

# Atividade de Banco de Dados 03

Crie um banco de dados SCA\_251.bd.

Nele defina as Tabela TB\_ALUNO, TB\_PROFESSOR, TB\_CURSO e TB\_DISCIPLINA.

```
CREATE DATABASE SCA_242.bd;

CREATE TABLE TB_ALUNO (
    Nome VARCHAR(255) NOT NULL;
    Matricula INT;
    Email VARCHAR(255) NOT NULL;
    PRIMARY KEY (Matricula)
);

CREATE TABLE TB_PROFESSOR (
    Nome VARCHAR(255) NOT NULL;
    SIAP INT;
    Email VARCHAR(255) NOT NULL;
    PRIMARY KEY (SIAP);
);

CREATE TABLE TB_CURSO (
    Nome VARCHAR(255) NOT NULL;
    Duração INT;
    Id INT;
    PRIMARY KEY (Id);
);

CREATE TABLE TB_DISCIPLINA (
    Nome VARCHAR(255) NOT NULL;
```

```
Duração INT;  
Id INT;  
PRIMARY KEY (Id);  
);
```

1) Identifique os possíveis relacionamentos com as suas cardinalidades envolvendo as tabelas.

TB\_ALUNO – Sala de aula – TB\_PROFESSOR (N : 1)

TB\_ALUNO – Cursando – TB\_CURSO (N : 1)

TB\_ALUNO – Matriculados – TB\_DISCIPLINA (N : N)

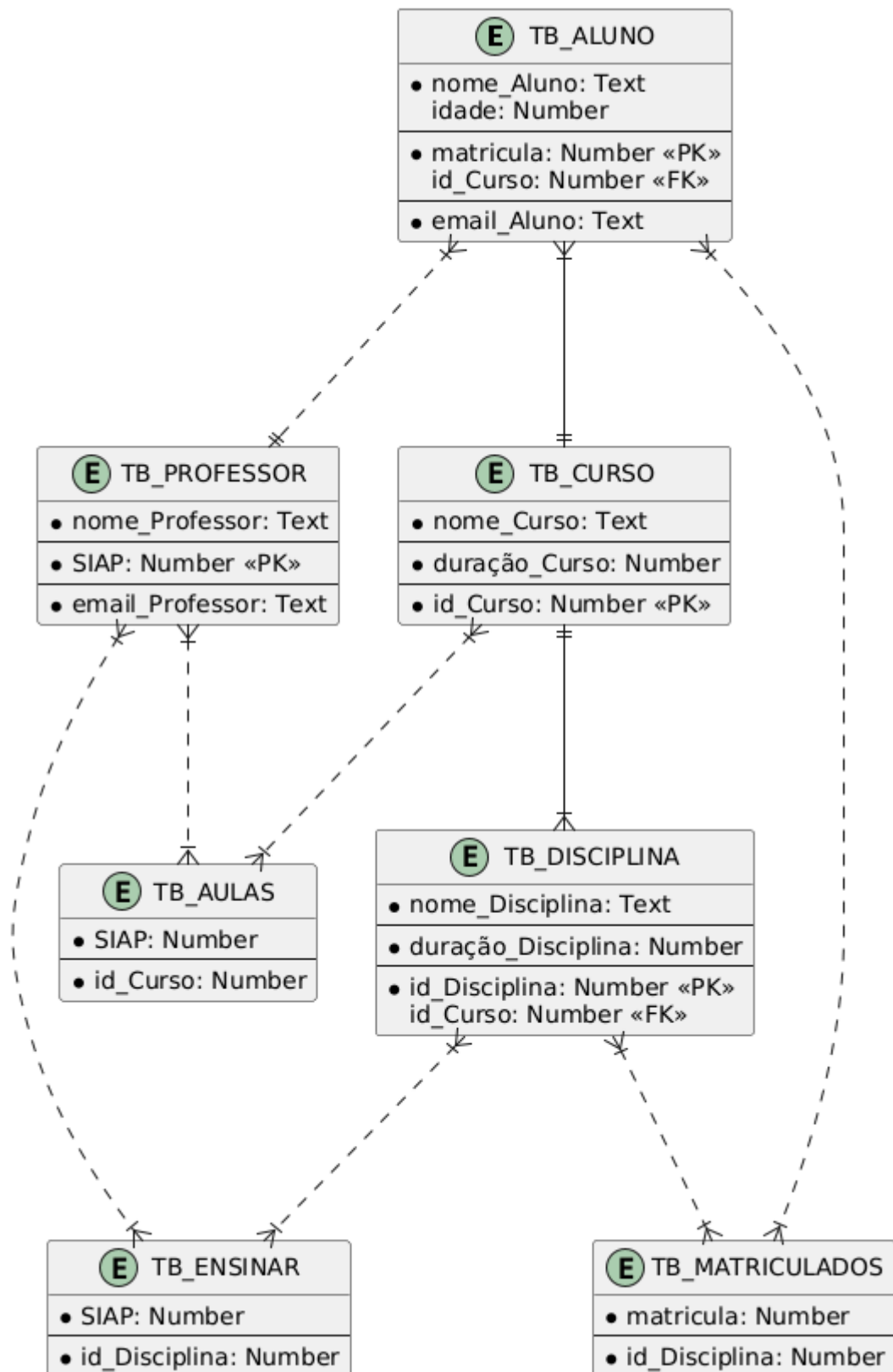
TB\_PROFESSOR – Dão aulas – TB\_CURSO (N : N)

TB\_PROFESSOR – Ensinam – TB\_DISCIPLINA (N : N)

TB\_CURSO – Matriz curricular – TB\_DISCIPLINA (1 : N)

2) Crie um Diagrama de um Modelo Lógico no qual são definidas chaves primárias (id) e chaves estrangeiras(tabelaId) seus relacionamentos e

suas cardinalidades. Onde existir relacionamentos N:N criar uma tabela TB\_E1\_TB\_E2 com chave primária concatenada envolvendo as chaves primárias de cada tabela



```
@startuml
entity TB_ALUNO {
  * nome_Aluno: Text
  idade: Number
  --
  * matricula: Number <<PK>>
  id_Curso: Number <<FK>>
  --
  *email_Aluno: Text
}
entity TB_PROFESSOR {
  * nome_Professor: Text
  --
  * SIAP: Number <<PK>>
  --
  * email_Professor: Text
}
entity TB_CURSO {
  * nome_Curso: Text
  --
  * duração_Curso: Number
  --
  * id_Curso: Number <<PK>>
}
entity TB_DISCIPLINA {
  * nome_Disciplina: Text
  --
  * duração_Disciplina: Number
  --
  * id_Disciplina: Number <<PK>>
  id_Curso: Number <<FK>>
}
entity TB_MATRICULADOS {
  * matricula: Number
  --
  * id_Disciplina: Number
}
```

```

entity TB_AULAS {
  * SIAP: Number
  --
  * id_Curso: Number
}
entity TB_ENSINAR {
  * SIAP: Number
  --
  * id_Disciplina: Number
}

```

```

TB_ALUNO }|...|| TB_PROFESSOR
TB_ALUNO }|---|| TB_CURSO
TB_ALUNO }|...|{ TB_MATRICULADOS
TB_DISCIPLINA }|...|{ TB_MATRICULADOS
TB_PROFESSOR }|...|{ TB_AULAS
TB_CURSO }|...|{ TB_AULAS
TB_PROFESSOR }|...|{ TB_ENSINAR
TB_DISCIPLINA }|...|{ TB_ENSINAR
TB_CURSO ||---|{ TB_DISCIPLINA
@enduml

```

3) Escreva o comando SQL cria esse banco de dados.

```

CREATE DATABASE SCA_242.bd

CREATE TABLE TB_ALUNO (
  Nome VARCHAR(255) NOT NULL
  Idade INT NOT NULL
  Matricula INT NOT NULL
  Email VARCHAR(255) NOT NULL
  Id_Curso INT
  PRIMARY KEY (Matricula)
  FOREIGN KEY (Id_Curso) REFERENCES TB_CURSO(Id_Curso)
)

```

```
CREATE TABLE TB_PROFESSOR (  
    Nome VARCHAR(255) NOT NULL  
    SIAP INT NOT NULL  
    Email VARCHAR(255) NOT NULL  
    PRIMARY KEY (SIAP)  
)  
  
CREATE TABLE TB_CURSO (  
    Nome VARCHAR(255) NOT NULL  
    Duração INT NOT NULL  
    Id_Curso INT NOT NULL  
    PRIMARY KEY (Id)  
)  
  
CREATE TABLE TB_DISCIPLINA (  
    Nome VARCHAR(255) NOT NULL  
    Duração INT NOT NULL  
    Id_Disciplina INT NOT NULL  
    Id_Curso INT NOT NULL  
    PRIMARY KEY (Id)  
    FOREIGN KEY (Id_Curso) REFERENCES TB_CURSO(Id_Curso)  
)  
  
CREATE TABLE TB_MATRICULADOS (  
    Matricula INT  
    id_Disciplina INT  
    PRIMARY KEY (Matricula, id_Disciplina)  
    FOREIGN KEY (Matricula) REFERENCES TB_ALUNO(Matricula)  
    FOREIGN KEY (id_Disciplina) REFERENCES  
TB_DISCIPLINA(id_Disciplina)  
)  
  
CREATE TABLE TB_AULAS (  
    SIAP INT  
    id_Curso INT  
    PRIMARY KEY (SIAP, id_Curso)
```

```

        FOREIGN KEY (SIAP) REFERENCES TB_PROFESSOR(SIAP)
        FOREIGN KEY (id_Curso) REFERENCES TB_CURSO(id_Curso)
    )

CREATE TABLE TB_ENSINAR (
    SIAP INT
    id_Disciplina INT
    PRIMARY KEY (SIAP, id_Disciplina)
    FOREIGN KEY (SIAP) REFERENCES TB_PROFESSOR(SIAP)
    FOREIGN KEY (id_Disciplina) REFERENCES
TB_DISCIPLINA(id_Disciplina)
)

```

4) Exemplifique o uso de comando SQL que executam operações CRUD(Insert, Select, Update e Delete) envolvendo todas as tabelas.

```

CREATE DATABASE SCA_242

CREATE TABLE TB_CURSO (
    Nome VARCHAR(255) NOT NULL,
    Duração INT NOT NULL,
    Id_Curso INT NOT NULL,
    PRIMARY KEY (Id_Curso)
);

CREATE TABLE TB_ALUNO (
    Nome VARCHAR(255) NOT NULL,
    Idade INT,
    Matricula INT NOT NULL,
    Email VARCHAR(255) NOT NULL,
    Id_Curso INT,
    PRIMARY KEY (Matricula),
    FOREIGN KEY (Id_Curso) REFERENCES TB_CURSO(Id_Curso)
);

CREATE TABLE TB_PROFESSOR (

```

```

    Nome VARCHAR(255) NOT NULL,
    SIAP INT NOT NULL,
    Email VARCHAR(255) NOT NULL,
    PRIMARY KEY (SIAP)
);

CREATE TABLE TB_DISCIPLINA (
    Nome VARCHAR(255) NOT NULL,
    Duração INT NOT NULL,
    Id_Disciplina INT NOT NULL,
    Id_Curso INT NOT NULL,
    PRIMARY KEY (Id_Disciplina),
    FOREIGN KEY (Id_Curso) REFERENCES TB_CURSO(Id_Curso)
);

CREATE TABLE TB_MATRICULADOS (
    Matricula INT,
    id_Disciplina INT,
    PRIMARY KEY (Matricula, id_Disciplina),
    FOREIGN KEY (Matricula) REFERENCES TB_ALUNO(Matricula),
    FOREIGN KEY (id_Disciplina) REFERENCES
TB_DISCIPLINA(id_Disciplina)
);

CREATE TABLE TB_AULAS (
    SIAP INT,
    id_Curso INT,
    PRIMARY KEY (SIAP, id_Curso),
    FOREIGN KEY (SIAP) REFERENCES TB_PROFESSOR(SIAP),
    FOREIGN KEY (id_Curso) REFERENCES TB_CURSO(id_Curso)
);

CREATE TABLE TB_ENSINAR (
    SIAP INT,
    id_Disciplina INT,
    PRIMARY KEY (SIAP, id_Disciplina),
    FOREIGN KEY (SIAP) REFERENCES TB_PROFESSOR(SIAP),

```



```
        FOREIGN KEY (id_Disciplina) REFERENCES
TB_DISCIPLINA(id_Disciplina)
);

-- CRUD

INSERT INTO TB_CURSO(Nome, Duração, Id_Curso)
VALUES('Informática', 2000, 60);

INSERT INTO TB_ALUNO(Nome, Idade, Matricula, Email, Id_Curso)
VALUES ('Lorran', 17, 20232011060777, 'lorran@gmail.com', 60),
       ('Alyne', 16, 20232011060666, 'alyne@gmail.com', 60);

INSERT INTO TB_PROFESSOR(Nome, SIAP, Email)
VALUES ('Taveira', 2000000000000, 'taveira@gmail.com'),
       ('Cesar', 3000000000000, 'cesar@gmail.com');

INSERT INTO TB_DISCIPLINA(Nome, Duração, Id_Curso,
Id_Disciplina)
VALUES('Matemática I', 20, 60, 1);

UPDATE TB_DISCIPLINA
SET Nome = 'Matemática II'
WHERE Id_Disciplina = 1;

DELETE FROM TB_ALUNO WHERE Matricula = 20232011060777;

DROP TABLE IF EXISTS TB_AULAS;
DROP TABLE IF EXISTS TB_ENSINAR;
DROP TABLE IF EXISTS TB_MATRICULADOS;

SELECT Nome, Email
FROM TB_PROFESSOR
WHERE Nome = 'Cesar'
LIMIT 1;
```