

Josivaldo de França



TDS – 2019.2

Banco de Dados



ESCOLA TÉCNICA ESTADUAL

PALMARES - PE

Informática





MySQL BÁSICO

BANCO DE DADOS

- O que é MySQL
- Criar bando de dados
- Nomenclatura de banco de dados
- Tipos de dados SQL
- Atributos SQL
- Criar Tabela
- Inserir, Alterar, Recuperar e Excluir registros

MySQL

- Utiliza a linguagem SQL comum a maioria dos bancos de dados relacionais.
- Desenvolvido pela *SUN*, o software é livre com código aberto.
- As informações contidas neste material referisse ao MySQL 5.1.

Como criar um banco de dados no MySQL

Esta é uma das atividades básicas de um administrador de banco de dados e tudo começa com esta tarefa – depois da instalação, claro.

Um banco de dados é um container, onde se armazenam informações dos mais variados tipos – contatos, vendedores, produtos, clientes etc. Qualquer tipo de informação que te ocorrer, pode ser organizada em um banco de dados.

Organização é a palavra. Por que os dados não são “jogados ao léu”, neste container. Um banco de dados é uma coleção de objetos que são usados para armazenar e manipular dados, tais como tabelas, views, gatilhos (triggers), stored procedures etc.



CRIAR BANDO DE DADOS

- Banco de dados também é conhecido pelo nome: **Schema**
- O sistema do MySQL pode suportar vários bancos de dados diferentes.
- Geralmente será criado um banco de dados para cada aplicação.
- Para criar um banco de dados no MySQL será utilizado a palavra-chave **CREATE DATABASE**.
- Sintaxe

```
CREATE DATABASE nome_do_banco_de_dados;
```

- Sintaxe alternativa

```
CREATE SCHEMA nome_do_banco_de_dados;
```

NOMENCLATURA

- Pode inserir no comando letras maiúsculas, mas o sistema interpreta e cria o banco com letras minúsculas.

Válido	Inválido
nome_banco	nome banco
3nome_banco	nome-banco
banco_pái	nome%banco
NomeBanco	

- Como boas práticas é preferível definir o nome todo em caixa baixa, sem acentos e se for necessário ter palavras compostas, separá-las com underline.

```
CREATE DATABASE nome_do_banco_de_dados;
```

O comando **CREATE DATABASE** faz o que o seu significado, em inglês sugere: cria um banco de dados;

Embora eu use letras maiúsculas para representar os comandos no MySQL, isto não é necessário. O interpretador do cliente MySQL não é sensível à caixa das letras;

O nome do seu banco de dados deve refletir com clareza o seu conteúdo.

Ainda sobre a nomenclatura dos bancos de dados, cabe ressaltar que aqui o MySQL é sensível às caixas de texto, sim. Ou seja, concessionaria é uma coisa, Concessionaria é outra;

Por último, **NÃO ESQUEÇA** de sempre concluir os comandos MySQL com um **ponto-e-vírgula**. Sem isto, o interpretador não entende que você terminou de dar um comando e vai continuar a espera de “algo mais”.

APRENDA FAZENDO

- Inicialize o Xampp;
- Start mysql;
- Acesse o mysql pelo console do Windows;
- C:\> cd xampp\mysql\bin
- C:\xampp\mysql\bin> mysql -u root -p -h 127.0.0.1
- Enter password:

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 53  
Server version: 10.1.31-MariaDB mariadb.org binary distribution  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

- Se tiver problema de senha deve-se editar o arquivo

Após a instalação do MySQL (se você não fizer nada), o usuário padrão será o “root” e a senha será vazia... E isso não é nada seguro, mesmo quando se desenvolve localmente... Hoje vou ensinar como mudar essa senha tanto no Windows quanto no Linux.

Mudando a senha do root no Windows

Iniciar > Executar, digite CMD e aperte enter

Navegue até o diretório bin dentro da pasta onde o MySQL está instalado, pode ser C:\mysql\bin ou se você estiver usando o XAMPP será C:\xampp\mysql\bin

Digite os seguintes comandos:

```
mysql -u root mysql
```

```
SET PASSWORD FOR root@localhost=PASSWORD('NOVASENHA');
```

APRENDA FAZENDO

- Crie um banco de dados com o nome: Carros

```
CREATE DATABASE carros;
```

TIPOS DE DADOS SQL

- Números Inteiros

- N = comprimento máximo de exibição

Tipo	Faixa	Descrição
TINYINT(N)	-128...127 ou 0...255	Inteiro muito pequeno – 1 byte
BIT		Sinônimo de TINYINT
BOOL		Sinônimo de TINYINT
SMALLINT(N)	-32768... 32767 ou 0... 65535	Inteiro pequeno – 2 bytes
MEDIUMINT(N)	-8388608...8388607 ou 0...16777215	Inteiro de tamanho médio – 3 bytes
INT(N)	-2147483648... 2147483647 ou 0... 4294967295	Inteiro regular – 4 bytes
INTEGER(N)		Sinônimo de INT
BIGINT(N)	-9223372036854775808... 9223372036854775807 ou 0... 18446744073709551615	Inteiro grande – 8 bytes

TIPOS DE DADOS SQL

- Números Flutuantes

Tipo	Faixa	Descrição
FLOAT		Utilizado para números de ponto flutuante duplo ou simples

TIPOS DE DADOS SQL

- Strings

Tipo	Faixa	Descrição
VARCHAR(N)	1 a 255 caracteres	
TEXT	0 a 65.535 caracteres	Campo TEXT de tamanho normal

ATRIBUTOS SQL

- Not null
- Auto-increment
- Primary key
- constraint

CRIAR TABELA

- Após criar um banco de dados, é necessário criar tabelas para atender certa demanda.
- Não é possível criar mais de uma tabela com mesmo nome em um banco de dados.
- Primeiro entre no BD com o comando: **USE** **nome_do_banco**;
- Para criar uma tabela em um banco de dados no MySQL será utilizado a palavra-chave **CREATE TABLE**.
- Sintaxe

```
CREATE TABLE nome_tabela  
(  
    <nome_da_coluna1> <tipo_da_coluna1> [<atributos_da_coluna1>],  
    ...  
    <nome_da_coluna> <tipo_da_coluna> [<atributos_da_coluna>]  
);
```


APRENDA FAZENDO

- Crie a tabela carro no banco de dados carros

```
CREATE TABLE carro
(
  id_carro int not null auto_increment,
  modelo varchar(50),
  cor varchar(15),
  ano smallint(4),

  primary key (id_carro)
)
```

- Para ver as suas tabelas digite: **Show Tables;**
- Para ver uma descrição digite: **desc nome_da_tabela;**

INSERIR DADOS

- Após criar uma tabela no banco de dados, é possível inserir um ou vários dados.
- Para inserir um registro na tabela serão utilizadas as palavras-chave **INSERT INTO** e **VALUES**.
- As strings devem ser incluídas em pares de aspas simples ou dupla.
- Números Inteiros ou Flutuantes não necessitam de aspas.
- Sintaxe

```
INSERT INTO <nome_tabela>  
( <campo1> , ... , <campoN> )  
VALUES  
( <valorCampo1> , ... , < valorCampoN> );
```

- Sintaxe alternativa

```
INSERT INTO <nome_tabela> VALUES ( <valorCampo1> , ... , < valorCampoN> );
```

APRENDA FAZENDO

- Inserir os seguintes registros na tabela carro.

```
INSERT INTO carro ( modelo , cor , ano ) VALUES ( 'Corsa' , 'Vermelho' , 2003 );  
INSERT INTO carro ( modelo , cor , ano ) VALUES ( 'Fusca' , 'Branco' , 1966 );  
INSERT INTO carro ( modelo , cor , ano ) VALUES ( 'Palio' , 'Prata' , 2009 );  
INSERT INTO carro ( modelo , cor , ano ) VALUES ( 'Gol' , 'Branco' , 2008 );
```

- Para ver todos os dados use o comando : **select *from carro;**

RECUPERAR DADOS ESPECÍFICOS

- Para recuperar registros específicos em uma tabela serão utilizadas as palavras-chave **WHERE**, **OR** e **AND**.

- Sintaxe

```
SELECT * FROM nome_tabela WHERE <condição>
```

- Sintaxe utilizando **AND**

- Todas as condições envolvidas devem ser verdadeiras.

```
SELECT * FROM nome_tabela WHERE <condição> AND <condição>
```

- Sintaxe utilizando **OR**

- Pelo menos uma condição envolvida deve ser verdadeira.

```
SELECT * FROM nome_tabela WHERE <condição> OR <condição>
```

OPERADORES DE COMPARAÇÃO PARA WHERE

Nome	Operador	Exemplo	Descrição
Igualdade	=	valor_coluna = 5	Verificar se os dois valores são iguais.
Maior que	>	valor_coluna > 30	Verificar se o valor da esquerda é maior que o da direita.
Menor que	<	valor_coluna < 45	Verificar se o valor da esquerda é menor que o da direita.
Maior ou igual	>=	valor_coluna >= 12	Verificar se o valor da esquerda é maior ou igual ao da direita.
Menor ou igual	<=	valor_coluna <= 94	Verificar se o valor da esquerda é menor ou igual ao da direita.
Desigualdade	!= ou <>	valor_coluna != 2009	Verificar se os dois valores são diferentes.

APRENDA FAZENDO

- Recuperar apenas os registros de carro brancos.

```
SELECT * FROM carro WHERE cor = 'Branco';
```

- Recuperar apenas os registros de carro brancos do ano 1966.

```
SELECT * FROM carro WHERE cor = 'Branco' AND ano = 1966;
```

- Recuperar apenas os registros de carro brancos ou pratas.

```
SELECT * FROM carro WHERE cor = 'Branco' OR cor = 'Prata';
```

- Recuperar apenas carros produzidos a partir do ano 2000.

```
SELECT * FROM carro WHERE ano >= 2000;
```

- Recuperar apenas carros que não foram produzidos em 2009.

```
SELECT * FROM carro WHERE ano != 2009;
```

ALTERAR DADOS

- Após inserir dados na tabela, podemos alterar os mesmos, caso necessário.
- Para altera registros em uma tabela serão utilizadas as palavras-chave **UPDATE** e **SET**.
- Sintaxe

```
UPDATE <Nome da Tabela>  
SET <Coluna 1> = 'valor Coluna 1' , <Coluna 1> = 'valor Coluna 1'  
WHERE <Condição>;
```

- Sintaxe alternativa

```
UPDATE <Nome da Tabela>  
SET <Coluna 1> = 'valor Coluna 1' , <Coluna 1> = 'valor Coluna 1'
```

APRENDA FAZENDO

- Alterar o modelo do Gol para Gol 2008.

```
UPDATE carro SET modelo = 'Gol 2008' WHERE id_carro = 4;
```

- Alterar a cor de todos os carros brancos para branco gelo.

```
UPDATE carro SET cor = 'Branco Gelo' WHERE cor = 'branco';
```

- Alterar a cor de branco gelo para branco neve nos carros fabricados a partir do ano 2000.

```
UPDATE carro SET cor = 'Branco Neve'  
WHERE cor = 'branco gelo' AND ano >= 2000;
```

- Alterar a cor de todos os carros para Cinza.

```
UPDATE carro SET cor = 'Cinza';
```


EXCLUIR REGISTRO

- Após inserir dados na tabela, podemos excluí-los se for necessário.
- Para excluir registro(s) em uma tabela será utilizada a palavra-chave **DELETE**.
- Sintaxe

```
DELETE FROM <Nome da Tabela> WHERE <Condição>
```

- Sintaxe alternativa

```
DELETE FROM <Nome da Tabela>
```

APRENDA FAZENDO

- Excluir apenas o carro que tenha o código 4.

```
DELETE FROM carro WHERE id_carro = 4;
```

- Excluir todos os registros de carro.

```
DELETE FROM carro
```

APRENDA FAZENDO

- Excluir uma tabela

```
DROP TABLE nome_da_tabela;
```

- Excluir um banco de dados.

```
DROP DATABASE nome_do_banco;
```

REFERÊNCIA BIBLIOGRÁFICAS

- <http://www.mysql.com> [acessado em 1 de maio de 2009 as 21Hrs]
- <http://dev.mysql.com> [acessado em 1 de maio de 2009 as 20Hrs]
- <http://www.w3schools.com/sql> [acessado em 1 de maio de 2009 as 21Hrs]

Bibliografia

<http://www.dicasdeprogramacao.com.br/como-criar-um-projeto-de-banco-de-dados/>