

# Correção Aula 1 - Introdução - Exercício 2

## 1 - Criação da tabela Aluno

```
create table aluno(  
    id_aluno integer not null primary key,  
    nome_aluno varchar(40) not null default 'NP',  
    sexo_aluno char(1) not null check (sexo_aluno in ('M','F')),  
    dt_nasc timestamp,  
    dt_alter_tab timestamp not null default now(),  
    observacoes varchar(250)  
);
```

## 2 - Inserindo tupla na tabela aluno, com sexo = 'X' e id\_aluno=1

```
insert into aluno(id_aluno, sexo_aluno) values (1,'X');
```

ERROR: new row for relation "aluno" violates check constraint "aluno\_sexo\_aluno\_check"

## 3 - Inserindo tupla na tabela aluno, com sexo = 'M' e id\_aluno=1

```
insert into aluno(id_aluno, sexo_aluno) values (1,'M');
```

INSERT 0 1

## 4 - Inserindo tupla na tabela aluno, com sexo = 'F' e id\_aluno=1

```
insert into aluno(id_aluno, sexo_aluno) values (1,'F');
```

ERROR: duplicate key value violates unique constraint "aluno\_pkey"

DETAIL: Key (id\_aluno)=(1) already exists.

## 5 - Acrescentar o campo cod\_sit\_aluno

```
alter table aluno add column cod_sit_aluno integer;
```

#### 6 - Acrescentar o campo id\_curso

```
alter table aluno add column id_curso integer;
```

#### 7 - Para testar NOT NULL, apagar campos

```
alter table aluno drop column cod_sit_aluno;
```

```
alter table aluno drop column id_curso;
```

#### 8 - Testando NOT NULL para campos cod\_sit\_aluno e id\_curso

```
alter table aluno add column cod_sit_aluno integer not null;
```

ERROR: column "cod\_sit\_aluno" contains null values

```
alter table aluno add column id_curso integer not null;
```

ERROR: column "id\_curso" contains null values

#### 9 - É necessário apagar as tuplas de aluno

```
delete from aluno;
```

DELETE 1

#### 10 - Acrescentando campos id\_curso e cod\_sit\_aluno NOT NUL

```
alter table aluno add column id_curso integer not null;
```

ALTER TABLE

```
alter table aluno add column cod_sit_aluno integer not null;
```

ALTER TABLE

#### 11 - Inserindo aluno teste

```
insert into aluno(id_aluno, sexo_aluno) values (1, 'M');
```

ERROR: null value in column "id\_curso" violates not-null constraint

## 12 - Criação da tabela curso

```
create table curso (  
    id_curso integer not null PRIMARY KEY,  
    nome_curso varchar(30) not null,  
    dt_implantacao timestamp not null,  
    dt_alteracao timestamp not null default now()  
);
```

## 13 - Inserindo tuplas na tabela curso

```
insert into curso values(1, 'Ciencia da Computacao', now());
```

INSERT 0 1

```
insert into curso values(2, 'Analise de Sistemas', now());
```

INSERT 0 1

```
insert into curso values(3, 'TADS', now());
```

INSERT 0 1

## 14 - Criação da tabela situacao\_aluno

```
CREATE TABLE situacao_aluno (  
    cod_sit_aluno integer not null primary key,  
    desc_sit_aluno varchar(30) not null,  
    dt_alteracao timestamp not null default now()  
);
```

## 15 - Inserindo tuplas na tabela situacao\_aluno

```
insert into situacao_aluno values (1, 'matriculado');
```

INSERT 0 1

```
insert into situacao_aluno values (2, 'desistente');
```

INSERT 0 1

```
insert into situacao_aluno values (3, 'excluido por diplomacao');
```

INSERT 0 1

#### 16 - Inserindo aluno teste em curso teste com situacao teste

```
insert into aluno(id_aluno, sexo_aluno,id_curso,cod_sit_aluno)
values (1,'M',1,1);
```

INSERT 0 1

#### 17 - Inserindo FK de id\_curso

```
alter table aluno add foreign key (id_curso) references
curso(id_curso) on delete restrict on update cascade;
```

ALTER TABLE

#### 18 - Inserindo FK de cod\_sit\_aluno

```
alter table aluno add foreign key(cod_sit_aluno) references
situacao_aluno(cod_sit_aluno) on delete restrict on update
cascade;
```

ALTER TABLE

## **19 - Consultas SQL**

### **19.1 - Obter todos os alunos do sexo "M"**

```
select *  
  
from aluno  
  
where sexo_aluno='M';
```

### **19.2 - Obter todos os alunos do sexo "F"**

```
select * from aluno where sexo_aluno='F';
```

### **19.3 - Contar o total de linhas da tabela aluno**

```
select count(*) from aluno;
```

### **19.4 - Contar a qtde de alunos do sexo "F" e "M"**

```
select sexo_aluno,count(*) from aluno group by sexo_aluno;
```

### **19.5 - Contar os alunos por curso**

```
select id_curso,count(*) from aluno group by id_curso;
```

### **19.6 - Contar os alunos por situação**

```
select cod_sit_aluno,count(*)  
  
from aluno  
  
group by cod_sit_aluno;
```

### **19.7 - Contar os alunos por curso e situação**

```
select id_curso,cod_sit_aluno,count(*)  
  
from aluno  
  
group by id_curso,cod_sit_aluno;
```

## 20 - PG\_DUMP

```
pg_dump -Fp db_hercules -U hercules -W >
```

```
                                /home/hercules/hercules_db.sql
```

```
vim /home/hercules/hercules_db.sql  (Para sair ESC,  :q! ENTER )
```

```
rm /home/hercules/hercules_db.sql
```

```
pg_dump --inserts -Fp db_hercules -U hercules -W >
```

```
                                /home/hercules/hercules_db.sql
```

```
vim /home/hercules/hercules_db.sql  (Para sair ESC,  :q! ENTER )
```

## 20 - Apagando o Banco de Dados

```
dropdb db_hercules -U hercules -W
```

## 21 - Criando um novo banco de dados VAZIO

```
createdb -T template0 -E UTF-8 db_hercules -U hercules -W
```

## 22 - RESTAURANDO O BANCO DE DADOS A PARTIR DO DUMP

```
psql -d db_hercules -U hercules -W < /home/hercules/hercules_db.sql
```

**Tem como fazer este processo de outras maneiras ? Como ? Pesquise !**