

Lista de exercícios 02 de Banco de Dados | BCD 29008

Semestre: 2022/2 Aluno: Guilherme medeiros

1. Para cada um dos exercícios da Lista 01, transforme o diagrama ER para o modelo relacional textual simplificado. É necessário que a resposta tenha o diagrama ER e o modelo relacional logo abaixo.

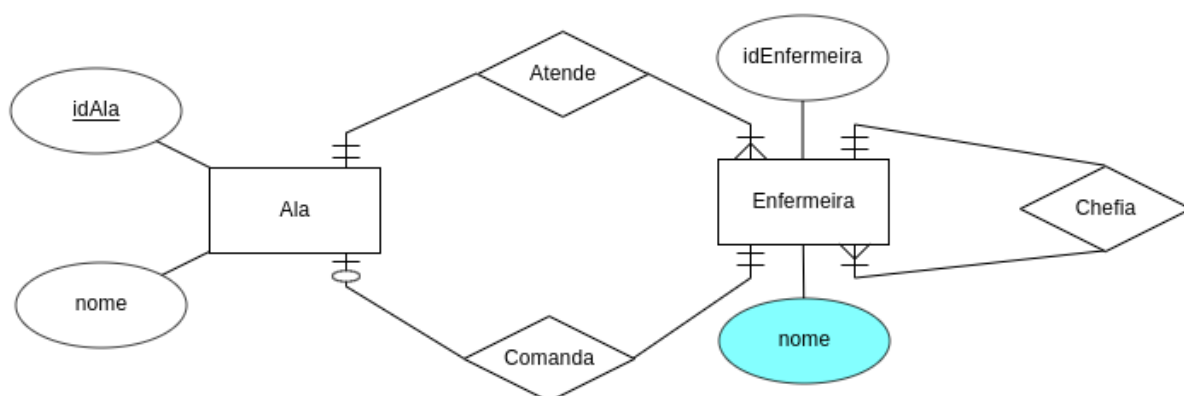
Legenda:

Negrito: Chave primária.

Itálico: Relação entre chave estrangeira e entidade.

Bloco descritivo: Descrição da cardinalidade das entidades.

1.1.



Relacionamento 1..1 __ 1..n entre Ala e Enfermeira: Adição de coluna

Relacionamento 0..1 __ 1..1 entre Ala e Enfermeira: Fusão de tabela

Relacionamento 1..1 __ 1..n entre Enfermeira e Enfermeira: Adição de coluna

Modelo relacional simplificado:

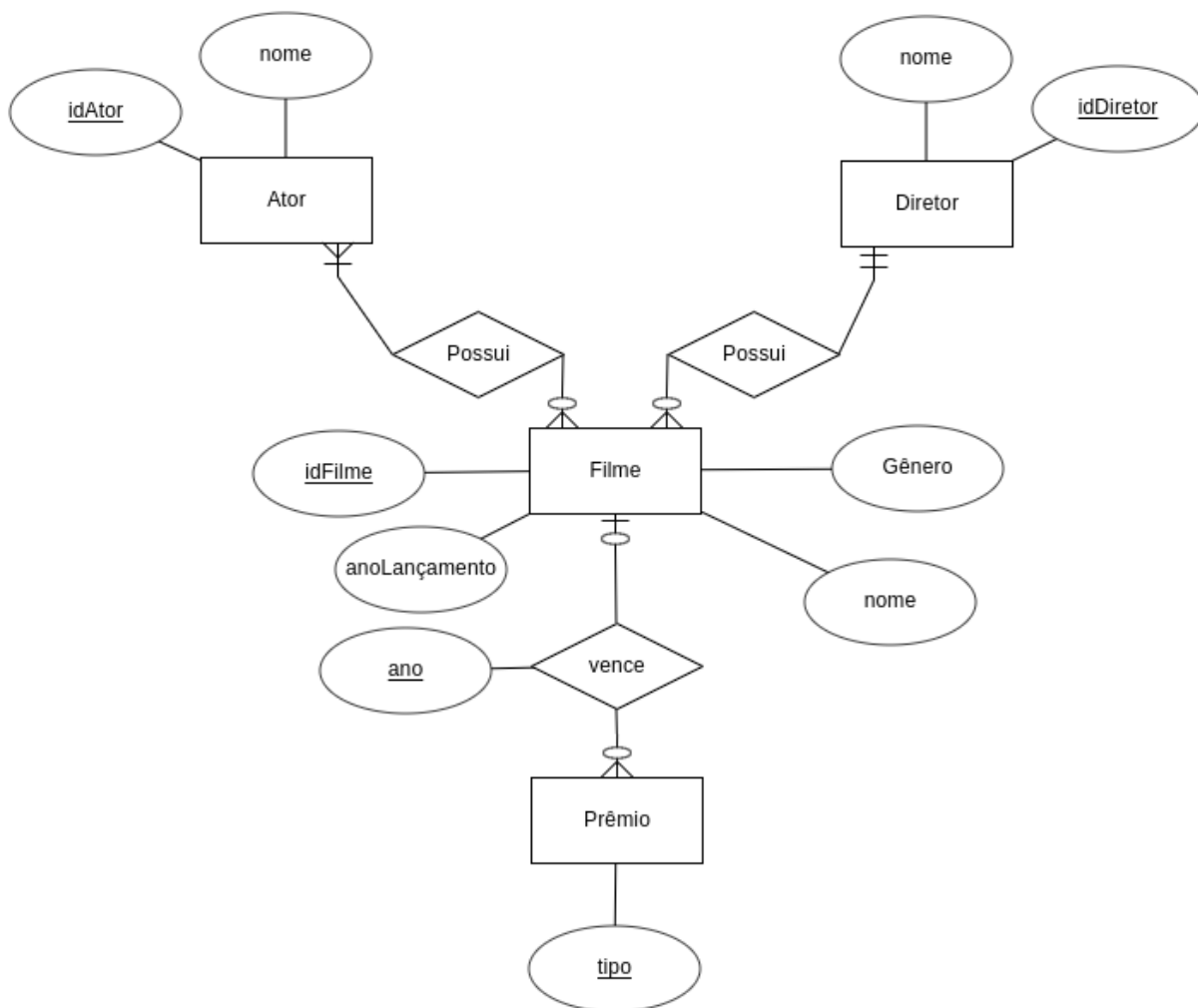
Ala(**idAla**, nome, idEnfermeiraChefe);

- idEnfermeiraChefe referencia idEnfermeira em Enfermeira.*

Enfermeira(**idEnfermeira**, nome, idAla, idEnfermeiraChefe);

- idEnfermeiraChefe referencia idEnfermeira em Enfermeira.*

1.2



Relacionamento 1..n __ 0..n entre Ator e Filme: Tabela própria
 Relacionamento 1..1 __ 0..n entre Diretor e Filme: Adição de coluna
 Relacionamento 0..1 __ 0..n entre Filme e Prêmio: Tabela própria

Modelo relacional simplificado:

Ator(**idAtor**, nome)

Diretor(**idDiretor**, nome)

Filme(**idFilme**, anoLançamento, nome, gênero, idDiretor)

- *idDiretor referencia Diretor*

AtorFilme(**idFilme**, **idAtor**)

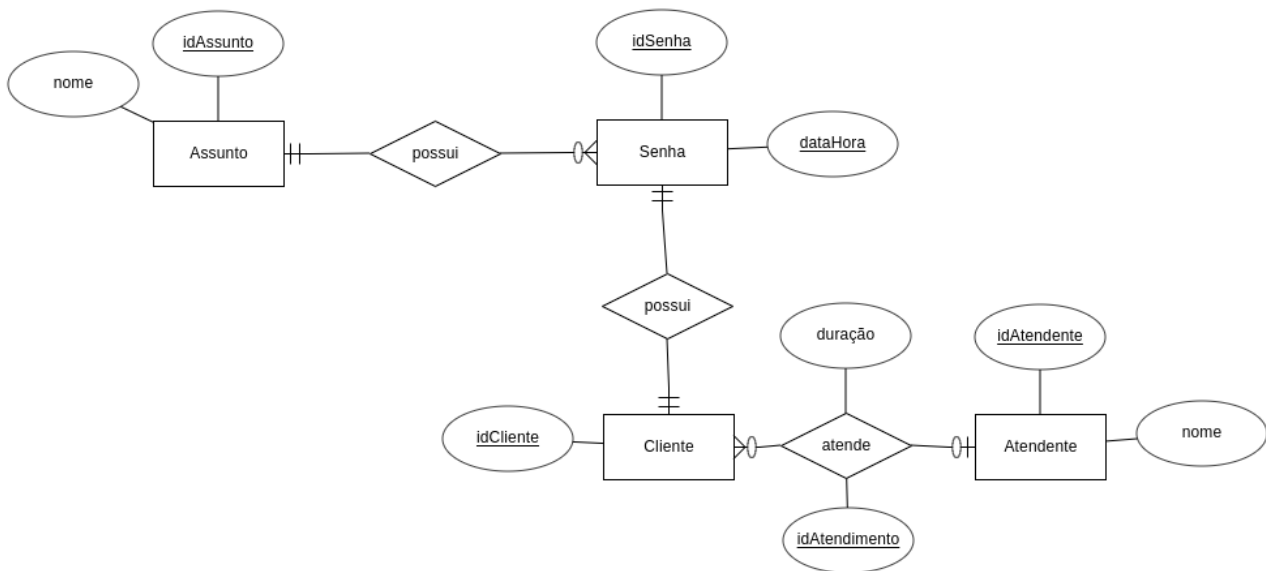
- *idFilme referencia Filme*
- *idAtor referencia Ator*

Premio(**tipo**)

Vencedor(**idFilme**,**idPremio**,ano)

- *idFilme* Referencia Filme
- *Premio* Referencia Premio

1.3



Relacionamento 1..1 __ 0..n entre Assunto e Senha: Adição de coluna
 Relacionamento 1..1 __ 1..1 entre Senha e Cliente: Fusão de tabelas
 Relacioanmento 0..n __ 0..1 entre Cliente e Atendente: Adição de coluna

Modelo relacional simplificado:

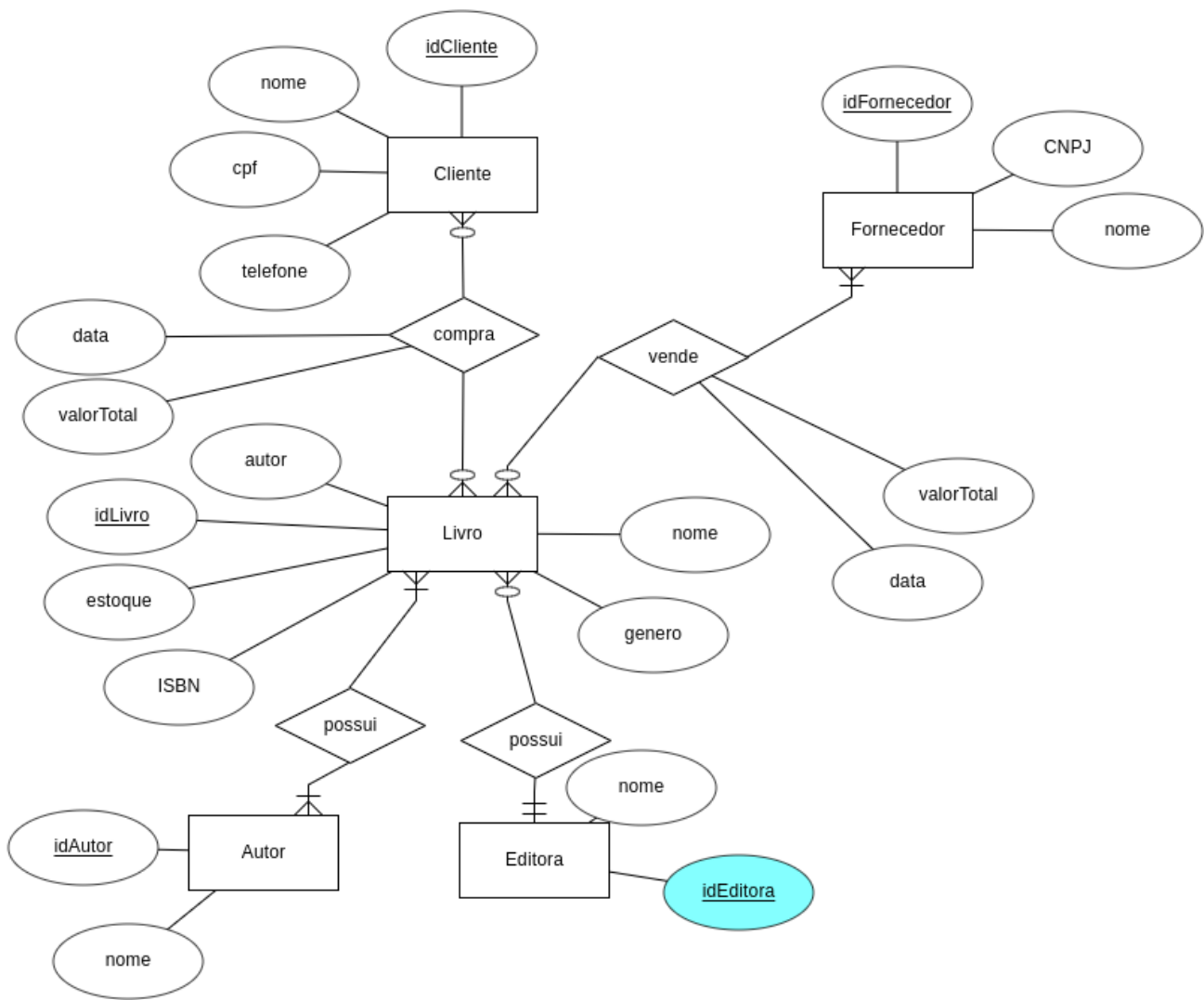
Assunto(**idAssunto**, nome)

Atendente(**idAtendente**, nome)

Senha(**idSenha**,**dataHora**, idAssunto, idCliente, **idAtendimento**, duracao, idAtendente)

- *idAssunto* referencia Assunto
- *idAtendente* referencia Atendente

1.4



Relacionamento 0..n __ 0..n entre Cliente e Livro: Tabela própria
 Relacionamento 1..n __ 0..n entre Fornecedor e Livro: Tabela própria
 Relacionamento 0..n __ 1..1 entre Livro e Editora: Adição de coluna
 Relacionamento 1..n __ 1..n entre Livro e Autor: Tabela própria

Modelo relacional simplificado:

Cliente(**idCliente**, nome, cpf, telefone)

Fornecedor(**idFornecedor**, CNPJ, nome)

Editora(**idEditora**, nome)

Livro(**idLivro**, autor, estoque, ISBN, genero, nome, idEditora)

- *idEditora referencia Editora*

VendaFornecedorLivro(**idFornecedor**, **idLivro**, data, valorTotal)

- *idFornecedor referencia Fornecedor*

- *idLivro referencia Livro*

CompraClienteLivro(**idCliente**, **idLivro**, data, valorTotal)

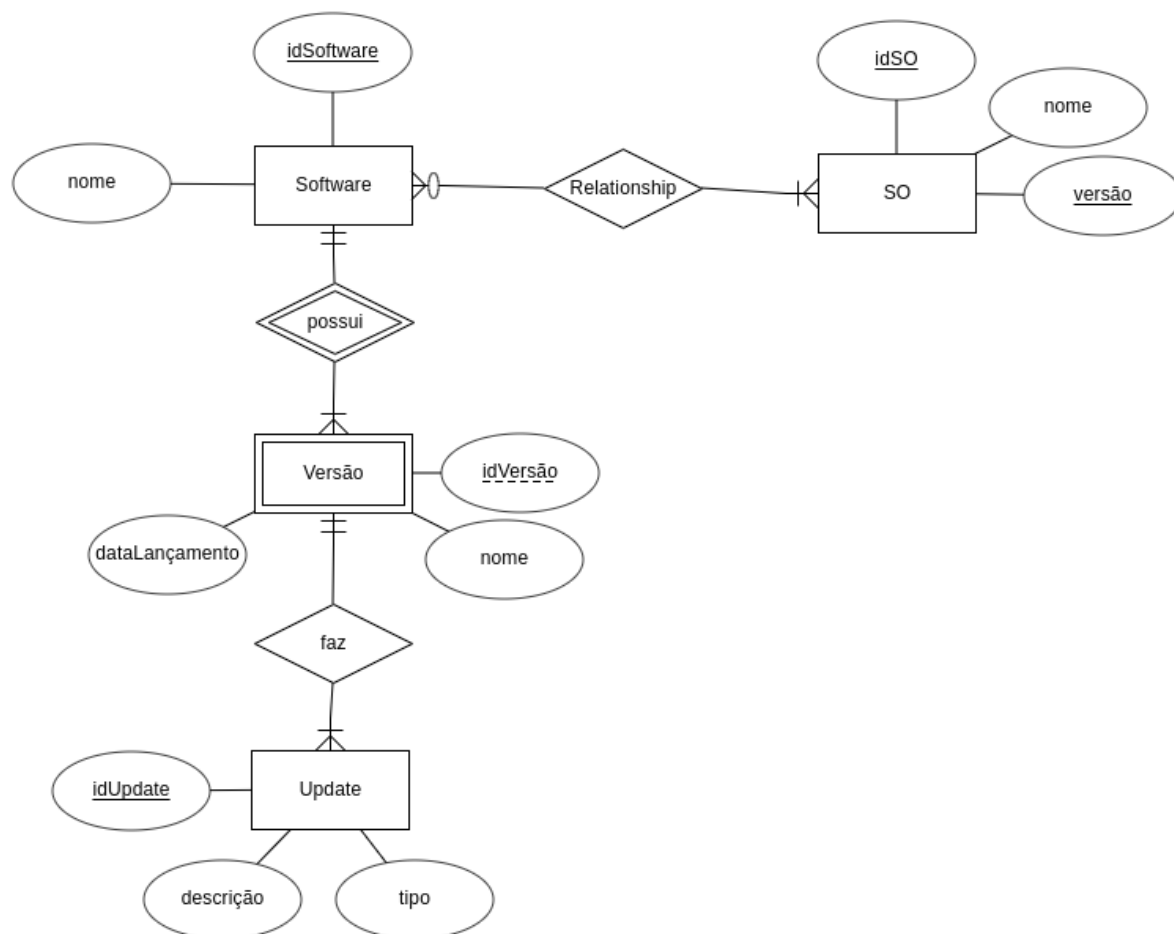
- *idCliente referencia Cliente*
- *idLivro referencia Livro*

Autor(**idAutor**, nome)

AutorLivro(**idAutor**, **idLivro**)

- *idAutor referencia Autor*
- *idLivro referencia Livro*

1.5



Relacionamento 1..n __ 0..n entre SO e Software: Tabela Própria
 Relacionamento 1..1 __ 1..n entre Software e Versão: Adição de coluna
 Relacionamento 1..1 __ 1..n entre Versão e Update: Adição de coluna

Modelo relacional simplificado:

SO(**idSO**, **versao**, nome)

Software(**idSoftware**, nome)

SoftwareSO(**idSoftware**, **idSO**, versaoSO)

- *idSoftware referencia software*
- *idSO referencia SO*
- *versaoSO referencia versao em SO*

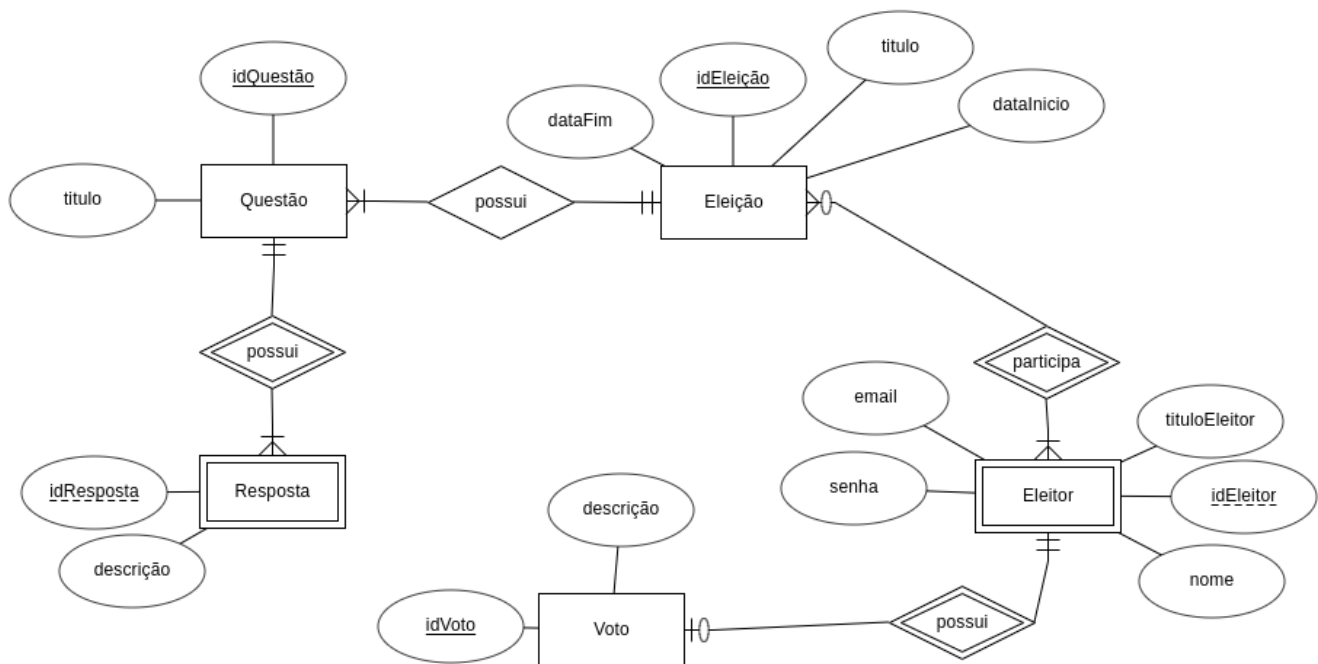
Versao(**idVersao**, nome, dataLancamento, idSoftware)

- *idSoftware referencia Software*

Update(**idUpdate**, descrição, tipo, idVersao)

- *idVersao referencia Versao*

1.6



Relacionamento 1..1 __ 1..n entre Questão e Resposta: Adição de coluna
 Relacionamento 1..n __ 1..1 entre Questão e Eleição: Adição de coluna
 Relacionamento 0..n __ 1..n entre Eleição e Eleitor: Tabela própria
 Relacionamento 1..1 __ 0..1 entre Eleitor e Voto: Fusão de tabelas

Modelo relacional simplificado:

Eleição(**idEleição**, dataFim, titulo, dataInicio)

Questão(**idQuestao**, titulo, **idEleição**)

- *idEleição referencia Eleição*

Resposta(**idResposta**, descrição, **idQuestao**)

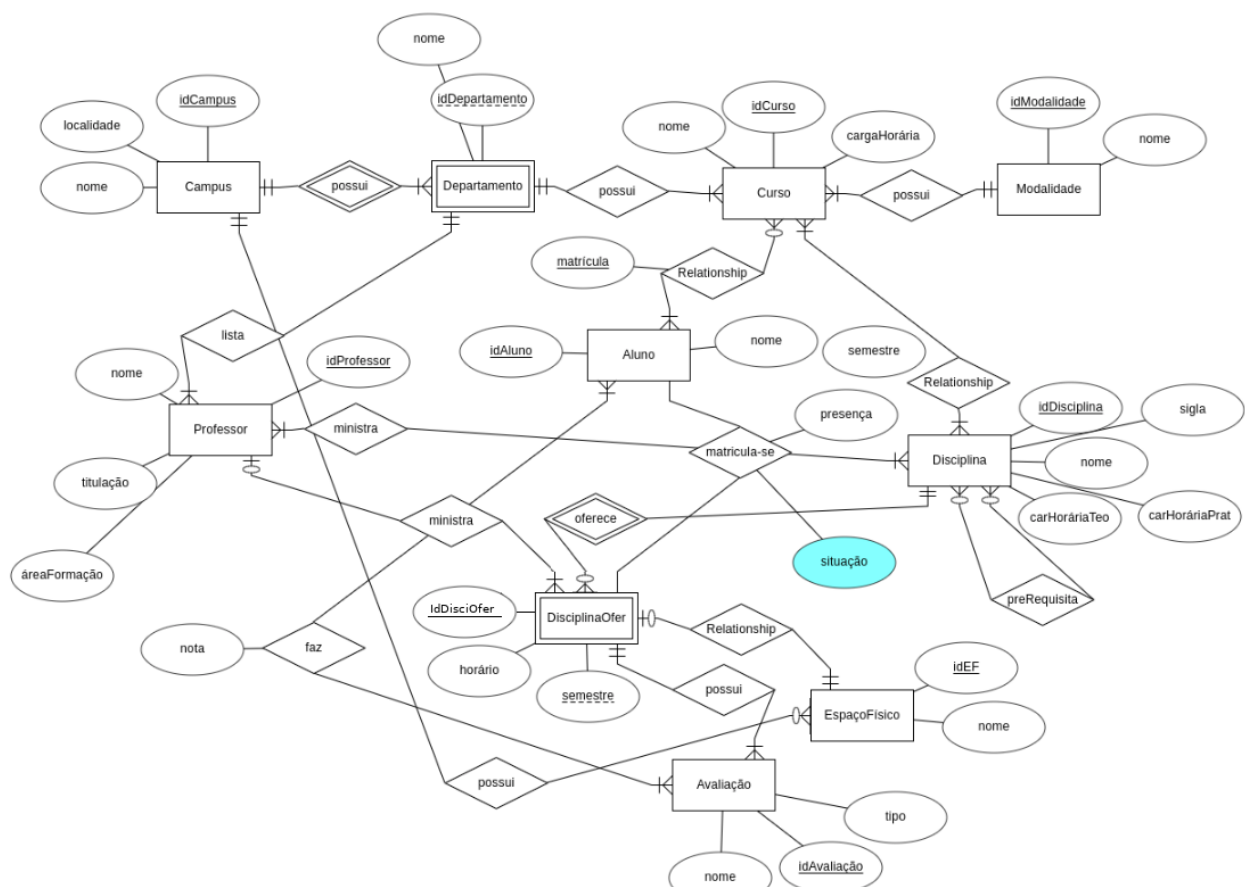
- *idQuestao referencia Questão*

Eleitor(**idEleitor**, nome, tituloEleitor, email, senha, idVoto, descrição)

EleitorEleição(**idEleitor**, **idEleição**)

- *idEleitor referencia Eleitor*
- *idEleição referencia Eleição*

1.7



Relacionamento 1..1 __ 1..n entre Campus e Departamento: adição de coluna
 Relacionamento 1..1 __ 1..n entre Departamento e Curso: adição de coluna
 Relacionamento 1..n __ 1..1 entre Curso e Modalidade: adição de coluna
 Relacionamento 1..1 __ 1..n entre Departamento e Professor: adição de coluna
 Relacionamento 0..n __ 1..n entre Curso e Aluno: Tabela própria
 Relacionamento 1..n __ 1..n entre Curso e Disciplina: Tabela própria
 Relacionamento 0..n __ 0..n entre Disciplina e Disciplina: Tabela própria
 Relacionamento 1..n __ 1..n entre Disciplina e professor: Tabela Própria
 Relacionamento 1..1 __ 0..n entre Disciplina e DisciplinaOfer: adição de colunas
 Relacionamento 0..1 __ 1..1 entre DisciplinaOfer e EspaçoFísico: Fusão de

```
Tabelas
Relacionamento 1..1 __ 1..n entre DisciplinaOfer e Avaliação: Adição de
coluna
Relacionamento 0..1 __ 1..n entre DisciplinaOfer e Professor: Adição de
coluna
Relacionamento 1..n __ 1..n entre Avaliação e Aluno: Tabela própria
Relacionamento 0..n __ 1..1 entre EspaçoFísico e Campus: Adição de coluna
```

Modelo relacional simplificado:

Campus(**idCampus**, localidade, nome)

Departamento(**idDepartamento**, nome, idCampus)

- *idCampus referencia Campus*

Modalidade(**idModalidade**, nome)

Curso(**idCurso**, nome, cargaHorária, idDepartamento, idModalidade)

- *idDepartamento referencia Departamento*
- *_idModalidade referencia Modalidade_*

Aluno(**idAluno**, nome)

AlunoCurso(**idAluno**, **idCurso**, **matricula**)

- *idAluno referencia Aluno*
- *idCurso referencia Curso*

Avaliação(**idAvaliação**, tipo, nome, idDisciOfer)

- *idDisciOfer referencia DisciplinaOfer*

AlunoAvaliação(**idAluno**, **idAvaliação**, nota)

- *idAluno referencia Aluno*
- *idAvaliação referencia Avaliação*

Disciplina(**idDisciplina**, sigla, nome, carHoráriaTeo, cargaHoráriaPrat)

DisciplinaPreReq(**idDisciplina**, **idDisciplina**)

- *idDisciplina referencia Disciplina*
- *idDisciplina referencia Disciplina*

DisciplinaCurso(**idDisciplina**, **idCurso**)

- *idDisciplina referencia Disciplina*
- *idCurso referencia Curso*

Professor(**idProfessor**, nome, titulação, áreaFormação, idDepartamento)

- *idDepartamento referencia Departamento*

ProfessorDisciplina(**idProfessor**, **idDisciplina**)

- *idProfessor* referencia Professor
- *idDisciplina* referencia Disciplina

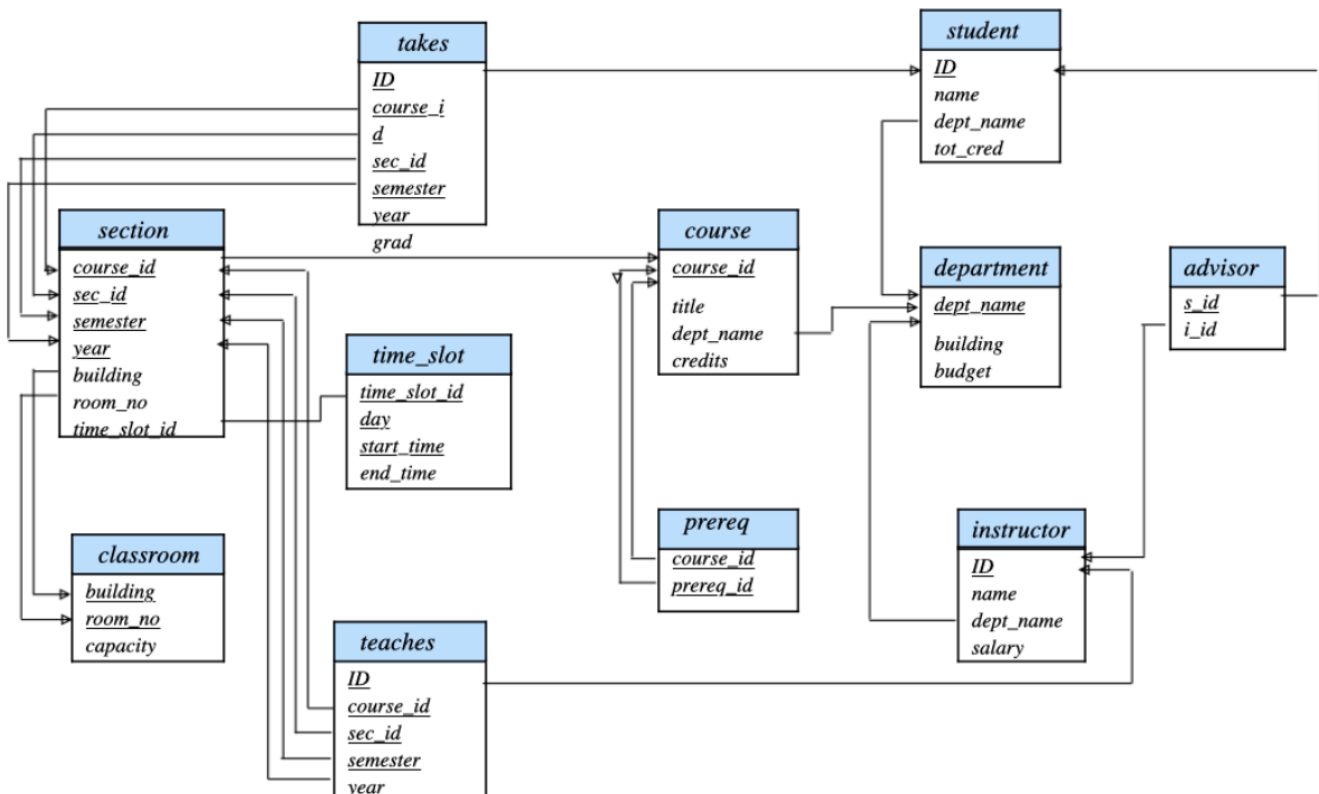
EspaçoFísico(**idEF**, nome, idCampus)

- *idCampus* referencia Campus

DisciplinaOfer(**idDisciOfer**, **semestre**, horário, idDisciplina, idEF, idProfessor)

- *idDisciplina* referencia Disciplina
- *idEF* referencia EspaçoFísico
- *idProfessor* referencia Professor

2. Considere o modelo relacional representado na figura abaixo e apresente as expressões em álgebra relacional para atender cada item desta questão. Faça uso da ferramenta disponível em <https://bcd29008.github.io/relax> e carregue o *dataset* chamado "Silberschatz - University DB".



(a) Da tabela *instructor*, liste todos os instrutores do departamento que tem o nome "Physics".

$\sigma_{dept_name = 'Physics'}(instructor)$

(b) Da tabela *instructor*, liste somente as seguintes colunas: *ID*, *name* e *salary*

$\pi_{ID, name, salary}(instructor)$

(c) Da tabela *instructor*, liste somente as seguintes colunas: *ID*, *name* e *salary* dos instrutores que possuírem salário maior que 66.000 e que trabalhem no departamento com nome "Com. Sci.".

pi ID,name,salary (*sigma* salary > 66000 (instructor) join *sigma* dept_name = 'Com.Sci.' (instructor))

(d) Liste a coluna *course_id*, da tabela *section*, de todos os cursos que foram ministrados no semestre: outono de 2009 (fall), ou na primavera de 2010 (spring) ou em ambos os semestres.

(e) Liste o nome do curso, o nome do departamento, o prédio onde o departamento está situado e o orçamento do departamento.

pi title, dept_name (course) join *pi* building, budget (department)