Bases de Dados - Teórico/Práticas

(2013/09/27 (v12))

Pedro Quaresma

Departamento de Matemática
Universidade de Coimbra

2013/2014

MySQL — Instalação

- **Linux (Debian)**
  - mysql-client — command-line client
  - mysql-server — server
  - mysql-workbench — GUI client
  - + dependências
  - php5-mysql — MySQL module for php5
- **MS-Windows**
  - `http://www.easyphp.org/` — WAMP
- **Para as todas as plataformas**
  - `http://dev.mysql.com/downloads/`
  - MySQL Community Server
  - MySQL Workbench
  - MySQL Utilities

MySQL — Servidor/Grupos de Trabalho

Cada grupo tem uma conta em "`rena4.mat.uc.pt`", `bdN`, e tem permissões para criar bases de dados cujos nomes têm de (obrigatoriamente) ter como prefixo o nome do grupo `bdN...`.

```
pedro@hilbert: mysql -u bd01 -p -h rena4.mat.uc.pt
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 5337
Server version: 5.0.45 SUSE MySQL RPM

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

```
show databases;

+------------+
| Database |
+------------+
| bd01Proj1 |
+------------+

1 row in set (0.00 sec)
```


MySQL — Meta-comandos

mysql — interpretador de linha de comando (`man mysql`)

```
$ mysql -p -u <nome_utilizador> -h <nome_de_basedados>
```

a opção `-p` é de "password".

- `use/connect <nome>` — estabelece a ligação a uma base de dados.
- `show databases/tables` — mostra as bases de dados/tabelas disponíveis.
- `describe <nome_tabela>` — mostra a estrutura interna de uma tabela.
- `source <nome_de_ficheiro>` — carrega um ficheiro e (tenta) executa os comandos SQL nele contidos.
- `quit` — sai do interpretador.

Um comando útil é o comando `mysqldump` o qual permite fazer cópias de segurança das bases de dados, por exemplo:

```
$ mysqldump -p --databases situacao3a > situacao3a.sql
```

cria o ficheiro "situacao3a.sql" no qual estão todos os comandos SQL necessários para (re)criar a base de dados "situacao3a".

Linguagem de Definição de Dados (DDL)

Linguagem de Definição de Dados (DDL) — Linguagem para especificar a informação acerca de cada relação, incluindo:

- O esquema de cada relação.
- O domínio de valores associados com cada atributo.
- Restrições de integridade
- O conjunto de índices a manter para cada relação.
- Informação de segurança e autorização para cada relação.
- As estruturas de armazenamento físico em disco de cada relação.

Criação de Bases Dados/Tabelas (Relações)

- Instrução `CREATE DATABASE <nome_da_bd>`, cria uma base de dados com o nome escolhido.
- Instrução:
  `CREATE TABLE <nome_da_tabela> (`
  `<campo_1 tipo_1>,`
  `...,`
  `<campo_n tipo_n>)`
  cria a tabela (relação).

Exemplo:

```
CREATE TABLE branch
(branch_name char(15) NOT NULL,
branch_city char(30),
assets integer)
```


Tipos em SQL

- `char(n)`. Cadeia de caracteres de comprimento fixo n.
- `varchar(n)`. Cadeia de caracteres de comprimento variável, com o máximo n especificado pelo utilizador.
- `int`. inteiro (um subconjunto finito dos inteiros, dependente da máquina).
- `smallint`. Inteiro pequeno (um subconjunto do tipo int).
- `numeric(p,d)`. Número de vírgula fixa, com precisão de p dígitos e com n casas decimais.
- `real, double precision`. Números de vírgula flutuante, com precisão dependente da máquina.
- `float(n)`. Número de vírgula flutuante, com um mínimo de precisão de n dígitos.
- Os valores nulos são permitidos em todos os tipos de dados. A declaração de um atributo como `not null` proíbe os valores nulos para esse atributo.

Tipos em SQL

- date. datas, contendo um ano com (4 dígitos), mês e dia
  - E.g. date 2001-7-27
- time. Tempo (diário), em horas, minutos e segundos.
  - E.g. time 09:00:30 time 09:00:30.75
- timestamp: data mais hora
  - E.g. timestamp 2001-7-27 09:00:30.75
- Interval: período de tempo
  - E.g. Interval 1 day
  - A subtracção de dois valores de date/time/timestamp devolve um intervalo
  - Os valores de intervalos podem ser adicionados a valores de date/time/timestamp
- Pode-se extrair campos do valor date/time/timestamp
  - E.g. extract (year from SYSDATE)

Alteração de Tabelas (Relações)

Instrução `ALTER TABLE` — a instrução `alter table` é utilizada para modificar o esquema, ou as restrições sobre relações já existente.

- Para adicionar novos atributos:

  ```
  ALTER TABLE r ADD A D
  ```

  Todos os tuplos existentes ficam com `null` no novo atributo.
- Para eliminar um atributo:

  ```
  ALTER TABLE r DROP A
  ```

  em que `A` é o nome de um atributo na relação `r`.
- Para modificar um atributo:

  ```
  ALTER TABLE r MODIFY A D
  ```

em que `A` é o nome do atributo a adicionar à relação `r` e `D` o domínio de `A`.

Destruição de Bases Dados/Tabelas (Relações)

- Instrução `DROP DATABASE` <nome_da_bd>, apaga a base de dados com o nome escolhido.
- Instrução `DROP TABLE` <nome_da_tabela>, apaga a tabela com o nome escolhido (da base de dados corrente).

A especificação completa destes comandos em MySQL pode ser obtida no manual de referência on-line (`http://dev.mysql.com/doc/`).

Bases de Dados e Páginas da Rede — Porquê?

Bases de Dados Informação:

- Banca: todas as transacções e movimentos;
- Companhias aéreas: reservas, horários;
- Universidades: inscrições, notas;
- Vendas: clientes, produtos, compras;
- Sistema de gestão de artigos numa conferência/revista;
- lojas "on-line";
- ...

Páginas da Rede Disseminação de Informação:

- Acesso Global (geográfico);
- Acesso Generalizado:
  - rede — larga cobertura a nível mundial;
  - navegadores — total.
- Interface uniforme e fácil de usar/programar.

Bases de Dados e Páginas da Rede — Como?

Linguagem de Programação

- Ligação à base de dados:
  - $\longrightarrow$ SQL;
  - $\longleftarrow$ estruturas de dados de transferência.
- Ligação à Página de Rede:
  - $\longrightarrow$ escrita de ficheiros HTML;
  - $\longleftarrow$ formulários HTML + estruturas de dados de transferência.

entre outras aproximações possíveis.