

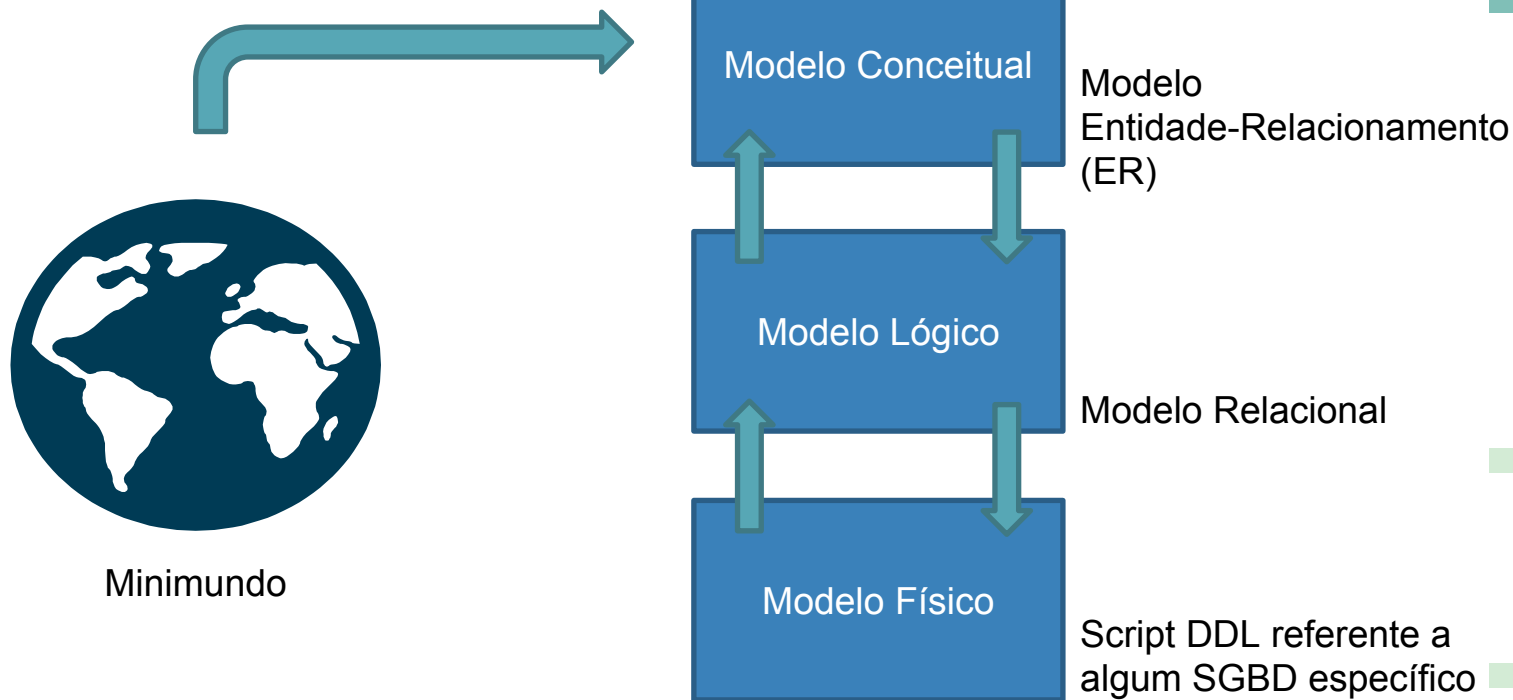
# Banco de Dados I 2018.2

Rebeca Barros  
16/11/2018

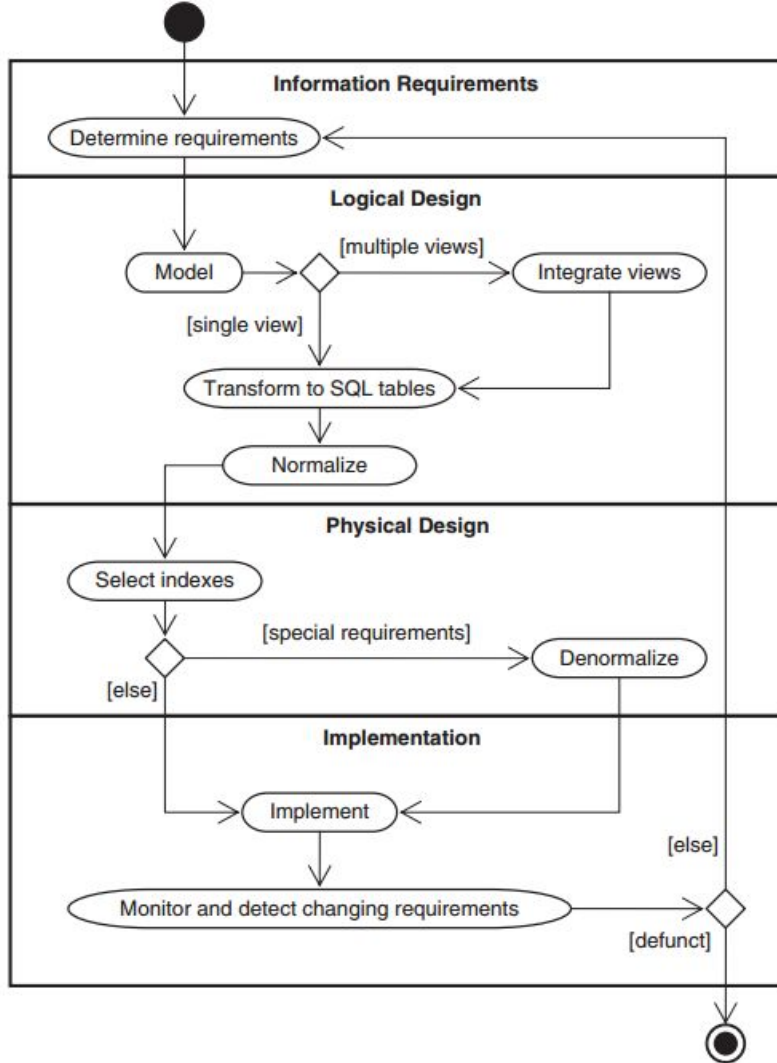


# Projeto de Banco de Dados

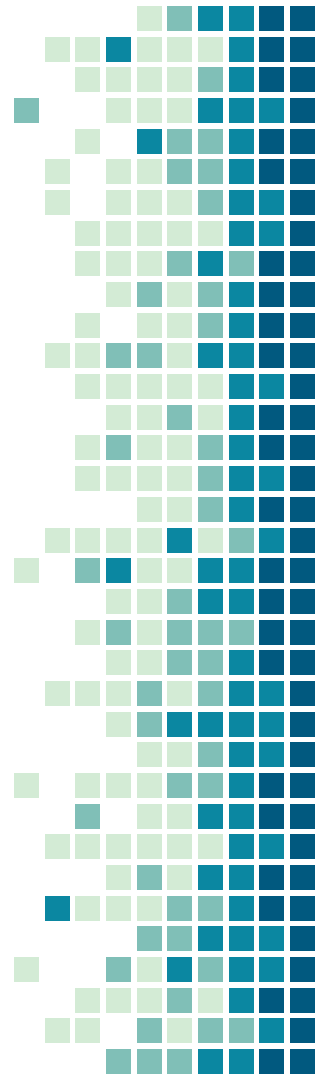




## Projeto de Banco de Dados



Ciclo de vida do Banco de Dados



# MINIMUNDO

- Porção da realidade, captada pelo analista, que a gestão de negócios de uma organização tem interesse em observar, controlar e administrar.



# Modelo Conceitual

- Representa e descreve a realidade do ambiente do problema. Visão global dos principais dados e seus relacionamentos.
- Completamente independente dos aspectos de implementação tecnológica

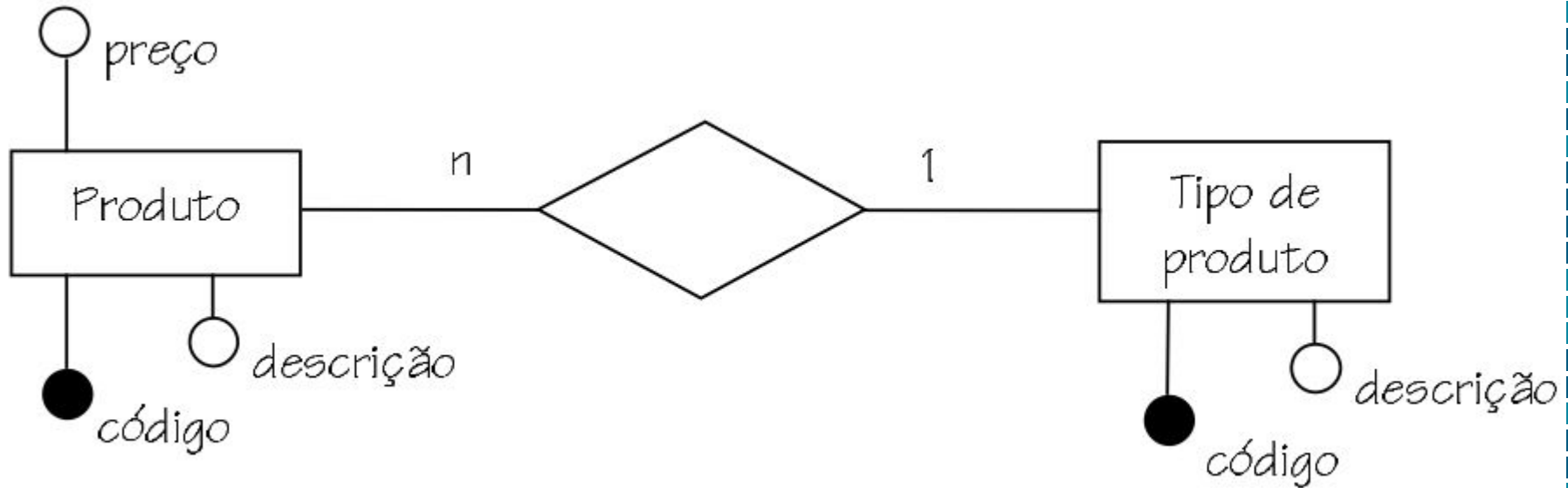


# Modelo Conceitual

- Primeira etapa do projeto de banco de dados.
- Descreve de forma simples e facilmente compreendida pelo usuário final as informações do contexto de um negócio.
- Preocupa-se em retratar a realidade de uma organização.



# Modelo Conceitual





# Modelo Lógico

- Estruturação e estabelecimento da lógica dos relacionamentos existentes entre os dados definidos no modelo conceitual.
- Adoção de alguma abordagem (relacional, hierárquica, rede ou orientada a objetos, por exemplo)



# Modelo Lógico

- Descreve as estruturas que estarão no banco de dados de acordo com a abordagem.
- Não considera nenhuma características específica de um SGDB.



# Modelo Lógico

- Forma textual do Modelo Lógico

TipoDeProduto(CodTipoProd,DescrTipoProd)

Produto(CodProd,DescrProd,PrecoProd,CodTipoProd)

CodTipoProd referencia TipoDeProduto

# Modelo Lógico

TipoDeProduto

<b>CodTipoProd</b>	<b>DescrTipoProd</b>
1	Computador
2	Impressora

Produto

<b>CodProd</b>	<b>DescrProd</b>	<b>PrecoProd</b>	<b>CodTipoProd</b>
1	PC desktop modelo X	2.500	1
2	PC notebook ABC	3.500	1
3	Impressora jato de tinta	600	2
4	Impressora laser	800	2

# Modelo Físico

- Construído a partir do modelo lógico.
- Descreve as estruturas físicas do armazenamento de dados, como:
  - Tipo e tamanho dos campos;
  - Índices;
  - Domínio de preenchimento dos campos;
  - Exigências de conteúdo;



# Modelo Lógico

- Etapa final do projeto de banco de dados;
- Script de criação do banco de dados é criado.
  - Conjunto de comandos que serão executados pelo SGBD para criar o banco correspondente ao modelo físico.



# O Modelo Entidade-Relaciona- mento



# Modelo ER

- A técnica de modelagem conceitual mais difundida e utilizada é a abordagem entidade-relacionamento (ER), criada em 1976 por Peter Chen.
- O modelo de dados é representado através de um modelo entidade-relacionamento (modelo ER).
- Um modelo ER é representado graficamente, através de um diagrama entidade- relacionamento (DER).



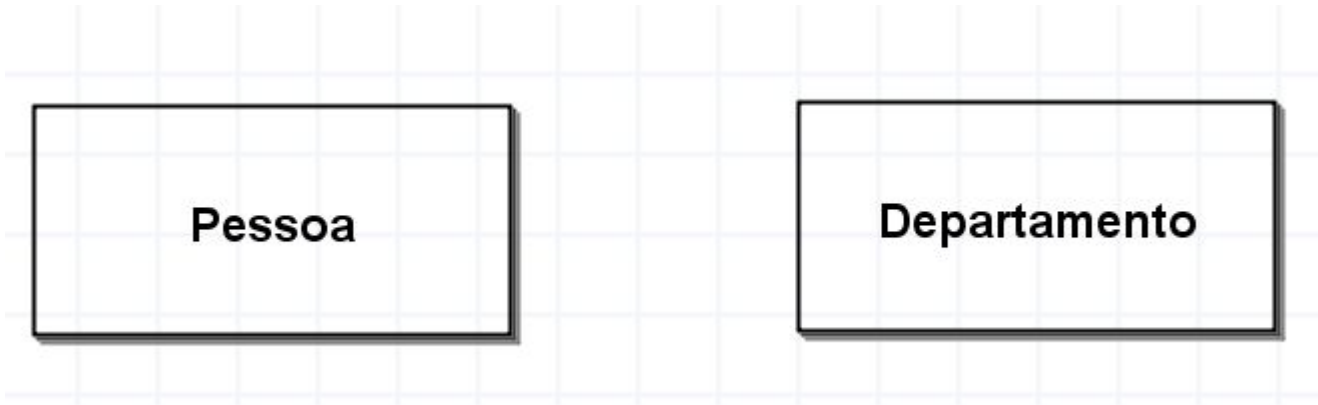
# Modelo ER - Entidades

- Objeto principal sobre o qual dados serão coletados; usualmente representam uma *pessoa*, *lugar*, *coisa* ou *evento* da qual se tem interesse.
- Uma ocorrência particular de uma entidade recebe o nome de instância.



# Modelo ER - Entidades

- Em um DER, uma entidade é representada através de um retângulo que contém o nome da entidade.



# Modelo ER - Relacionamentos

- Representam associações entre uma ou mais entidades.
- São descritos em termos de: grau, conectividade e existência



# Modelo ER - Relacionamentos

- No DER, um relacionamento é representado através de um losango, ligado por linhas aos retângulos representativos das entidades que participam do relacionamento.



# Modelo ER - Relacionamentos

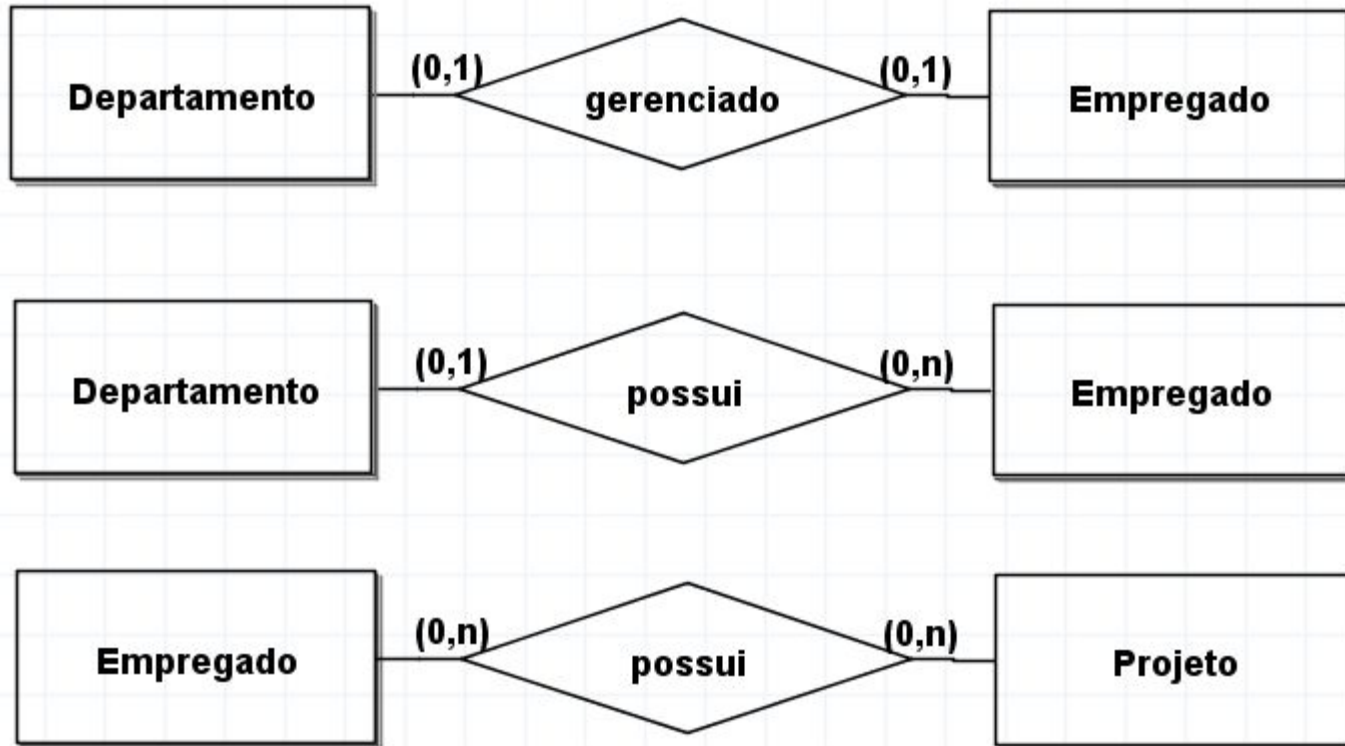
- Cardinalidade é quantas ocorrências de uma entidade podem estar associadas a uma determinada ocorrência através do relacionamento.
  - Máxima: número máximo de ocorrências
    - 1 para uma única ocorrência
    - N para mais de uma ocorrência
  - Mínima: número mínimo de ocorrências
    - 0 para relacionamento opcional
    - 1 para relacionamento obrigatório

# Modelo ER - Relacionamentos

- No DER a cardinalidade é representada com o par (mínima, máxima) próximo a linha entre o relacionamento e a entidade. Por convenção, ela vai anotada “no outro lado” do relacionamento a qual se refere. Assim, a figura abaixo indica que Departamento possui 0 ou n ocorrências de Empregado; enquanto empregado está associado a 0 ou 1 departamento.

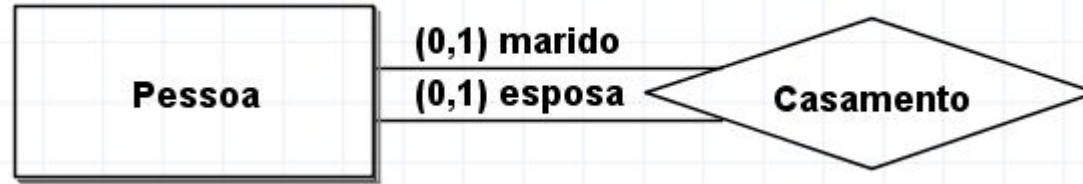


# Modelo ER - Relacionamentos



# Modelo ER - Relacionamentos

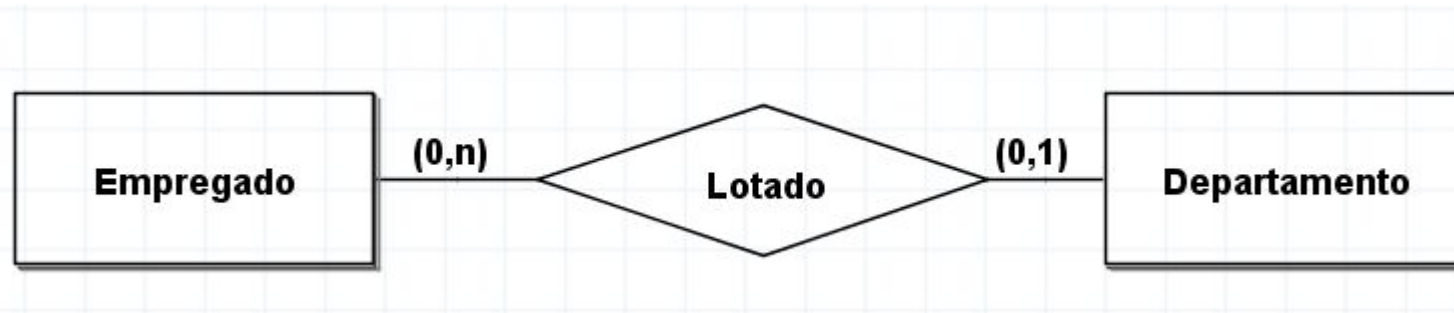
- Quanto ao grau um relacionamento pode ser:
  - Auto-relacionamento ou binário recursivo





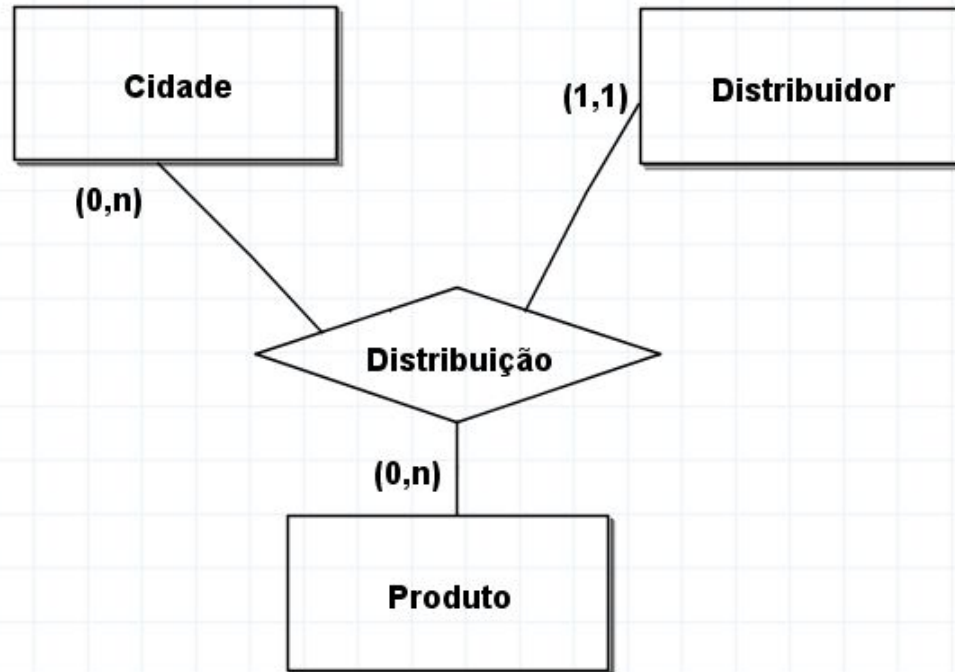
# Modelo ER - Relacionamentos

- Quanto ao grau um relacionamento pode ser:
  - Binário



# Modelo ER - Relacionamentos

- Quanto ao grau um relacionamento pode ser:
  - Ternário



# Modelo ER - Exemplo

- Crie um DER para um banco de dados de controle acadêmico de uma universidade seguindo os seguintes requisitos:
  - Deseja-se manter informações sobre alunos, cursos, disciplinas e departamentos.
  - Deseja-se manter informações sobre a associação de alunos a cursos, de disciplinas a cursos, de disciplinas a departamentos, bem como de disciplinas a suas disciplinas pré-requisitos.

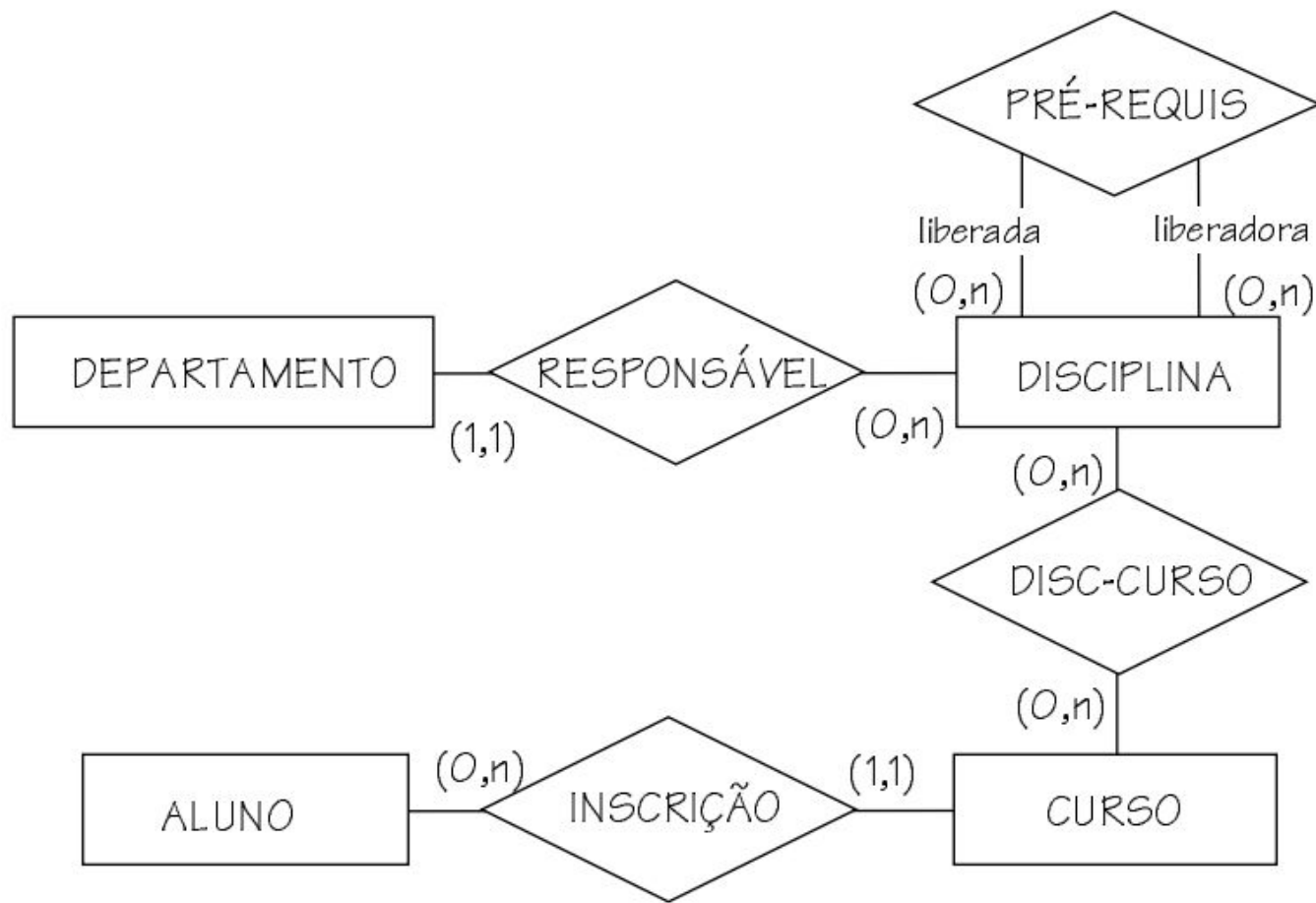
# Modelo ER - Exemplo

- Cada disciplina possui exatamente um departamento responsável, e um departamento é responsável por muitas disciplinas, inclusive por nenhuma.
- Uma disciplina pode possuir diversos pré-requisitos, inclusive nenhum. Uma disciplina pode ser pré-requisito de muitas outras disciplinas, inclusive de nenhuma.

# Modelo ER - Exemplo

- Uma disciplina pode aparecer no currículo de muitos cursos (inclusive de nenhum) e um curso pode possuir muitas disciplinas em seu currículo (inclusive nenhuma).
- Um aluno está inscrito em exatamente um curso e um curso pode ter nele inscritos muitos alunos (inclusive nenhum).





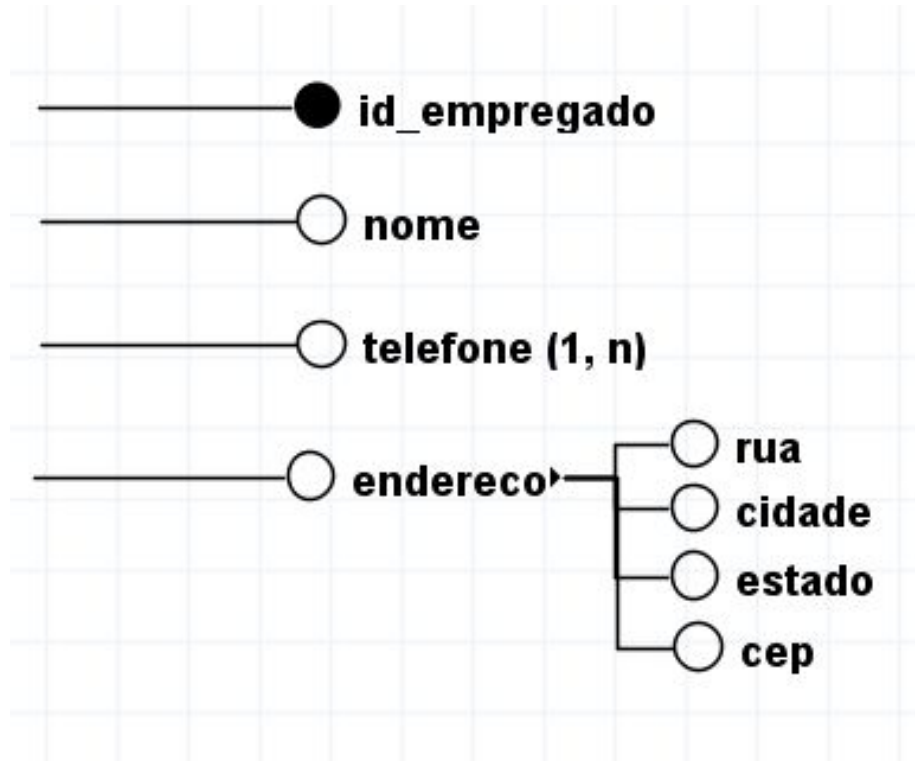
DER para controle acadêmico de uma universidade

# Modelo ER - Atributos

- Características das entidades que fornecem detalhes descritivos sobre as mesmas. Podem ser divididos em 2 tipos:
  - identificadores: usados para determinar unicamente a instância de uma entidade. Exemplos: id, cpf, rg;
  - descritores: usado para especificar uma características não única de uma instância de uma entidade. Exemplos: nome, cargo, endereço;



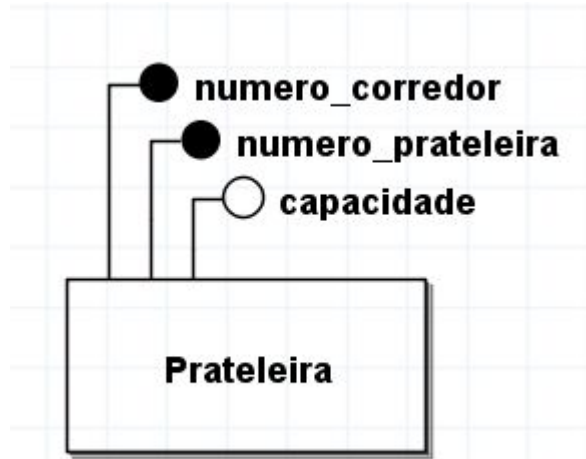
# Modelo ER - Atributos





# Modelo ER – Atributos Compostos

- O caso mais simples é a entidade possuir um único atributo como identificador.
- Mas podem ocorrer casos em que o identificador é composto por mais de um atributo.

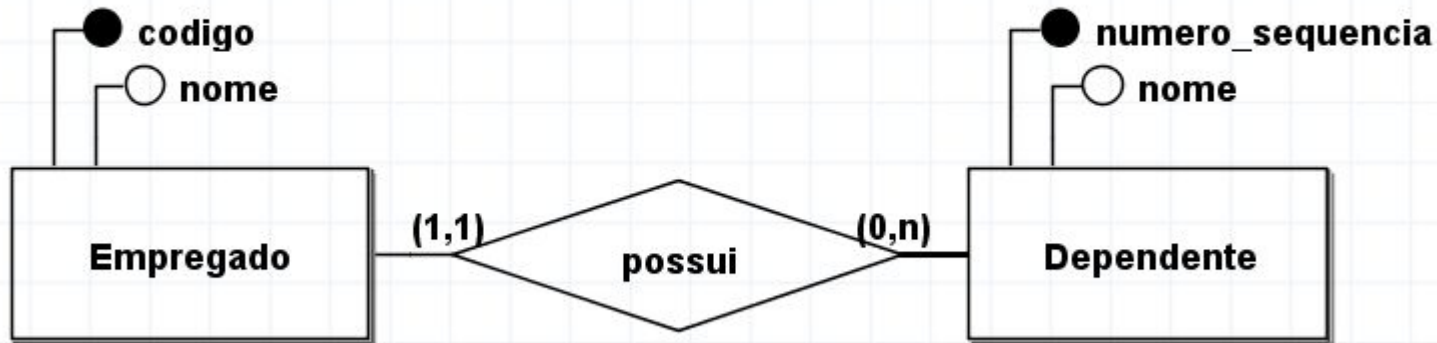


# Modelo ER – Entidade Fraca

- Há casos onde o identificador de uma entidade é composto não somente de atributos mas também por relacionamentos dos quais a entidade participa.
- É o chamado relacionamento identificador.
- No DER, o relacionamento usado como identificador é indicado por uma linha mais densa.



# Modelo ER – Entidade Fraca

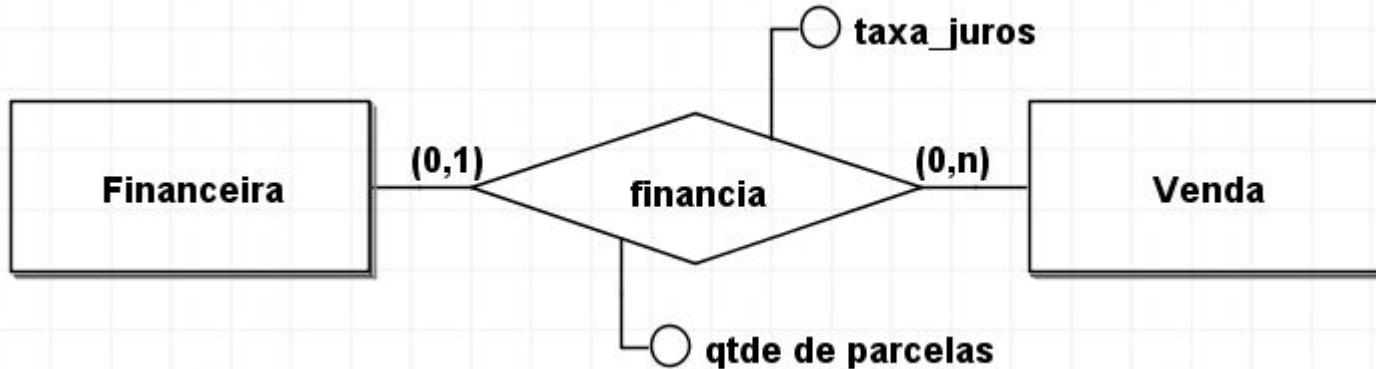


# Modelo ER - Atributos

- O identificador de uma entidade deve ser:
  - Mínimo, isso significa que retirando um de seus componentes ele deixa de ser identificador.
  - Único

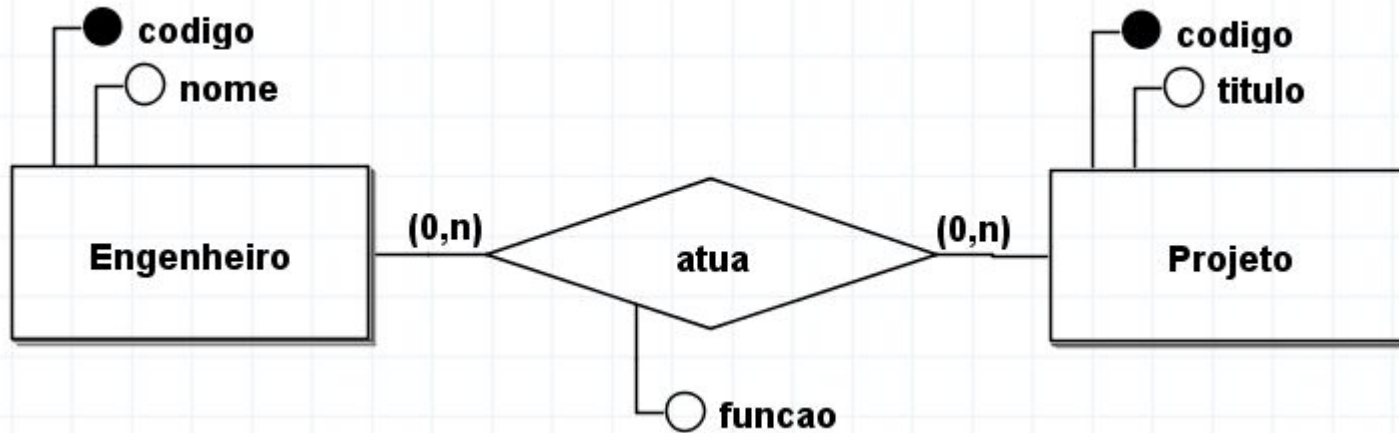
# Modelo ER - Atributos

- Assim como as entidades, relacionamentos também podem ter atributos.



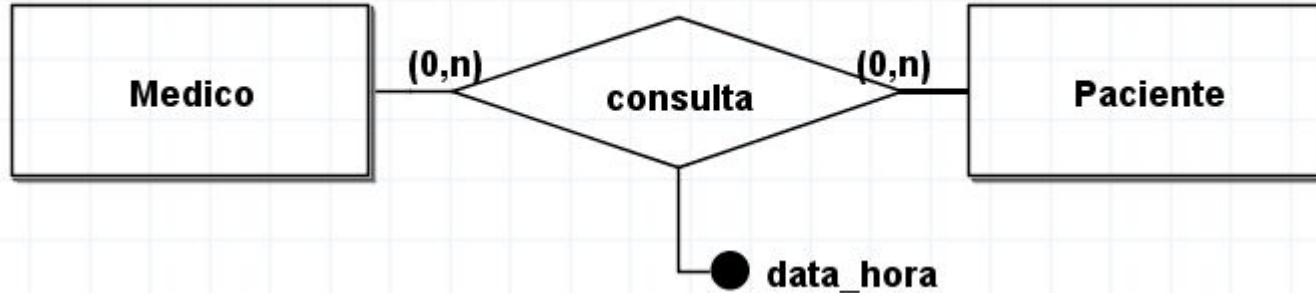
# Modelo ER - Atributos

- Assim como as entidades, relacionamentos também podem ter atributos.



# Modelo ER - Atributos

- Assim como as entidades, relacionamentos também podem ter atributos.



# O Modelo Entidade-Relaciona- mento

Conceitos Avançados





# Modelo ER –

## Generalização/Especialização

- Generalização especifica uma relação onde vários tipos de entidades com um certo número de atributos em comum podem ser generalizadas em uma entidade de nível mais alto - uma entidade genérica conhecida como entidade supertipo.
- Os níveis mais abaixo são chamados de entidades especializadas ou subtipos.

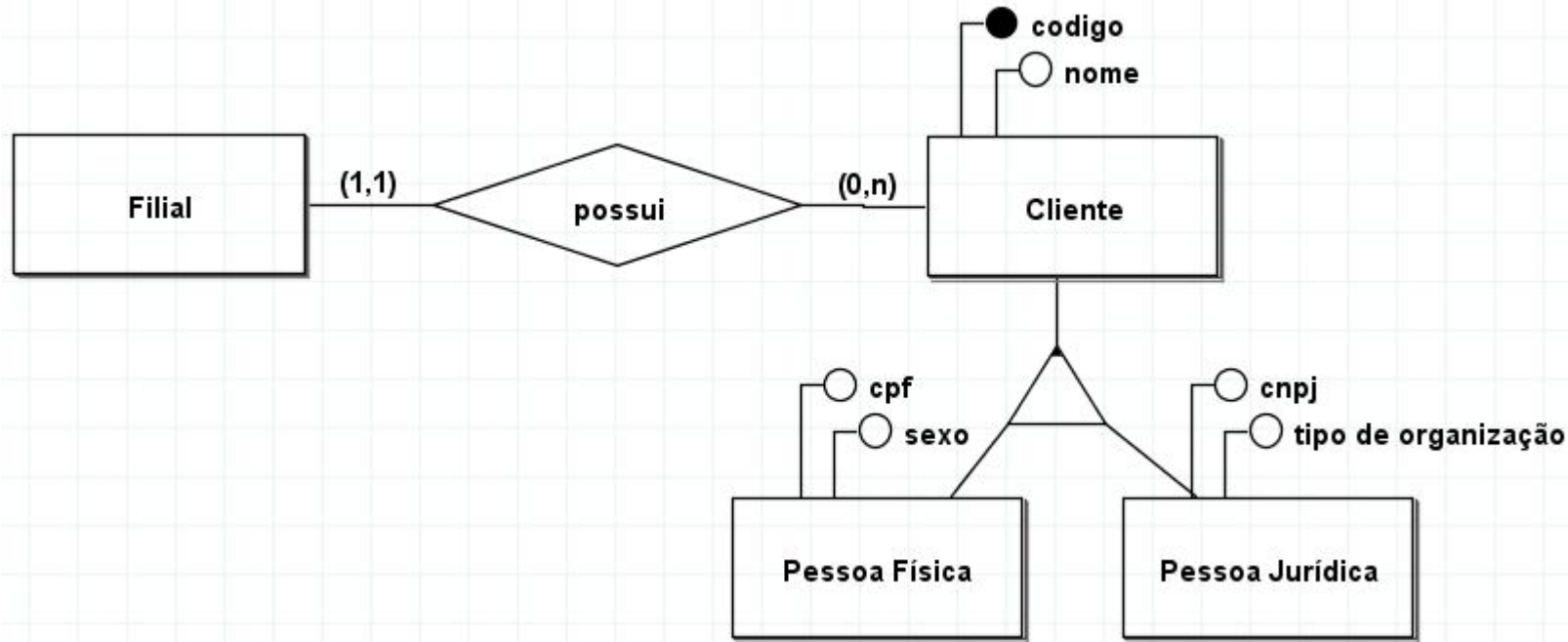


# Modelo ER – Generalização/Especialização

- O símbolo para representar generalização/especialização no DER é um triângulo isósceles.
- Generalização pode ser descrita em termos do conceito de *herança*, que especifica que todos os atributos de um supertipo são propagados para as entidades de tipo inferior.



# Modelo ER – Generalização/Especialização



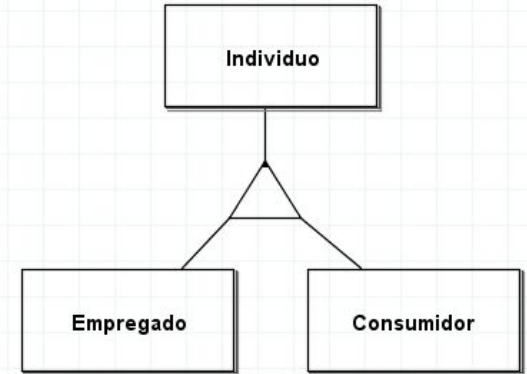
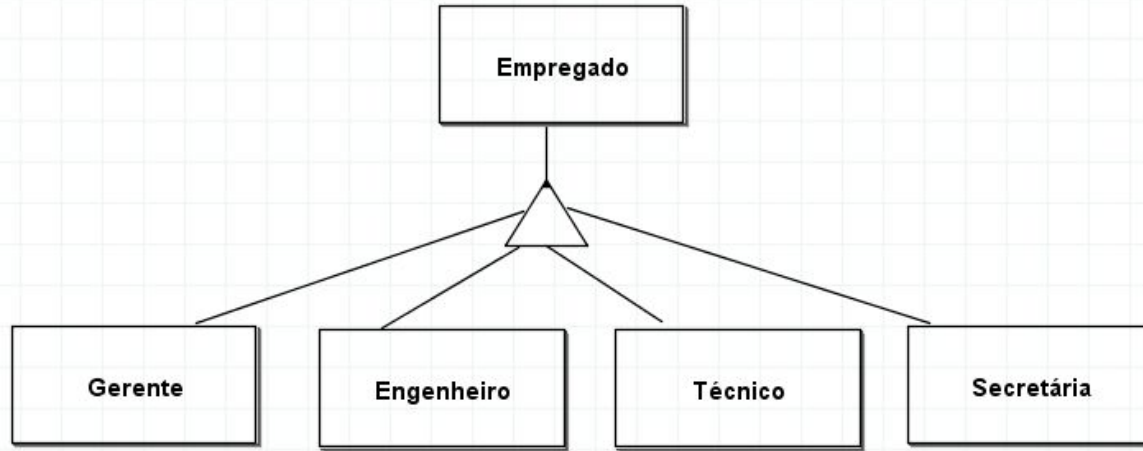
# Modelo ER –

## Generalização/Especialização

- Generalização também pode ser classificada em dois tipos de restrições importantes: *disjunção* e *completude*.
- Disjunção requer que os subtipos sejam mutualmente exclusivos, ou seja, se uma instância pertence a um desses subtipos não poderá pertencer a nenhum outro; ou podem ser sobrepostos, o que significa que uma instância pode pertencer a mais de um subtipo.



# Modelo ER – Generalização/Especialização

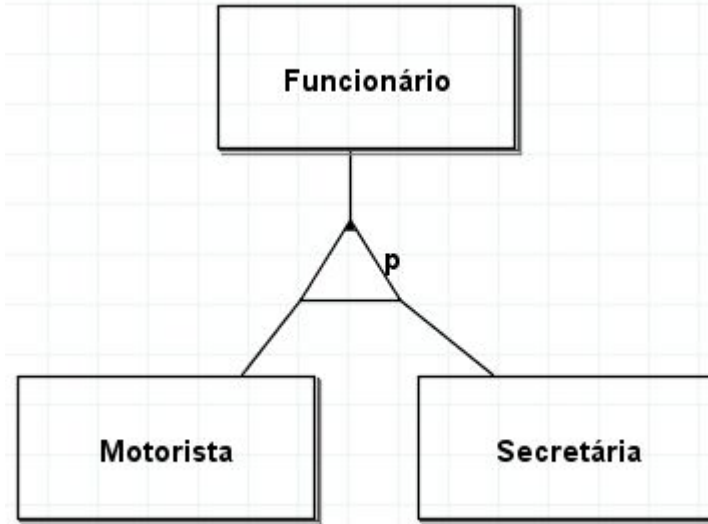
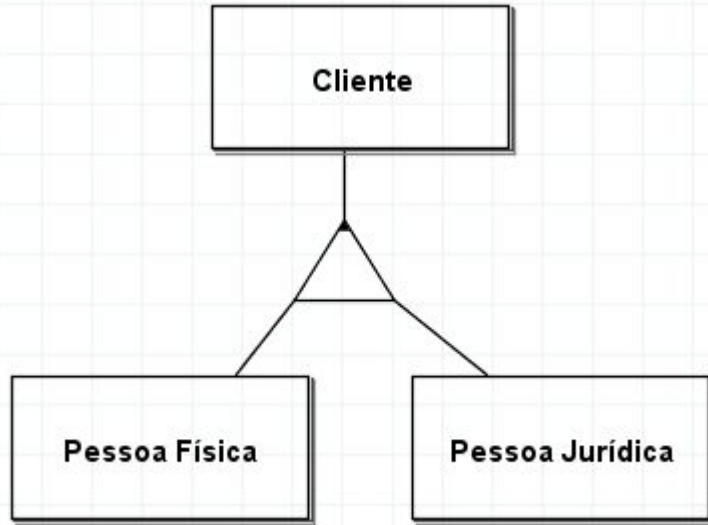


# Modelo ER – Generalização/Especialização

- Generalização também pode ser classificada em dois tipos de restrições importantes: disjunção e *completude*.
- Completude: total ou parcial. Total, quando a instância precisa ser de um das especializações; parcial, quando se admite a existência de uma instância não especializada.

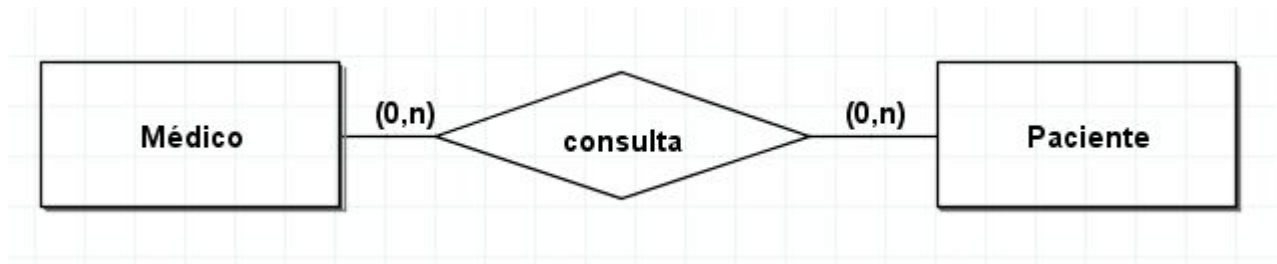


# Modelo ER – Generalização/Especialização



# Modelo ER – Entidade Associativa

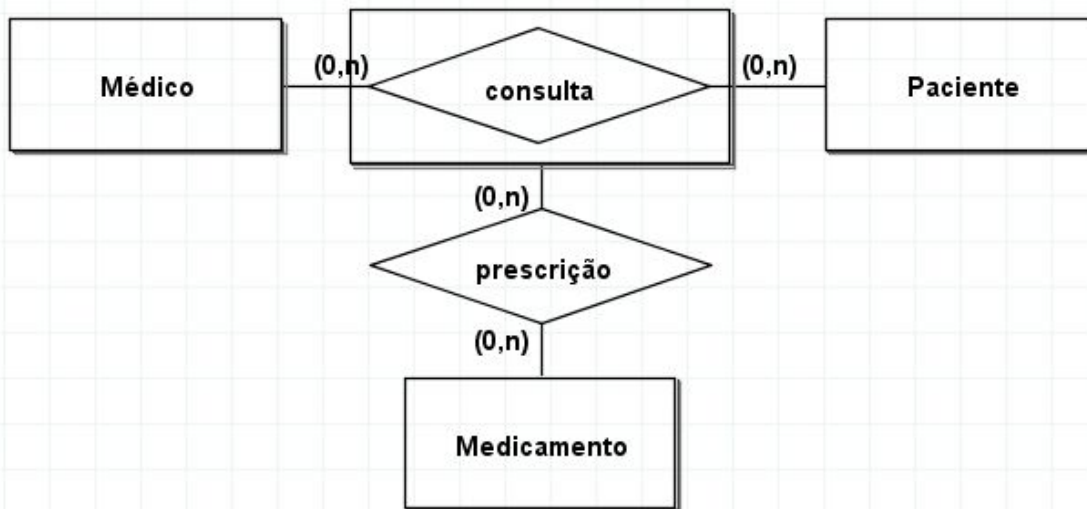
- Na modelagem ER não foi prevista a possibilidade de associar uma entidade com um relacionamento ou então de associar dois relacionamentos entre si.



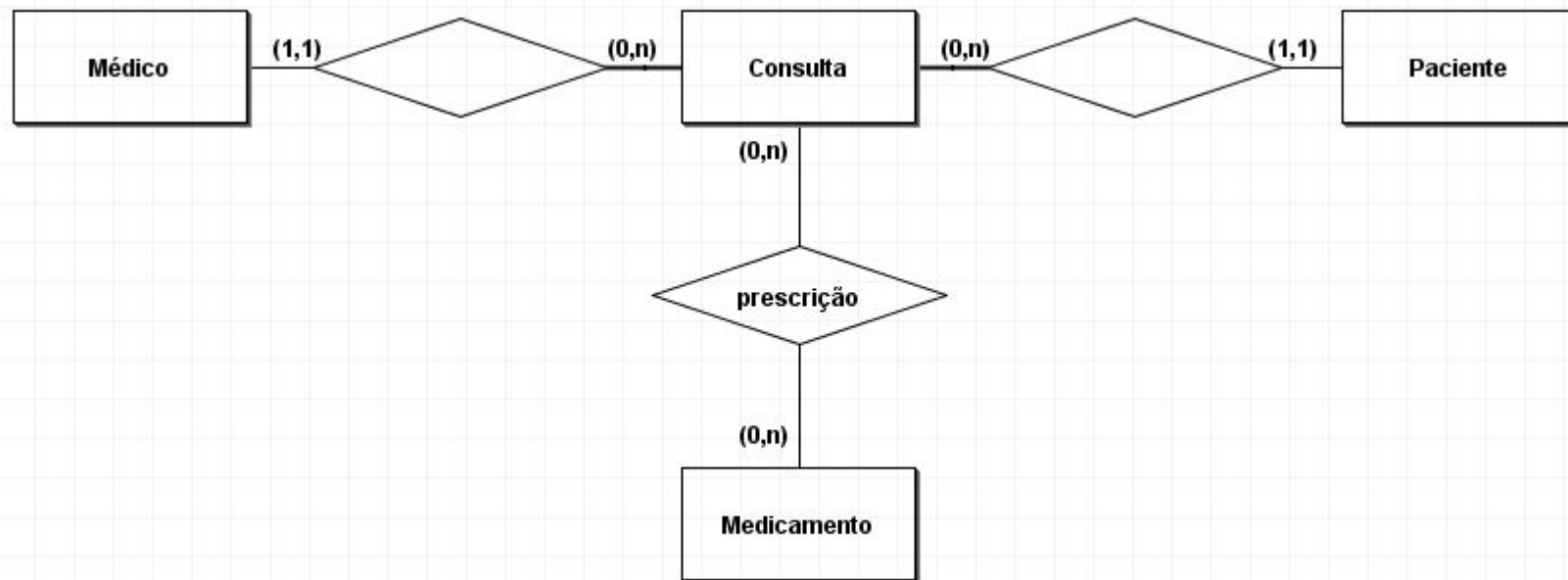


# Modelo ER – Entidade Associativa

- Para tal, foi criado um conceito especial, o de entidade associativa. Ou seja, a redefinição de um relacionamento, que passa a ser tratado como se fosse também uma entidade.



# Modelo ER – Entidade Associativa



# O Modelo Entidade-Relaciona- mento

Notação Textual



# ERD

- O utilitário erd é uma ferramenta código livre escrita em Haskell que ler uma descrição textual de entidades, atributos e relacionamentos entre essas entidades e produz um DER a partir dessa descrição.
- Disponível no link ->

<https://github.com/BurntSushi/erd>

# ERD

- Para criar sua modelagem ER usando o erd é preciso criar um arquivo de texto com a extensão .er.
- A descrição dos elementos de dados deve seguir as seguintes notações:



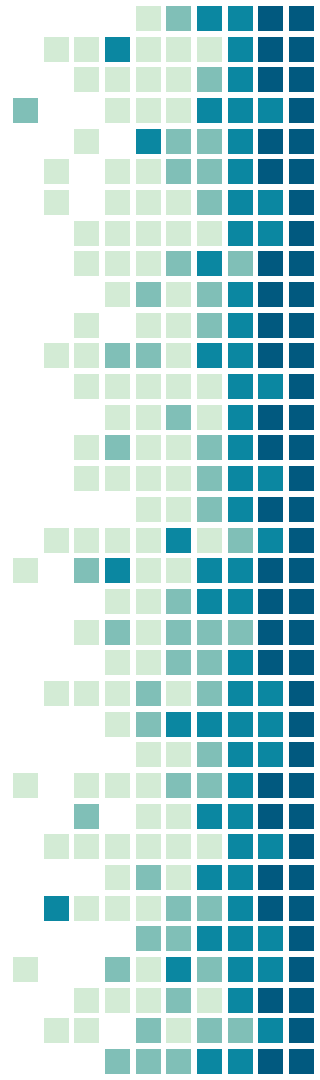
# ERD

A descrição dos elementos de dados deve seguir as seguintes notações:

Entidades são declaradas entre colchetes []:

[Pessoa]

[Local]



# ERD

- Atributos são declarados abaixo da entidade. Considera-se atributo pertencente a essa entidade todos que estiverem abaixo dela até o fim do arquivo ou até a declaração de uma nova entidade;
- Cada atributo deve ser declarado em uma linha;
- Um asterisco (\*) deve ser usado antes do nome do atributo caso ele seja o atributo identificador.

# ERD – Exemplo Entidades e Atributos

[Pessoa]

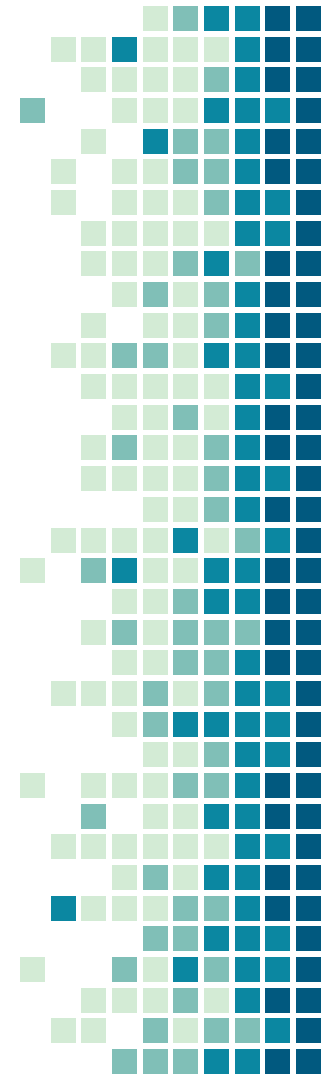
\*nome

altura

peso

[Local]

\*id





# ERD

- Relacionamentos são descritos no seguinte formato:

Entidade1 cardinalidade -- cardinalidade Entidade 2

Os símbolos que representam os 4 tipos de cardinalidade possíveis são:

Cardinalidade	Sintaxe
0 ou 1	?
Exatamente 1	1
0 ou mais	*
1 ou mais	+

# ERD – Exemplo Relacionamentos

Pessoa \*--1 Local

A linha acima indica que uma Pessoa está atrelada a apenas um Local; enquanto que um Local pode ser várias pessoas.

# ERD – Exemplo Relacionamentos

O comando {label: “nome do relacionamento”} deve ser adicionado após o nome da segunda entidade caso deseje-se atribuir um nome ao relacionamento. Assim, no exemplo anterior poderia ser:

Pessoa \*--1 Local {label: “nasce”}

Para representar que a relação entre Pessoa e Local é a de nascer.

# Exercícios



## 1 – Parte A

Construa um modelo ER para uma empresa. A empresa está organizada em Departamentos. Cada Departamento pode ter vários funcionários lotados no mesmo, sendo que cada funcionário sempre estará lotado em um único departamento. Um funcionário da empresa pode estar alocado à vários projetos ou a nenhum e em um projeto pode ter vários funcionários alocados ou mesmo nenhum funcionário alocado.



## 1 – Parte B

- Para a solução da parte A, coloque os atributos abaixo nas respectivas entidades e relacionamentos:
- data de nascimento do funcionário
- sigla do departamento
- nome do departamento
- código do projeto
- nome do projeto
- data de início da alocação do funcionário no projeto
- data de início da lotação do funcionário no departamento

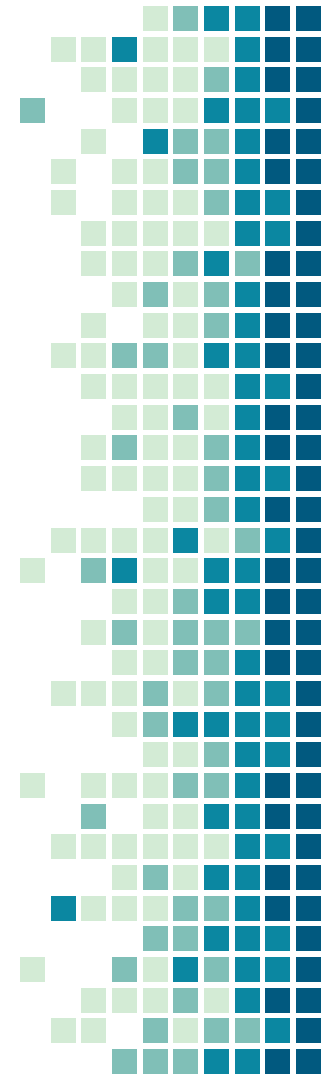


## 1 – Parte C

- Com base nas soluções encontradas nas partes A e B, defina o atributo chave de cada entidade e relacionamento. Caso ache necessário, crie novos atributos para o papel de identificador.



2 – Crie um modelo ER que represente um banco de dados para armazenar detalhes de uma biblioteca musical. Como esse banco terá informações de uma coleção pessoal, ele será simples e armazenará informações apenas entre artistas, álbuns e faixas.





Alguns requisitos são:

- Uma coleção é composta por álbuns;
- Um álbum é feito por exatamente um artista;
- Um artista pode fazer um ou mais álbuns;
- Um álbum contém um ou mais faixas;
- Artistas, álbuns e faixas, todos possuem um nome;
- Cada faixa está em apenas um álbum;
- Cada faixa tem uma duração, medida em segundos.
- Quando uma faixa é executada, a data e horário que a execução começou deve ser armazenada no banco. Essa informação é usada para saber a última vez que a faixa foi executada assim como o número de vezes que uma faixa foi executada.

