Nome: Renan De Souza Rodrigues

Ao longo do semestre foi visto os tipos de dados do MySQL como:

* INT: armazena números inteiros, sem vírgulas ou pontos
* FLOAT: armazena números não inteiros, com vírgulas ou pontos, decimais
* CHAR: armazena caracteres de um comprimento fixo
* VARCHAR: armazena caracteres de um comprimento variável
* TEXT: armazena texto de comprimento variável, utilizado para um texto longo
* DATE: armazena data sendo ANO/MÊS/DIA
* BOOLEAN: armazena TRUE ou FALSE

Também o que são (DDL) linguagem de definição de dados e (DML) linguagem de manipulação de dados e (DQL) uma subcategoria dos comandos DML, linguagem de Consulta.

Os comandos DDL que vimos foram:

* CREATE TABLE:
  + O comando “CREATE TABLE” é usado para criar uma nova tabela em um banco de dados. Ele define a estrutura da tabela, especificando os nomes das colunas, os tipos de dados que cada coluna pode conter e outras características, como chaves primárias e restrições de integridade. Isso prepara a tabela para armazenar dados de acordo com a estrutura definida.

CREATE TABLE Exemplo (

ID INT NOT NULL PRIMARY KEY AUTO\_INCREMENT,

Nome VARCHAR(50),

Idade INT

);

Vai criar uma tabela chamada exemplo com colunas de “ID,NOME e IDADE”

com ID sendo um valor “INT” e chave primária (PRIMARY KEY), não podendo ser nula pois (NOT NULL) e de auto incremento (AUTO\_INCREMENT)

* DROP TABLE e DROP DATABASES

DROP Modifica a estrutura de um objeto existente (adiciona ou remove colunas de uma tabela, por exemplo:

* + DROP TABLE nome Tabela;

Apagar uma tabela do banco

* + DROP DATABASE nome Banco;

Apaga todo o banco de dados

* ALTER TABLE Modifica a estrutura de um objeto existente (adiciona ou remove colunas de uma tabela, por exemplo:
  + ALTER TABLE [nomeTabela]

RENAME TO [novoNome];

Este comando é usado para renomear uma tabela existente. Substitua [nomeTabela] pelo nome atual da tabela e [novoNome] pelo novo nome desejado.

* + ALTER TABLE [nomeTabela]

MODIFY COLUMN [nomeColuna] [novosValores];

Este comando é usado para modificar a definição de uma coluna específica em uma tabela. Substitua [nomeTabela] pelo nome da tabela, [nomeColuna] pelo nome da coluna que está sendo modificada e [novosValores] pelos novos valores ou características da coluna.

* + ALTER TABLE [nomeTabela]

CHANGE COLUMN [nomeColuna] [novoNome] [novosValores];

Este comando realiza duas ações: altera o nome da coluna e modifica seus valores ou características. Substitua [nomeTabela] pelo nome da tabela, [nomeColuna] pelo nome atual da coluna, [novoNome] pelo novo nome desejado e [novosValores] pelos novos valores ou características da coluna.

* + ALTER TABLE [nomeTabela]

ADD COLUMN [nome] [valores]:

Este comando é usado para modificar a estrutura de uma tabela, adicionando uma nova coluna. [nomeTabela] é o nome da tabela à qual você está adicionando a coluna, [nome] é o nome da nova coluna, e [valores] representa o tipo de dados da nova coluna.

* + ALTER TABLE [nomeTabela]

DROP COLUMN [nomeColuna];

Este comando é utilizado para remover uma coluna específica de uma tabela. Substitua [nomeTabela] pelo nome da tabela e [nomeColuna] pelo nome da coluna que você deseja remover.

* + ALTER TABLE [nomeTabela]

ADD FOREIGN KEY ([nomeColuna]) REFERENCES [nomeTabela\_referencia] ([nomeColuna\_referencia]);

Este comando adiciona uma chave estrangeira à tabela. [nomeTabela] é o nome da tabela que terá a chave estrangeira, [nomeColuna] é a coluna na tabela atual que se tornará a chave estrangeira, e [nomeTabela\_referencia] e [nomeColuna\_referencia] são o nome da tabela e coluna referenciadas pela chave estrangeira.

Os comandos DML que vimos foram:

* SELECT:
  + O “SELECT” é utilizado para recuperar dados de um banco de dados, permitindo a consulta de dados específicos de uma tabela sobre condições específicas também se for necessário
  + SELECT coluna 1, coluna 2, FROM nome Tabela

WHERE condição;

Vai selecionar a coluna 1 e coluna 2 de uma tabela se a condição do “WHERE” for atendida, não é necessário colocar o “WHERE”, sem isso retornará tudo das colunas 1 e 2

* + SELECT \* FROM nome tabela

Vai retornar todas as colunas de uma tabela pois (\*) significa tudo

* + SELECT coluna 1, coluna 2

FROM tabela1

INNER JOIN tabela2 ON tabela1.coluna = tabela2.coluna;

O “INNER JOIN” é uma cláusula usada para combinar colunas de duas ou mais tabelas com base em uma condição específica. Ele cria um resultado que inclui apenas as colunas que atendem à condição de junção, eliminando as colunas que não correspondem.

* + SELECT Nome AS 'Nome do Cliente'

FROM Clientes;

O “AS”(Alias) é utilizado para apelidar uma coluna ou uma tabela

* + SELECT T.coluna 1, T.coluna 2 FROM Tabela T;

Assim Estou apelidando a Tabela de “T”

* + SELECT COUNT(\*) FROM [tabela];

COUT(\*) Para contar quantas linhas tem, MAX(nome\_coluna) para saber qual o maior valor, MIN(nome\_coluna) para saber qual o menor valor, SUM (nome coluna) Para somar, AVG(nome coluna) soma tudo e divide pela quantidade 'média'

* UPDATE:
  + O comando “UPDATE” é usado para modificar registros existentes em uma tabela de banco de dados. Ele permite atualizar os valores das colunas em linhas específicas com base em uma condição. O comando “UPDATE” é uma operação de alteração de dados que permite fazer modificações em registros já existentes em uma tabela, como atualizar valores de colunas.
  + UPDATE Exemplo

SET nome Coluna = ‘novo valor’,

WHERE Idade = ‘valor atual’;

Ele vai atualizar a tabela exemplo na coluna idade para um novo valor

e com o “WHERE” é uma condição geralmente se coloca a coluna de chave primária para identificar a coluna

* DELETE:
  + DELETE FROM nome Tabela

WHERE nome Coluna;

Apaga uma coluna de uma determinada tabela e sem o “WHERE” apagará

todas, demissão por justa causa

* INSERT
  + O comando “INSERT” é usado para adicionar novos registros a uma tabela em um banco de dados. Ele permite especificar os valores a serem inseridos nas colunas da tabela. Sua estrutura básica inclui o nome da tabela e os valores a serem inseridos. É usado para adicionar dados a uma tabela existente.

INSERT INTO Clientes (Nome, Email)

VALUES ('João Silva', 'joao@email.com');

INSERT INTO Clientes VALUES

('João Silva', 'joao@email.com');

Vimos também os comando:

* USE:
  + USE nome Banco de dados;

Seta o Banco de dados para a utilização

* DESC:
  + DESC nome tabela;

Descreve as informações de uma tabela como nome da colunas e seu tipo

de dado

* LIKE
  + SELECT \* FROM Tabela

WHERE Coluna LIKE 'A%';

* + [%A] Inicia com letra A
  + [A%] terminam com letra A
  + [%A%] tem letra A em alguma parte
* IF EXISTS e IF NOT EXISTS:
  + ALTER TABLE Nome\_Tabela

DROP NomeColuna IF EXISTS;

Se existir ele irá apagar

* + CREATE TABLE IF NOT EXISTS (valores);

Ele irá criar se não existir

* SHOW TABLES STATUS [nome\_Tabela:
  + mostra status da tabela
* SHOW DATABASES:
  + mostra os bancos disponíveis