

```

@startuml
entity TB_ALUNO {
+alunoId : int <<PK>>
nome : varchar
email : varchar
cursoId : int <<FK>>
}

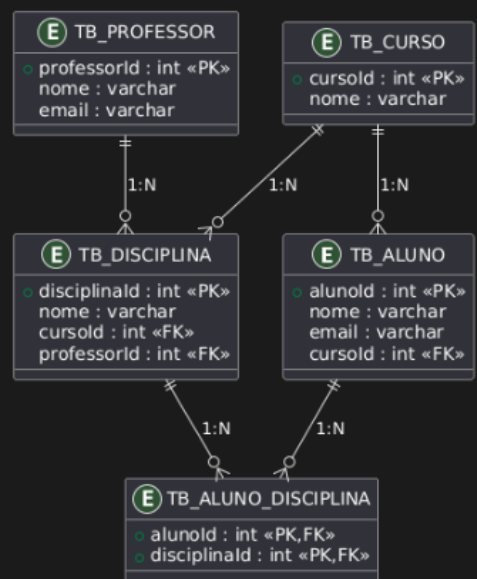
entity TB_PROFESSOR {
+professorId : int <<PK>>
nome : varchar
email : varchar
}

entity TB_CURSO {
+cursoId : int <<PK>>
nome : varchar
}

entity TB_DISCIPLINA {
+disciplinaId : int <<PK>>
nome : varchar
cursoId : int <<FK>>
professorId : int <<FK>>
}

entity TB_ALUNO_DISCIPLINA {
+alunoId : int <<PK,FK>>
+disciplinaId : int <<PK,FK>>
}

```



```

' Relacionamentos
TB_CURSO ||--o{ TB_ALUNO : "1:N"
TB_CURSO ||--o{ TB_DISCIPLINA : "1:N"
TB_PROFESSOR ||--o{ TB_DISCIPLINA : "1:N"
TB_ALUNO ||--o{ TB_ALUNO_DISCIPLINA : "1:N"
TB_DISCIPLINA ||--o{ TB_ALUNO_DISCIPLINA : "1:N"
@enduml

```

```

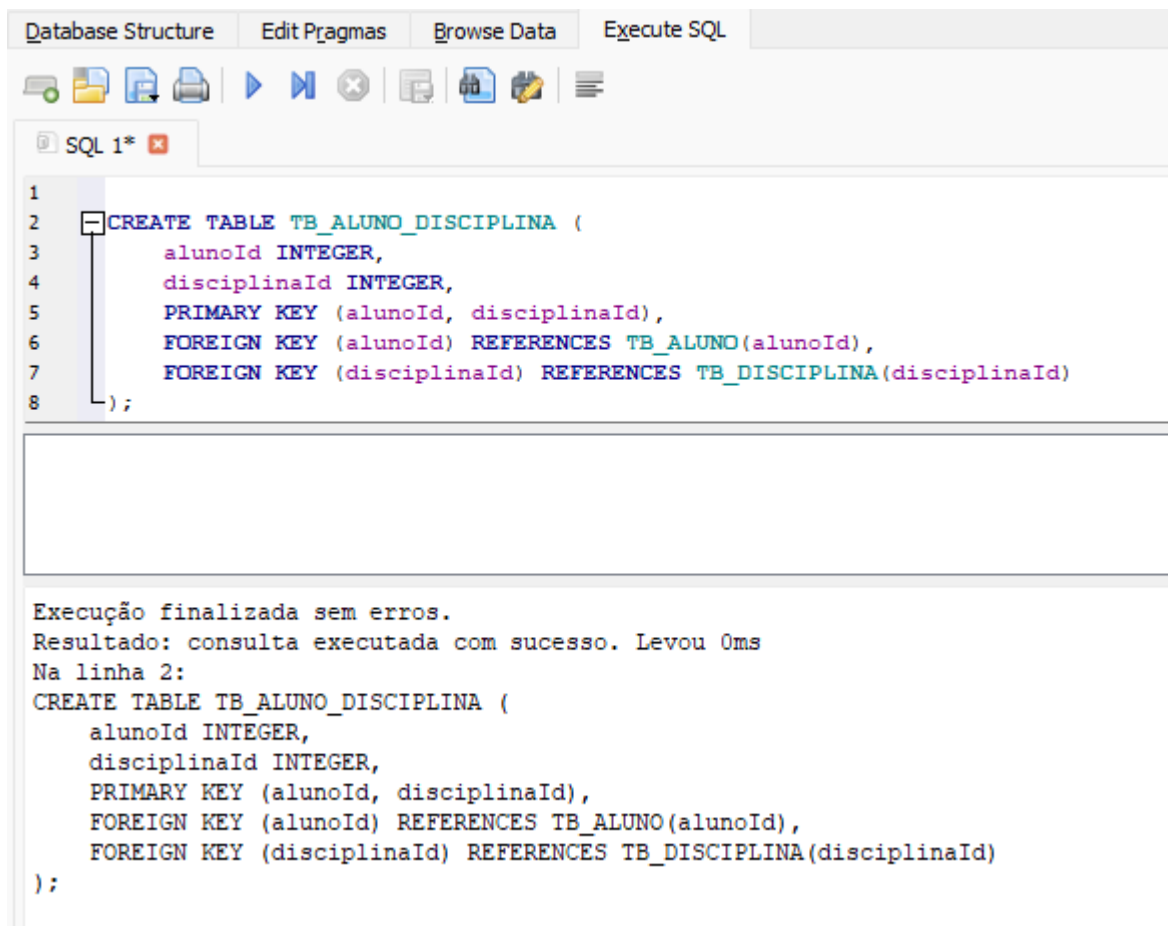
SQL 1*
1 CREATE TABLE TB_CURSO (
2     cursoId INTEGER PRIMARY KEY AUTOINCREMENT,
3     nome TEXT NOT NULL
4 );
5 CREATE TABLE TB_PROFESSOR (
6     professorId INTEGER PRIMARY KEY AUTOINCREMENT,
7     nome TEXT NOT NULL,
8     email TEXT NOT NULL
9 );
10 CREATE TABLE TB_ALUNO (
11     alunoId INTEGER PRIMARY KEY AUTOINCREMENT,
12     nome TEXT NOT NULL,
13     email TEXT NOT NULL,
14     cursoId INTEGER,
15     FOREIGN KEY (cursoId) REFERENCES TB_CURSO(cursoId)
16 );
17 CREATE TABLE TB_DISCIPLINA (
18     disciplinaId INTEGER PRIMARY KEY AUTOINCREMENT,
19     nome TEXT NOT NULL,
20     cursoId INTEGER,
21     professorId INTEGER,
22     FOREIGN KEY (cursoId) REFERENCES TB_CURSO(cursoId),
23     FOREIGN KEY (professorId) REFERENCES TB_PROFESSOR(professorId)
24 );

```

```

disciplinaId INTEGER PRIMARY KEY AUTOINCREMENT,
nome TEXT NOT NULL,
cursoId INTEGER,
professorId INTEGER,
FOREIGN KEY (cursoId) REFERENCES TB_CURSO(cursoId),
FOREIGN KEY (professorId) REFERENCES TB_PROFESSOR(professorId)

```



The screenshot shows a database management tool with tabs for 'Database Structure', 'Edit Pragmas', 'Browse Data', and 'Execute SQL'. The 'Execute SQL' tab is active, displaying a SQL editor with the following code:

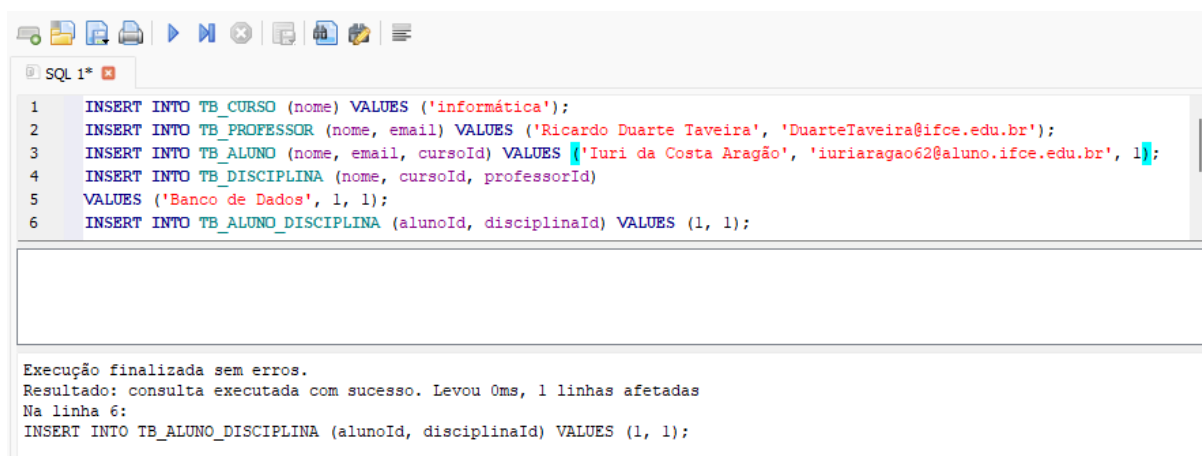
```
1  
2 CREATE TABLE TB_ALUNO_DISCIPLINA (  
3     alunoId INTEGER,  
4     disciplinaId INTEGER,  
5     PRIMARY KEY (alunoId, disciplinaId),  
6     FOREIGN KEY (alunoId) REFERENCES TB_ALUNO(alunoId),  
7     FOREIGN KEY (disciplinaId) REFERENCES TB_DISCIPLINA(disciplinaId)  
8 );
```

Below the editor, the execution results are shown:

Execução finalizada sem erros.  
Resultado: consulta executada com sucesso. Levou 0ms  
Na linha 2:  
CREATE TABLE TB\_ALUNO\_DISCIPLINA (  
 alunoId INTEGER,  
 disciplinaId INTEGER,  
 PRIMARY KEY (alunoId, disciplinaId),  
 FOREIGN KEY (alunoId) REFERENCES TB\_ALUNO(alunoId),  
 FOREIGN KEY (disciplinaId) REFERENCES TB\_DISCIPLINA(disciplinaId)  
);

4.

## Insert



The screenshot shows the same database management tool interface, but with the 'Execute SQL' tab displaying a different SQL statement:

```
1 INSERT INTO TB_CURSO (nome) VALUES ('informática');  
2 INSERT INTO TB_PROFESSOR (nome, email) VALUES ('Ricardo Duarte Taveira', 'DuarteTaveira@ifce.edu.br');  
3 INSERT INTO TB_ALUNO (nome, email, cursoId) VALUES ('Iuri da Costa Aragão', 'iuriaragao62@aluno.ifce.edu.br', 1);  
4 INSERT INTO TB_DISCIPLINA (nome, cursoId, professorId)  
5 VALUES ('Banco de Dados', 1, 1);  
6 INSERT INTO TB_ALUNO_DISCIPLINA (alunoId, disciplinaId) VALUES (1, 1);
```

The execution results are shown below:

Execução finalizada sem erros.  
Resultado: consulta executada com sucesso. Levou 0ms, 1 linhas afetadas  
Na linha 6:  
INSERT INTO TB\_ALUNO\_DISCIPLINA (alunoId, disciplinaId) VALUES (1, 1);

## Select

Database Structure Edit Pragma Browse Data Execute SQL

SQL 1\*

```

1 SELECT * FROM TB_CURSO;
2
3 SELECT A.alunoId, A.nome AS aluno, A.email, C.nome AS curso
4 FROM TB_ALUNO A
5 JOIN TB_CURSO C ON A.cursoId = C.cursoId;
6
7 SELECT D.nome AS disciplina, C.nome AS curso, P.nome AS professor
8 FROM TB_DISCIPLINA D
9 JOIN TB_CURSO C ON D.cursoId = C.cursoId
10 JOIN TB_PROFESSOR P ON D.professorId = P.professorId;
11
12 SELECT A.nome AS aluno, D.nome AS disciplina
13 FROM TB_ALUNO_DISCIPLINA AD
14 JOIN TB_ALUNO A ON AD.alunoId = A.alunoId
15 JOIN TB_DISCIPLINA D ON AD.disciplinaId = D.disciplinaId
16 WHERE A.alunoId = 1;

```

|   | aluno                | disciplina     |
|---|----------------------|----------------|
| 1 | Iuri da Costa Aragão | Banco de Dados |

Execução finalizada sem erros.  
 Resultado: 1 linhas retornadas em 9 ms  
 Na linha 12:  
 SELECT A.nome AS aluno, D.nome AS disciplina  
 FROM TB\_ALUNO\_DISCIPLINA AD  
 JOIN TB\_ALUNO A ON AD.alunoId = A.alunoId  
 JOIN TB\_DISCIPLINA D ON AD.disciplinaId = D.disciplinaId  
 WHERE A.alunoId = 1;

## Update

Database Structure Edit Pragma Browse Data Execute SQL

SQL 1\*


```

1 UPDATE TB_CURSO SET nome = 'Ciência da Computação e Dados' WHERE cursoId = 1;
2 UPDATE TB_PROFESSOR SET email = 'novo.email@ifce.edu.br' WHERE professorId = 2;
3 UPDATE TB_ALUNO SET cursoId = 2 WHERE alunoId = 1;

```

Execução finalizada sem erros.  
 Resultado: consulta executada com sucesso. Levou 0ms, 1 linhas afetadas  
 Na linha 3:  
 UPDATE TB\_ALUNO SET cursoId = 2 WHERE alunoId = 1;

## Delete



SQL 1\*

```
1 DELETE FROM TB_ALUNO_DISCIPLINA WHERE alunoId = 1 AND disciplinaId = 1;
2 DELETE FROM TB_ALUNO_DISCIPLINA WHERE alunoId = 1;
3 DELETE FROM TB_ALUNO WHERE alunoId = 1;
4 DELETE FROM TB_DISCIPLINA WHERE disciplinaId = 2;
```

---

Execução finalizada sem erros.  
Resultado: consulta executada com sucesso. Levou 0ms, 0 linhas afetadas  
Na linha 4:  
DELETE FROM TB\_DISCIPLINA WHERE disciplinaId = 2;