

Aula 5 - Comandos de SQL

SUBQUERY ou SUBCONSULTAS

Prof. Rui Silvestrin

1 - Definição

É um comando Select aninhado em outro comando Select e que retorna resultados intermediários. A Subquery, ou a query mais interna (Sub-select), é geralmente executada primeiro e sua saída é usada para completar a condição Query principal ou mais externa. Ela pode retornar um valor só ou uma lista de valores para ser usada numa comparação.

Uma Subquery pode ser muito útil quando precisamos selecionar linhas de uma tabela com uma condição que dependa de um dado da própria tabela.

- . Podemos usar uma Subquery em uma cláusula HAVING;
- . Não podemos usar a cláusula ORDER BY em uma SUBQUERY;
- . Subqueries podem ser aninhadas.

Subconsultas são *sempre* colocadas entre parênteses e podem retornar no resultado apenas uma coluna (embora em alguns casos podem retornar mais de uma linha).

```
select nome 'Funcionário', Salario 'Salário',  
Porcentagem = 100*(salario*1.0/  
(select sum(salario) from funcionario) )  
from funcionario
```

Subconsultas com operadores

Uma sub-consulta pode ser inserida nos resultados, como acima, ou pode ser usada numa expressão WHERE, com um operador de comparação, como =, <, >, <=, >= ou <>. Nesse caso ela deve retornar apenas um valor. Esse valor é substituído na consulta principal no momento da execução.

Listas de valores

Uma sub-consulta pode retornar uma lista de valores e essa lista de valores pode ser usada em comparações com o operador IN. Por exemplo, para saber quais são os funcionarios que possuem cargos, pode ser usado:

```
select funcionario.nome  
from funcionario  
where codcargo in (select codcargo from cargo)
```

Uma sub-consulta pode ser usada também com um operador de comparação modificado com as palavras ANY [qualquer] ou ALL [todos]. Por exemplo, > ALL [maior que todos] significa que para que a condição seja satisfeita, o valor comparado deve ser maior que todos os elementos da lista:

O exemplo abaixo utiliza o banco de dados Pubs.

```
select title
from titles
where advance > ALL
(select advance
 from publishers, titles
 where titles.pub_id = publishers.pub_id
 and pub_name = 'Algodata Infosystems')
```

A subconsulta retorna uma lista de valores de 'advance' [adiantamento] que contém todos os valores de adiantamento de livros publicados pela editora 'Algodata Infosystems'. Seu resultado é (5000, 5000, 5000, 7000, 8000) A consulta externa retorna os livros cujo 'advance' é maior do que todos os itens dessa lista.

As seguintes combinações que podem ser usadas:

- > ALL maior que todos os elementos da lista
- < ALL menor que todos
- <> ALL diferente de todos (o mesmo que NOT IN)
- = ANY igual a *algum* dos elementos da lista (o mesmo que IN)
- > ANY maior que algum dos elementos da lista
- < ANY menor que algum dos elementos
- <> ANY diferente de algum dos elementos da lista (falso se igual a todos)

Além disso, podem ser usadas combinações com >= e <=, de forma análoga. Note que não é permitido o uso de = ALL.

Testes de existência

Um teste de existência é uma condição que envolve a palavra EXISTS e uma sub-consulta. A condição é verdadeira se a sub-consulta retorna alguma linha e é falsa se ela retorna zero linhas. Por exemplo, para saber quais os departamentos que possui funcionários cujo cargo é igual a cargo1, utilize o banco de dados Exemplos e execute o seguinte comando:

```
select d.nome
from departamento d
where exists (select *
 from funcionario f, cargo c
 where f.codcargo = c.codcargo
 and c.nome = 'Cargo1')
```

and d.coddepartamento = f.coddepartamento)

O resultado da sub-consulta não importa, pois está apenas sendo testada a existência de um resultado. Nesse caso, a lista de colunas é sempre um asterisco (*).

Subconsultas correlacionadas

As sub-consultas que foram vistas até agora nos exemplos podem ser avaliadas uma vez só e depois substituídas no corpo da consulta principal. Já uma sub-consulta *correlacionada* [correlated subquery] depende dos valores da consulta principal onde ela está alinhada, por isso deve ser avaliada uma vez para cada linha do resultado externo.

Por exemplo, a seguinte utilize o banco de dados Pubs para consultar a lista, para cada livro, o autor que tem a maior porcentagem de royalties sobre o livro (a tabela 'titleauthor' relaciona livros e autores de forma N x N):

```
select title_id, au_id, royaltyper
from titleauthor ta
where royaltyper = (select max(royaltyper)
                    from titleauthor
                    where title_id = ta.title_id)
```

Essa é uma sub-consulta correlacionada porque ela faz referência a uma tabela da consulta mais externa. A sub-consulta é avaliada repetidas vezes, uma para cada linha da tabela 'titleauthor'.