



TABELAS, Constraints, Sequences, Triggers

create, alter, drop

Objetivos

- Descrever os detalhes e sintaxe para a criação de:
 - Tabelas
 - Constraints
 - Sequences
 - Triggers
- Outros comandos
alter, drop

Para criar uma TABELA

precisamos saber:

- Tabela:
 <NOME>
- Colunas:
 <NOME>
 - Tipo de dados (data type)
 - [valor default diferente de null?]
 - [não permite nulo? → default é aceitar nulo]
 - [constraints]
 - Check
 - Primary Key
 - Foreign Key

create TABLE syntax

```
▪ create table NOME_TABELA (  
    col1    tipo    default <valor>    [null|not null]    [constraint],  
    col2    tipo    default <valor>    [null|not null]    [constraint],  
    col3    tipo    default <valor>    [null|not null]    [constraint],  
    col4    tipo    default <valor>    [null|not null]    [constraint],  
    ...  
    ...  
);
```

Exemplo de CREATE TABLE

(constraints não nomeadas)

```
create table FUNCIONARIOS (  
    cod_empr      number          NOT NULL PRIMARY KEY,  
    nome          varchar2(100)   NOT NULL,  
    cpf           varchar2(14)    NOT NULL,  
    data_nasc     date            NOT NULL,  
    nro_fone      varchar2(20)    NOT NULL,  
    tipo_fone     char(3)         NOT NULL check  
                                     (tipo_fone in ('FIX', 'MOB')),  
    salario       number(10,2)    NOT NULL,  
    foto          BLOB,  
    curriculo     CLOB  
);
```

Exemplo de CREATE TABLE com PK nomeada

```
create table FUNCIONARIOS (  
  cod_empr      number          NOT NULL,  
  nome          varchar2(100)  NOT NULL,  
  cpf           varchar2(14)   NOT NULL,  
  data_nasc     date           NOT NULL,  
  nro_fone      varchar2(20)   NOT NULL,  
  tipo_fone     char(3)        NOT NULL check  
                                     (tipo_fone in ('FIX', 'MOB')),  
  salario       number(10,2)   NOT NULL,  
  foto          BLOB,  
  curriculo     CLOB,  
  constraint FUNCIONARIOS_PK primary key (cod_empr)  
);
```

Alterando Tabela

```
alter table FUNCIONARIOS drop column tipo_fone;
```

Table altered.

```
alter table FUNCIONARIOS add data_contrat DATE not null;
```

Table altered.

```
alter table FUNCIONARIOS modify nro_fone varchar2(30);
```

Table altered.

Valores Default

```
create table FUNCIONARIOS (  
    cod_empr      number          NOT NULL PRIMARY KEY,  
    nome          varchar2(100)  NOT NULL,  
    cpf           varchar2(14)   NOT NULL,  
    data_nasc     date           NOT NULL,  
    nro_fone      varchar2(20)   NOT NULL,  
    tipo_fone     char(3)        NOT NULL check  
                                (tipo_fone in ('FIX', 'MOB')),  
    salario       number(10,2)   default 0 NOT NULL,  
    foto          BLOB,  
    curriculo     CLOB  
);
```

```
alter table FUNCIONARIOS modify salario default 0;
```


FOREIGN KEY

```
create table CARGOS (  
    cod_cargo  number(5) NOT NULL primary key,  
    descr      varchar2(20) NOT NULL  
);  
  
alter table FUNCIONARIOS add cargo NUMBER(5) NOT NULL;  
  
alter table FUNCIONARIOS add constraint FUNCIONARIOS_CARGOS_FK  
    FOREIGN KEY (cargo) references CARGOS(cod_cargo);
```

Gerando um ID sequencia

```
create sequence SEQ_FUNCIONARIOS;
```

```
1 create sequence SEQ_FUNCIONARIOS;
```

```
Sequence created.
```

Gerando um ID sequencia

```
create or replace trigger TRG_FUNCIONARIOS_SEQ
before insert on FUNCIONARIOS
for each row
BEGIN
    :new.cod_empr := SEQ_FUNCIONARIOS.nextval;
END;
/
```

```
1 v create or replace trigger TRG_FUNCIONARIOS_SEQ
2   before insert on FUNCIONARIOS
3   for each row
4   BEGIN
5       :new.cod_empr := SEQ_FUNCIONARIOS.nextval;
6   END;
7   /
```

Trigger created.

Regras

- Nomes

- Não pode ser palavra reservada
(ex. TABLE, CREATE, CHAR, VARCHAR2, ...)
- Tamanho máximo
30 caracteres
- Começar com 1 letra
- Permitido letras numeros # \$ _
- Nome da tabela é único no schema
(nome do objeto é unico no schema)
- Nome da Coluna
(unico na tabela)

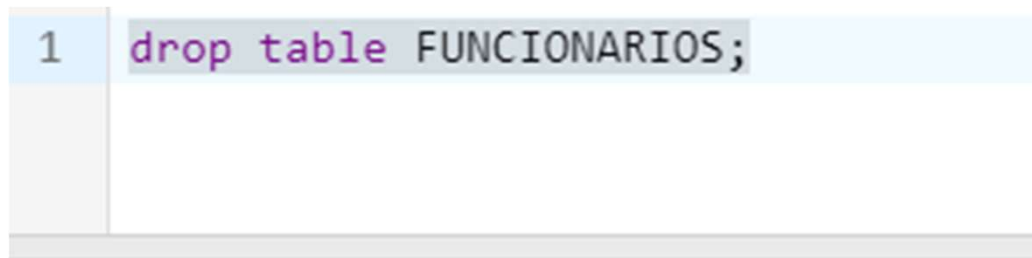
Comandos de Eliminação → DROP

- **CUIDADO !!!!!** Risco de perda de dados!
 - Comandos de DROP
 - Eliminam objetos (e seus dados)
- Cuidado com todo comando de DROP!

ELIMINANDO Tabela

- **CUIDADO !!!!!** Risco de perda de dados!

```
drop table FUNCIONARIOS;
```



```
1 drop table FUNCIONARIOS;
```

```
Table dropped.
```

Consultando Dicionário de Dados

ALL_TABLES

ALL_TAB_COLUMNS

ALL_CONSTRAINTS

ALL_OBJECTS

Resumo e Dúvidas

- Dúvidas ou comentários ... ?

