

**Banco de Dados**

**Aluno:**

**Matrícula:**

**SEMESTRE. 202X.X**

**BACHARELADO EM CIÊNCIAS DA COMPUTAÇÃO**

**ATIVIDADE 8**



CREATE TABLE Professor(matricula numeric(10),

nome varchar(255) not null,

RG bigint unique not null,

-- se 'M' é igual a 'm'? é diferente!

-- check(sexo in ('M','F'))

sexo varchar(1) check(sexo = 'M' or sexo = 'F' or sexo = 'm' or sexo = 'f'),

-- check(idade between 21 and 80)

idade int check(idade >=21 and idade<=80),

-- check(titulação in ('graduado','especialista','mestre','doutor'))

titulacao varchar(15) check(titulacao = 'graduado' or

titulacao = 'especialista' or

titulacao = 'mestre' or

titulacao = 'doutor'),

categoria varchar(15) check(categoria in ('auxiliar','assistente',

'adjunto','titular')),

nroTurmas int check(nroTurmas >=0),

primary key(matricula));

INSERT INTO Professor VALUES (01, 'Henrique', 1111111, 'M', 33, 'doutor', 'adjunto', 4),

(02, 'Henrique', 11111111111111, 'm', 33, 'doutor', 'adjunto', 4);

SELECT \* FROM Professor;

CREATE TABLE Cursos (codigo int,

nome varchar(255) not null,

duracao int check(duracao >= 4 and duracao <=12),

coordenador numeric(10),

foreign key(coordenador) references Professor(matricula)

on delete cascade on update cascade,

primary key(codigo));

INSERT INTO Cursos VALUES (120, 'BCC', 8, 01),

(121, 'BCC2', 8, 01);

SELECT \* FROM Cursos;

CREATE TABLE Disciplinas(codigo int,

nome varchar not null,

creditos int check(creditos between 2 and 8),

primary key(codigo));

INSERT INTO Disciplinas VALUES (333, 'BDI', 4);

SELECT \* FROM Disciplinas;

CREATE TABLE Curriculos(curso int,

disciplina int,

fase int check(fase between 1 and 12),

foreign key(curso) references Cursos(codigo)

on update cascade on delete cascade,

foreign key(disciplina) references Disciplinas(codigo)

on update cascade on delete cascade,

primary key(curso,disciplina));

INSERT INTO Curriculos VALUES (120, 333, 3);

SELECT \* FROM Curriculos;

CREATE TABLE Turmas(disciplina int,

codigo int,

vagas int check(vagas > 0),

professor numeric(10),

primary key(disciplina,codigo),

foreign key(disciplina) references Disciplinas(codigo)

on update cascade on delete cascade,

foreign key(professor) references Professor(matricula)

on update cascade on delete cascade);