

Bases de Dados 2005/2006

Aula 2

Sumário

0. A base de dados Projecto
 1. SQL Query Analyzer – executar comandos (e scripts).
 2. SQL – Structured Query Language.
-

0. A base de dados Projecto

Na máquina *ferro.win.di.ubi.pt* encontra-se a base de dados Projecto. Estabeleça uma ligação com o servidor e veja alguns dos elementos desta base de dados.

A base de dados Projecto foi criada a partir da execução de scripts usando a ferramenta *SQL Query Analyzer*.

Os scripts são muito úteis quando se pretende criar a base de dados, os seus objectos (tabelas, restrições, índices, procedimentos armazenados, triggers, etc.) e carregar os dados iniciais (de arranque) de uma forma (quase) automática. Esta situação pode surgir, por exemplo, quando se pretende instalar uma solução informática composta por uma base de dados e uma (ou mais) aplicação(ões) que a usa(m).

1. SQL Query Analyzer – executar comandos (e scripts)

Pretende-se que estabeleça uma ligação ao servidor de base de dados local e crie uma base de dados semelhante à base de dados Projecto. Para isso deve:

1. Aceder à página www.di.ubi.pt/~pprata/bd_04_05.htm e descarregar os scripts: [criar a base de dados](#); [criar as tabelas e estabelecer relacionamentos e restrições](#); [Inserir Dados](#); [criar views](#); [criar procedimentos armazenados](#) e [criar triggers](#);
3. Adaptar o scrip [criar a base de dados](#) de modo a usar para nome da base de dados o seu número de aluno. Adaptar também de modo adequado a directoria onde vai colocar os ficheiros de dados (data) e de log.
4. Alterar a instrução *Use database*, nos restantes scripts, para *Use n_aluno*.
5. Executar os scripts.
6. Verificar que as tabelas têm dados.

2. SQL – Structured Query Language

A SQL (*Structured Query Language*) é uma linguagem padrão para a interrogação de bases de dados relacionais.

A SQL é uma linguagem declarativa (não procedimental). Uma pesquisa (*query*) em SQL especifica o que (*what*) deve ser procurado mas não como (*how*) deve ser procurado. A SQL não é uma linguagem de programação completa; por exemplo, não tem instruções de controlo nem de iteração.

A SQL consiste numa linguagem de definição de dados (DDL – *Data Definition Language*) e numa linguagem de manipulação de dados (DML – *Data Manipulation Language*) (e numa linguagem de controlo de acesso a dados (DCL – *Data Control Language*)).

DDL – para criar/alterar/eliminar objectos (tabelas, restrições, índices, etc.) da base de dados.

DML – para manipular dados (inserir dados nas tabelas, actualizar dados existentes e eliminar dados das tabelas) e para interrogar (baseado na álgebra relacional) a base de dados.

DCL – para tratar os aspectos relacionados com a autorização de acesso aos dados. Permite que o utilizador controle quem tem acesso para visualizar ou manipular dados dentro da base de dados (comandos: GRANT e REVOKE)).

2.1 SQL – DML – Interrogação

A documentação do *SQL Server 2000* fornece-nos a seguinte estrutura para o comando Select:

Transact-SQL Reference

SELECT

Retrieves rows from the database and allows the selection of one or many rows or columns from one or many tables. The full syntax of the SELECT statement is complex, but the main clauses can be summarized as:

```
SELECT [ ALL | DISTINCT ] select_list
[ INTO new_table ]
FROM table_source
[ WHERE search_condition ]
[ GROUP BY group_by_expression ]
[ HAVING search_condition ]
[ ORDER BY order_expression [ ASC | DESC ] ]
```

Considerando a base de dados criada no ponto anterior e o programa *SQL Query Analyzer*, pretende-se que experimente os seguintes comandos:

- a) `Select * From Empregado; ----- o que faz o '*'?`
- b) `Select EmpNum, Nome From Empregado;`
- c) `Select EmpNum From Empregado; ----- não aparecem tuplos duplicados, porquê?`
- d) `Select DepNum From Empregado; ----- aparecem duplicados!`
- e) `Select Distinct DepNum From Empregado;`
- f) `Select * From Empregado Where Salario > 200000 and Salario < 300000;`
- g) `Select * From Empregado Where Salario Between 200000 and 30000;`
- h) `Select * From Empregado Where (Salario Between 200000 and 30000) or EmpNum = 1;`
- i) `Select * From Empregado Order By Nome Desc;`
- j) `Select * From Empregado Where Salario Between 200000 and 30000 Order By Salario;`
- k) `Select * From Empregado Where Nome Like 'D%';`
- l) `Select * From Empregado Where EmpNum > 3;`
- m) `Select * From empregado Where EmpNum = 3 or EmpNum = 5;`
- n) `Select * From Empregado Where EmpNum IN (3,5);`
- o) `Select * From Empregado Where EmpNum IN (Select EmpNum From Empregado Where Nome Like 'D%');`