

1) Notação textual resumida:

Assunto(idAssunto, desc)
Atendente(idAtendente, nome)
Senha(numSenha, horario, idAssunto)
 idAssunto referencia Assunto
Atendimento(numSenha, idAtendente, início, fim)
 idAtendente referencia Atendente
 numSenha referencia Senha

2) Notação textual resumida:

Empregado(idEmpregado, nome)
Dependente(idDependente, nome, idEmpregado)
 idEmpregado referencia Empregado

Instruções DDL:

```
CREATE TABLE Empregado(  
    idEmpregado INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,  
    nome TEXT  
);  
  
CREATE TABLE Dependente(  
    idDependente INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,  
    nome TEXT,  
    idEmpregado INTEGER NOT NULL,  
    FOREIGN KEY(idEmpregado) REFERENCES Empregado(idEmpregado)  
);
```

3) Notação textual resumida:

Chefe(idChefe, nome)
Piloto(idPiloto, nome)
Equipe(idEquipe, nome)
Temporada(idTemporada, dataInicio, dataFim)
Corrida(idCorrida, idTemporada, circuito, idPosição)
 idTemporada referencia Temporada
 idPosição referencia Posição
Posição(idPosição, idCorrida, idPiloto)
Cadastro(idCadastro, idChefe, idPiloto1, idPiloto2, idEquipe, idTemporada)
 idPiloto1 referencia Piloto
 idPiloto2 referencia Piloto
 idEquipe referencia Equipe
 idTemporada referencia Temporada

4) Notação textual resumida:

Aluno(idAluno)
Professor(idProfessor)
Disciplina(idDisciplina, idProfessor)
 idProfessor referencia Professor
Prova(idProva, nota, idAluno, idDisciplina)

```

        idAluno referencia Aluno
        idDisciplina referencia Disciplina
Assiste(data, idAluno, idDisciplina, faltas)
        idAluno referencia Aluno
        idDisciplina referencia Disciplina

```

Instruções DDL:

```

CREATE TABLE Professor(
    idProfessor INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT
);

CREATE TABLE Aluno(
    idAluno INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT
);

CREATE TABLE Disciplina(
    idDisciplina INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT
    idProfessor INTEGER,
    FOREIGN KEY(idProfessor) REFERENCES Professor(idProfessor)
);

CREATE TABLE Prova(
    idProva INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    nota INTEGER,
    idAluno INTEGER,
    FOREIGN KEY(idAluno) REFERENCES Aluno(idAluno),
    idDisciplina INTEGER,
    FOREIGN KEY(idDisciplina) REFERENCES Disciplina(idDisciplina)
);

CREATE TABLE Assiste(
    data INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    faltas INTEGER NOT NULL,
    idAluno INTEGER,
    FOREIGN KEY(idAluno) REFERENCES Aluno(idAluno),
    idDisciplina INTEGER,
    FOREIGN KEY(idDisciplina) REFERENCES Disciplina(idDisciplina)
);

```

5) Notação textual resumida:

```

Cargo(idCargo, nome)
Pesquisador(idPesquisador, nome, dNasc)
InstituiçãoAcademica(idInstAcad, país, nome, sigla)
DepartamentoAcademico(idDeptoAcad, nome, idInstAcad)
    idInstAcad referencia InstituiçãoAcademica
HistoricoCargo(idHistoricoCargo, idInstAcad, idPesquisador, idCargo, dEntrada, dSaída)
    idInstAcad referencia InstituiçãoAcademica
    idPesquisador referencia Pesquisador
    idCargo referencia Cargo
Projeto(idProjeto, título, início, término)
Coordenação(idCoordenação, idProjeto, idPesquisador)

```

```

        idProjeto referencia Projeto
        idPesquisador referencia Pesquisador
Publicação(idPublicação, título)
Autoria(idAutoria, idPublicação, idPesquisador)
        idPublicação referencia Publicação
        idPesquisador referencia Pesquisador
Titulação(Título, data, idInstAcad, idProfessor)
        idInstAcad referencia InstituiçãoAcademica
        idProfessor referencia Pesquisador

```

Instruções DDL:

```

CREATE TABLE Cargo(
    idCargo INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    nome TEXT NOT NULL
);

CREATE TABLE Pesquisador(
    idPesquisador INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    nome TEXT NOT NULL,
    dNasc INTEGER NOT NULL
);

CREATE TABLE InstituiçãoAcademica(
    idInstAcad INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    país TEXT NOT NULL,
    nome TEXT NOT NULL,
    sigla TEXT NOT NULL
);

CREATE TABLE Departamento(
    idDeptoAcad INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    nome TEXT NOT NULL,
    idInstAcad INTEGER NOT NULL,
    FOREIGN KEY(idInstAcad) REFERENCES InstituiçãoAcademica(idInstAcad)
);

CREATE TABLE HistoricoCargo(
    idHistoricoCargo INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    idInstAcad INTEGER NOT NULL,
    FOREIGN KEY(idInstAcad) REFERENCES InstituiçãoAcademica(idInstAcad),
    idPesquisador INTEGER NOT NULL,
    FOREIGN KEY(idPesquisador) REFERENCES Pesquisador(idPesquisador),
    idCargo INTEGER NOT NULL,
    FOREIGN KEY(idCargo) REFERENCES Cargo(idCargo),
    dEntrada TEXT NOT NULL,
    dSaída TEXT NOT NULL
);

CREATE TABLE Projeto(
    idProjeto INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    título TEXT NOT NULL,
    início TEXT NOT NULL,

```

```

        término TEXT NOT NULL
    );

CREATE TABLE Coordenação(
    idCoordenação INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    idProjeto INTEGER NOT NULL,
    FOREIGN KEY(idProjeto) REFERENCES Projeto(idProjeto),
    idPesquisador INTEGER NOT NULL,
    FOREIGN KEY(idPesquisador) REFERENCES Pesquisador(idPesquisador)
);

CREATE TABLE Publicação(
    idPublicação INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    título TEXT NOT NULL
);

CREATE TABLE Autoria(
    idAutoria INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    idPublicação INTEGER NOT NULL,
    FOREIGN KEY(idPublicação) REFERENCES Publicação(idPublicação),
    idPesquisador INTEGER NOT NULL,
    FOREIGN KEY(idPesquisador) REFERENCES Pesquisador(idPesquisador)
);

CREATE TABLE Titulação(
    Título INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    data INTEGER NOT NULL,
    idInstAcad INTEGER NOT NULL,
    FOREIGN KEY(idInstAcad) REFERENCES InstituiçãoAcademica(idInstAcad),
    idProfessor INTEGER NOT NULL,
    FOREIGN KEY(idProfessor) REFERENCES Pesquisador(idPesquisador)
);

```

6) Relax (dataset Silberschatz - UniversityDB)

- a) σ dept_name = 'Physics' instructor
- b) π ID, name, salary instructor
- c) π ID, name, salary (σ salary > 6600 \wedge dept_name = 'Comp. Sci.' instructor)
- d) π course_id (σ semester = 'Fall' \vee semester = 'Spring' section)
- e) π course.title, course.dept_name, department.building, department.budget (σ course.dept_name = department.dept_name (π title, dept_name course \times π dept_name, building, budget department))