

MYSQL O MySQL é uma implementação de SGDB que utiliza o padrão SQL. É o mais popular entre os bancos de dados com código fonte aberto. Largamente utilizado em aplicações Internet, conta com mais de 5.000.000 de instalações.

Apresenta diversos tipos de tecnologia na hora de criar e manter tabelas, a saber (cláusula type=xxx no create table):

BDB Berkeley DB: tabelas de transação segura com bloqueio de página. Mais moderno e bom para recuperar dados com rapidez.

heap Dados armazenados apenas na memória, com registro de tamanho fixo. Busca apenas por igual e diferente. Chave do tipo inteiro e UNIQUE. Não aceita BLOB e TEXT.

ISAM Era o mecanismo de armazenamento original. Obsoleto-se.

InnoDB Tabelas com transações seguras, com bloqueio de linha. Gasta mais espaço que o MyISAM. Aceita integridade referencial por software.

Merge Uma coleção de tabelas MyISAM usadas como uma tabela.

MyISAM O mecanismo de armazenamento que substituiu o ISAM. As meta-tabelas do MySQL estão aqui. Permite indexar BLOB e TEXT. É o método mais rápido, mas não implementa roll-back e COMMIT.

Um passeio pelo MySQL Para criar um BD chama-se o utilitário *mysql*. Ao chamá-lo deve-se passar o nome do computador onde o MySQL está rodando (se for neste mesmo computador, esta cláusula pode ser omitida), um username (opção -u) e uma senha de acesso (opção -p) que foram estabelecidos pelo administrador durante a geração do SGDB. O comando de criação é CREATE DATABASE. Veja-se então um exemplo completo:

```
mysql -h nome-servidor -u nome-usuário -p  
(...segue-se a senha)  
> CREATE DATABASE laranja;  
> USE laranja;
```

Para criar uma tabela o comando é CREATE TABLE, cuja sintaxe é
CREATE TABLE nome-tabela, { nome-campo, tipo, [NULL | NOT NULL] [DEFAULT valor], ... };
Os tipos de campo serão vistos a seguir. Para visualizar as tabelas criadas no banco de dados o comando é SHOW TABLES; e para visualizar a estrutura de uma determinada tabela o comando é DESC nome-tabela;

Para inserir registros dentro de uma tabela o comando é INSERT INTO. Se os conteúdos a inserir estão na mesma ordem que os campos criados não é necessário fornecer seus nomes. Senão é necessário fornecê-los.

Para alterar um ou mais registros de uma tabela o comando é o UPDATE, cujo formato é UPDATE nome-tabela SET campo1=valor1 [...] [WHERE condição]; Se a cláusula WHERE está ausente, todos os registros são alterados.

Para alterar a estrutura de uma tabela o comando é ALTER TABLE.

Para excluir registros de uma tabela o comando é DELETE FROM, cujo formato é DELETE FROM nome-tabela [WHERE condição];

Para eliminar uma tabela o comando é DROP TABLE e para eliminar um banco de dados o comando é DROP DATABASE.

Para consultar o banco de dados o comando é SELECT, que será estudado adiante.

Usuários e privilégios O usuário root geralmente é utilizado para propósitos de administração (e só – razões de segurança). Um privilégio é o direito de realizar uma ação particular, num objeto particular associado a um usuário particular.

O princípio do menor privilégio diz que um usuário (ou um processo) deve ter o nível mais baixo de privilégio exigido para realizar a tarefa a ele atribuída.

os comandos para manusear privilégios são GRANT e REVOKE. Eles se aplicam a 4 níveis: global, banco de dados, tabela e coluna. O comando GRANT tem a seguinte sintaxe:

```
GRANT privilégios [coluna ou colunas]  
ON item TO [usuário]  
[IDENTIFIED BY senha] [WITH GRANT OPTION]
```

Privilégios indicam o que usuário pode fazer e são: SELECT, INSERT, UPDATE, DELETE, INDEX, ALTER, CREATE e DROP. Há ainda os seguintes geralmente reservados aos administradores: RELOAD, SHUTDOWN, PROCESS e FILE. Há ainda ALL (concede TUDO) e USAGE (não concede nada). Item pode ser *.* (global), namedb.* (todo o db), namedb.nometabela (tabela). Usuário pode ser fulano@dominio e a cláusula with grant option dá ao usuário o direito dele criar novos privilégios a outrem.

Todas estas informações são guardadas no banco de dados mysql que é um bd comum. Para este arquivo muito cuidado ao aplicar privilégios. As suas tabelas são mysql.user, mysql.db, mysql.tables_priv e mysql.columns_priv (os 4 níveis de privilégio). Finalmente, o comando REVOKE revoga os privilégios e tem sintaxe muito parecida ao GRANT.

Para desligar sua sessão no mysql, o comando é quit. Após entrar no MySQL, o comando para indicar qual bd quer trabalhar é USE nome-db; Opcionalmente isto pode ser dito no logon, escrevendo a cláusula -D nome-db.

Criando uma tabela - exemplo Seja o comando

```
create table ALUNO ( ID-ALUNO int unsigned not  
null auto_increment primary key, NOME char(40)  
not null, CURSO char(5) not null);  
create table CURSO (ID-CURSO char(5) not null  
primary key, NOME-CURSO char(30), ...);
```

NOT NULL significa que o campo não pode ficar vazio. AUTO_INCREMENT pode ser usado em colunas inteiras e indica que cada registro receberá um número novo. Esta coluna deve ser indexada. As entradas de PRIMARY KEY devem ser únicas. Escrever PRIMARY KEY ao final da lista de colunas é usado quando a chave é composta de 2 ou mais colunas.

Os tipos de campos podem ser (lista parcial) INT, BIGINT, FLOAT, DOUBLE, DECIMAL(M,D), DATE, TIME, DATETIME, CHAR, VARCHAR(M), BLOB, TEXT,...

Após criar uma tabela a mesma poderá ser vista com os comandos

```
show databases;  
use nome-db;  
show tables;  
describe nome-tabela;
```

Inserindo dados, um exemplo

```
insert [into] ALUNO (CURSO, NOME)  
values ("AA001", "Jose da Silva");
```

Várias linhas podem ser incluídas em um único comando. Cada conjunto de dados deve estar entre seus próprios parênteses.

Recuperando dados O comando básico é o SELECT que tem o formato

```
SELECT items FROM tabelas [WHERE condição]  
[GROUP BY tipo_grupo] [HAVING definição_where]  
[ORDER BY tipo_pedido] [LIMIT critério];
```

a lista de items pode ser substituída por *, o que significa tudo. Na condição, o SQL aceita: =, >, <, >=, <=, != ou <>, IS NOT NULL, IS NULL, BETWEEN, IN, NOT IN, LIKE, NOT LIKE, REGEXP. Em LIKE, os caracteres do string são % curinga de tamanho qualquer e _ para coringa de 1

caracter. As REGEXP são expressões POSIX. Diversos critérios podem ser unidos com AND e OR.

A pesquisa pode ser feita em mais de uma tabela, veja o exemplo

```
select cliente.nome_cliente, notafiscal.valor,  
notafiscal.data  
from cliente, notafiscal,  
where cliente.cliente.id_cliente = 33 and  
cliente.id_cliente = notafiscal.id_cliente;
```

Note a especificação de ponto. Parece ser uma boa idéia utilizá-la sempre, embora o Mysql só a exija quando houver ambiguidade (duas colunas com o mesmo nome em tabelas diferentes).

A cláusula ORDER recupera as linhas na ordem especificada pela(s) coluna (s) citada (s). A ordem ascendente é ASC e a descendente é DESC.

Há algumas funções que podem ser solicitadas no comando SELECT e são: AVG(coluna), COUNT(DISTINCT)(coluna), MIN(coluna), MAX(coluna), STD(coluna), SUM(coluna) entre outros.

Quando estas funções são usadas junto com a cláusula GROUP BY, as funções têm seu comportamento alterado. Ao invés de efetuar a função em toda a tabela ela se aplica apenas ao grupo. Há ainda a cláusula HAVING que é um WHERE que se aplica aos grupos. Veja o exemplo: seja mostrar os clientes cujo valor medio de notas fiscais seja maior que R\$100,00.

```
select notafiscal.id_cliente,  
avg(notafiscal.valor)  
from notafiscal group by notafiscal.cliente  
having avg(notafiscal.valor) > 50;
```

Finalmente a cláusula LIMIT indica quais saídas mostrar. Tem 2 parâmetros: a partir de qual linha mostrar e quantas devem ser mostradas. (É usado em servidores web, para paginar um determinado arquivo, por exemplo).

O comando para criar um índice por uma coluna é ALTER TABLE tabela ADD INDEX (coluna); O comando para defragmentar uma tabela é OPTIMIZE TABLE tabela;

Alguns utilitários são bastante úteis, a saber: mysisamchk = analisa e depura tabelas MyISAM. mysql = interage com o servidor, aceitando comandos SQL.

mysqladmin = tarefas administrativas do administrador.

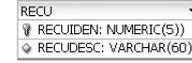
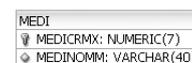
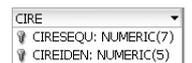
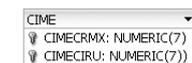
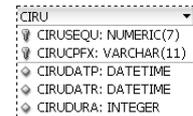
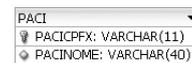
mysqldump = extrai a estrutura e os dados de um banco de dados.

mysqlimport = carrega os dados a partir de arquivos de texto.

mysqlshow = mostra informações sobre o bd.

🔗 Para você fazer

1. Crie um banco de dados de nome CIRURGIA



2. Crie 5 linhas em cada tabela
3. Recupere a informação completa de uma cirurgia

