

Atividade Prática – Laboratório E/R

Objetivo:

Praticar a criação de Modelos Entidade-Relacionamento (E/R) e o mapeamento desses modelos para esquemas de banco de dados relacionais utilizando SQL.

Instruções:

1. **Modele o Diagrama Entidade-Relacionamento (E/R):**
 - Identifique as entidades principais.
 - Identifique os atributos de cada entidade, classificando-os (simples, composto, multivalorado, chave, etc.).
 - Identifique os relacionamentos entre as entidades, especificando suas cardinalidades (mínima e máxima).
 - Utilize o conceito de especialização/generalização onde for apropriado para representar os diferentes tipos de pessoas, alunos e servidores. Preste atenção à possibilidade de sobreposição e totalidade/parcialidade da especialização.
2. **Mapeie o Modelo E/R para SQL:**
 - Converta o seu diagrama E/R em um conjunto de comandos SQL CREATE TABLE.
 - Para cada tabela, defina:
 - As colunas correspondentes aos atributos, escolhendo tipos de dados SQL apropriados (ex: VARCHAR, INT, etc.).
 - A chave primária (PRIMARY KEY).
 - Outras restrições que julgar necessárias (NOT NULL, UNIQUE, CHECK).

Descrição do Sistema:

1. Uma pessoa possui CPF, nome e endereço.
2. Pessoas podem ser servidores, alunos ou egressos, ou mais de um tipo ao mesmo tempo.
 - Servidores possuem salário.



- [illegible]

- SELECT
- FROM
- WHERE
- ORDER BY
- LIMIT



-- Cria tabela Pessoa

```
CREATE TABLE Pessoa (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Nome VARCHAR(100) NOT NULL,  
    Rua VARCHAR(100),  
    Numero VARCHAR(10),  
    Bairro VARCHAR(50),  
    Cidade VARCHAR(50)  
);
```

-- Cria tabela Servidor

```
CREATE TABLE Servidor (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Salario DECIMAL(10,2) NOT NULL,  
    FOREIGN KEY (CPF) REFERENCES Pessoa(CPF)  
);
```

-- Cria tabela Aluno

```
CREATE TABLE Aluno (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Departamento VARCHAR(50) NOT NULL,  
    FOREIGN KEY (CPF) REFERENCES Pessoa(CPF)  
);
```

-- Cria tabela Egresso

```
CREATE TABLE Egresso (  
    CPF VARCHAR(11) PRIMARY KEY,  
    FOREIGN KEY (CPF) REFERENCES Pessoa(CPF)  
);
```

-- Cria tabela Titulo (multivalorado para Egresso)

```
CREATE TABLE Titulo (  
    ID_Titulo INTEGER PRIMARY KEY,  
    CPF_Egresso VARCHAR(11),  
    Nome_Titulo VARCHAR(100) NOT NULL,  
    Departamento VARCHAR(50) NOT NULL,  
    Ano_Obtencao INT NOT NULL,  
    FOREIGN KEY (CPF_Egresso) REFERENCES Egresso(CPF)  
);
```

-- Cria tabela Tecnico_Administrativo

```
CREATE TABLE Tecnico_Administrativo (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Especialidade VARCHAR(100) NOT NULL,  
    FOREIGN KEY (CPF) REFERENCES Servidor(CPF)  
);
```



-- Cria tabela Docente

```
CREATE TABLE Docente (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Classe VARCHAR(10) NOT NULL,  
    Nivel INT NOT NULL,  
    FOREIGN KEY (CPF) REFERENCES Servidor(CPF)  
);
```

-- Cria tabela Aluno_Assistente

```
CREATE TABLE Aluno_Assistente (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Horas_Auxilio INT NOT NULL,  
    FOREIGN KEY (CPF) REFERENCES Servidor(CPF)  
);
```

-- Cria tabela Aluno_Graduacao

```
CREATE TABLE Aluno_Graduacao (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Semestre INT NOT NULL,  
    FOREIGN KEY (CPF) REFERENCES Aluno(CPF)  
);
```

-- Cria tabela Aluno_Pos_Graduacao

```
CREATE TABLE Aluno_Pos_Graduacao (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Programa VARCHAR(100) NOT NULL,  
    FOREIGN KEY (CPF) REFERENCES Aluno(CPF)  
);
```

-- Cria tabela Aluno_Pesquisa

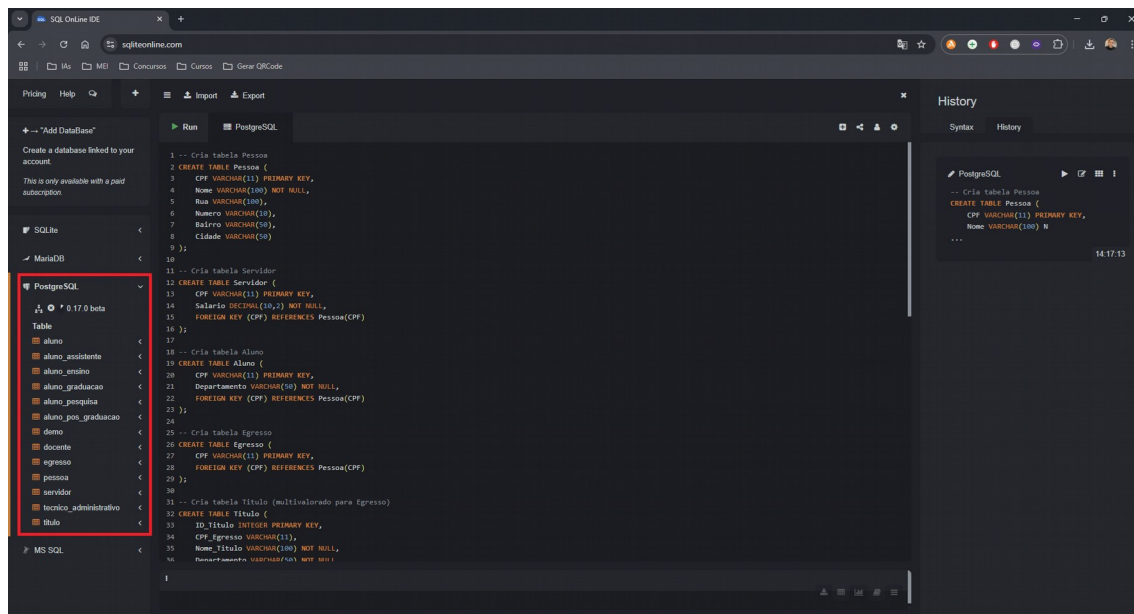
```
CREATE TABLE Aluno_Pesquisa (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Projeto_Pesquisa VARCHAR(100) NOT NULL,  
    FOREIGN KEY (CPF) REFERENCES Aluno(CPF)  
);
```

-- Cria tabela Aluno_Ensino

```
CREATE TABLE Aluno_Ensino (  
    CPF VARCHAR(11) PRIMARY KEY,  
    Disciplina_Associada VARCHAR(100) NOT NULL,  
    FOREIGN KEY (CPF) REFERENCES Aluno(CPF)  
);
```



Testado os comandos SQL em: <https://sqliteonline.com/>



The screenshot displays the SQLite Online IDE interface. On the left, a sidebar lists various databases, with 'PostgreSQL' selected and highlighted by a red rectangle. The main area shows a series of SQL commands for creating tables: 'Pessoa', 'Servidor', 'Aluno', 'Egresso', and 'Titulo'. The 'Pessoa' table is created with columns for CPF, Nome, Rua, Numero, Bairro, and Cidade. The 'Servidor' table includes CPF, Salarío, and a foreign key to 'Pessoa'. The 'Aluno' table has CPF, Departamento, and a foreign key to 'Pessoa'. The 'Egresso' table includes CPF and a foreign key to 'Pessoa'. The 'Titulo' table has ID_Titulo, CPF_Egresso, and Nome_Titulo, with a foreign key to 'Egresso'. The right sidebar shows a 'History' panel with the executed SQL commands.

```
1 -- Cria tabela Pessoa
2 CREATE TABLE Pessoa (
3   CPF VARCHAR(11) PRIMARY KEY,
4   Nome VARCHAR(100) NOT NULL,
5   Rua VARCHAR(100),
6   Numero VARCHAR(10),
7   Bairro VARCHAR(50),
8   Cidade VARCHAR(50)
9 );
10
11 -- Cria tabela Servidor
12 CREATE TABLE Servidor (
13   CPF VARCHAR(11) PRIMARY KEY,
14   Salarío DECIMAL(10,2) NOT NULL,
15   FOREIGN KEY (CPF) REFERENCES Pessoa(CPF)
16 );
17
18 -- Cria tabela Aluno
19 CREATE TABLE Aluno (
20   CPF VARCHAR(11) PRIMARY KEY,
21   Departamento VARCHAR(50) NOT NULL,
22   FOREIGN KEY (CPF) REFERENCES Pessoa(CPF)
23 );
24
25 -- Cria tabela Egresso
26 CREATE TABLE Egresso (
27   CPF VARCHAR(11) PRIMARY KEY,
28   FOREIGN KEY (CPF) REFERENCES Pessoa(CPF)
29 );
30
31 -- Cria tabela Titulo (multivalorado para Egresso)
32 CREATE TABLE Titulo (
33   ID_Titulo INTEGER PRIMARY KEY,
34   CPF_Egresso VARCHAR(11),
35   Nome_Titulo VARCHAR(100) NOT NULL,
36   FOREIGN KEY (CPF_Egresso) REFERENCES Egresso(CPF)
37 );
```

