

Definição:
Conjunto de comandos SQL que permite definir, modificar e estruturar a estrutura de um banco de dados.

- Principais Comandos:**
- **CREATE:**
 - Cria objetos como tabelas, índices e visões.
 - **ALTER:**
 - Modifica a estrutura existente do banco de dados, como adicionar ou remover colunas.
 - **DROP:**
 - Remove objetos do banco de dados, como tabelas ou índices.

- Objetos DDL:**
- **Tabela:**
 - Estrutura fundamental para armazenar dados.
 - **Índice:**
 - Melhora a velocidade de recuperação de dados.
 - **Visão:**
 - Tabela virtual baseada em uma ou mais tabelas físicas.
 - **Procedimento Armazenado:**
 - Conjunto de instruções SQL que pode ser executado.

(DDL)

LINGUAGEM DE DEFINIÇÃO DE DADOS:



- Restrições (Constraints):**
- **PRIMARY KEY:**
 - Identifica exclusivamente cada registro em uma tabela.
 - **FOREIGN KEY:**
 - Garante integridade referencial entre duas tabelas.
 - **UNIQUE:**
 - Garante que os valores em uma coluna ou conjunto de colunas sejam exclusivos.
 - **CHECK:**
 - Garante que os valores em uma coluna atendam a uma condição específica.

- Transações DDL:**
- **COMMIT:**
 - Salva as alterações no banco de dados.
 - **ROLLBACK:**
 - Desfaz as alterações desde o último COMMIT.

- Exemplos Práticos:**
- **CREATE TABLE** Employee (...);
 - **ALTER TABLE** Customer **ADD COLUMN** Phone VARCHAR(15);
 - **DROP INDEX** idx_Name **ON** Person;

01

Definição:

Conjunto de comandos SQL usados para manipular dados em um banco de dados.

02

Principais Comandos

- **SELECT:**
 - Recupera dados de uma ou mais tabelas.
- **INSERT:**
 - Adiciona novos registros a uma tabela.
- **UPDATE:**
 - Modifica os valores em uma ou mais colunas de uma tabela.
- **DELETE:**
 - Remove registros de uma tabela.

03

Claúsulas select:

- **FROM:**
 - Especifica a tabela da qual os dados serão recuperados.
- **WHERE:**
 - Filtra os registros com base em uma condição.
- **ORDER BY:**
 - Classifica os resultados em uma ordem específica.
- **GROUP BY:**
 - Agrupa os resultados com base em uma ou mais colunas.
- **HAVING:**
 - Filtra grupos resultantes da cláusula GROUP BY.

04

Joins:

- **INNER JOIN:**
 - Retorna apenas registros que têm correspondência em ambas as tabelas.
- **LEFT (OUTER) JOIN:**
 - Retorna todos os registros da tabela à esquerda e os registros correspondentes da tabela à direita.
- **RIGHT (OUTER) JOIN:**
 - Retorna todos os registros da tabela à direita e os registros correspondentes da tabela à esquerda.
- **FULL (OUTER) JOIN:**
 - Retorna registros quando há uma correspondência em uma das tabelas.

05

Transações DML:

- **COMMIT:**
 - Salva as alterações feitas no banco de dados.
- **ROLLBACK:**
 - Desfaz as alterações desde o último COMMIT.
- **SAVEPOINT:**
 - Define um ponto no qual você pode rolar para trás.

06

Exemplos Práticos:

- `SELECT FirstName, LastName FROM Employees WHERE Department = 'IT';`
- `INSERT INTO Products (ProductName, Price) VALUES ('Laptop', 1200);`
- `UPDATE Customers SET City = 'New York' WHERE CustomerID = 101;`
- `DELETE FROM Orders WHERE OrderID = 1001;`

(DML)

Linguagem de Manipulação de Dados: