

Subconsultas e Junções

Diogo Cezar Teixeira Batista

`diogocezar@utfpr.edu.br`

Universidade Tecnológica Federal do Paraná
Campus Cornélio Procópio
UTFPR-CP

Cornélio Procópio - 2008

Agenda I

- 1 Subconsultas e Junções
 - Exists
 - In e Not In
 - Junções
 - Inner Join
 - Left Join
 - Right Join

Exists

Caracterização

Verifica se um valor existe no *resultset* retornado por um *select*.

- Se a subconsulta retornar pelo menos uma linha, o resultado dela é TRUE, caso contrário será FALSE;
- Observe a ligação entre a consulta externa e a interna.

Código 1: Exemplo de utilização do Exists

```
1 SELECT col1 FROM tab1
2 WHERE EXISTS(SELECT 1 FROM tab2 WHERE col2 = tab1.col2);
```

In e Not In

Caracterização

Verifica se um valor está contido em um grupo de valores.

Código 2: Exemplo de utilização do in e not in

```
1 SELECT DISTINCT nome_cliente
2 from devedor
3 Where nome_cliente in (select nome_cliente from depositante)
4
5 SELECT DISTINCT nome_cliente
6 from devedor
7 Where nome_cliente not in ('Smith', 'Jones')
```

Formas Alternativas para In e Not In

Código 3: Exemplo de utilização do in e not in de formas alternativas

```
1 SELECT * FROM produtos
2 WHERE cod_produto = (SELECT cod_produto FROM item_pedido)
3
4 SELECT * FROM Table1
5     WHERE (column1, column2) = (SELECT column1, column2 FROM
        Table2)
```

Inner Join

Caracterização

Somente as linhas/registros que satisfaçam a ligação determinada pelo *JOIN* serão recuperados pelo *SELECT*, sendo assim, os registros que não se enquadram no relacionamento definido pelo *JOIN* não serão recuperados.

Exemplo

Código 4: Exemplo de utilização do Inner Join

```
1 SELECT p.uname, p.nome, a.qtde
2 FROM pessoas p
3 INNER JOIN acessos a
4 ON p.uname = a.pessoa
5 ORDER BY p.uname;
6
7 O mesmo JOIN:
8
9 SELECT p.uname, p.nome, a.qtde
10 FROM pessoas p, acessos a
11 WHERE p.uname = a.pessoa
12 ORDER BY p.uname;
```

Left Join

Caracterização

Através do uso do *LEFT*, todos os registros na tabela à esquerda da *query* serão listados, independente de terem ou não registros relacionados na tabela à direita. Nesse caso, as colunas relacionadas com a tabela da direita voltam nulos (*NULL*).

Exemplo

Código 5: Exemplo de utilização do Left Join

```
1 SELECT p.uname, p.nome, a.pessoa, a.qtde
2 FROM pessoas p LEFT JOIN acessos a
3 ON p.uname=a.pessoa
4 ORDER BY p.uname;
```

Left Join

Caracterização

É o inverso do *Left Join*, ou seja, todos os registros da tabela à direita serão listados, independente de terem ou não registros relacionados na tabela à esquerda.

Exemplo

Código 6: Exemplo de utilização do Right Join

```
1 SELECT p.uname, p.nome, a.pessoa, a.qtde
2 FROM pessoas p
3 RIGHT JOIN acessos a
4 ON p.uname=a.pessoa
5 ORDER BY p.uname;
```
