

-- 1) Cadastro de clientes:

```
CREATE TABLE cliente (  
    clino NUMBER (2,0),  
    clinome VARCHAR2 (30),  
    clicpf NUMBER (11),  
    clinasc DATE,  
    cliemail VARCHAR2 (30),  
  
    CONSTRAINT pk_cliente PRIMARY KEY (clino)  
);
```

```
CREATE OR REPLACE PROCEDURE sp_cadastrar_cliente  
    (p_cod IN cliente.clino%TYPE,  
    p_nome IN cliente.clinome%TYPE,  
    p_cpf IN cliente.clicpf%TYPE),  
    p_nascimento IN cliente.clinasci%TYPE),  
    p_email IN cliente.cliemail%TYPE)  
IS  
BEGIN  
    INSERT INTO cliente (clino, clinome, clicpf, clinasc, cliemail)  
    VALUES (p_cod, p_nome, p_cpf, p_nascimento, p_email);  
END sp_cadastrar_cliente;  
/  
  
BEGIN
```

```
sp_cadastrar_cliente (01, 'Matheus Silva', '291.602.916-50', TO_DATE ('22-FEV-1981',  
'DD-MON-YYYY'), 'msilva123@gmail.com');  
END;  
/
```

-- 2) Cadastro de Produtos:

```
CREATE OR REPLACE PROCEDURE criar_tabela_produto AS
```

```
BEGIN
```

```
-- Cria a tabela "produto" se ainda não existir
```

```
EXECUTE IMMEDIATE 'CREATE TABLE produto (
```

```
    id NUMBER PRIMARY KEY,
```

```
    nome VARCHAR2(100),
```

```
    preco NUMBER,
```

```
    categoria_id NUMBER,
```

```
    feedback_id NUMBER,
```

```
    datafabricacao DATE,
```

```
    FOREIGN KEY (categoria_id) REFERENCES categoria(id),
```

```
    FOREIGN KEY (feedback_id) REFERENCES feedback(id)
```

```
);
```

```
EXCEPTION
```

```
    WHEN OTHERS THEN
```

```
        IF SQLCODE = -955 THEN
```

```
            NULL; -- Tabela já existe
```

```
        ELSE
```

```
            RAISE;
```

```
        END IF;
```

```
END;
```

```
/
```

-- 3) Cadastro de Feedbacks:

```
CREATE TABLE feedbacks (
```

```
    id INT PRIMARY KEY,
```

```
cliente VARCHAR2(100),  
produto VARCHAR2(100),  
comentario VARCHAR2(100)  
);
```

```
-- Criar a sequência para gerar o ID da tabela "feedbacks"
```

```
CREATE SEQUENCE feedbacks_seq START WITH 1 INCREMENT BY 1;
```

```
-- Criar a trigger para automatizar o cadastro de feedbacks
```

```
CREATE OR REPLACE TRIGGER trg_cadastro_feedback
```

```
AFTER INSERT ON clientes
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO feedbacks (id, cliente, produto, comentario)
```

```
VALUES (feedbacks_seq.NEXTVAL, :new.nome, :new.produto, 'Sem comentário');
```

```
END;
```

```
/
```

```
-- 4) Criação de Tabelas
```

```
-- Criação da procedure para criar a tabela
```

```
CREATE OR REPLACE PROCEDURE criar_tabela AS
```

```
BEGIN
```

```
-- Criação da tabela Usuário
```

```
EXECUTE IMMEDIATE 'CREATE TABLE usuario
```

```
clino NUMBER (2,0),
```

```
clinome VARCHAR2 (30),
```

```
clicpf NUMBER (11),
```

```
clinasc DATE,
```

```
cliemail VARCHAR2 (30),
```

```
CONSTRAINT pk_cliente PRIMARY KEY (clino)
```

-- Criação da tabela Produto

```
EXECUTE IMMEDIATE 'CREATE TABLE produto (  
    id_produto NUMBER PRIMARY KEY,  
    nome VARCHAR2(100),  
    preco NUMBER,  
    categoria_id NUMBER,  
    feedback_id NUMBER,  
    datafabricacao DATE,  
    FOREIGN KEY (categoria_id) REFERENCES categoria(id),  
    FOREIGN KEY (feedback_id) REFERENCES feedback(id)  
);
```

-- Criação da tabela Feedback

```
EXECUTE IMMEDIATE 'CREATE TABLE feedback (  
    id_feedback NUMBER PRIMARY KEY,  
    cliente VARCHAR2(100),  
    produto VARCHAR2(100),  
    comentario VARCHAR2(100)  
    FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario),  
    FOREIGN KEY (id_produto) REFERENCES produto(id_produto)  
);
```

-- Mensagem de confirmação

```
DBMS_OUTPUT.PUT_LINE('As tabelas foram criadas com sucesso!');
```

END;

/

-- Chamada da procedure

BEGIN

criar_tabela;

END;

/