**PROJETO**

**Banco de Dados I**

NOME:

Eric Dias

**2024**

**-- A. Inserir no mínimo 5 registros em cada tabela**

**-- O código insere 5 registros em cada tabela: Permissao, Usuario, Setor, Cargo, Funcionario, Projeto, Tarefa, FuncionarioProjeto, PermissaoUsuario e FuncionarioTarefa, preenchendo os dados com exemplos variados.**

*INSERT INTO Permissao (nome) VALUES*

*('Admin'), ('Editor'), ('Visualizador'), ('Analista'), ('Gerente');*

*INSERT INTO Usuario (senha, login, nome) VALUES*

*(MD5('senha1'), 'usuario1', 'João Silva'),*

*(MD5('senha2'), 'usuario2', 'Maria Oliveira'),*

*(MD5('senha3'), 'usuario3', 'Pedro Santos'),*

*(MD5('senha4'), 'usuario4', 'Ana Costa'),*

*(MD5('senha5'), 'usuario5', 'Lucas Lima');*

*INSERT INTO Setor (nome) VALUES*

*('Financeiro'), ('TI'), ('Recursos Humanos'), ('Marketing'), ('Operações');*

*INSERT INTO Cargo (salario, nome) VALUES*

*(3500.00, 'Analista'), (5000.00, 'Gerente'), (4500.00, 'Coordenador'),*

*(4000.00, 'Supervisor'), (3000.00, 'Assistente');*

*INSERT INTO Funcionario (setor\_id, user\_id, cargo\_id, cpf, nome) VALUES*

*(1, 1, 2, '12345678901', 'João Silva'),*

*(2, 2, 1, '23456789012', 'Maria Oliveira'),*

*(3, 3, 3, '34567890123', 'Pedro Santos'),*

*(4, NULL, 4, '45678901234', 'Ana Costa'),*

*(5, 4, 5, '56789012345', 'Lucas Lima');*

*INSERT INTO Projeto (nome, descricao, data\_inicio, data\_final) VALUES*

*('Projeto A', 'Projeto estratégico A', '2024-01-01', '2024-06-01'),*

*('Projeto B', 'Projeto estratégico B', '2024-02-01', '2024-07-01'),*

*('Projeto C', 'Projeto de expansão C', '2024-03-01', '2024-08-01'),*

*('Projeto D', 'Projeto interno D', '2024-04-01', '2024-09-01'),*

*('Projeto E', 'Projeto técnico E', '2024-05-01', '2024-10-01');*

*INSERT INTO Tarefa (projeto\_id, status, prazo, descricao) VALUES*

*(1, 'Pendente', '2024-02-01', 'Tarefa 1 do Projeto A'),*

*(1, 'Em Progresso', '2024-03-01', 'Tarefa 2 do Projeto A'),*

*(2, 'Concluido', '2024-02-15', 'Tarefa 1 do Projeto B'),*

*(3, 'Pendente', '2024-05-01', 'Tarefa 1 do Projeto C'),*

*(4, 'Em Progresso', '2024-06-01', 'Tarefa 1 do Projeto D');*

*INSERT INTO FuncionarioProjeto (func\_id, projeto\_id) VALUES*

*(1, 1), (2, 1), (3, 2), (4, 3), (5, 4);*

*INSERT INTO PermissaoUsuario (user\_id, permissao\_id) VALUES*

*(1, 1), (2, 2), (3, 3), (4, 4), (5, 5);*

*INSERT INTO FuncionarioTarefa (func\_id, tarefa\_id) VALUES*

*(1, 1), (2, 2), (3, 3), (4, 4), (5, 5);*

**-- B. Consulta envolvendo uma tabela**

**-- Retorna todos os registros da tabela Funcionario onde o campo "ativo" é verdadeiro.**

*SELECT \* FROM Funcionario WHERE ativo = TRUE;*

**-- C. Consulta envolvendo duas tabelas com equi-join**

**-- Exibe o nome do funcionário e o nome de seu cargo, relacionando as tabelas Funcionario e Cargo pela coluna cargo\_id.**

*SELECT*

*f.nome AS funcionario,*

*c.nome AS cargo*

*FROM*

*Funcionario f JOIN Cargo c ON f.cargo\_id = c.id;*

**-- D. Consulta envolvendo duas tabelas com inner-join**

**-- Mostra o nome do usuário e sua respectiva permissão, unindo as tabelas Usuario, PermissaoUsuario e Permissao.**

*SELECT*

*u.nome AS usuario,*

*p.nome AS permissao*

*FROM*

*Usuario u INNER JOIN PermissaoUsuario pu ON u.id = pu.user\_id*

*INNER JOIN Permissao p ON pu.permissao\_id = p.id;*

**-- E. Consulta envolvendo três tabelas com equi-join**

**-- Lista o nome do funcionário, o nome do projeto e a descrição da tarefa associada, usando as tabelas Funcionario, FuncionarioProjeto, Projeto e Tarefa.**

*SELECT*

*f.nome AS funcionario,*

*p.nome AS projeto,*

*t.descricao AS tarefa*

*FROM*

*Funcionario f JOIN FuncionarioProjeto fp ON f.id = fp.func\_id*

*JOIN Projeto p ON fp.projeto\_id = p.id*

*JOIN Tarefa t ON p.id = t.projeto\_id;*

**-- F. Consulta envolvendo três tabelas com inner-join**

**-- Apresenta o nome do usuário, do funcionário e do setor ao qual ele pertence, unindo Usuario, Funcionario e Setor.**

*SELECT*

*u.nome AS usuario,*

*f.nome AS funcionario,*

*s.nome AS setor*

*FROM*

*Usuario u INNER JOIN Funcionario f ON u.id = f.user\_id*

*INNER JOIN Setor s ON f.setor\_id = s.id;*

**-- G. Consulta envolvendo mais de três tabelas**

**-- Mostra o nome do funcionário, do projeto, da tarefa e do cargo associado, unindo Funcionario, FuncionarioProjeto, Projeto, Tarefa e Cargo.**

*SELECT*

*f.nome AS funcionario,*

*p.nome AS projeto,*

*t.descricao AS tarefa,*

*c.nome AS cargo*

*FROM*

*Funcionario f JOIN FuncionarioProjeto fp ON f.id = fp.func\_id*

*JOIN Projeto p ON fp.projeto\_id = p.id*

*JOIN Tarefa t ON p.id = t.projeto\_id*

*JOIN Cargo c ON f.cargo\_id = c.id;*

**-- H. Consulta usando NULL ou NOT NULL**

**-- Lista o nome do funcionário e o login do usuário, exibindo apenas os registros onde o login não é nulo.**

*SELECT*

*f.nome,*

*u.login*

*FROM*

*Funcionario f LEFT JOIN Usuario u ON f.user\_id = u.id*

*WHERE*

*u.login IS NOT NULL;*

**-- I. Consulta usando LIKE com %**

**-- Busca os usuários cujo login contém a palavra "usuario", retornando nome e login.**

*SELECT*

*nome, login*

*FROM*

*Usuario*

*WHERE*

*login LIKE '%usuario%';*

**-- J. Consulta usando LIKE com \_**

**-- Filtra usuários cujo login começa com "usuario" seguido por um único caractere adicional, exibindo nome e login.**

*SELECT*

*nome, login*

*FROM*

*Usuario*

*WHERE*

*login LIKE 'usuario\_';*

**-- K. Consulta com subconsulta usando IN**

**-- Retorna o nome de funcionários que pertencem a setores cujo nome começa com a letra "T".**

*SELECT*

*nome*

*FROM*

*Funcionario*

*WHERE*

*setor\_id IN (SELECT id FROM Setor WHERE nome LIKE 'T%');*

**-- L. Consulta com subconsulta usando ANY**

**-- Lista o nome de funcionários cujo cargo possui um salário maior que 4000, utilizando a subconsulta na tabela Cargo.**

*SELECT*

*nome*

*FROM*

*Funcionario*

*WHERE*

*cargo\_id = ANY (SELECT id FROM Cargo WHERE salario > 4000);*

**-- M. Consulta com subconsulta usando EXISTS**

**-- Mostra o nome de funcionários que possuem associação com pelo menos um projeto na tabela FuncionarioProjeto.**

*SELECT*

*nome*

*FROM*

*Funcionario*

*WHERE*

*EXISTS (SELECT 1 FROM FuncionarioProjeto WHERE func\_id = Funcionario.id);*

**-- N. Consulta usando UNION**

**-- Combina e retorna os nomes das tabelas Setor e Cargo, sem duplicar resultados.**

*SELECT*

*nome*

*FROM*

*Setor*

*UNION SELECT*

*nome*

*FROM*

*Cargo;*

**-- O. Consulta usando GROUP BY e função agregada**

**-- Agrupa os funcionários por setor e calcula o total de funcionários em cada setor.**

*SELECT*

*s.nome AS setor,*

*COUNT(f.id) AS total\_funcionarios*

*FROM*

*Funcionario f JOIN Setor s ON f.setor\_id = s.id*

*GROUP BY*

*s.nome;*

**-- P. Consulta usando GROUP BY e HAVING**

**-- Agrupa os funcionários por cargo e calcula o salário médio, exibindo apenas os cargos com salário médio maior que 3500.**

*SELECT*

*c.nome AS cargo,*

*AVG(c.salario) AS salario\_medio*

*FROM*

*Funcionario f JOIN Cargo c ON f.cargo\_id = c.id*

*GROUP BY*

*c.nome*

*HAVING*

*AVG(c.salario) > 3500;*

**-- Q. Consulta usando DELETE**

**-- Remove o registro do funcionário com id igual a 5.**

*DELETE FROM Funcionario WHERE id = 5;*

**-- R. Consulta usando UPDATE**

**-- Atualiza o nome do funcionário com id 1 para "João Atualizado" e altera seu setor para o de id 3.**

*UPDATE*

*Funcionario*

*SET*

*nome = 'João Atualizado',*

*setor\_id = 3*

*WHERE*

*id = 1;*

**-- S. Consulta usando DROP**

**-- Exclui a tabela FuncionarioProjeto caso ela exista no banco de dados.**

*DROP TABLE IF EXISTS FuncionarioProjeto;*

**-- T. Criação de visão envolvendo três tabelas**

**-- Cria uma visão chamada VisaoProjetos que exibe o nome do funcionário, o nome do projeto e a descrição da tarefa associada.**

*CREATE VIEW VisaoProjetos AS*

*SELECT*

*f.nome AS funcionario,*

*p.nome AS projeto,*

*t.descricao AS tarefa*

*FROM*

*Funcionario f JOIN FuncionarioProjeto fp ON f.id = fp.func\_id*

*JOIN Projeto p ON fp.projeto\_id = p.id*

*JOIN Tarefa t ON p.id = t.projeto\_id;*