

```

-- =====
-- EXEMPLO COMPLETO - SQL SERVER
-- SUBSELECTS | SWITCH (CASE) | VIEWS
-- =====

-- Apagar tabelas antigas (caso existam)
DROP TABLE IF EXISTS Projeto;
DROP TABLE IF EXISTS Funcionario;
DROP TABLE IF EXISTS Departamento;

-- =====
-- 1. Criação das Tabelas
-- =====

-- Tabela Departamento
CREATE TABLE Departamento (
    DepartamentoID INT PRIMARY KEY,
    Nome VARCHAR(100)
);

-- Tabela Funcionario
CREATE TABLE Funcionario (
    FuncionarioID INT PRIMARY KEY,
    Nome VARCHAR(100),
    Salario DECIMAL(10,2),
    DepartamentoID INT,
    FOREIGN KEY (DepartamentoID) REFERENCES Departamento(DepartamentoID)
);

-- Tabela Projeto
CREATE TABLE Projeto (
    ProjetoID INT PRIMARY KEY,
    Nome VARCHAR(100),
    ResponsavelID INT,
    FOREIGN KEY (ResponsavelID) REFERENCES Funcionario(FuncionarioID)
);

-- =====
-- 2. Inserção de Dados
-- =====

-- Departamentos
INSERT INTO Departamento VALUES
(1, 'TI'),
(2, 'Financeiro'),
(3, 'RH');

```

-- Funcionários

INSERT INTO Funcionario VALUES

(1, 'João', 9500.00, 1),  
(2, 'Maria', 12000.00, 2),  
(3, 'Carlos', 4800.00, 2),  
(4, 'Ana', 7000.00, 3),  
(5, 'Pedro', 8600.00, 2);

-- Projetos

INSERT INTO Projeto VALUES

(1, 'Sistema ERP', 1),  
(2, 'Relatório Financeiro', 2),  
(3, 'Campanha de Recrutamento', 4);

-- =====

-- 3. Exemplo de SUBSELECT

-- =====

-- Funcionários com salário acima da média:

SELECT Nome, Salario

FROM Funcionario

WHERE Salario > (SELECT AVG(Salario) FROM Funcionario);

-- Projetos com responsáveis do departamento de TI:

SELECT Nome

FROM Projeto

WHERE ResponsavelID IN (

    SELECT FuncionarioID FROM Funcionario WHERE DepartamentoID = 1

);

-- Funcionários sem projetos atribuídos:

SELECT Nome

FROM Funcionario

WHERE FuncionarioID NOT IN (

    SELECT ResponsavelID FROM Projeto

);

-- Funcionários do mesmo departamento da Maria

SELECT Nome

FROM Funcionario

WHERE DepartamentoID = (

    SELECT DepartamentoID

```
FROM Funcionario
WHERE Nome = 'Maria'
);
```

```
-- =====
-- 4. Exemplo de SWITCH (CASE WHEN)
-- =====
```

```
-- Classificar faixa salarial
SELECT
    Nome,
    Salario,
    CASE
        WHEN Salario >= 10000 THEN 'Alto'
        WHEN Salario >= 7000 THEN 'Médio'
        ELSE 'Baixo'
    END AS FaixaSalarial
FROM Funcionario;
```

```
-- Definir a situação do bônus com múltiplas faixas de valores.
```

```
SELECT
    Nome,
    Salario,
    CASE
        WHEN Salario >= 10000 THEN 'Bônus Completo'
        WHEN Salario >= 8000 AND Salario < 10000 THEN 'Bônus Parcial'
        WHEN Salario >= 5000 AND Salario < 8000 THEN 'Bônus Simples'
        ELSE 'Sem Bônus'
    END AS SituacaoBonus
FROM Funcionario;
```

```
-- =====
-- 5. Criação de VIEW
-- =====
```

```
-- View com nome do projeto, responsável e departamento:
CREATE OR ALTER VIEW vw_ProjetosDetalhados AS
SELECT
    p.Nome AS Projeto,
    f.Nome AS Responsavel,
    d.Nome AS Departamento
FROM Projeto p
INNER JOIN Funcionario f ON p.ResponsavelID = f.FuncionarioID
INNER JOIN Departamento d ON f.DepartamentoID = d.DepartamentoID;
```

```
-- Consulta:  
SELECT * FROM vw_ProjetosDetalhados;
```