que é utilizada pela classe Cliente.
- A classe Fornecedor tem existência independente da classe Cliente. A classe Cliente, por sua vez, depende da existência da classe Fornecedor.
- Uma alteração na classe Fornecedor pode levar a erros na classe Cliente.
- No UML: ----->

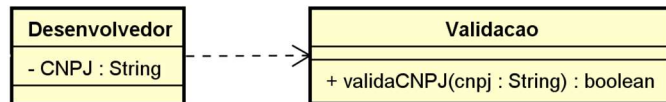
Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
34

Dependência



- A classe Desenvolvedor tem um campo CNPJ, cujos dígitos de verificação devem ser validados. Essa classe utiliza um método disponibilizado por outra classe, Validacao, que recebe um CNPJ e retorna se o dígito é válido ou não. A classe Validacao é fornecedora de recursos para a classe Desenvolvedor.



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
35

Exercícios

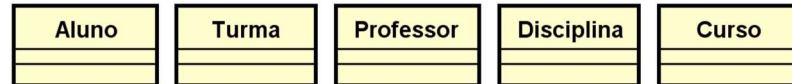


Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
36

Exercício

- 1) A partir das classes definidas abaixo, crie um Diagrama de Classes que identifique as relações existentes no modelo.



Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
37

Exercício

- 2) Considere as relações entre os objetos abaixo. Identifique o tipo de relação, sua cardinalidade e navegabilidade.
- a) Time de Futebol – Jogador
 - b) Empresa – Funcionário – Pessoa
 - c) Grupo de Trabalho – Aluno
 - d) Nota Fiscal – Itens da Nota Fiscal – Produtos – Departamento
 - e) Prova - Questão
 - f) Conta Bancária – Transação
 - g) Sistema – DBMS

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
38

Exercício

- 3) A partir do DCU criado para o problema abaixo, crie um Diagrama de Classes correspondente.

Cliente	Vendas	Financeiro
Perguntar sobre pacotes disponíveis	Fornecer informações sobre o pacote de viagens	Receber o pagamento de um pacote de viagens
Saber os preços dos pacotes	Vender um pacote de viagens	No caso de cancelamento do pacote, restituir o valor pago ao cliente
Obter informações detalhadas sobre os pacotes	Cadastrar o cliente no sistema, tanto na hora da venda ou somente para contatos futuros	Cobrar as parcelas do pacote que estão em atraso
Comprar um pacote	Cancelar uma viagem	
Pagar o pacote de viagem		
Cancelar uma viagem		

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
39


Exercício



- 4) Crie um Diagrama de Classes que modele o sistema de matrículas da sua universidade. Considere o uso que você faz do sistema e projete um modelo de *software* que ofereça os recursos que você acessa.

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
40

Obrigado.



 PROGRAMADOR RAIZ	 PROGRAMADOR NUTELLA
<ul style="list-style-type: none">- Alterava código em produção- Renomeava para <code>_old</code>- Testava com print- Tinha e-mail no BOL- Subia o site com FTP linha de comando<ul style="list-style-type: none">- Usava XP SP2- Usava VI- Se reunia com a equipe para programar- Usava metodologia gohorse, mas entregava	<ul style="list-style-type: none">- Passa por 15 ambientes de homologação e não implanta- Usa Git e ainda perde informações- Gasta 90% do orçamento criando testes- Tem e-mail no Gmail- Sobe projeto na nuvem<ul style="list-style-type: none">- Usa MacOS- Usa IDEs cheias de frescuras- Tem a sua equipe toda no Slack- Usa 3 métodos ágeis e atrasa a entrega

Copyright © 2018 Prof. Ricardo Drudi

Todos direitos reservados. A reprodução ou divulgação total ou parcial deste documento é expressamente proibida sem o consentimento formal, por escrito, do professor Ricardo Drudi.

Unified Modeling Language
Diagrama de Classes

Prof. Ricardo Drudi
41