

Banco de Dados I

Professor Msc. Aparecido Vilela Junior
aparecido.vilela@unicesumar.edu.br

Exercícios- Atributos

- Uma firma vende produtos de limpeza, e deseja melhor controlar os produtos que vende, seus clientes e os pedidos.
- Cada produto é caracterizado por um código, nome do produto, categoria (ex. detergente, sabão em pó, sabonete, etc), e seu preço. A categoria é uma classificação criada pela própria firma. A firma possui informações sobre todos seus clientes. Cada cliente é identificado por um código, nome, endereço, telefone, status ("bom", "médio", "ruim"), e o seu limite de crédito.
- Guarda-se igualmente a informação dos pedidos feitos pelos clientes. Cada pedido possui um número e guarda-se a data de elaboração do pedido. Cada pedido pode envolver de um a vários produtos, e para cada produto, indica-se a quantidade deste pedida.

- Lista das entidades encontradas:
- **Produtos**
- **Pedidos**
- **Categoria**
- **Clientes**

- **Produtos (código, nome, categoria, preço)**
- **Pedidos (numero, cliente, data_pedido, (produto, quantidade_pedida)).**
- **Categoria (código, nome)**
- **Clientes (código, nome, endereço, status, limite_credito).**

Exercícios Relacionamentos

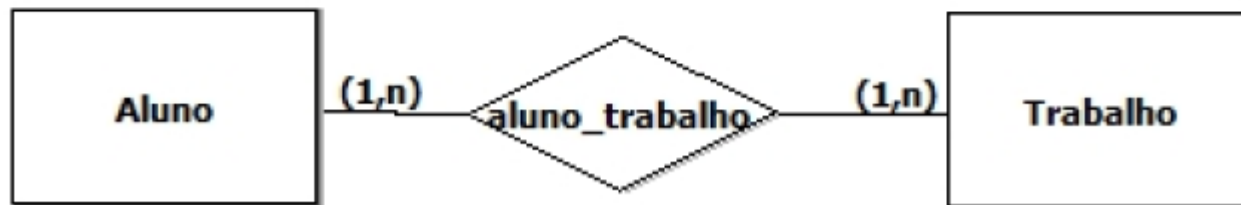
Exercícios – Relacionamentos

- 1) Um aluno realiza vários trabalhos. Um trabalho é realizado por um ou mais alunos.
- 2) Um diretor dirige no máximo um departamento. Um departamento tem no máximo um diretor.
- 3) Um autor escreve vários livros. Um livro pode ser escrito por vários autores.
- 4) Uma equipe é composta por vários jogadores. Um jogador joga apenas em uma equipe.
- 5) Um cliente realiza várias encomendas. Uma encomenda diz respeito apenas a um cliente.

Exercícios – Relacionamentos

1) Um aluno realiza vários trabalhos. Um trabalho é realizado por um ou mais alunos.

- Entidades: Aluno e Trabalho.
- Relacionamento: aluno_trabalho.
- Cardinalidades:
 - Aluno realiza vários trabalhos (1,n).
 - Trabalho é realizado por um ou mais alunos (1,n).

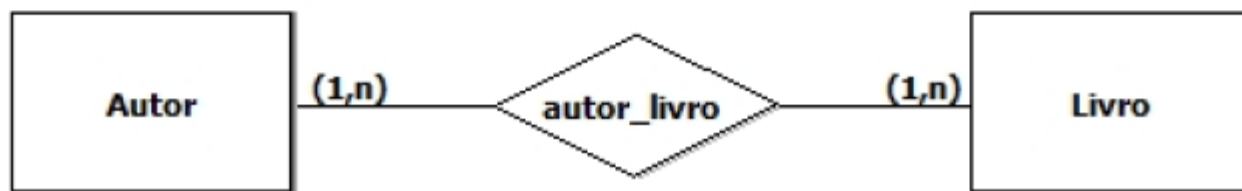


2) Um diretor dirige no máximo um departamento. Um departamento tem no máximo um diretor.

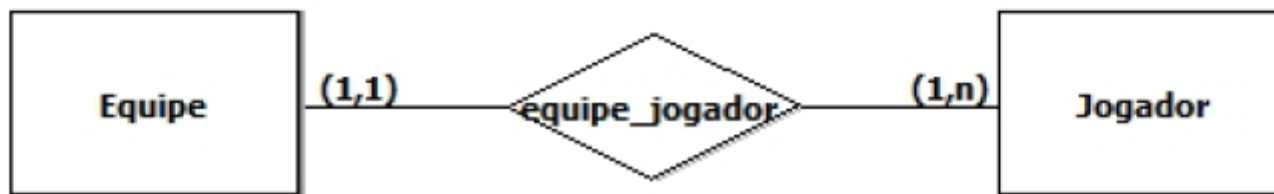
- Entidades: Diretor e Departamento.
- Relacionamento: diretor_depto.
- Cardinalidades:
 - diretor dirige no máximo um departamento (1,1).
 - Um departamento tem no máximo um diretor (1,1).



1. Um **autor** escreve vários **livros**. Um livro pode ser escrito por vários autores.
 - Entidades: Autor e Livros.
 - Relacionamento: autor_livro.
 - Cardinalidades:
 - Um autor escreve vários livros (1,1).
 - Um livro pode ser escrito por vários autores (1,1).

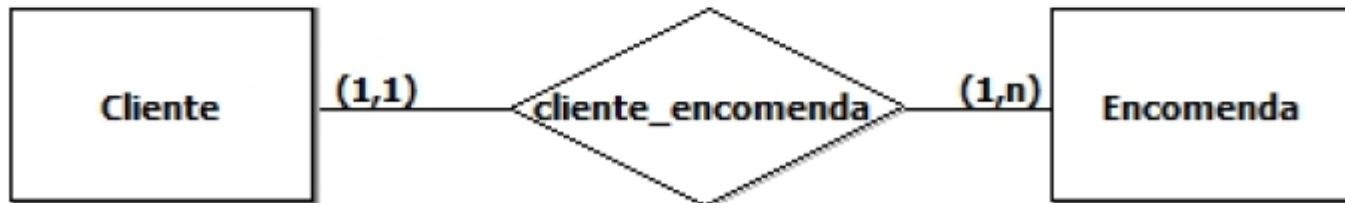


1. Uma **equipe** é composta por vários **jogadores**.
Um jogador joga apenas em uma equipe.
 - Entidades: Equipe e Jogador.
 - Relacionamento: equipe_jogador.
 - Cardinalidades:
 - Uma equipe é composta por vários jogadores (1,n).
 - Um jogador joga apenas em uma equipe (1,1).



5) Um cliente realiza várias encomendas. Uma encomenda diz respeito apenas a um cliente.

- Entidades: Cliente e Encomenda.
- Relacionamento: cliente_encomenda.
- Cardinalidades:
 - Um cliente realiza várias encomendas (1,n).
 - Uma encomenda diz respeito apenas a um cliente (1,1).



Um berçário deseja informatizar suas operações. Quando um bebê nasce, algumas informações são armazenadas sobre ele, tais como: nome, data do nascimento, peso do nascimento, altura, a mãe deste bebê e o médico que fez seu parto.

Para as mães, o berçário também deseja manter um controle, guardando informações como: nome, endereço, telefone e data de nascimento.

Para os médicos, é importante saber: CRM, nome, telefone celular e especialidade.

Quais Entidades encontrou?

Quais Atributos dessas entidades?

Quais relacionamentos?

Um berçário deseja informatizar suas operações. Quando um bebê nasce, algumas informações são armazenadas sobre ele, tais como: nome, data do nascimento, peso do nascimento, altura, a mãe deste bebê e o médico que fez seu parto.

Para as mães, o berçário também deseja manter um controle, guardando informações como: nome, endereço, telefone e data de nascimento.

Para os médicos, é importante saber: CRM, nome, telefone celular e especialidade.

Quais Entidades encontrou?

Quais Atributos dessas entidades?

Quais relacionamentos?

Liste das entidades encontradas

Bebê (nome, data do nascimento, peso do nascimento, altura, a mãe, médico)

Mãe (nome, endereço, telefone e data de nascimento)

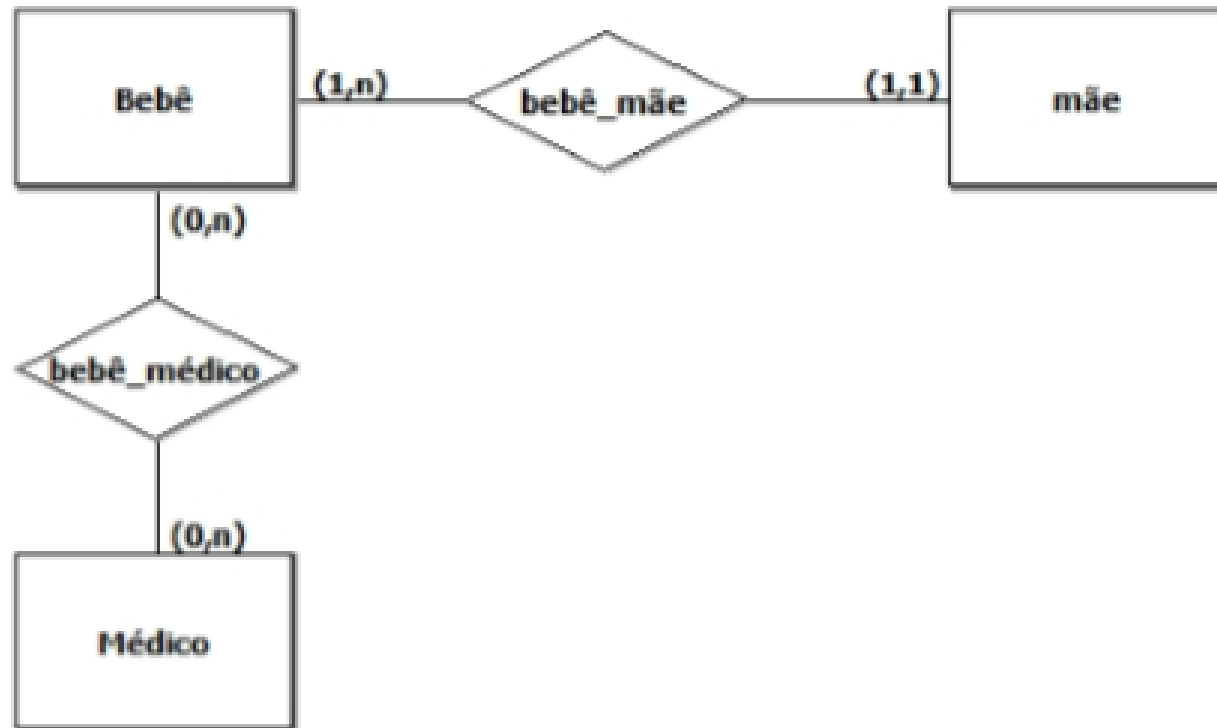
Médico (CRM, nome, telefone celular e especialidade)

Liste os relacionamentos encontrados

mãe_bebê: relacionamento entre Mãe e Bebê.

Médico_bebê: relacionamento entre Médico e Bebê

- Modelo Entidade Relacionamento



Uma biblioteca deseja manter informações sobre seus livros. Inicialmente, quer armazenar para os livros as seguintes características: ISBN, título, ano editora e autores deste livro. Para os autores, deseja manter: nome e nacionalidade. Cabe salientar que um autor pode ter vários livros, assim como um livro pode ser escrito por vários autores. Cada livro da biblioteca pertence a uma categoria. A biblioteca deseja manter um cadastro de todas as categorias existentes, com informações como: código da categoria e descrição. Uma categoria pode ter vários livros associados a ela.

Livro

Autor

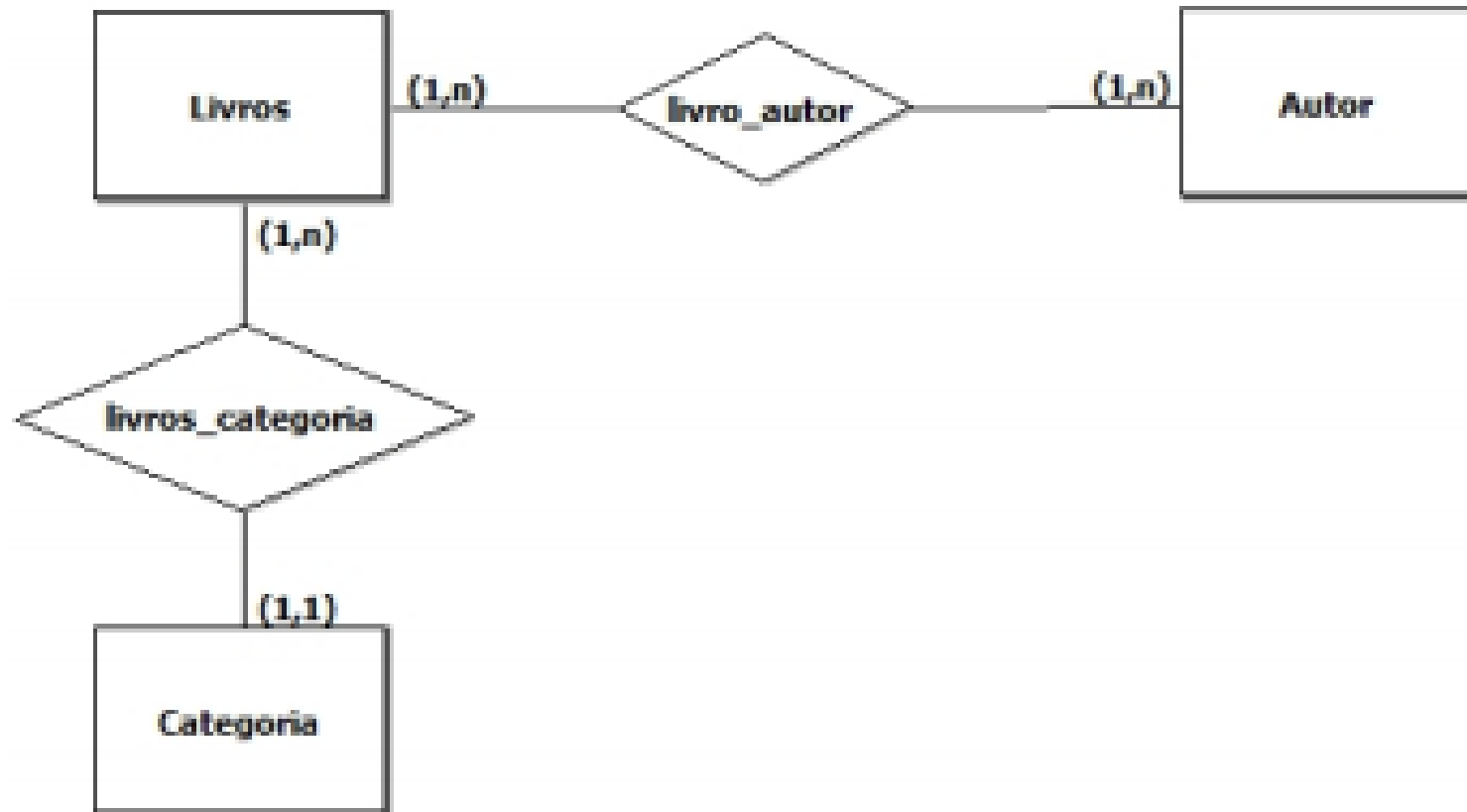
Categoria

Liste os relacionamentos encontrados

livros_autor: relacionamento entre Autor e Livros.

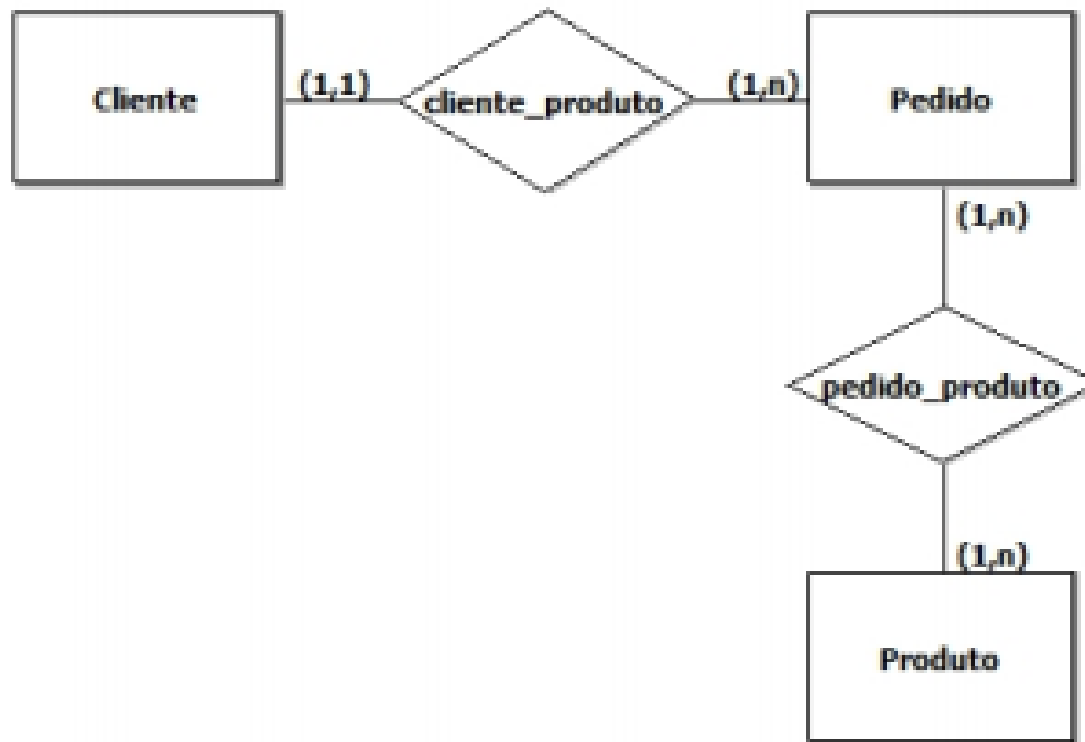
livros_categorias: relacionamento entre Livros e Categorias.

Resolução



Uma floricultura deseja informatizar suas operações. Inicialmente, deseja manter um cadastro de todos os seus clientes, mantendo informações como: RG, nome, telefone e endereço. Deseja também manter um cadastro contendo informações sobre os produtos que vende, tais como: nome do produto, tipo (flor, vaso, planta,...), preço e quantidade em estoque. Quando um cliente faz uma compra, a mesma é armazenada, mantendo informação sobre o cliente que fez a compra, a data da compra, o valor total e os produtos comprados.

- Modelo Entidade Relacionamento



Exercício – Modelo Conceitual

Uma agência de turismo deseja controlar a reserva de pacotes de viagem por seus clientes.

Cada cliente deve ser cadastrado na agência, sendo armazenados, seu CPF (usado como identificador), seu nome e seus vários telefones. A agência cadastra o maior número de telefones de contato possíveis, para cada um deles o próprio número e o tipo (comercial, residencial, celular, recados).

Para cada tipo, um cliente pode ter vários telefones.

Além dos clientes, a agência registra pacotes de viagem. Cada pacote é identificado por um código (numérico) e devem ser armazenados uma descrição do pacote (texto livre), a data de partida e a duração (em número de dias).

Finalmente, devem ser registradas as reservas para os pacotes. Cada reserva é feita por um cliente e refere-se a um pacote. É necessário saber a data em que foi feita a reserva. Um cliente pode reservar um pacote para várias pessoas (familiares, amigos, etc.). Neste caso, somente o cliente estará cadastrado, e é necessário armazenar o número de pessoas que participam da reserva.

Projete um esquema textual para a base de dados em questão. Esta base de dados deve refletir exatamente o enunciado acima e não deve conter redundâncias.

Neste esquema, deve-se apresentar o Modelo Conceitual – Entidades, Atributos e Relacionamentos.

Resposta:

Cliente (CPF, NomeCli)

Telefone(CPF, NoTel, TipoTel)

CPF referencia Cliente

Pacote(CodPac, DescricaoPac, DataPac, DuracaoPac)

Reserva

(CodCPF, CodPac, DataReserva, NoPessoasReserva)

Uma Escola tem várias turmas. Uma turma tem vários professores, sendo que um professor pode ministrar aulas em mais de uma turma. Uma turma tem sempre aulas na mesma sala, mas uma sala pode estar associada a várias turmas (com horários diferentes).

Resolução

Turmas

Professores

Aulas

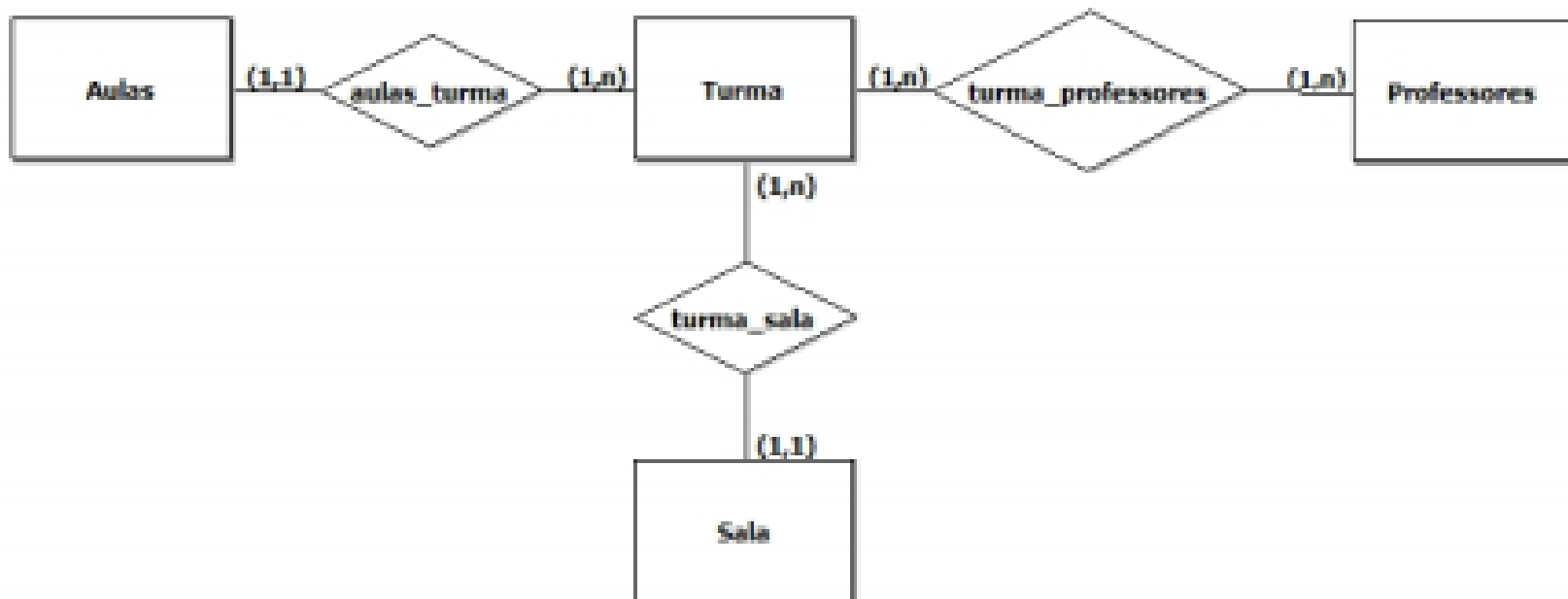
Sala

turma_professores: relacionamento em que uma turma ter vários professores.

turma_professores: relacionamento em que um professor poder ministrar aulas em mais de uma turma

turma_sala: relacionamento em que uma turma está associada a uma e somente uma sala.

Resolução



Exercício

Projete uma base de dados relacional para armazenar dados sobre pessoas ligadas ao Departamento de Informática do Unicesumar.

Cada *pessoa* é identificada por um *número de cartão* e a base de dados deve manter o *nome da pessoa*, bem como seu *sexo*. A base de dados deve manter os *e-mails* da pessoa. Cada pessoa pode ter vários e-mails, mas há somente um *e-mail preferencial* que será usado para comunicar-se com a pessoa.

Caso a pessoa for aluno do Unicesumar estará vinculada a um *curso*. Cada curso possui um *código numérico* e um *nome*. Uma pessoa pode estar vinculada a no máximo um curso.

As pessoas que trabalham em projetos de pesquisa (docentes, alunos e bolsistas) devem estar vinculadas a estes projetos. Cada *projeto* está cadastrado na base de dados com seu *código numérico* e seu *nome*. Para cada participante é necessário conhecer seu *papel no projeto* (o papel pode ser docente, aluno ou bolsistas). Observar que uma pessoa pode participar de vários projetos, inclusive com diferentes papéis.

Projete uma base de dados relacional que armazena os dados acima sem redundância de dados. Enumere as tabelas, suas colunas, as chaves primárias e as chaves estrangeiras.

Resolução

Pessoa (NumeroCartao, NomePess, Sexo, EMailPref,
CodCurso)

(CodCurso) referencia Curso

OutroEMail (NumeroCartao, EMail)

(NumeroCartao) referencia Pessoa

Curso (CodCurso, NomeCurso)

Projeto (CodProj, NomeProj)

ProjetoPessoa (CodProj, NumeroCartao, PapelPessProj)

(CodProj) referencia Projeto

(NumeroCartao) referencia Pessoa

Exercício – Modelo Conceitual

Projete uma base de dados relacional para armazenar dados sobre pessoas ligadas ao Departamento de Informática do Unicesumar.

Cada *pessoa* é identificada por um *número de cartão* e a base de dados deve manter o *nome da pessoa*, bem como seu *sexo*. A base de dados deve manter os *e-mails* da pessoa. Cada pessoa pode ter vários e-mails, mas há somente um *e-mail preferencial* que será usado para comunicar-se com a pessoa.

Caso a pessoa for aluno do Unicesumar estará vinculada a um *curso*. Cada curso possui um *código numérico* e um *nome*. Uma pessoa pode estar vinculada a no máximo um curso.

As pessoas que trabalham em projetos de pesquisa (docentes, alunos e bolsistas) devem estar vinculadas a estes projetos. Cada *projeto* está cadastrado na base de dados com seu *código numérico* e seu *nome*. Para cada participante é necessário conhecer seu *papel no projeto* (o papel pode ser docente, aluno ou bolsistas). Observar que uma pessoa pode participar de vários projetos, inclusive com diferentes papéis.

Projete uma base de dados relacional que armazena os dados acima sem redundância de dados. Enumere as tabelas, suas colunas, as chaves primárias e as chaves estrangeiras.

Resolução

Pessoa (NumeroCartao, NomePess, Sexo, EMailPref, CodCurso)

(CodCurso) referencia Curso

OutroEMail (NumeroCartao, EMail)

(NumeroCartao) referencia Pessoa

Curso (CodCurso, NomeCurso)

Projeto (CodProj, NomeProj)

ProjetoPessoa (CodProj, NumeroCartao, PapelPessProj)

(CodProj) referencia Projeto

(NumeroCartao) referencia Pessoa

Exercícios – Modelo Conceitual

Uma companhia de aviação deseja montar uma base de dados para divulgação em um site Web. A base de dados deve conter as seguintes informações.

A companhia oferece vários voos. Cada voo é identificado por um código numérico e a base de dados deve registrar o tipo de aeronave que é usada para o voo. Cada aeronave é identificada por uma sigla alfanumérica, como "B-767" e possui uma descrição. Obviamente, um tipo de aeronave pode ser usado em muitos voos.

Além dos dados acima, é necessário saber os trechos que compõe o voo.

Um voo é composto de vários trechos, numerados de um em diante, na ordem em que são voados. Para cada trecho é necessário saber o horário de saída, o horário de chegada, o aeroporto de origem e o aeroporto de destino.

Um aeroporto é identificado por uma sigla de três letras, como "MGF" ou "CWB" e tem um nome.

Projete uma base de dados relacional que armazene os dados acima sem redundância de dados.

Resolução

Voo (CodVoo,SiglaTipoAero)

CodTipoAero referencia TipoAeronave

TipoAeronave (SiglaTipoAero,DescricaoTipoAero)

Trecho (CodVoo,NumTrecho,SiglaAeroSaida,HorarioSaida,
SiglaAeroChegada,HorarioChegada)

SiglaAeroSaida referencia Aeroporto

SiglaAeroChegada referencia Aeroporto

CodVoo referencia Voo

Aeroporto (SiglaAero,NomeAero)

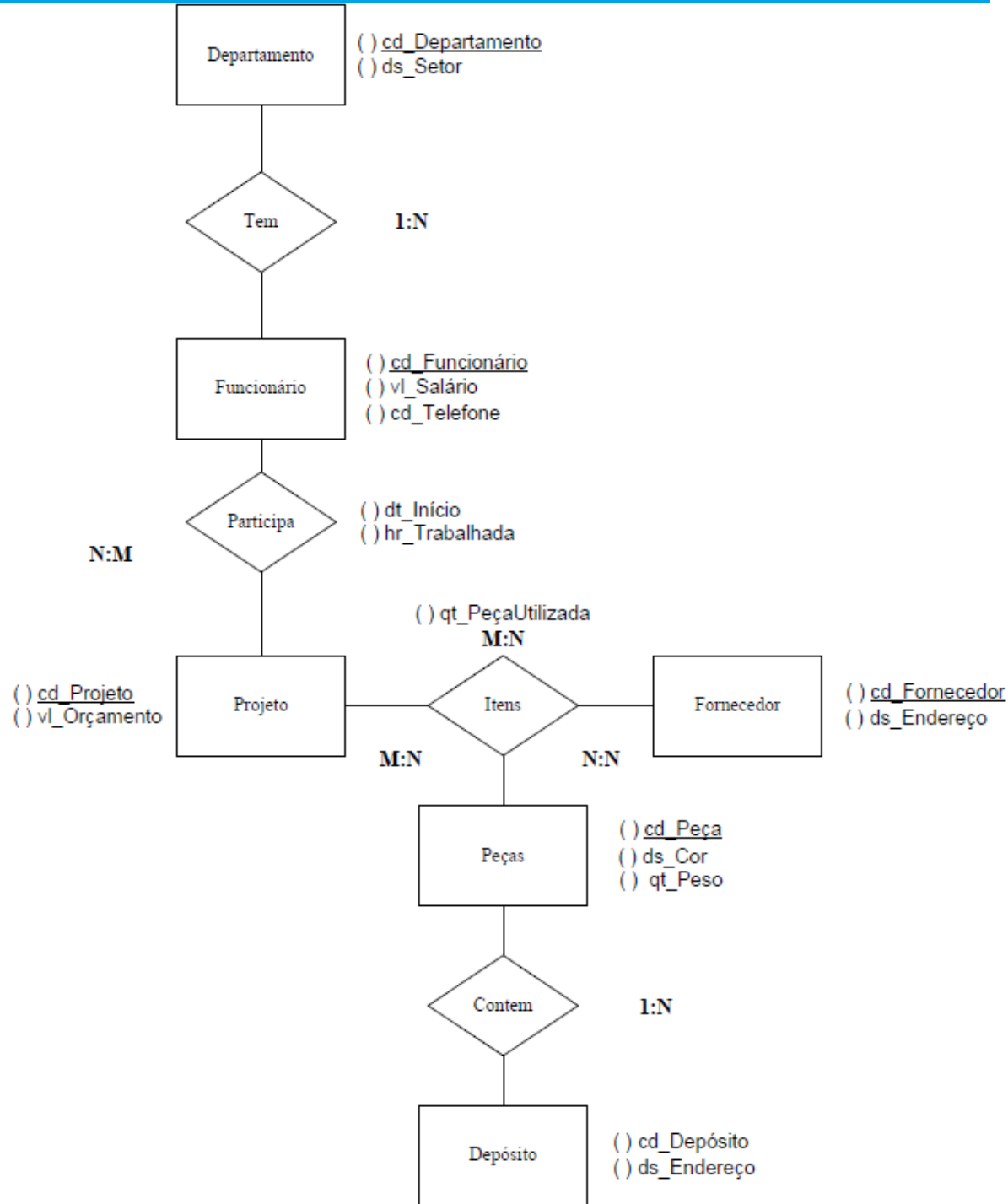
Elaborar um diagrama para uma Indústria.

- **Requisitos:**

- a) Cada Funcionário pode estar alocado a somente um Departamento;
- b) Cada Funcionário pode pertencer a mais de um Projeto;
- c) Um projeto pode utilizar-se de vários Fornecedores e de várias Peças;
- d) Uma Peça pode ser fornecida por vários Fornecedores e atender a vários Projetos;
- e) Um Fornecedor pode atender a vários Projetos e fornecer várias Peças;
- f) Um Depósito pode conter várias Peças;
- g) Deseja-se ter um controle do material utilizado por cada Projeto, identificando inclusive o seu Fornecedor. Gravar as informações de data de Início e Horas Trabalhadas no Projeto.

- Peças;
- Depósitos;
- Fornecedor;
- Projeto;
- Funcionário;
- Departamento.

- a) Peças: Número, Peso e Cor;
- b) Depósito: Número e Endereço;
- c) Fornecedor: Número e Endereço;
- d) Projeto: Número e Orçamento;
- e) Funcionário: Número, Salário e Telefone;
- f) Departamento: Número e Setor.



Exercícios ER

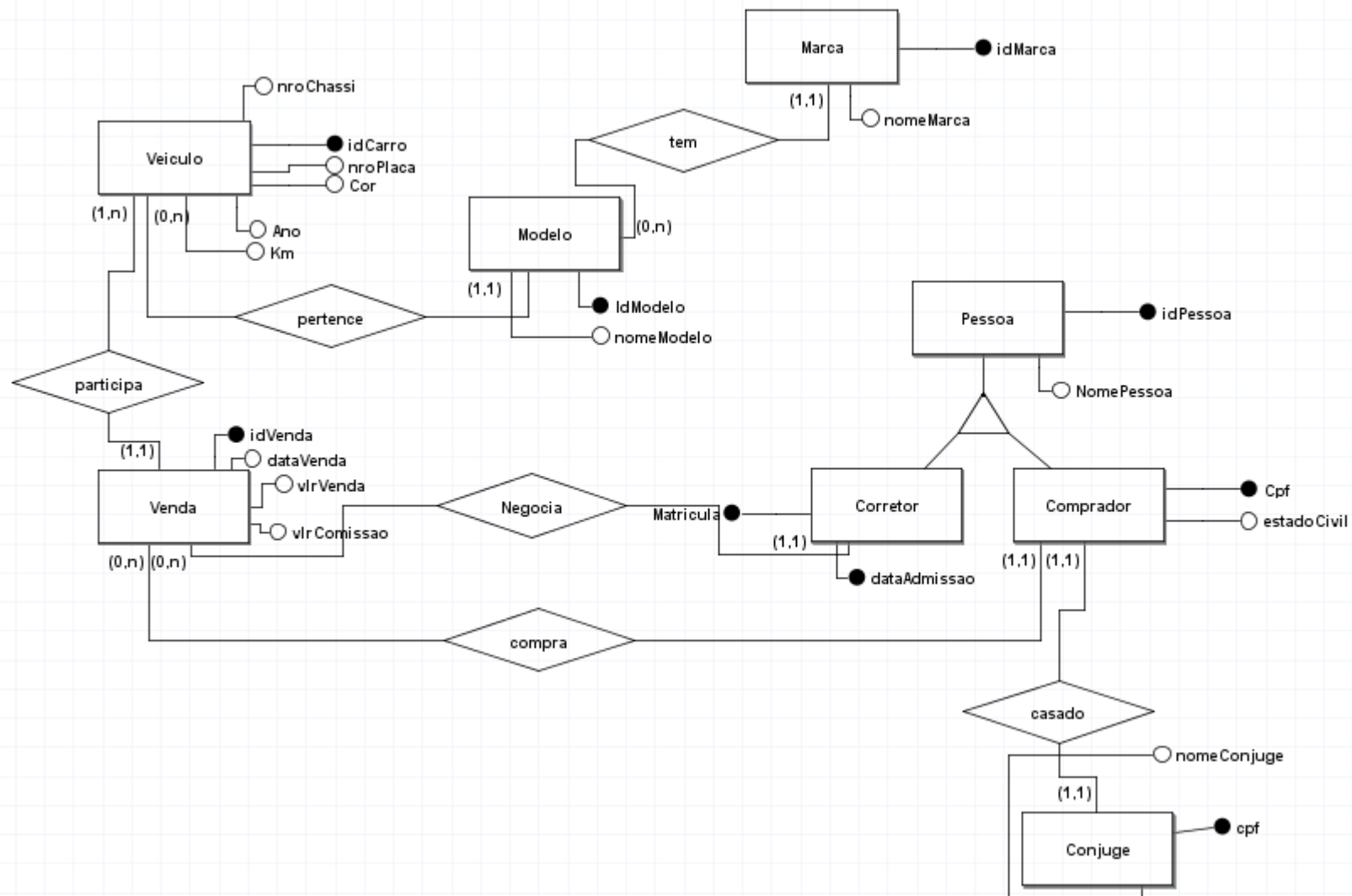
Exercícios

1. A empresa XYZ pretende armazenar os dados sobre os seus projetos e sobre os funcionários lotados nesses projetos. Cada projeto possui uma sigla e um nome. A sigla de um projeto o identifica unicamente em relação a outros projetos. Cada funcionário tem um código de matrícula, um CPF e um nome. Além disso, um funcionário pode estar lotado em vários projetos, bem como um projeto pode contar com a participação de vários funcionários. Baseado nessas informações, modele o Diagrama Entidade-Relacionamento (DER) para o problema proposto

Exercícios

2. Uma concessionária que trabalha com venda de veículos deseja criar uma base de dados para o seu negócio. Essa base deve atender aos seguintes requisitos: a). Para qualquer veículo, devemos saber o número do chassi, número da placa, cor, ano de fabricação, quilometragem, marca e modelo. b). Sobre marca e modelo, basta sabermos seus códigos e nomes. c). Todo carro pertence a um modelo e este modelo pertence a uma marca. d). Uma pessoa pode assumir um dos seguintes papéis em relação a concessionária: corretor ou comprador. e). Sobre o comprador do veículo, tem-se CPF, nome, estado civil e, se for casado, os dados do cônjuge (como nome e CPF). f). Sobre os corretores, tem-se número da matrícula, nome e data de admissão. g). Um corretor negocia com um comprador a venda de um veículo. Sobre a venda são necessárias as seguintes informações: data, valor da venda e valor da comissão do corretor.

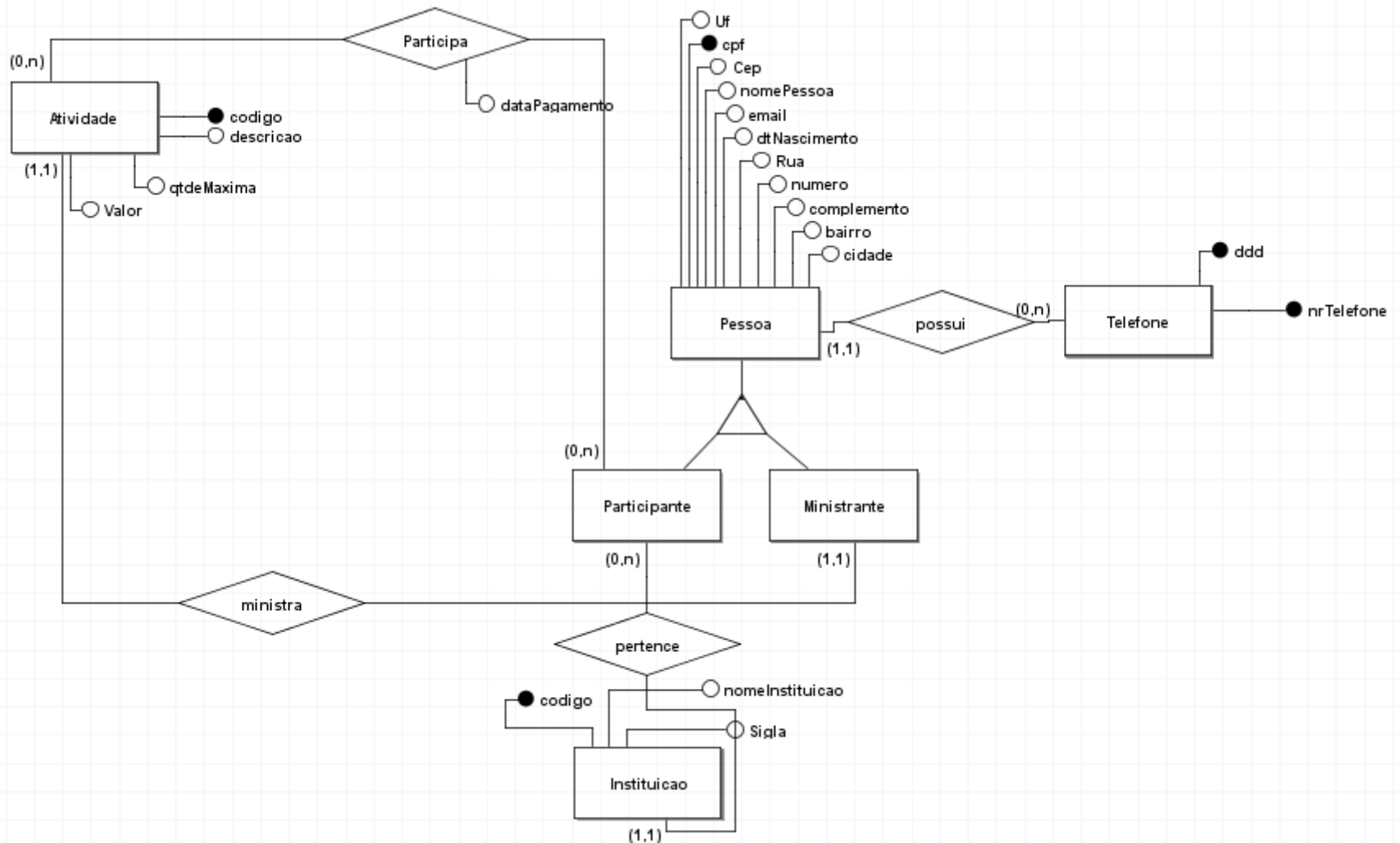
Resolução



Exercícios

3. Os alunos da Disciplina de Banco de Dados I do Curso de Engenharia de Software foram convidados para projetar o banco de dados que será utilizado para gerenciar as atividades realizadas durante o evento DEV PARANÁ 2023. Para isso, crie um DER capaz de representar os seguintes requisitos: a). Cada atividade do evento possui: código, descrição, quantidade máxima de participantes e valor. b). Há dois tipos de pessoas no evento: participantes e ministrantes de atividades. c). Todas essas pessoas possuem os seguintes atributos em comum: cpf, nome, data de nascimento, endereço, telefone e email. d) O endereço é composto de rua, número, complemento, bairro, cidade, UF e CEP. e). É possível cadastrar vários telefones para uma mesma pessoa. f). Cada participante do evento pode realizar várias atividades. Além disso, cada participante está vinculado a uma única instituição. Uma instituição tem um código, uma sigla e um nome. g). Cada atividade só pode ter um único ministrante.. h). É necessário armazenar a data de pagamento em que cada participante pagou cada uma de suas atividades. Um único participante pode pagar cada uma de suas atividades em datas diferentes.

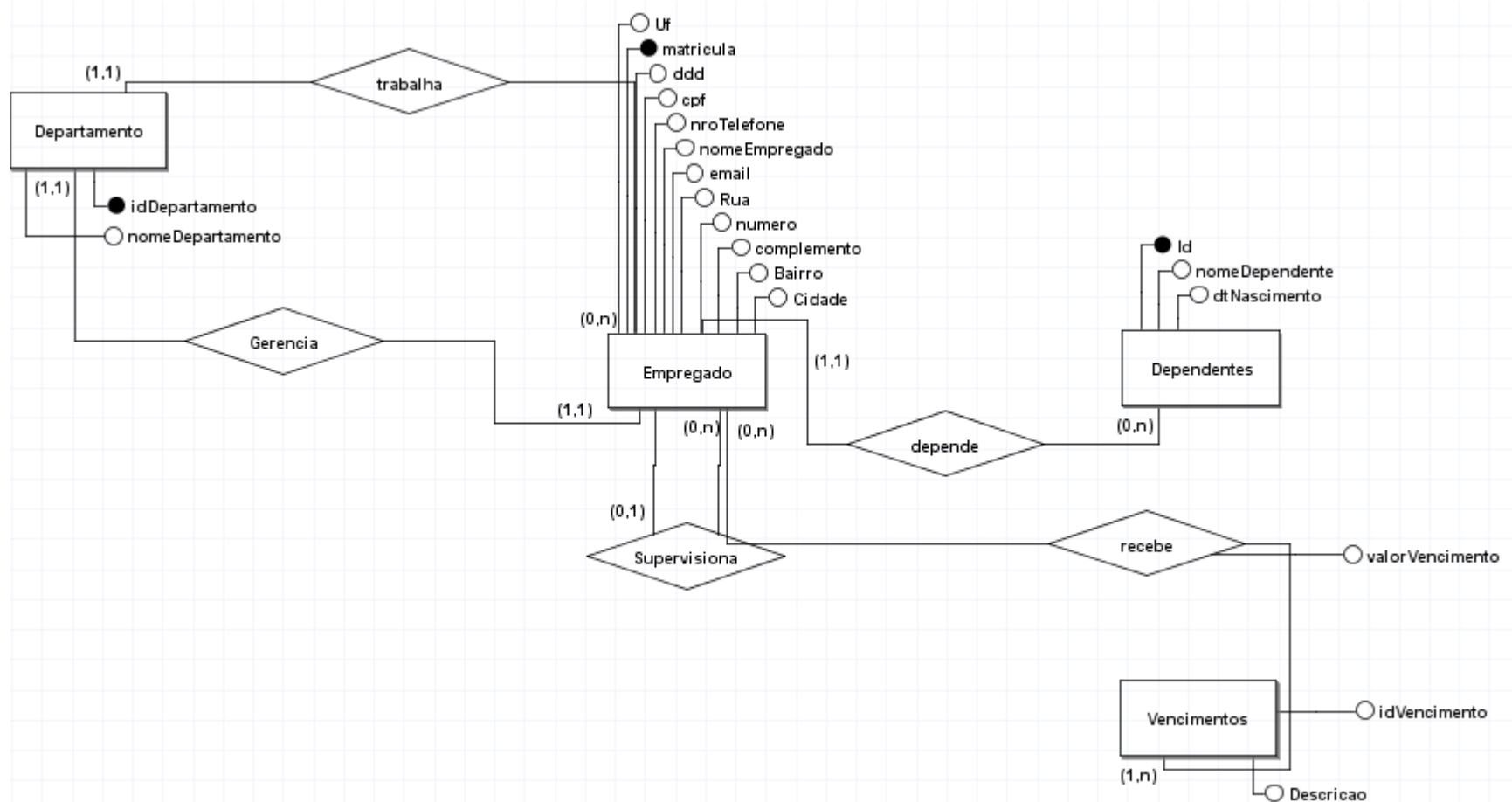
Resolução



Exercícios

4. Crie um DER para representar os dados da empresa Xing-Ling de Maringá que tem seus dados organizados da seguinte forma: a). Cada empregado trabalha em um único departamento da empresa e possui: matrícula, cpf, nome, endereço, telefone e email. b) Funcionários são diretamente chefiados por um único supervisor que também é um funcionário. É possível haver mais de um supervisor em um mesmo departamento. c). Um departamento possui no mínimo 5 funcionários, onde somente um deles é o gerente do departamento. O gerente do departamento também é um funcionário. Além disso, cada departamento possui um código e um nome. d). Os dependentes dos funcionários devem possuir como atributos: id, nome e data de nascimento. e) O salário de um empregado é calculado com base nos seus diversos vencimentos. Para cada tipo de vencimento, existe uma descrição e o valor correspondente.

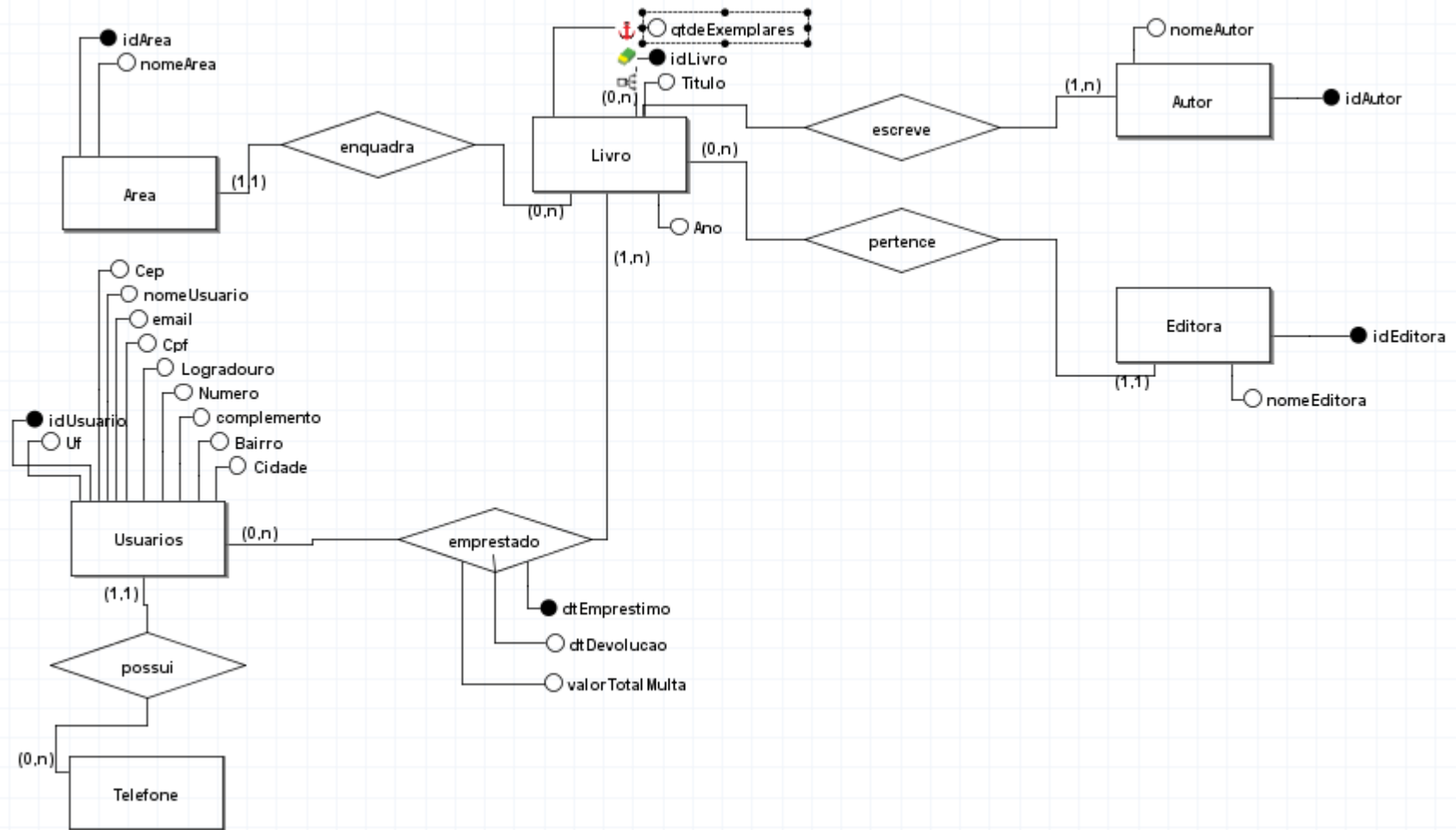
Resolução



Exercícios

5. Você foi convidado a desenvolver um sistema para controlar uma biblioteca que vai funcionar assim: a). Os livros serão separados por área (exemplo: romance, ficção, policial, etc.) b) Sobre o livro é necessário saber: título, autores, editora, ano de publicação, área e ISBN. c). É possível ter vários exemplares de um mesmo livro na biblioteca. d). Sobre os usuários da biblioteca, precisamos saber seu CPF, nome, telefone e endereço. Um aluno pode ter vários telefones. e). Sobre autor, editora e área, basta sabermos seus códigos e nomes. f). Os usuários da biblioteca podem pegar livros emprestados. Quando isso ocorre é necessário saber a data do empréstimo, a data da devolução e o valor total da multa cobrada, caso o livro seja devolvido com atraso.

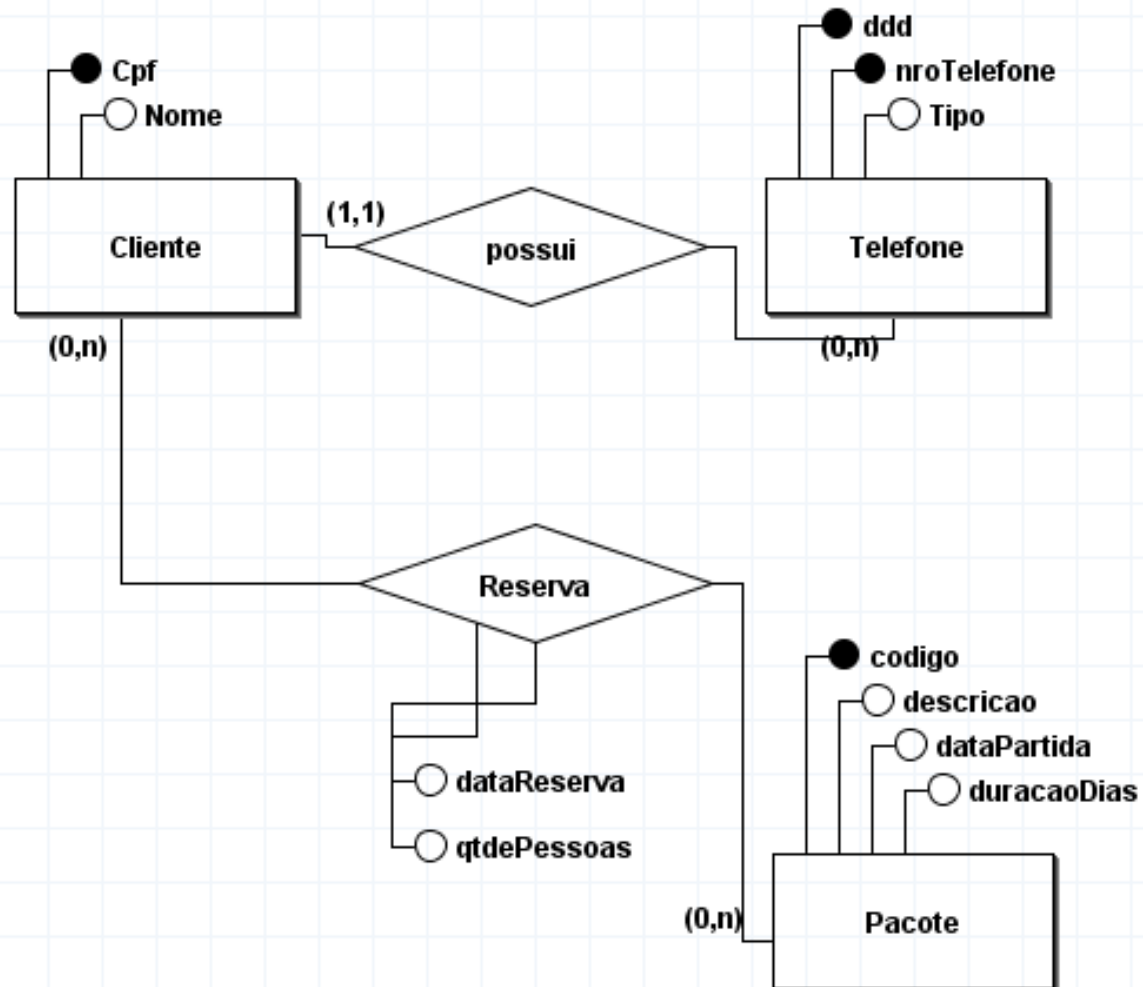
Resolução



Exercícios

7. Uma agência de turismo deseja controlar a reserva de pacotes de viagem por seus clientes. Cada cliente deve ser cadastrado na agência, sendo armazenados, seu CPF (usado como identificador), seu nome e seus vários telefones. A agência cadastra o maior número de telefones de contato possíveis, para cada um deles o próprio número e o tipo (comercial, residencial, celular, recados). Para cada tipo, um cliente pode ter vários telefones. Além dos clientes, a agência registra pacotes de viagem. Cada pacote é identificado por um código (numérico) e devem ser armazenados uma descrição do pacote (texto livre), a data de partida e a duração (em número de dias). Finalmente, devem ser registradas as reservas para os pacotes. Cada reserva é feita por um cliente e refere-se a um pacote. É necessário saber a data em que foi feita a reserva. Um cliente pode reservar um pacote para várias pessoas (familiares, amigos, etc.). Neste caso, somente o cliente estará cadastrado, e é necessário armazenar o número de pessoas que participam da reserva. Projete um esquema textual para a base de dados em questão. Esta base de dados deve refletir exatamente o enunciado acima e não deve conter redundâncias.

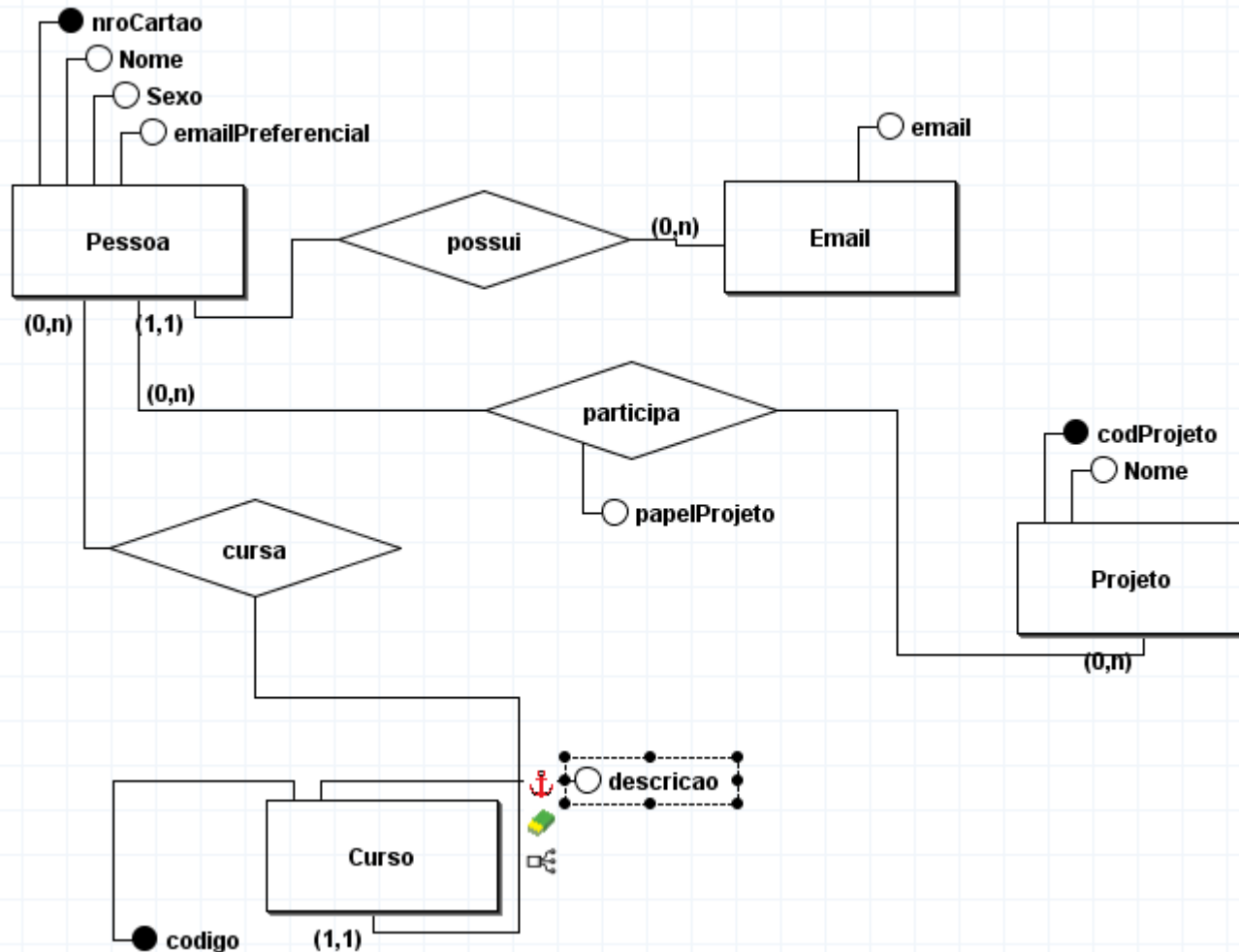
Resolução



Exercícios

8. Projete uma base de dados relacional para armazenar dados sobre pessoas ligadas ao Departamento de Informática do Unicesumar. Cada pessoa é identificada por um número de cartão e a base de dados deve manter o nome da pessoa, bem como seu sexo. A base de dados deve manter os e-mails da pessoa. Cada pessoa pode ter vários e-mails, mas há somente um e-mail preferencial que será usado para comunicar-se com a pessoa. Caso a pessoa for aluno do Unicesumar estará vinculada a um curso. Cada curso possui um código numérico e um nome. Uma pessoa pode estar vinculada a no máximo um curso. As pessoas que trabalham em projetos de pesquisa (docentes, alunos e bolsistas) devem estar vinculadas a estes projetos. Cada projeto está cadastrado na base de dados com seu código numérico e seu nome. Para cada participante é necessário conhecer seu papel no projeto (o papel pode ser docente, aluno ou bolsistas). Observar que uma pessoa pode participar de vários projetos, inclusive com diferentes papéis. Projete uma base de dados relacional que armazena os dados acima sem redundância de dados. Enumere as tabelas, suas colunas, as chaves primárias e as chaves estrangeiras.

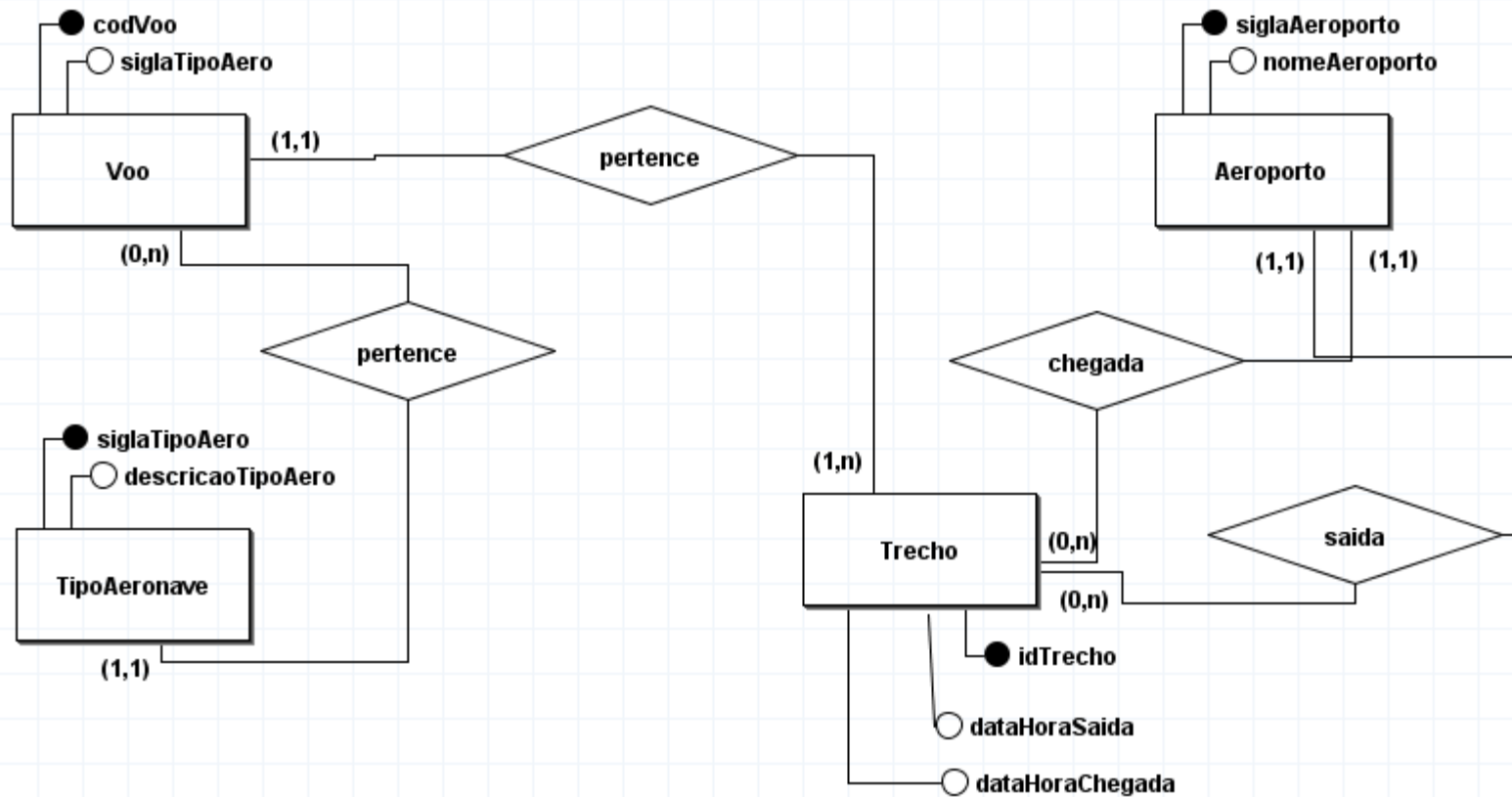
Resolução



Exercícios

9. Uma companhia de aviação deseja montar uma base de dados para divulgação em um site Web. A base de dados deve conter as seguintes informações. A companhia oferece vários voos. Cada voo é identificado por um código numérico e a base de dados deve registrar o tipo de aeronave que é usada para o voo. Cada aeronave é identificada por uma sigla alfanumérica, como "B-767" e possui uma descrição. Obviamente, um tipo de aeronave pode ser usado em muitos voos. Além dos dados acima, é necessário saber os trechos que compõe o voo. Um voo é composto de vários trechos, numerados de um em diante, na ordem em que são voados. Para cada trecho é necessário saber o horário de saída, o horário de chegada, o aeroporto de origem e o aeroporto de destino. Um aeroporto é identificado por uma sigla de três letras, como "MGF" ou "CWB" e tem um nome. Projete uma base de dados relacional que armazene os dados acima sem redundância de dados.

Resolução



Exercícios Generalização/ Especialização

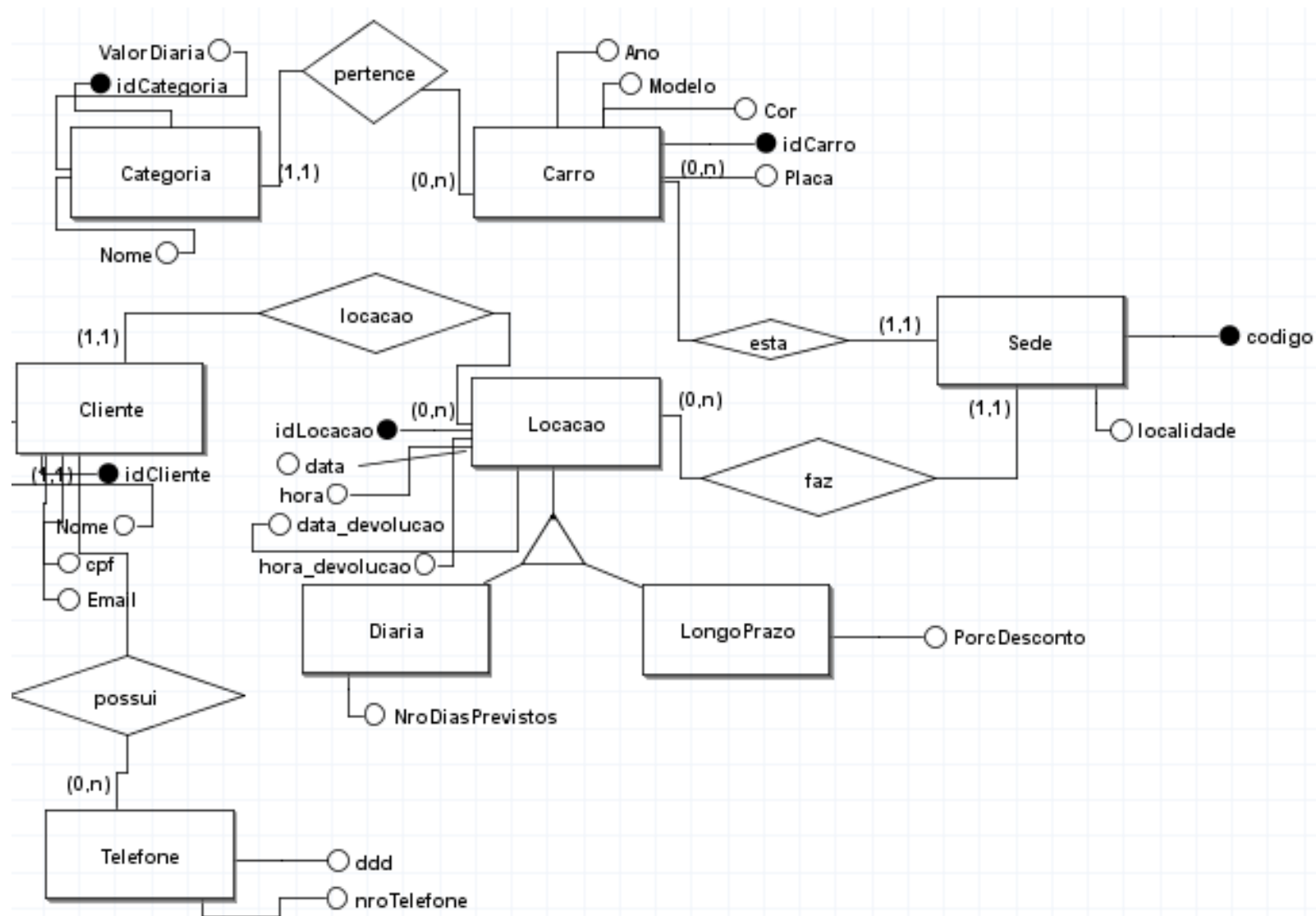
Exercício – Generalização/Especialização

- Uma locadora de carros deseja fazer um sistema para armazenar as informações das locações que os clientes fazem.
- A locadora possui diversas sedes, sendo que cada sede possui um código e uma localidade, sendo que uma localidade é armazenada internamente por dois números de ponto flutuante (exemplo: a localidade 18°56'04.4"S 48°17'29.7"W é armazenada como -18.9345654, 48.2915811).
- Para cada locação é necessário registrar a data e horário em que o cliente pega o carro, bem como a sede da qual o carro foi retirado. Há dois tipos de locação: a diária e a de longo período.
- Quando a locação for diária, deve-se armazenar o número de dias previstos para devolução do carro. Quando a locação for de longo período, deve-se armazenar a porcentagem de desconto dada no valor da diária. Em todos os casos, a data e horário em que o carro foi devolvido devem ser armazenados.
- Os dados de cada carro são modelo, placa, cor, ano e data de aquisição. A locadora trabalha somente com carros de cor branca, preta, cinza e vermelha. Os dados dos clientes são nome, cpf, e-mail e seus telefones. Cada carro pertence a uma categoria, sendo que cada categoria de carro possui um valor de diária de locação.

Exercício 1: Uma locadora de **carros** deseja fazer um sistema para armazenar as informações das **locações** que os **clientes** fazem. A locadora possui diversas **sedes**, sendo que cada sede possui um **código** e uma **localidade**, sendo que uma localidade é armazenada internamente por **dois números de ponto flutuante** (exemplo: a localidade $18^{\circ}56'04.4''S$ $48^{\circ}17'29.7''W$ é armazenada como -18.9345654, -48.2915811). Para cada locação é necessário registrar a **data e horário** em que o cliente pega o carro, bem como a sede da qual o carro foi retirado. Há dois tipos de locação: a **diária** e a de **longo período**. Quando a locação for diária, deve-se armazenar o número de **dias previstos** para devolução do carro. Quando a locação for de longo período, deve-se armazenar a **porcentagem de desconto dada no valor da diária**. Em todos os casos, a **data e horário em que o carro foi devolvido** devem ser armazenados. Os dados de cada carro são **modelo**, **placa**, **cor**, **ano** e **data de aquisição**. A locadora trabalha somente com carros de cor branca, preta, cinza e vermelha. Os dados dos clientes são **nome**, **cpf**, **email** e seus **telefones**. Cada carro pertence a uma **categoria**, sendo que cada categoria de carro possui um valor de **diária de locação**.

Instância mínima: 1 cliente, 2 carros, 3 locações.

Resolução



Exercício –

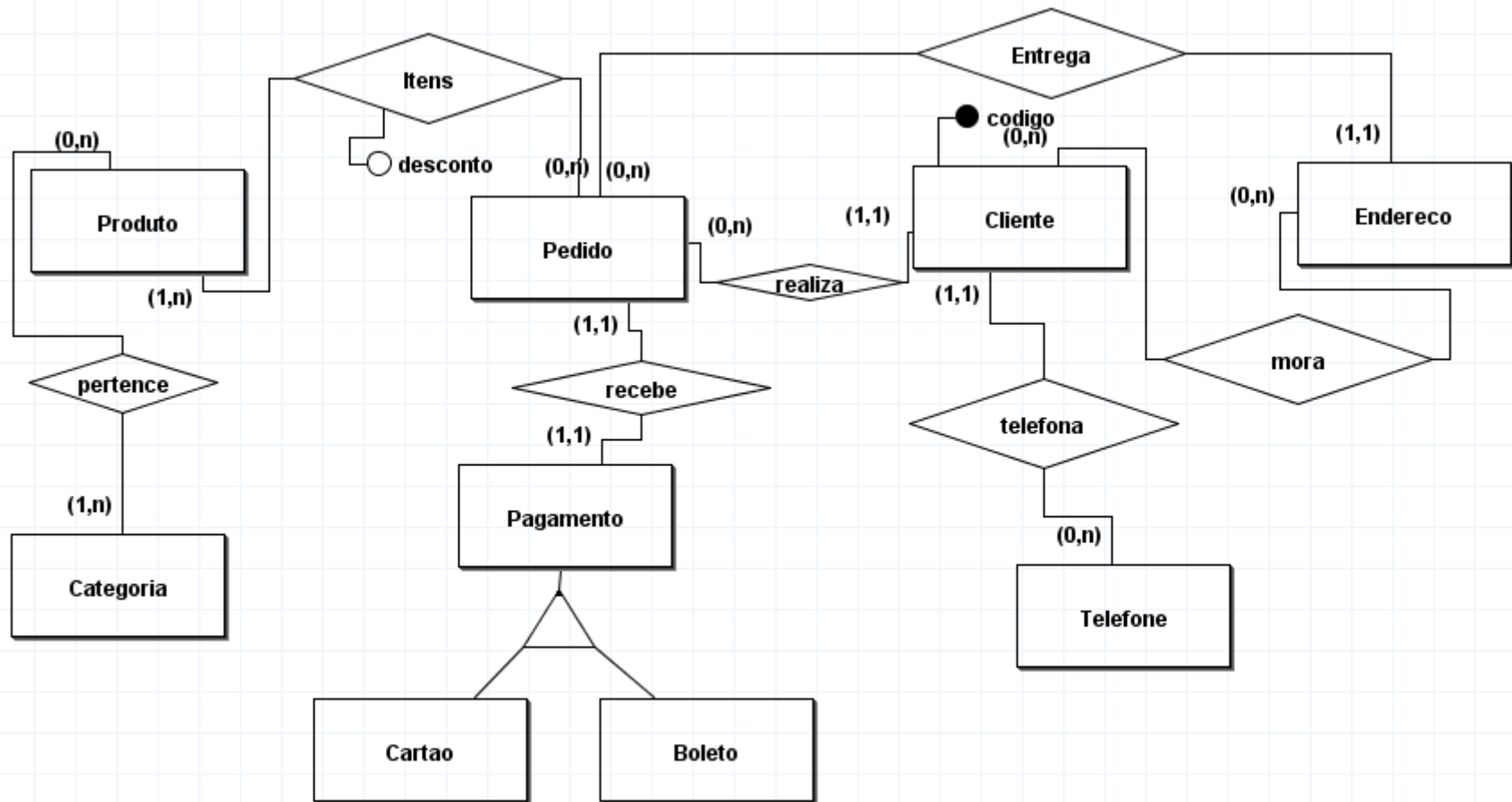
Generalização/Especialização

- Exercício 2: Deseja-se fazer um sistema de pedidos. Um ou mais produtos podem ser vendidos em cada pedido, sendo que a cada produto pode ser dado um desconto diferente, e também cada produto pode ser vendido em uma ou mais unidades. Cada produto possui nome e preço, e pode pertencer a várias categorias. Cada pedido é feito por um cliente, que deve ter em seu cadastro nome, telefones, e-mail, cpf ou cnpj, e um ou mais endereços, sendo que o cliente deve especificar um endereço para entrega na hora de comprar. Para um pedido, deve ser registrado o instante em que é realizado e o endereço de entrega. Um pedido deve ser pago ou por boleto, ou por cartão de crédito. No caso de boleto, deve-se armazenar a data de vencimento e a data de pagamento. No caso de cartão de crédito, deve-se armazenar o número de parcelas. Todo pagamento possui um estado (pendente, quitado ou cancelado).
- Instância mínima: 1 cliente, 3 produtos, 2 pedidos, pelo menos um pedido com mais de um produto, pelo menos um pedido pago com cartão de crédito e um com boleto.
-

Exercício 2: Deseja-se fazer um sistema de pedidos. Um ou mais produtos podem ser vendidos em cada pedido, sendo que a cada produto pode ser dado um desconto diferente, e também cada produto pode ser vendido em uma ou mais unidades. Cada produto possui nome e preço, e pode pertencer a várias categorias. Cada pedido é feito por um cliente, que deve ter em seu cadastro nome, telefones, email, cpf ou cnpj, e um ou mais endereços, sendo que o cliente deve especificar um endereço para entrega na hora de comprar. Para um pedido, deve ser registrado o instante em que é realizado e o endereço de entrega. Um pedido deve ser pago ou por boleto, ou por cartão de crédito. No caso de boleto, deve-se armazenar a data de vencimento e a data de pagamento. No caso de cartão de crédito, deve-se armazenar o número de parcelas. Todo pagamento possui um estado (pendente, quitado ou cancelado).

Instância mínima: 1 cliente, 3 produtos, 2 pedidos, pelo menos um pedido com mais de um produto, pelo menos um pedido pago com cartão de crédito e um com boleto.

Exercícios



Exercícios Normalização

Exercícios

a) Acidente (numero_placa_carro, cpf_motorista, nome_motorista, total_danos_acidente, data_acidente)

Acidente(numero_placa*, cpf_motorista*, total_danos_acidente, data_acidente)

Motorista(cpf_motorista*, nome_motorista)

Exercícios

b) Paciente (cod_paciente, nome_paciente, (fone_paciente), (crm_medico, nome_medico, data_consulta), cod_convenio, nome_convenio, (cod_exame, nome_exame, diagnostico))

Paciente(cod_paciente*, nome_paciente, cod_convenio**)

Convenio(cod_convenio*, nome_convenio)

Telefone(cod_paciente*, fone_paciente*)

Consulta(cod_paciente*, crm_medico*, data_consulta)

Medico(crm_medico*, nome_medico)

Diagnostico(cod_paciente*, cod_exame*, diagnostico)

Exame(cod_exame*, nome_exame)

Exercícios

- c) Aluno (cod_aluno, nome_aluno, sexo_aluno, dt_nasc_aluno, cod_curso, nome_curso, nome_diretor, (cod_disciplina, nome_disciplina, nota_disciplina))

d) NotaFiscal (num_nota, dt_emissao_nota, cod_cliente, nome_cliente, endereco_cliente: rua; numero; complemento, cpf_cliente, (cod_produto, nome_produto, qtd_vendida_produto, valor_unitario_produto, valor_total_por_item_vendido), valor_total_da_nota)

1Fn

NotaFiscal(num_nota*, dt_emissao_nota, cod_cliente, nome_cliente, rua_cliente, numero_cliente, complemento_cliente, cpf_cliente, valor_total_da_nota)

ItemNota(num_nota*, cod_produto*, nome_produto, qtd_vendida_produto, valor_unitario_produto, valor_total_item_vendido)

2Fn

NotaFiscal(num_nota*, dt_emissao_nota, cod_cliente, nome_cliente, rua_cliente, numero_cliente, complemento_cliente, cpf_cliente, valor_total_da_nota)

ItemNota(num_nota*, cod_produto*, qtd_vendida_produto, valor_unitario_produto, valor_total_item_vendido)

Produto(cod_produto*, nome_produto, valor_unitario_produto)

3FN

NotaFiscal(num_nota*, dt_emissao_nota, cod_cliente, valor_total_da_nota)

ItemNota(num_nota*, cod_produto*, qtd_vendida_produto, valor_unitario_produto, valor_total_item_vendido)

Produto(cod_produto*, nome_produto, valor_unitario_produto)

Cliente(cod_cliente, nome_cliente, rua_cliente, numero_cliente, complemento_cliente, cpf_cliente)

Exercícios

e) Departamento (cod_departamento, nome_departamento, (matr_empregado, nome_empregado, endereço_empregado: rua; numero; complemento, dt_nasc_empregado, salario_empregado))

e)

1FN

Departamento(cod_departamento*, nome_departamento)

Empregado(cod_departamento*, matr_empregado*, nome_empregado, end_rua_empregado, end_numero_empregado, end_complemento_empregado,

dt_nasc_empregado, salario_empregado)

Exercícios

f) ItemVenda(NumeroNF, CodigoTipoProd, NumeroProd, DescricaoProd, DataVenda, CodReg, CodEmp, QtdeItem, PrecoItem, NomeEmp, DescricaoTipoProd)

1FN - Já está

2FN

ItemVenda(NumeroNF*,codigotipoProd*,NumeroProd*,DataVenda,QtdeItem,CodReg,CodEmp,NomeEmp)

Produto(NumeroProd,DescricaoProd)

TipoProd(CodigoTipoProd, DescricaoTipoProd)

3FN

ItemVenda(NumeroNF*,codigotipoProd*,NumeroProd*,DataVenda,QtdeItem,CodReg,CodEmp**)

Empregado(CodEmp*,NomeEmp)

Produto(NumeroProd*,DescricaoProd)

TipoProd(CodigoTipoProd*, DescricaoTipoProd)

Exercícios

g) `Matricula(CodAluno, CodTurma, CodDisciplina, NomeDisciplina, NomeAluno, CodLocalNascAluno, NomeLocalNascAluno)`

1FN - Já está

2FN

`Matricula(codAluno*,CodTurma*,CodDisciplina**)`

`Disciplina(codDisciplina*,nomeDisciplina)`

`Aluno(codAluno*,NomeAluno,codLocalNascAluno, NomeLocalNascAluno)`

Exercícios

g) `Matricula(CodAluno, CodTurma, CodDisciplina, NomeDisciplina, NomeAluno, CodLocalNascAluno, NomeLocalNascAluno)`

1FN - Já está

2FN

`Matricula(codAluno*,CodTurma*,CodDisciplina**)`

`Disciplina(codDisciplina*,nomeDisciplina)`

`Aluno(codAluno*,NomeAluno,codLocalNascAluno, NomeLocalNascAluno)`

3FN

`Matricula(codAluno*,CodTurma*,CodDisciplina**)`

`Disciplina(codDisciplina*,nomeDisciplina)`

`Aluno(codAluno*,NomeAluno, codLocalNascAluno**)`

`Local (codLocalNascAluno*, NomeLocalNascAluno)`

Exercícios

Pedido

Pedido: (---noPed---) **Data do pedido:** (---data---)

Código do Cliente: (---codClie---)

Nome do Cliente: (---NomeCliente)

Telefones p/ contato: (---noTel---) (---noTel---) (---noTel---)

Código Peça	Descrição Peça	Quantidade Pedido
(---)	(---)	(---)
(---)	(---)	(---)

Pedido(noPed,data,codClie,NomeClie(noTel),(codPeca,DescricaoPeca,QtdePedido));

1FN

Pedido(noPed*,data,CodClie,NomeClie

Telefone(CodClie*,noTel*)

PecasVenda(noPed*,codPeca*,DescricaoPeca,QtdePedido)

Exercícios

Pedido

Pedido: (---noPed---) **Data do pedido:** (---data---)

Código do Cliente: (---codClie---)

Nome do Cliente: (---NomeCliente)

Telefones p/ contato: (---noTel---) (---noTel---) (---noTel---)

Código Peça	Descrição Peça	Quantidade Pedido
(---)	(---)	(---)
(---)	(---)	(---)

2FN

Pedido(noPed*,data,CodClie,NomeClie)

Telefone(CodClie*,noTel*)

PecasVenda(noPed*,codPeca*,QtdePedido)

Peca(codPeca*,descricaoPeca)

Exercícios

Pedido

Pedido: (---noPed---) **Data do pedido:** (---data---)

Código do Cliente: (---codClie---)

Nome do Cliente: (---NomeCliente)

Telefones p/ contato: (---noTel---) (---noTel---) (---noTel---)

Código Peça	Descrição Peça	Quantidade Pedido
(---)	(---)	(---)
(---)	(---)	(---)

3FN

Pedido(noPed*,data,CodClie**)

Telefone(CodClie*,noTel*)

PecasVenda(noPed*,codPeca*,QtdePedido)

Peca(codPeca*,descricaoPeca)

Cliente(CodClie*,NomeClie)

Exercícios DML

Respostas

- 1) Liste todas as colunas da tabela EMPREGADO
(utilizar o *)

```
SELECT * FROM EMPREGADO
```

- 2) Liste somente o nome do empregado e o
salario .(projeção de colunas)

```
SELECT NOMEEMPREGADO, SALARIO FROM  
EMPREGADO;
```

Respostas

- 3) Liste o código do empregado ao invés de exibir o nome da coluna IDEMP, exita Codigo (utilizando apelido)

```
SELECT IDEMP CODIGO FROM EMPREGADO;
```

- 4) Liste o nome do empregado e faça a operação de soma para o salario + comissão de o nome dessa operação de Salario_Bruto (projeção com operadores)

```
SELECT NOMEEMPREGADO, SALARIO+COALESCE(COMISSAO,0)  
SALARIO_BRUTO FROM EMPREGADO;
```

Respostas

5) Liste os códigos dos departamentos existentes na tabela empregado (sem repetir os códigos -> Distinct)

```
SELECT DISTINCT DEPTNO FROM EMPREGADO;
```

6) Liste todos os empregados do departamento 10 (Operador =)

```
SELECT * FROM EMPREGADO = 10;
```

7) Liste todos os empregados do departamento maior ou igual a 30 (Operador >=)

```
SELECT * FROM EMPREGADO WHERE DEPTNO >= 30;
```

8) Liste todos os empregados cujo departamento não seja o 10 (Operador <> ou !=)

```
SELECT * FROM DEPARTAMENTO WHERE DEPTNO <> 10;
```

Respostas

9) Liste todos os empregados do departamento 10 ou 20
(Operador condicional OR)

```
SELECT * FROM EMPREGADO WHERE DEPTNO = 10 OR DEPTNO  
= 20;
```

10) Liste todos os empregados que têm um salário entre 1000 e
2000 (Operador \geq \leq e condicional AND)

```
SELECT * FROM EMPREGADO WHERE SALARIO  $\geq$  1000 AND  
SALARIO  $\leq$  2000;
```

11) Liste todos os empregados que não possuem gerente
cadastrado (IS NULL ou IS NOT NULL)

```
SELECT * FROM EMPREGADO WHERE GERENTE IS NULL;
```

Respostas

12) Liste todos os empregados do departamento 20 ou 30
(Operador de conjunto IN)

```
SELECT * FROM EMPREGADO WHERE DEPTNO IN (20,30);
```

13) Liste todos os empregados que têm um salário entre 1000 e
2000 (Operador de intervalo de valores BETWEEN)

```
SELECT * FROM EMPREGADO WHERE SALARIO BETWEEN 1000  
AND 2000;
```

14) Liste os números e nomes dos departamentos ordenados
pelo nome do departamento.

```
SELECT IDDEPTO, NOMEDEPARTAMENTO FROM DEPARTAMENTO  
ORDER BY NOMEDEPARTAMENTO;
```

Respostas

15) Liste todos os códigos diferentes de funções desempenhadas pelos empregados.

```
SELECT DISTINCT IDFUNCAO FROM EMPREGADO;
```

16) Liste a informação detalhada dos empregados dos departamentos 10 e 20 por ordem alfabética do nome.

```
SELECT * FROM EMPREGADO WHERE DEPTNO IN (10,20) ORDER BY  
NOMEEMPREGADO;
```

17) Liste os nomes e funções de todos os empregados do departamento 20 e função (Auxiliar)

```
UPDATE TBFUNCAO SET NOMEFUNCAO = 'Auxiliar' where IDFUNCAO = 1;  
SELECT * FROM EMPREGADO WHERE DEPTNO = 20 AND ID_FUNCAO =  
1;
```


Respostas

18) Apresente todos os nomes de empregados que tenham TH ou LL.

```
SELECT * FROM EMPREGADO WHERE NOMEEMPREGADO LIKE  
'%TH%' OR NOMEMPREGADO LIKE '%LL%';
```

19) Liste o nome, função e salário para todos os empregados que tenham um chefe (Gerente).

```
SELECT NOMEEMPREGADO, IDFUNCAO, SALARIO FROM  
EMPREGADO WHERE GERENTE IS NOT NULL;
```

Respostas

20) Apresente o nome e a remuneração total (14 vezes o salário mais a comissão) para todos os empregados.

```
SELECT NOMEEMPREGADO, SALARIO * 14 + COALESCE(COMISSAO,0)  
      REMUNERACAO_TOTAL  
FROM EMPREGADO;
```

21) Liste o nome, salário anual (14 vezes o salário mensal) e comissão para todo o pessoal de vendas (VENDEDORES) cujo salário mensal for maior que a comissão.

```
SELECT NOMEEMPREGADO, SALARIO * 14 + COALESCE(COMISSAO,0)  
      REMUNERACAO_TOTAL  
FROM EMPREGADO  
WHERE ID_FUNCAO = 2 AND SALARIO > COMISSAO;
```

Elmasri & Navathe – Fundamentos de Bancos de Dados

Carlos Alberto Heuser – Projeto de Banco de Dados

Korth e Silberchatz – Sistema de Bancos de Dados