

- 1) Crie o usuário nomeado de **exercicios** (sem acentuação), identificado pela senha **123456**, por meio da DDL, para a resolução dos exercícios do treinamento. Na sequência, através da DCL, conceda as permissões necessárias ao usuário **exercicios**.

-- Criação do usuário "exercicios" para elaboração dos exercícios

CREATE USER exercicios **IDENTIFIED BY** 123456;

-- Concedendo permissão "resource + connect" para o usuário "exercicios"

GRANT resource, connect **TO** exercicios;

ALTER USER exercicios **quota** 100M **on USERS**;

- 2) Crie a tabela **tb_regiao**, conforme a instrução SQL abaixo.

OBSERVAÇÃO: criamos explicitamente:

- **Constraint NOT NULL** para a coluna **id_regiao**
- **Índice** para a coluna **id_regiao**
- **Constraint PRIMARY KEY** para a coluna **id_regiao**

CREATE TABLE tb_regiao(
id_regiao

NUMBER CONSTRAINT nn_id_regiao NOT NULL,

nm_regiao **VARCHAR2(25));**

CREATE UNIQUE INDEX pk_id_regiao

ON tb_regiao (id_regiao);

ALTER TABLE tb_regiao

ADD (CONSTRAINT pk_id_regiao

PRIMARY KEY (id_regiao));

- 3) Crie a tabela **tb_pais**, conforme a instrução SQL abaixo.

OBSERVAÇÃO: criamos explicitamente:

- **Constraint NOT NULL** para a coluna **id_pais**
- **Constraint PRIMARY KEY** para a coluna **id_pais**
- **Constraint FOREIGN KEY** para a coluna **id_regiao**

CREATE TABLE tb_pais(
id_pais

CHAR(2) CONSTRAINT nn_id_pais NOT NULL,

nm_pais **VARCHAR2(40),**

id_regiao **NUMBER,**

CONSTRAINT pk_id_pais **PRIMARY KEY** (id_pais));

ALTER TABLE tb_pais

ADD (CONSTRAINT fk_regiao_pais

FOREIGN KEY (id_regiao) **REFERENCES** tb_regiao(id_regiao));

4) Crie a tabela **tb_localizacao**, conforme a instrução SQL abaixo.

OBSERVAÇÃO: criamos explicitamente:

- **Constraint NOT NULL** para a coluna **cidade**
- **Índice** para a coluna **id_localizacao**
- **Constraint PRIMARY KEY** para a coluna **id_localizacao**
- **Constraint FOREIGN KEY** para a coluna **id_pais**
- Uma **seqüência** nomeada de **seq_localizacao**

```
CREATE TABLE tb_localizacao(  
id_localizacao      NUMBER(4),  
id_pais             CHAR(2),  
endereco            VARCHAR2(40),  
cep                 VARCHAR2(12),  
cidade              VARCHAR2(30) CONSTRAINT nn_loc_cidade NOT NULL,  
estado              VARCHAR2(25));
```

```
CREATE UNIQUE INDEX pk_id_localizacao  
ON tb_localizacao (id_localizacao);
```

```
ALTER TABLE tb_localizacao  
ADD ( CONSTRAINT pk_id_loc  
      PRIMARY KEY (id_localizacao),  
      CONSTRAINT fk_id_pais  
      FOREIGN KEY (id_pais)  
      REFERENCES tb_pais(id_pais));
```

```
CREATE SEQUENCE seq_localizacao  
START WITH          3300  
INCREMENT BY        100  
MAXVALUE            9900  
NOCACHE  
NOCYCLE;
```

5) Crie a tabela **tb_departamento**, conforme a instrução SQL abaixo.

OBSERVAÇÃO: criamos explicitamente:

- **Constraint NOT NULL** para a coluna **nm_departamento**
- **Índice** para a coluna **id_departamento**
- **Constraint PRIMARY KEY** para a coluna **id_departamento**
- **Constraint FOREIGN KEY** para a coluna **id_localizacao**
- Uma **seqüência** nomeada de **seq_departamento**

```
CREATE TABLE tb_departamento(  
id_departamento     NUMBER(4),  
nm_departamento     VARCHAR2(30) CONSTRAINT nn_nm_depto NOT NULL,
```

```
id_gerente          NUMBER(6),
id_localizacao      NUMBER(4));
```

```
CREATE UNIQUE INDEX pk_id_departamento
ON tb_departamento (id_departamento);
```

```
ALTER TABLE tb_departamento
ADD ( CONSTRAINT pk_id_departamento
      PRIMARY KEY (id_departamento),
      CONSTRAINT fk_loc_departamento
      FOREIGN KEY (id_localizacao) REFERENCES tb_localizacao (id_localizacao));
```

```
CREATE SEQUENCE seq_departamento
START WITH          280
INCREMENT BY        10
MAXVALUE            9990
NOCACHE
NOCYCLE;
```

6) Crie a tabela **tb_funcao**, conforme a instrução SQL abaixo.

OBSERVAÇÃO: criamos explicitamente:

- **Constraint NOT NULL** para a coluna **ds_funcao**
- **Índice** para a coluna **id_funcao**
- **Constraint PRIMARY KEY** para a coluna **id_funcao**

```
CREATE TABLE tb_funcao(
id_funcao          VARCHAR2(10),
ds_funcao          VARCHAR2(35) CONSTRAINT nn_ds_funcao NOT NULL,
base_salario       NUMBER(8,2),
teto_salario       NUMBER(8,2));
```

```
CREATE UNIQUE INDEX pk_id_funcao
ON tb_funcao (id_funcao);
```

```
ALTER TABLE tb_funcao
ADD ( CONSTRAINT pk_id_funcao
      PRIMARY KEY(id_funcao));
```

7) Crie a tabela **tb_empregado**, conforme a instrução SQL abaixo.

OBSERVAÇÃO: criamos explicitamente:

- **Constraint NOT NULL** para a coluna **sobrenome**
- **Constraint NOT NULL** para a coluna **email**
- **Constraint NOT NULL** para a coluna **data_admissao**
- **Constraint NOT NULL** para a coluna **id_funcao**
- **Constraint CHECK** para a coluna **salario**
- **Constraint unique** para a coluna **email**

- **Índice** para a coluna **id_empregado**
- **Constraint PRIMARY KEY** para a coluna **id_empregado**
- **Constraint FOREIGN KEY** para a coluna **id_departamento**
- **Constraint FOREIGN KEY** para a coluna **id_funcao**
- **Constraint FOREIGN KEY** para a coluna **id_gerente**
- **Constraint FOREIGN KEY** para a tabela **tb_departamento** para a coluna **id_gerente**
- Uma **sequência** nomeada de **seq_empregado**

```
CREATE TABLE tb_empregado(
id_empregado          NUMBER(6),
nome                  VARCHAR2(20),
sobrenome              VARCHAR2(25) CONSTRAINT nn_emp_sobrenome NOT NULL,
email                  VARCHAR2(25) CONSTRAINT nn_emp_email NOT NULL,
telefone              VARCHAR2(20),
data_admissao          DATE CONSTRAINT nn_emp_dt_adm NOT NULL,
id_funcao              VARCHAR2(10) CONSTRAINT nn_emp_funcao NOT NULL,
salario                NUMBER(8,2),
percentual_comissao    NUMBER(2,2),
id_gerente             NUMBER(6),
id_departamento       NUMBER(4),
CONSTRAINT min_emp_salario CHECK (salario > 0),
CONSTRAINT un_emp_email UNIQUE (email));
```

```
CREATE UNIQUE INDEX pk_id_emp
ON tb_empregado (id_empregado);
```

```
ALTER TABLE tb_empregado
ADD ( CONSTRAINT pk_id_emp
      PRIMARY KEY (id_empregado),
      CONSTRAINT fk_emp_depto
      FOREIGN KEY (id_departamento) REFERENCES tb_departamento,
      CONSTRAINT fk_emp_funcao
      FOREIGN KEY (id_funcao) REFERENCES tb_funcao (id_funcao),
      CONSTRAINT fk_emp_gerente
      FOREIGN KEY (id_gerente) REFERENCES tb_empregado);
```

```
ALTER TABLE tb_departamento
ADD ( CONSTRAINT fk_gerente_depto
      FOREIGN KEY (id_gerente) REFERENCES tb_empregado (id_empregado));
```

```
CREATE SEQUENCE seq_empregado
START WITH            207
INCREMENT BY          1
NOCACHE
NOCYCLE;
```

8) Crie a tabela **tb_historico_funcao**, conforme a instrução SQL abaixo.

OBSERVAÇÃO: criamos explicitamente:

- **Constraint NOT NULL** para a coluna **id_empregado**
- **Constraint NOT NULL** para a coluna **data_inicio**
- **Constraint NOT NULL** para a coluna **data_termino**
- **Constraint NOT NULL** para a coluna **id_funcao**
- **Constraint CHECK** para a coluna **data_termino**
- **Índice** para as colunas **id_empregado** e **data_inicio**
- **Constraint PRIMARY KEY** para as colunas **id_empregado** e **data_inicio**
- **Constraint FOREIGN KEY** para a coluna **id_funcao**
- **Constraint FOREIGN KEY** para a coluna **id_empregado**
- **Constraint FOREIGN KEY** para a coluna **id_departamento**

```
CREATE TABLE tb_historico_funcao(  
id_empregado          NUMBER(6) CONSTRAINT nn_hist_emp_id_emp NOT NULL,  
data_inicio           DATE CONSTRAINT nn_hist_emp_dt_inicio NOT NULL,  
data_termino          DATE CONSTRAINT nn_hist_emp_dt_termino NOT NULL,  
id_funcao             VARCHAR2(10) CONSTRAINT nn_hist_emp_id_funcao NOT NULL,  
id_departamento      NUMBER(4),  
CONSTRAINT ck_hist_emp_data_intervalo CHECK (data_termino > data_inicio));
```

```
CREATE UNIQUE INDEX pk_hist_emp_id_emp  
ON tb_historico_funcao (id_empregado, data_inicio);
```

```
ALTER TABLE tb_historico_funcao  
ADD ( CONSTRAINT pk_hist_emp_id_emp  
      PRIMARY KEY (id_empregado, data_inicio),  
      CONSTRAINT fk_hist_funcao_funcao  
      FOREIGN KEY (id_funcao) REFERENCES tb_funcao,  
      CONSTRAINT fk_hist_funcao_emp  
      FOREIGN KEY (id_empregado) REFERENCES tb_empregado,  
      CONSTRAINT fk_hist_funcao_depto  
      FOREIGN KEY (id_departamento) REFERENCES tb_departamento);
```