**Atividade de Banco de Dados: Projeto Completo e Detalhado**

**Objetivo:** Desenvolver um sistema de gerenciamento de dados utilizando o MySQL, abordando desde a criação do banco de dados e tabelas até a implementação de operações CRUD (Create, Read, Update, Delete), normalização de dados, diferentes tipos de JOINs, agregeações e documentação técnica completa.

**Parte 1: Criação e Configuração do Banco de eeDados**

**1. Criação do Banco de Dados**

* Crie um banco de dados chamado sistema\_gestao.

**2. Criação de Tabelas**

* Crie as seguintes tabelas no banco de dados sistema\_gestao:
  + **clientes**: id, nome, telefone, data\_nascimento, estado\_civil, sexo, cidade, idade.
  + **pedidos**: id, cliente\_id (chave estrangeira para a tabela clientes), produto, quantidade.
  + **produtos**: id, nome, preco.
  + **estoque**: produto\_id (chave estrangeira para a tabela produtos), quantidade\_disponivel.

**3. Inserção de Dados**

* Insira dados nas tabelas conforme os exemplos fornecidos no documento.

**Parte 2: Operações CRUD e Consultas SQL**

**4. Operações CRUD**

* Realize operações de inserção, leitura, atualização e exclusão de dados em cada uma das tabelas.

INSERÇÃO

INSERT INTO clientes ( id,nome,telefone,data\_nascimento,estado\_civil,sexo,cidade,idade) values

(‘1’,‘miguel sousa silva’, ‘86994911088’,’20040225’,’C’,’M’,’São Paulo’,’20’);

SELEÇÃO

select \* from clientes;

ATUALIZAÇÃO

update clientes set nome = ‘Miguel de sousa da silva’ where id = ‘1’

DELETE

delete from estoque where produto\_id = ‘11’

**5. Consultas SQL Avançadas**

* Realize as seguintes consultas:
  + Selecione todos os clientes que têm mais de 30 anos.

SELECT \* FROM clientes WHERE idade > 30;

* + Atualize o endereço de um cliente específico.
  + Delete todos os pedidos feitos por um cliente específico.

Delete from pedidos where cliente\_id = ‘3’ ;

* + Insira múltiplos produtos no estoque de uma só vez.

INSERT INTO estoque VALUES

(2,20),

(3,10),

(1,20),

(3,15),

(5,10),

(6,30);

* + Atualize o preço de todos os produtos com um aumento de 10%.

update produtos set preco = preco + (preco \* 0.1);

**6. Consultas Específicas**

* Utilize filtros e ordenações para:
  + Encontrar clientes cujo nome começa com 'A'.
  + Listar pedidos feitos em um intervalo de datas.
  + Ordenar produtos pelo preço, do mais caro ao mais barato.

select \* from produtos where preco order by preco desc;

* + Buscar produtos com quantidade disponível menor que 10.

select \* from estoque where quantidade\_disponivel < 10;

**Parte 3: Normalização de Dados**

**7. Normalização**

* Normalize as tabelas de acordo com as três primeiras Formas Normais (1FN, 2FN e 3FN):
  + Reestruturação para Primeira Forma Normal (1FN)
  + Reestruturação para Segunda Forma Normal (2FN)
  + Reestruturação para Terceira Forma Normal (3FN)

**Parte 4: Join e Relacionamentos**

**8. Utilização de JOINs**

* Utilize diferentes tipos de JOINs para combinar dados das tabelas:
  + **INNER JOIN**: Selecione todos os clientes que fizeram pedidos, juntamente com os detalhes dos pedidos.
  + **LEFT JOIN**: Liste todos os clientes, independentemente de terem feito pedidos ou não.
  + **RIGHT JOIN**: Encontre todos os produtos em estoque, juntamente com os detalhes dos produtos.
  + **FULL JOIN**: Combine os resultados de clientes e pedidos, mesmo aqueles sem correspondências (simulado com UNION).

**Parte 5: Agregações e Funções**

**9. Funções de Agregação**

* Utilize funções de agregação para:
  + Contar o número total de clientes.

select count(\*) as 'QUANTIDADE DE CLIENTES' from clientes;

* + Calcular a soma total de todas as vendas.

select sum(quantidade) as 'SOMA DOS PEDIDOS' from pedidos;

* + Encontrar a média de idade dos clientes.

select avg(idade) as 'média da idades' from clientes;

* + Determinar a idade mínima e máxima dos clientes.

SELECT MIN(idade) as 'Idade Mínima' FROM clientes;

SELECT MAX(idade) as 'Idade Máxima' FROM clientes;

* + Contar quantas cidades distintas existem na tabela de clientes.

select count(distinct cidade)as 'Quantidade de cidades' from clientes;

**Parte 6: Documentação Técnica Completa**

**10. Documentação Técnica**

* Crie a documentação completa do sistema, incluindo:
  + **Requisitos do Sistema**: Descrição das funcionalidades esperadas.
  + **Diagrama Entidade-Relacionamento (ERD)**: Representação visual das tabelas e seus relacionamentos.
  + **Especificações de Tabelas**: Detalhamento de cada tabela com seus campos, tipos de dados, chaves primárias e estrangeiras, e quaisquer restrições.
  + **Plano de Backup e Recuperação de Dados**: Procedimentos para garantir a segurança dos dados.(pesquisar)
  + **Controle de Acesso e Segurança**: Definição de permissões e segurança para os usuários do banco de dados.(Pesquisar)

**Parte 7: Exercícios Práticos**

**11. Exercícios Práticos**

* Responda as seguintes perguntas:
  1. Explique, com suas palavras, o que é um banco de dados e por que ele é importante.
  2. Compare bancos de dados relacionais e não-relacionais, destacando suas principais diferenças e exemplos de uso.
  3. Descreva as especificações de uma tabela Produto com as colunas id, nome, preco e quantidade.
  4. Pesquise sobre as diferenças entre MySQL e PostgreSQL e liste suas vantagens e desvantagens.
  5. Crie uma documentação básica para um sistema de gerenciamento de estoque, incluindo requisitos do sistema e diagrama ERD.
  6. Explique a função de um SGBD e cite três exemplos de SGBDs populares.
  7. Descreva como implementar um controle de acesso básico no MySQL.(Pesquisar)
  8. Explique a importância da normalização de dados e dê exemplos práticos.

**Entrega**

* **Código SQL**: Incluindo a criação do banco de dados, tabelas, inserção de dados, operações CRUD, normalização, JOINs e agregações.
* **Documentação**: Arquivo contendo os requisitos do sistema, diagrama ERD, especificações das tabelas, plano de backup e recuperação de dados, e controle de acesso e segurança.
* **Respostas dos Exercícios**: com as respostas às perguntas práticas.