

Readme.rmd

Sergio Pedro R Oliveira

2022-03-16

Objetivo

Estudo dirigido de MySQL.

Referência

Vídeo aulas “O curso completo de Banco de Dados e SQL, sem mistérios” - Udemey.

Módulo 2 - Teoria

Modelagem

1. Analise de requisitos

- Modelo das necessidades do Cliente, o que é do interesse do cliente e o que ele precisa no banco de dados.
- Processos a serem controlados pelo sistema.
- É uma fase de muita conversa e reunião com o cliente para investigar as regras do negocio.

2. processos de modelagem

- Fases 01 e 02 do projeto de banco de dados são feitos pelo administrador de dados:

i. Modelo conceitual

- Rascunho dos requisitos do projeto.
- Desenho conceitual.
- Modelo **entidades-relacionamentos**, define os relacionamentos entre os agentes.

ii. Modelo lógico

- Coloca os requisitos num programa de diagramas.
 - Cria **entidades**, posteriormente serão tabelas.
 - Cria **atributos**, posteriormente serão campos, colunas nas tabelas.
- Fase 03 do projeto de banco de dados é feita tanto pelo administrador de bancos de dados(DBA) quanto administrador de dados(AD):

iii. Modelo físico

- Criando banco de dados.
CREATE DATABASE *nome_do_banco_de_dados*;
- Conectando-se a um dos banco de dados do sistema.
USE *nome_do_banco_de_dados*;
- Criando tabela.
CREATE TABLE *nome_da_tabela*(
 coluna1 *tipo*(*tamanho*),
 coluna2 *tipo*(*tamanho*),

...
);

- Verificando os banco de dados no sistema.
SHOW DATABASES;

- Verificando as tabelas do banco de dados.
SHOW TABLES;

- *Descrevendo* como é a estrutura de uma tabela, verificando quais são as colunas.
DESC nome_da_tabela;

Tipagem de campos

A tipagem correta diminui o tempo de resposta, otimiza os processos.

1. Tipo caracteres

- **CHAR**

- Usado quando o numero de caracteres não varia, separa na memoria um espaço determinado para ser preenchido.

- **VARCHAR**

- Usado quando o numero de caracteres varia, dependendo da entrada adapta o espaço separado na memoria para caber os characters.

2. Tipo **ENUM**

- Conjunto de dados enumerados, ou seja, um conjunto fixo de dados.
- Limita dados em uma coluna, lista de opções.
- tipo característico do **MySQL**.

3. Tipo numerico

- **INT**

- Para numeros inteiros.
- Numero maximo de 11 digitos, para numeros maiores que isso usar **VARCHAR**.

- **FLOAT**

- Ponto flutuante, ou seja, numeros reais.
- Para numeros com casas decimais.
FLOAT(*total*, *virgula*)

4. Para fotos e documentos

- **BLOB**

5. Tipo textos

- **TEXT**

Módulo 3 - Comandos

Inserir registros na tabela - INSERT

- Existem diversas formas de inserir dados na tabela, entre eles temos:
 - Omitindo colunas/campos.
 - * Determina apenas a tabela, que puxa todos os campos para serem preenchidos, na ordem que aparece na tabela.
 - * Sintaxe:
INSERT INTO *nome_da_tabela*
VALUES (*valor_na_coluna_1*, *valor_na_coluna_2*,...);
 - Colocando as colunas.
 - * Especifica a ordem das entradas e os campos a serem preenchidos.
 - * Sintaxe:
INSERT INTO *nome_da_tabela*(*coluna_3*, *coluna_1*, *coluna_2*,...) **VALUES** (*valor_na_coluna_3*, *valor_na_coluna_1*,...);
 - INSERT COMPACTO, somente **MySQL**.
 - * Insere diversos registros de uma vez, na ordem que aparecem na tabela.
 - * Sintaxe:
INSERT INTO *nome_da_tabela*
VALUES (*valor_na_coluna_1_registro1*, *valor_na_coluna_2_registro1*,...),
(*valor_na_coluna_1_registro2*, *valor_na_coluna_2_registro2*,...),
...;

Consultando campos na tabela - SELECT

- O comando **SELECT** serve para projeção, seleção e junção.
- O comando **SELECT** seleciona os campos/colunas a serem mostrados.
- Projeta/constroi o que deve ser mostrado, não apenas os dados da tabela.
 - Exemplo de código:
SELECT 'SERGIO PEDRO' AS MEU_NOME;
 - Sintaxe:
SELECT 'algo a mostrar' AS alias_da_coluna;
- Seleciona o que deve ser mostrado da tabela.
 - Exemplo de código:
SELECT NOME, SEXO, EMAIL, ENDERECO FROM CLIENTE;
 - Sintaxe:
SELECT coluna_1, coluna_6, coluna_3, coluna_5 FROM tabela;
 - Seleciona todas as colunas da tabela:
SELECT * FROM tabela;
Obs.: '*', Diminui a eficiência da pesquisa na tabela.

Consultando registros na tabela - WHERE

- O comando **WHERE** serve para filtrar os registros/linhas da tabela, antes de mostrar.
 - Sintaxe:
SELECT *coluna_1*, *coluna_2* **FROM** *tabela*
WHERE *coluna_1* = *critério*;
- O comando **WHERE** não precisa ter haver com a seleção **SELECT**.
 - Sintaxe:
SELECT *coluna_1*, *coluna_3* **FROM** *tabela*
WHERE *coluna_2* = *critério*;
- Para trabalhar com *strings*, é útil usar o comando **LIKE** e os *caracteres coringas*.
 - Caracteres coringas:
 - * *%*
Qualquer coisa.
 - * *_*
Um único caracter.
 - Sintaxe:
SELECT *coluna_1*, *coluna_3* **FROM** *tabela*
WHERE *coluna_2* **LIKE** '*string_procurada*';
Obs.: Os caracteres coringas podem entrar em qualquer lugar da string para complementar o texto a procurar.

Detalhes

- **Comentarios** no **MySQL**, diferente do **SQL** onde comentarios são `'/**/'`, no **MySQL** é `'#'`.
- O que são e o que fazem os administradores:
 - Administrador de dados(AD):

O Administrador de Dados (AD) tem o objetivo de gerenciar o Modelo de Dados Corporativo, contribuindo para assegurar a qualidade das informações, a integração dos sistemas, a retenção e a disseminação do conhecimento dos negócios.

Cabe a ele, guiado por certos princípios e através de atividades de planejamento, organização e controle dos dados corporativos, gerenciar os dados como recursos de uso comum da organização, promovendo-lhes os valores de autenticidade, autoridade, precisão, acessibilidade, seguridade e inteligibilidade.

Tem como função o planejamento central, a documentação e o gerenciamento dos dados a partir da perspectiva de seus significados e valores para a organização como um todo.
 - Administrador de banco de dados (DBA):

O DBA (database administrator), sigla em inglês para Administrador de Banco de Dados, é um profissional da área de tecnologia responsável pela criação, instalação, monitoramento, reparos e análise de estruturas de um banco de dados.

O banco de dados fica sob análise periódica do DBA, que trabalha para que não haja sobrecargas do sistema e que as informações inseridas tenham destino correto nos servidores. Outras funções também importantes são analisar o espaço em disco, buscar melhorias para os sistemas e realizar backups.
- Acesso ao **MySQL** pelo terminal é necessário usar o comando:
`mysql -u root -p`
 - Depois colocar a senha.
- Ao final dos comandos do **SQL** e do **MySQL**, usar o `;`(delimitador), ele informa que o comando acabou e deve ser executado.

Andamento dos Estudos

Assunto em andamento:

Atualmente estou estudando Seção 3 - Comandos.