

Trabalho Prático 01

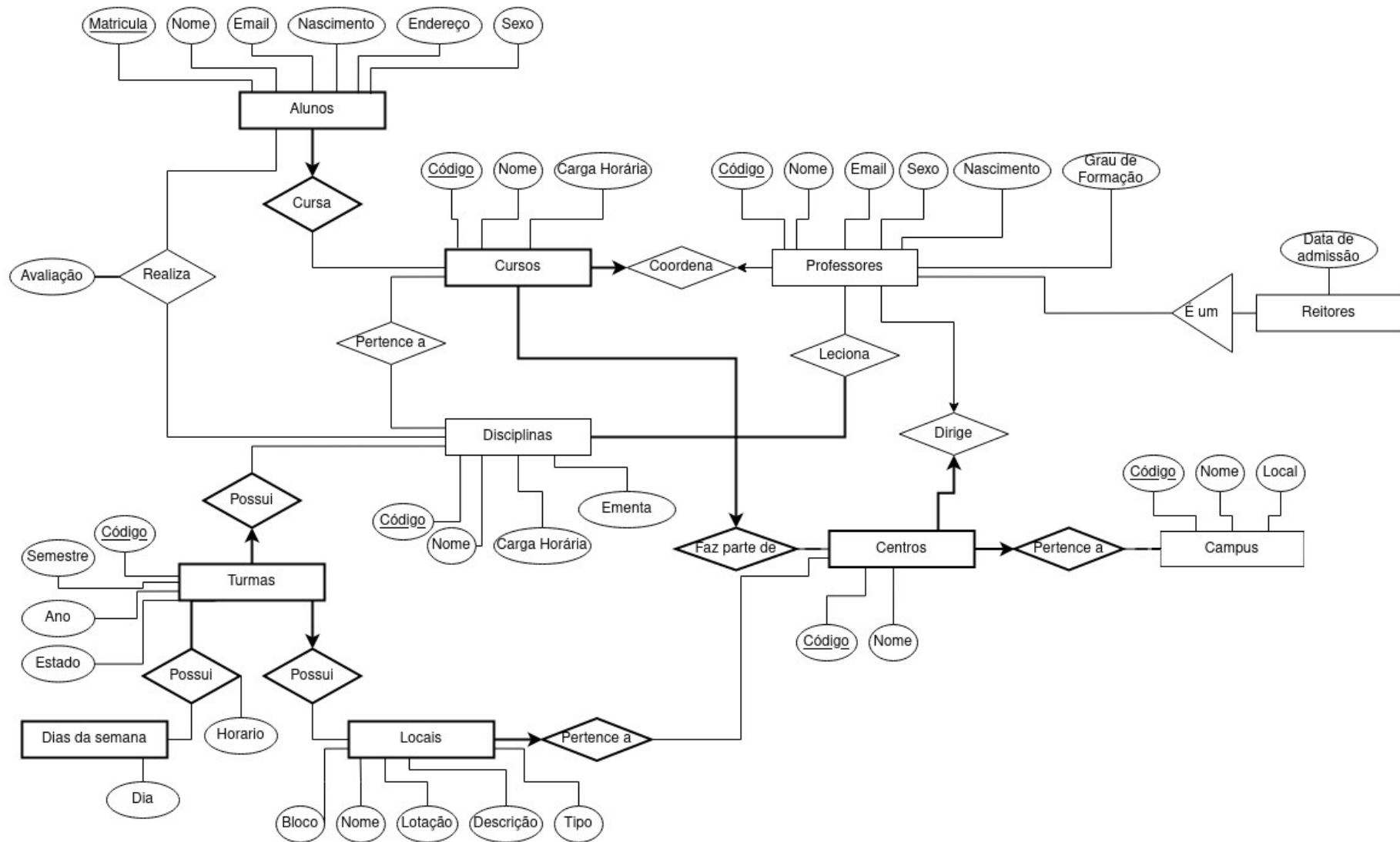
Fundamentos de Bancos de Dados

Professor: Javam de Castro Machado

Dupla: Breno Macêdo de Brito e Gustavo Wendell Moreira Valença

30 de setembro de 2022

Diagrama do Sistema Universitário



Dicionário de dados: Sistema Universitário

Relacionamentos

Entidade	Relacionamento	Nome do Relacionamento	Descrição
Aluno	Curso	Cursa	Tabela para registro dos alunos da universidade.
	Disciplina	Realiza	
Curso	Aluno	Cursa	Tabela para registro dos cursos da universidade.
	Disciplina	Pertence a	
	Professor	Coordena	
	Centro	Faz parte de	
Professor	Curso	Coordena	Tabela para registro dos professores da universidade.
	Disciplina	Leciona	
	Centro	Dirige	
	Reitor	É um	
Reitor	Professor	É um	Tabela para registro dos reitores da universidade.
Disciplina	Curso	Pertence a	Tabela para registro das disciplinas da universidade.
	Aluno	Realiza	
	Professor	Leciona	
	Turma	Possui	
Centro	Professor	Dirige	Tabela para registro dos centros da universidade.
	Curso	Faz parte de	
	Campus	Pertence a	
	Local	Pertence a	

Dicionário de dados: Sistema Universitário

Campus	Centro	Pertence a	Tabela para registro dos campi da universidade.
Turma	Disciplina	Possui	Tabela para registro das turmas da universidade.
	Local	Possui	
	Dia da semana	Possui	
Local	Turma	Possui	Tabela para registro dos locais da universidade.
	Centro	Pertence a	
Dia da semana	Turma	Possui	Tabela para registro de dias da semana.

Atributos

Entidade Aluno				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
matricula	Inteiro	4 bytes	PK, NOT NULL	Matricula do aluno.
nome	Caractere	255 bytes	NOT NULL	Nome do aluno.
email	Caractere	255 bytes	NOT NULL	E-mail do aluno.
nascimento	Data	4 bytes		Data de nascimento do aluno.
endereço	Caractere	255 bytes		Endereço do aluno.
sexo	Genero	9 bytes		Sexo do aluno.

Dicionário de dados: Sistema Universitário

cursoId	Inteiro	4 bytes	PK, FK, NOT NULL	Código de identificação do curso que o aluno realiza.
---------	---------	---------	------------------	---

Entidade Curso				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
codigo	Inteiro	4 bytes	PK, NOT NULL	Código do curso.
nome	Caractere	255 bytes	NOT NULL	Nome do curso.
cargaHoraria	Inteiro	4 bytes		Carga horaria do curso.
coordenadorId	Inteiro	4 bytes	PK, FK, NOT NULL	Código identificador do coordenador do curso.
centroId	Inteiro	4 bytes	PK, FK, NOT NULL	Código identificador do centro do curso.

Entidade Disciplina				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
codigo	Inteiro	4 bytes	PK, NOT NULL	Código da disciplina.
nome	Caractere	255 bytes	NOT NULL	Nome da disciplina.

Dicionário de dados: Sistema Universitário

cargaHoraria	Inteiro	4 bytes	CHECK (cargaHoraria >= 32 AND cargaHoraria <= 128)	Carga horaria da disciplina.
ementa	Caractere	255 bytes		Ementa da disciplina.

Entidade Professor				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
codigo	Inteiro	4 bytes	PK, NOT NULL	Matricula do aluno.
nome	Caractere	255 bytes	NOT NULL	Nome do aluno.
email	Caractere	255 bytes	CHECK (cargaHoraria >= 32 AND cargaHoraria <= 128)	E-mail do aluno.
sexo	Genero	9 bytes		Data de nascimento do aluno.
formacao	Formacao	9 bytes	NOT NULL	Grau máximo de formação do professor.
nascimento	Data	8 bytes		Data de nascimento do professor.

Dicionário de dados: Sistema Universitário

Entidade Centro				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
codigo	Inteiro	4 bytes	PK, NOT NULL	Código identificador do centro.
nome	Caractere	255 bytes	NOT NULL	Nome do centro.
campusId	Inteiro	4 bytes	PK, FK, NOT NULL	Código de identificação do campus a qual o centro pertence.
diretorId	Inteiro	4 bytes	PK, FK, NOT NULL	Código de identificação do diretor do centro.

Entidade Campus				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
codigo	Inteiro	4 bytes	PK, NOT NULL	Código identificador do campus.
nome	Caractere	255 bytes	NOT NULL	Nome do campus.
localizacao	Caractere	255 bytes	NOT NULL	Localização do campus.

Dicionário de dados: Sistema Universitário

Entidade Reitor				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
dataDeAdmissao	Data	8 bytes		Data de admissão do reitor.
professorId	Inteiro	4 bytes	PK, FK, NOT NULL	Código de identificação do professor que o reitor se refere.

Entidade Turma				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
codigo	Inteiro	4 bytes	PK, NOT NULL	Código da turma.
periodo	Caractere	7 bytes		Período da turma.
estado	Turma_Estado	9 bytes		Estado da turma.
disciplinaId	Inteiro	4 bytes	PK, FK, NOT NULL	Código identificador da disciplina que a turma realiza.
localId	Inteiro	4 bytes	PK, FK, NOT NULL	Código identificador do local da turma.

Dicionário de dados: Sistema Universitário

Entidade Local				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
codigo	Inteiro	4 bytes	PK, NOT NULL	Código do local.
nome	Caractere	255 bytes	NOT NULL	Nome do local.
bloco	Inteiro	4 bytes		Bloco que o local está localizado.
lotacao	Inteiro	4 bytes	NOT NULL	Lotação máxima do local.
descricao	Caractere	255 bytes		Descrição do local.
tipo	Local_Tipo	12 bytes		Tipo do local.
centroId	Inteiro	4 bytes	PK, FK, NOT NULL	Código identificador do centro que o local pertence.

Entidade Dia da Semana				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
Id	Inteiro	4 bytes	PK, NOT NULL, SERIAL	Código identificador do dia da semana.
dia	Dia_da_Semana	7 bytes	UNIQUE	Nome do dia da semana.

Dicionário de dados: Sistema Universitário

Entidade relacao_alunos_disciplinas				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
Id	Inteiro	4 bytes	PK, NOT NULL, SERIAL	Código identificador da relação.
alunold	Inteiro	4 bytes	PK, FK, NOT NULL, UNIQUE (alunold, disciplinald)	Código identificador do aluno.
disciplinald	Inteiro	4 bytes	PK, FK, NOT NULL, UNIQUE (alunold, disciplinald)	Código identificador da disciplina.
avaliacao	Real	4 bytes	CHECK (avaliacao >= 0 AND avaliacao <= 10);	Resultado final da disciplina.

Dicionário de dados: Sistema Universitário

Entidade relacao_cursos_disciplinas				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
Id	Inteiro	4 bytes	PK, NOT NULL, SERIAL	Código identificador da relação.
cursoId	Inteiro	4 bytes	PK, FK, NOT NULL, UNIQUE (cursoId, disciplinaId)	Código identificador do curso.
disciplinaId	Inteiro	4 bytes	PK, FK, NOT NULL, UNIQUE (cursoId, disciplinaId)	Código identificador da disciplina.

Entidade relacao_professores_disciplinas				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
Id	Inteiro	4 bytes	PK, NOT NULL, SERIAL	Código identificador da relação.
professorId	Inteiro	4 bytes	PK, FK, NOT NULL, UNIQUE (professorId, disciplinaId), CHECK (verificaDisciplinas (professorId))	Código identificador do professor.
disciplinaId	Inteiro	4 bytes	PK, FK, NOT NULL, UNIQUE (professorId, disciplinaId)	Código identificador da disciplina.

Dicionário de dados: Sistema Universitário

Entidade relacao_turmas_diasSemana				
Atributo	Tipos de dados	Comprimento	Restrições	Descrição
Id	Inteiro	4 bytes	PK, NOT NULL, SERIAL	Código identificador da relação.
turmaId	Inteiro	4 bytes	PK, FK, NOT NULL, UNIQUE (turmaId, diaSemanaId)	Código identificador da turma.
diaSemanaId	Inteiro	4 bytes	PK, FK, NOT NULL, UNIQUE (turmaId, diaSemanaId)	Código identificador do dia da semana.
horarioDeInicio	Time	8 bytes	NOT NULL	Horario de inicio da turma nesse dia.
horarioDeTermino	Time	8 bytes	NOT NULL	Horario de termino da turma nesse dia.

Comandos SQL usados na implementação

```
1  -- ALUNOS
2
3  CREATE TYPE GENERO AS ENUM ('masculino', 'feminino', 'nao_binario', 'outro');
4  CREATE TYPE FORMACAO AS ENUM ('mestrado', 'doutorado');
5  CREATE TYPE TURMA_ESTADO AS ENUM ('aberta', 'concluida');
6  CREATE TYPE LOCAL_TIPO AS ENUM ('bloco', 'sala_de_aula', 'auditorio', 'laboratorio');
7  CREATE TYPE DIAS_SEMANA AS ENUM ('segunda', 'terca', 'quarta', 'quinta', 'sexta', 'sabado');
8
9  CREATE TABLE alunos (
10     matricula INTEGER PRIMARY KEY,
11     nome VARCHAR(255) NOT NULL,
12     email VARCHAR(255) NOT NULL,
13     nascimento DATE,
14     endereco VARCHAR(255),
15     sexo GENERO,
16     cursoId INTEGER NOT NULL
17 );
18
19 -- CURSOS
20
21 CREATE TABLE cursos (
22     codigo INTEGER PRIMARY KEY,
23     nome VARCHAR(255) NOT NULL,
24     cargaHoraria INTEGER,
25     coordenadorId INTEGER NOT NULL,
26     centroId INTEGER NOT NULL
27 );
28
29 -- DISCIPLINAS
30
31 CREATE TABLE disciplinas (
32     codigo INTEGER PRIMARY KEY,
33     nome VARCHAR(255) NOT NULL,
34     cargaHoraria INTEGER,
35     ementa VARCHAR(255),
36     CONSTRAINT cargaHoraria_intervalo CHECK (cargaHoraria >= 32 AND cargaHoraria <= 128)
37 );
38
```

Comandos SQL usados na implementação

```
39 -- PROFESSORES
40
41 CREATE TABLE professores (
42     codigo INTEGER PRIMARY KEY,
43     nome VARCHAR(255) NOT NULL,
44     email VARCHAR(255) NOT NULL,
45     sexo GENERO,
46     formacao FORMACAO NOT NULL,
47     nascimento DATE
48 );
49
50 -- CENTROS
51
52 CREATE TABLE centros (
53     codigo INTEGER PRIMARY KEY,
54     nome VARCHAR(255) NOT NULL,
55     campusId INTEGER NOT NULL,
56     diretorId INTEGER NOT NULL
57 );
58
59 -- CAMPUS
60
61 CREATE TABLE campi (
62     codigo INTEGER PRIMARY KEY,
63     nome VARCHAR(255) NOT NULL,
64     localizacao VARCHAR(255) NOT NULL
65 );
66
67 -- REITORES
68
69 CREATE TABLE reitores (
70     dataDeAdmissao DATE,
71     professorId INTEGER NOT NULL
72 );
73
```

Comandos SQL usados na implementação

```
74 -- TURMAS
75
76 CREATE TABLE turmas (
77     codigo INTEGER PRIMARY KEY,
78     periodo VARCHAR(7), -- ____-__
79     estado TURMA_ESTADO,
80     disciplinaId INTEGER NOT NULL,
81     localId INTEGER NOT NULL
82 );
83
84 -- LOCAIS
85
86 CREATE TABLE locais (
87     codigo INTEGER PRIMARY KEY,
88     nome VARCHAR(255) NOT NULL,
89     bloco INTEGER,
90     lotacao INTEGER NOT NULL,
91     descricao VARCHAR(255),
92     tipo LOCAL_TIPO,
93     centroId INTEGER NOT NULL
94 );
95
96 -- RELACAO CURSA: ALUNOS -- N:1 -- CURSOS
97
98 ALTER TABLE alunos
99     ADD CONSTRAINT fk_curso_aluno FOREIGN KEY (cursoId)
100     REFERENCES cursos ON DELETE CASCADE;
101
```

Comandos SQL usados na implementação

```
102 -- RELACAO REALIZA: ALUNOS -- N:N -- DISCIPLINAS
103
104 CREATE TABLE relacao_alunos_disciplinas (
105     id SERIAL PRIMARY KEY,
106     alunoId INTEGER NOT NULL,
107     disciplinaId INTEGER NOT NULL,
108     UNIQUE (alunoId, disciplinaId),
109     CONSTRAINT fk_aluno FOREIGN KEY (alunoId) REFERENCES alunos ON DELETE CASCADE,
110     CONSTRAINT fk_disciplina FOREIGN KEY (disciplinaId) REFERENCES disciplinas ON DELETE CASCADE,
111     avaliacao REAL,
112     CONSTRAINT avaliacao_intervalo CHECK (avaliacao >= 0 AND avaliacao <= 10)
113 );
114
115 -- RELACAO PERTENCE A: CURSOS -- N:N -- DISCIPLINAS
116
117 CREATE TABLE relacao_cursos_disciplinas (
118     id SERIAL PRIMARY KEY,
119     cursoId INTEGER NOT NULL,
120     disciplinaId INTEGER NOT NULL,
121     UNIQUE (cursoId, disciplinaId),
122     CONSTRAINT fk_curso FOREIGN KEY (cursoId) REFERENCES cursos ON DELETE CASCADE,
123     CONSTRAINT fk_disciplina FOREIGN KEY (disciplinaId) REFERENCES disciplinas ON DELETE CASCADE
124 );
125
126 -- RELACAO COORDENA: CURSOS -- 1:1 -- PROFESSORES
127
128 ALTER TABLE cursos
129     ADD CONSTRAINT fk_curso_coordenador FOREIGN KEY (coordenadorId)
130     REFERENCES professores ON DELETE CASCADE;
131
132 -- RELACAO FAZ PARTE DE: CURSOS N:1 CENTROS
133 ALTER TABLE cursos
134     ADD CONSTRAINT fk_curso_centro FOREIGN KEY (centroId)
135     REFERENCES centros ON DELETE CASCADE;
```


Comandos SQL usados na implementação

```
138 -- RELACAO LECIONA: PROFESSRES N:N DISCIPLINAS
139
140 CREATE FUNCTION verificaDisciplinas (profId INTEGER)
141 RETURNS BOOLEAN
142 LANGUAGE plpgsql
143 AS
144 $$
145 DECLARE
146     quantidadeDisciplinas INTEGER;
147 BEGIN
148     SELECT COUNT(*) INTO quantidadeDisciplinas FROM relacao_professores_disciplinas WHERE professorId = profId;
149     RETURN quantidadeDisciplinas <= 4;
150 END;
151 $$;
152
153 CREATE TABLE relacao_professores_disciplinas (
154     id SERIAL PRIMARY KEY,
155     professorId INTEGER NOT NULL,
156     disciplinaId INTEGER NOT NULL,
157     UNIQUE (professorId, disciplinaId),
158     CONSTRAINT fk_professor FOREIGN KEY (professorId) REFERENCES professores ON DELETE CASCADE,
159     CONSTRAINT fk_disciplina FOREIGN KEY (disciplinaId) REFERENCES disciplinas ON DELETE CASCADE,
160     CONSTRAINT maximo_disciplinas CHECK (verificaDisciplinas (professorId))
161 );
162
163 -- RELACAO É PROFESSOR: REITORES 1:1 PROFESSORES
164
165 ALTER TABLE reitores
166     ADD CONSTRAINT fk_professor FOREIGN KEY (professorId)
167     REFERENCES professores ON DELETE CASCADE;
168
169 -- RELACAO POSSUI: TURMAS N:1 DISCIPLINAS
170
171 ALTER TABLE turmas
172     ADD CONSTRAINT fk_disciplina FOREIGN KEY (disciplinaId)
173     REFERENCES disciplinas ON DELETE CASCADE;
```

Comandos SQL usados na implementação

```
175 -- RELACAO POSSUI: TURMAS N:1 LOCAL
176
177 ALTER TABLE turmas
178     ADD CONSTRAINT fk_local FOREIGN KEY (localId)
179     REFERENCES locais ON DELETE CASCADE;
180
181 -- RELACAO POSSUI: TURMAS N:N DIAS DA SEMANA
182
183 CREATE TABLE diasSemana (
184     id SERIAL PRIMARY KEY,
185     dia DIAS_SEMANA UNIQUE
186 );
187
188 INSERT INTO diasSemana (dia) VALUES ('segunda');
189 INSERT INTO diasSemana (dia) VALUES ('terca');
190 INSERT INTO diasSemana (dia) VALUES ('quarta');
191 INSERT INTO diasSemana (dia) VALUES ('quinta');
192 INSERT INTO diasSemana (dia) VALUES ('sexta');
193 INSERT INTO diasSemana (dia) VALUES ('sabado');
194
195 CREATE TABLE relacao_turmas_diasSemana (
196     id SERIAL PRIMARY KEY,
197     turmaId INTEGER NOT NULL,
198     diasSemanaId INTEGER NOT NULL,
199     UNIQUE (turmaId, diasSemanaId),
200     horarioDeInicio TIME NOT NULL,
201     horarioDeTermino TIME NOT NULL,
202     CONSTRAINT fk_turma FOREIGN KEY (turmaId) REFERENCES turmas ON DELETE CASCADE,
203     CONSTRAINT fk_diasSemana FOREIGN KEY (diasSemanaId) REFERENCES diasSemana ON DELETE CASCADE
204 );
205
206 -- RELACAO PERTENCE A: LOCAIS N:1 CENTROS
207
208 ALTER TABLE locais
209     ADD CONSTRAINT fk_centro FOREIGN KEY (centroId)
210     REFERENCES centros ON DELETE CASCADE;
```

Comandos SQL usados na implementação

```
212 -- RELACAO DIRIGE: PROFESSORES 1:1 CENTROS
213
214 ALTER TABLE centros
215     ADD CONSTRAINT fk_diretor FOREIGN KEY (diretorId)
216     REFERENCES professores ON DELETE SET NULL;
217
218 -- RELACAO PERTENCE A: CENTRO N:1 CAMPI
219
220 ALTER TABLE centros
221     ADD CONSTRAINT fk_campus FOREIGN KEY (campusId)
222     REFERENCES campi ON DELETE CASCADE;
```

Comandos SQL usados no preenchimento das tabelas

```
1  -- Inserindo campus -----
2
3  INSERT INTO campi (codigo, nome, localizacao) VALUES (69420, 'Campus do Pici', 'Fortaleza');
4  INSERT INTO campi (codigo, nome, localizacao) VALUES (111111, 'Campus de Sobral', 'Sobral');
5  INSERT INTO campi (codigo, nome, localizacao) VALUES (444444, 'Campus do Quixada', 'Quixada');
6
7  -- Inserindo professores -----
8
9  INSERT INTO professores (codigo, nome, email, sexo, formacao, nascimento)
10 VALUES (13, 'Javam de Castro Machado', 'javam@ufc.br', 'masculino', 'doutorado', '1972-02-22');
11
12 INSERT INTO professores (codigo, nome, email, sexo, formacao, nascimento)
13 VALUES (22, 'Joaquim Bento Cavalcante Neto', 'joaquimb@dc.ufc.br', 'masculino', 'doutorado', '1984-09-11');
14
15 INSERT INTO professores (codigo, nome, email, sexo, formacao, nascimento)
16 VALUES (123, 'Creto Augusto Vidal', 'cvidal@lia.ufc.br ', 'masculino', 'doutorado', '1970-09-11');
17
18 INSERT INTO professores (codigo, nome, email, sexo, formacao, nascimento)
19 VALUES (10, 'Pablo Mayckon Silva Farias', 'pablo@dc.ufc.br ', 'masculino', 'doutorado', '1990-09-12');
20
21 -- Inserindo centros -----
22
23 INSERT INTO centros (codigo, nome, campusId, diretorId) VALUES (69, 'Centro de Ciências', 69420, 13);
24 INSERT INTO centros (codigo, nome, campusId, diretorId) VALUES (70, 'Centro de Tecnologia', 69420, 22);
25
26 -- Inserindo disciplinas
27
28 INSERT INTO disciplinas (codigo, nome, cargaHoraria, ementa) VALUES (123, 'Felicidade', 64, 'Você será feliz nessa disciplina (eu acho)');
29 INSERT INTO disciplinas (codigo, nome, cargaHoraria, ementa) VALUES (555, 'Computação Gráfica', 64, 'kkkkkkkkkkkkkkkkkkkkkkkkkkkkkk');
30 INSERT INTO disciplinas (codigo, nome, cargaHoraria, ementa) VALUES (444, 'Fundamentos de Banco de Dados', 64, 'Professor Javam gente boa');
31
32 INSERT INTO relacao_professores_disciplinas (professorId, disciplinaId) VALUES (123, 123);
33 INSERT INTO relacao_professores_disciplinas (professorId, disciplinaId) VALUES (123, 555);
34 INSERT INTO relacao_professores_disciplinas (professorId, disciplinaId) VALUES (13, 444);
35 INSERT INTO relacao_professores_disciplinas (professorId, disciplinaId) VALUES (22, 444);
```


Comandos SQL usados no preenchimento das tabelas

```
37 -- Inserindo locais
38
39 INSERT INTO locais (codigo, nome, bloco, lotacao, descricao, tipo, centroId)
40 VALUES (12345, 'Bloco da Computação - 910', NULL, 100, 'Bloco da computação, onde tem a sala do PET, ar condicionado mt bom', 'bloco', 69);
41
42 INSERT INTO locais (codigo, nome, bloco, lotacao, descricao, tipo, centroId)
43 VALUES (12346, 'LEC', 910, 40, 'Laboratório de Computação, só no computer', 'laboratorio', 69);
44
45 INSERT INTO locais (codigo, nome, bloco, lotacao, descricao, tipo, centroId)
46 VALUES (12347, 'Sala 1045', 915, 40, 'Sala de aula do bloco 915 - Antiga sala do juvencio', 'sala_de_aula', 69);
47
48 INSERT INTO locais (codigo, nome, bloco, lotacao, descricao, tipo, centroId)
49 VALUES (12348, 'Auditorio do 915', 915, 50, 'Auditorio do 915, aquele lugar tem uma acustica horrivel :(', 'auditorio', 69);
50
51 -- Inserindo o reitor
52
53 INSERT INTO reitores (dataDeAdmissao, professorId) VALUES ('2022-02-22', 10);
54
55 -- Inserindo cursos
56
57 INSERT INTO cursos (codigo, nome, cargaHoraria, coordenadorId, centroId) VALUES (6551, 'Ciência da Computação', 3000, 10, 69);
58 INSERT INTO cursos (codigo, nome, cargaHoraria, coordenadorId, centroId) VALUES (3213, 'Biologia', 3000, 13, 69);
59
60 INSERT INTO relacao_cursos_disciplinas (cursoId, disciplinaId) VALUES (3213, 123);
61 INSERT INTO relacao_cursos_disciplinas (cursoId, disciplinaId) VALUES (6551, 123);
62 INSERT INTO relacao_cursos_disciplinas (cursoId, disciplinaId) VALUES (6551, 555);
63 INSERT INTO relacao_cursos_disciplinas (cursoId, disciplinaId) VALUES (6551, 444);
64
65 -- Inserindo turmas
66
67 INSERT INTO turmas (codigo, periodo, estado, disciplinaId, localId) VALUES (321, '2022.2', 'aberta', 123, 12347);
68 INSERT INTO turmas (codigo, periodo, estado, disciplinaId, localId) VALUES (123, '2022.2', 'aberta', 555, 12347);
69
70 INSERT INTO relacao_turmas_diasSemana (turmaId, diaSemanaId, horarioDeInicio, horarioDeTermino) VALUES (321, 2, '08:00', '12:00');
71 INSERT INTO relacao_turmas_diasSemana (turmaId, diaSemanaId, horarioDeInicio, horarioDeTermino) VALUES (123, 1, '10:00', '12:00');
72 INSERT INTO relacao_turmas_diasSemana (turmaId, diaSemanaId, horarioDeInicio, horarioDeTermino) VALUES (123, 3, '10:00', '12:00');
```

Comandos SQL usados no preenchimento das tabelas

```
74 -- Inserindo alunos
75
76 INSERT INTO alunos (matricula, nome, email, nascimento, endereco, sexo, cursoId)
77 VALUES (508682, 'Renan Xerez Marques', 'renanxerez@gmail.com', '2003-05-10', 'Avenida Paulo Muzy Numero 22', 'masculino', 6551);
78
79 INSERT INTO alunos (matricula, nome, email, nascimento, endereco, sexo, cursoId)
80 VALUES (123456, 'Maria Eduarda Leandro', 'mariaeduarda@gmail.com', '2003-12-22', 'Rua dos bobos Numero 0', 'nao_binario', 3213);
81
82 INSERT INTO alunos (matricula, nome, email, nascimento, endereco, sexo, cursoId)
83 VALUES (514513, 'Breno Macêdo de Brito', 'brenomacedo@gmail.com', '2003-01-14', 'Avenida Brasil Numero 777', 'masculino', 6551);
84
85 INSERT INTO alunos (matricula, nome, email, nascimento, endereco, sexo, cursoId)
86 VALUES (508016, 'Gustavo Wendell Moreira Valença', 'gustavovalenca13@gmail.com', '2003-08-29', 'Rua Francesa Numero 92', 'masculino', 6551);
87
88 INSERT INTO alunos (matricula, nome, email, nascimento, endereco, sexo, cursoId)
89 VALUES (509298, 'Tiago Brandão Forte', 'bigbrandon@gmail.com', '2003-05-10', 'ALDEOTA', 'masculino', 6551);
90
91 INSERT INTO relacao_alunos_disciplinas (alunoId, disciplinaId, avaliacao) VALUES (508682, 555, 4);
92 INSERT INTO relacao_alunos_disciplinas (alunoId, disciplinaId, avaliacao) VALUES (123456, 123, 10);
93 INSERT INTO relacao_alunos_disciplinas (alunoId, disciplinaId, avaliacao) VALUES (514513, 555, 4);
94 INSERT INTO relacao_alunos_disciplinas (alunoId, disciplinaId, avaliacao) VALUES (508016, 555, 7);
95 INSERT INTO relacao_alunos_disciplinas (alunoId, disciplinaId, avaliacao) VALUES (509298, 555, 10);
```