

1) Defina o que é Banco de Dados

Um banco de dados é uma coleção organizada de informações ou dados, geralmente armazenados eletronicamente em um sistema de computador. É controlado por um sistema de gerenciamento de banco de dados (DBMS) e inclui os dados, o DBMS e os aplicativos associados a eles, formando o sistema de banco de dados. Os dados nos bancos de dados são estruturados em tabelas com linhas e colunas, permitindo um processamento eficiente e consultas aos dados. Esses dados podem ser facilmente acessados, gerenciados, modificados, atualizados, controlados e organizados. A linguagem de consulta estruturada (SQL) é frequentemente utilizada para escrever e consultar dados em bancos de dados relacionais.

2) Defina SGBD e dê exemplos

SGBD (Sistema de Gerenciamento de Banco de Dados) é um software responsável pelo gerenciamento e controle dos dados em um banco de dados por meio de uma interface gráfica. Ele permite a organização, acesso, manipulação, e proteção das informações de uma empresa.

Um SGBD facilita o gerenciamento do banco de dados, controla o acesso de usuários, define privilégios, realiza backups e restaurações, além de fornecer informações e auxiliar nas operações com os dados.

Existem diferentes tipos de SGBDs, sendo os principais:

SGBDs Relacionais: São os mais conhecidos e utilizados há algum tempo. Eles organizam os dados em tabelas, colunas e registros, e utilizam a linguagem SQL para executar comandos. Exemplos de SGBDs relacionais são o MySQL, PostgreSQL e Microsoft SQL Server.

SGBDs NoSQL: Esses sistemas, também conhecidos como "Not Only SQL", estão ganhando espaço nas organizações. Eles são utilizados principalmente quando há necessidade de escalabilidade e manipulação de grandes volumes de dados. Alguns exemplos de SGBDs NoSQL são o MongoDB e o Apache Cassandra.

É importante destacar que os SGBDs são diferentes dos bancos de dados em si. O termo "Banco de Dados" é frequentemente usado para se referir aos SGBDs, mas tecnicamente o SGBD é o software responsável pelo gerenciamento dos bancos de dados.

Em resumo, um SGBD é um sistema de software que gerencia e controla bancos de dados, oferecendo recursos para facilitar o acesso, a organização e a proteção das informações, com diferentes tipos de SGBDs disponíveis, como os relacionais e os NoSQL.

3) Defina DDL e DML

DDL (Data Definition Language) e DML (Data Manipulation Language) são duas sublinguagens da linguagem SQL (Structured Query Language) que são usadas para interagir com bancos de dados relacionais.

DDL (Linguagem de Definição de Dados) é responsável pela definição da estrutura dos dados no banco de dados. As principais instruções DDL incluem:

CREATE: usada para criar objetos de banco de dados, como tabelas, índices, visões, procedimentos armazenados, funções e gatilhos.

ALTER: usada para alterar a estrutura dos objetos existentes no banco de dados.

DROP: usada para excluir objetos do banco de dados.

TRUNCATE: usada para remover todos os registros de uma tabela, liberando o espaço alocado para os registros.

RENAME: usada para renomear um objeto.

Por outro lado, DML (Linguagem de Manipulação de Dados) é usada para manipular os dados em si. As principais instruções DML incluem:

SELECT: usada para recuperar dados de uma ou mais tabelas.

INSERT: usada para inserir novos registros em uma tabela.

UPDATE: usada para atualizar registros existentes em uma tabela.

DELETE: usada para excluir registros de uma tabela.

MERGE: usada para realizar operações de inserção ou atualização (UPSERT).

CALL: usada para chamar um subprograma em PL/SQL ou Java.

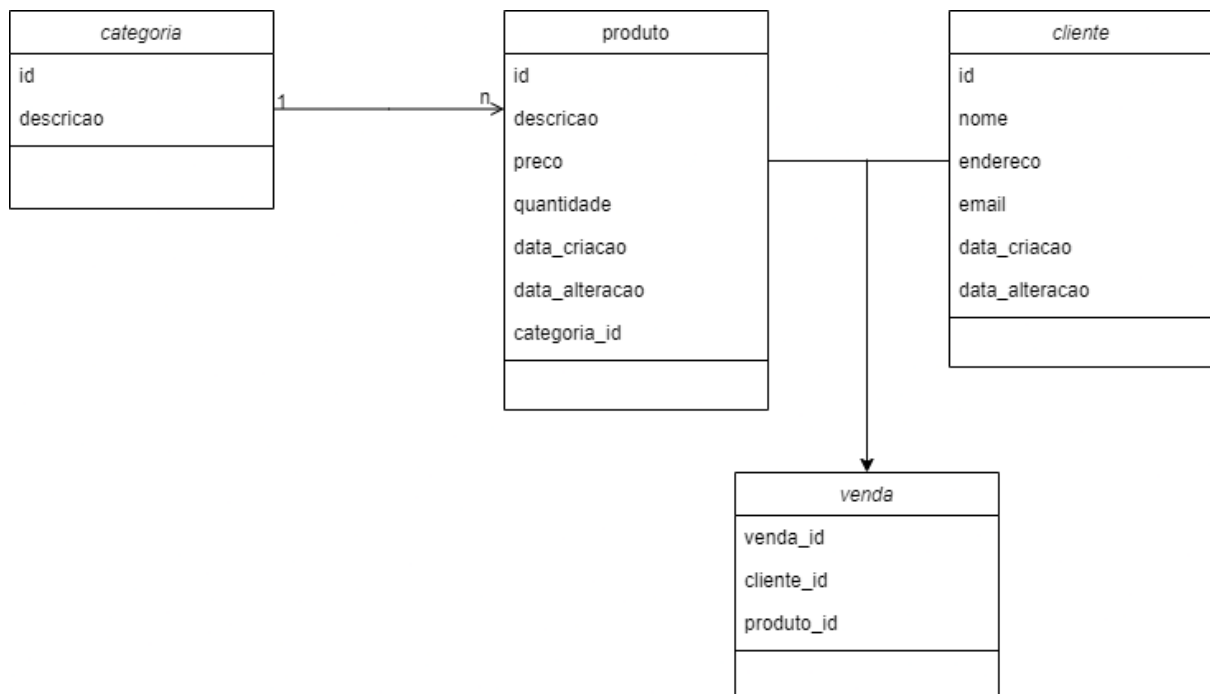
EXPLAIN PLAN: usada para interpretar o plano de acesso aos dados.

LOCK TABLE: usada para controle de concorrência.

Essas instruções DDL e DML permitem que os usuários definam a estrutura dos dados e realizem operações de manipulação nesses dados, respectivamente. É importante notar que existem outras sublinguagens na linguagem SQL, como DQL (Data Query Language) e DCL (Data Control Language), que também desempenham papéis específicos na interação com bancos de dados.

4) Faça o Diagrama Entidade-Relacionamentos

Faça o Diagrama Entidade-Relacionamentos do estudo de caso a seguir: Uma empresa pretende vender produtos pela internet através de uma loja virtual. Portanto, está precisando de um sistema para gerenciar os produtos, as vendas e seus futuros clientes. Cada produto tem uma categoria, que possui: o id e a descrição. Deve-se fazer um cadastro de clientes contendo: id, nome, endereço, email e datas de criação e alteração das suas informações. Os produtos devem ser gerenciados pelo sistema com os dados: id, descrição, preço, quantidade, categoria e datas de criação e alteração das suas informações. As vendas são geradas entre clientes e produtos. Onde um cliente compra vários produtos e um tipo de produto pode ser comprado por inúmeros clientes.



5) Use os comandos DDL disponíveis no SQL para criação das tabelas da questão 5.

-- Tabela de Categorias

```
CREATE TABLE categorias (  
  id INT PRIMARY KEY,  
  descricao VARCHAR(50)  
);
```

-- Tabela de Produtos

```
CREATE TABLE produtos (  
  id INT PRIMARY KEY,  
  descricao VARCHAR(100),  
  preco DECIMAL(10, 2),  
  quantidade INT,  
  categoria_id INT,  
  data_criacao DATETIME,  
  data_alteracao DATETIME,  
  FOREIGN KEY (categoria_id) REFERENCES categorias (id)  
);
```

-- Tabela de Clientes

```
CREATE TABLE clientes (  
  id INT PRIMARY KEY,  
  nome VARCHAR(50),  
  endereco VARCHAR(100),  
  email VARCHAR(50),  
  data_criacao DATETIME,  
  data_alteracao DATETIME  
);
```

-- Tabela de Vendas

```
CREATE TABLE Vendas (  
  id INT PRIMARY KEY,  
  cliente_id INT,  
  produto_id INT,  
  data_venda DATETIME,  
  FOREIGN KEY (cliente_id) REFERENCES clientes (id),  
  FOREIGN KEY (produto_id) REFERENCES produtos (id)  
);
```

6) Use os comandos DML para “popular” e “explorar” as informações da base

-- Inserindo dados na tabela categorias

```
INSERT INTO categorias (id, descricao)
VALUES (1, 'Eletrônicos'),
       (2, 'Roupas'),
       (3, 'Acessórios');
```

-- Inserindo dados na tabela produtos

```
INSERT INTO produtos (id, descricao, preco, quantidade, categoria_id, data_criacao,
data_alteracao)
VALUES (1, 'Smartphone', 1500.00, 10, 1, '2023-06-01 09:00:00', '2023-06-02 14:30:00'),
       (2, 'Notebook', 2500.00, 5, 1, '2023-06-03 11:45:00', '2023-06-04 16:20:00'),
       (3, 'Camiseta', 50.00, 20, 2, '2023-06-05 08:15:00', '2023-06-06 10:10:00'),
       (4, 'Calça', 80.00, 15, 2, '2023-06-07 13:20:00', '2023-06-08 09:45:00');
```

-- Inserindo dados na tabela clientes

```
INSERT INTO clientes (id, nome, endereco, email, data_criacao, data_alteracao)
VALUES (1, 'João Silva', 'Rua das Flores, 123', 'joao@email.com', '2023-06-01 09:00:00',
'2023-06-01 09:00:00'),
       (2, 'Maria Santos', 'Av. Principal, 456', 'maria@email.com', '2023-06-02 14:30:00',
'2023-06-03 16:00:00');
```

-- Inserindo dados na tabela vendas

```
INSERT INTO vendas (id, cliente_id, produto_id, data_venda)
VALUES (1, 1, 1, '2023-06-03 10:30:00'),
       (2, 1, 3, '2023-06-05 15:20:00'),
       (3, 2, 2, '2023-06-07 11:00:00');
```