

Banco de Dados - Ciclo 3 - Webaula 23



WEBAULA 23

☰ Seleção de Dados com Várias Tabelas

☰ Referências

QUESTION BANKS

Seleção de Dados com Várias Tabelas

A partir de agora vamos estudar as **subconsultas** na linguagem SQL. Outra denominação atribuída à subconsulta é “consulta interna” – mas vamos adotar a primeira. Uma subconsulta nada mais é do que uma consulta inserida dentro de outra, ou seja, a presença do comando SELECT (com suas cláusulas necessárias ao contexto) servindo de restrição/filtro para a consulta principal. Em geral, dizemos que precisamos utilizar uma subconsulta quando não podemos recuperar o que desejamos através de um único comando. Em outras palavras, é quando temos duas ou mais variáveis a serem consideradas no problema.

Dica

Assim como nas demais seções de conteúdo, recomendamos que você vá executando as sentenças SQL para colocar em prática os conceitos apresentados.

Para compreender a funcionalidade de uma subconsulta, considere o seguinte cenário apresentado na Figura 66, onde se deseja conhecer quais funcionários têm o salário maior que o salário de Jones.

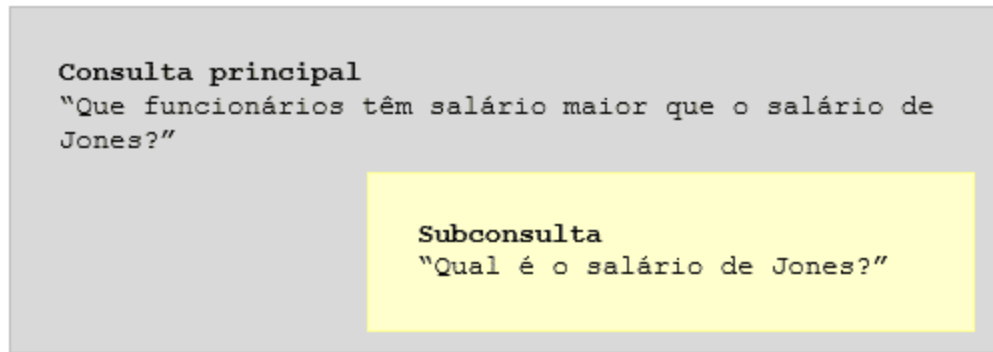


Figura 66

Suponha que você deseja criar uma consulta para descobrir quem recebe um salário maior que o empregado Jones. Para resolver este problema são necessárias duas consultas: (i) uma consulta para descobrir o quanto Jones recebe, e (ii) a outra para descobrir quem recebe mais do que esta quantia. Você pode resolver esse problema combinando duas consultas, colocando uma consulta dentro da outra. A consulta interna, ou a subconsulta, retorna um valor que é usado pela consulta externa, ou pela consulta principal, como também é conhecida. Usar uma subconsulta equivale a executar duas consultas sequenciais e usar o resultado da primeira como o valor de pesquisa na segunda consulta. Vamos conhecer agora a sintaxe do uso de subconsultas (Quadro 95).

Quadro 95 – Sintaxe de utilização de subconsultas

```
SELECT lista_select
FROM tabela(s)
WHERE colunax operador
      (SELECT colunax FROM tabela);
```

Observe na sintaxe de utilização de uma subconsulta alguns detalhes importantes:

- A subconsulta (consulta interna) é executada uma vez antes da consulta principal.
- O resultado da subconsulta é usado pela consulta principal (consulta externa).
- A subconsulta pode ser utilizada nas cláusulas WHERE, HAVING e FROM.
- A subconsulta deve estar sempre entre parênteses e no lado direito do operador de comparação.
- Não adicione uma cláusula ORDER BY a uma subconsulta.
- Utilize operadores de uma única linha com subconsultas de uma única linha.
- Utilize operadores de várias linhas com subconsultas de várias linhas.

Como vimos, uma subconsulta é uma instrução SELECT que é incorporada em uma cláusula de outra instrução SELECT. Você pode desenvolver instruções sofisticadas a partir de instruções simples usando subconsultas. Elas podem ser muito úteis quando você precisa selecionar linhas de uma tabela com uma condição que dependa dos dados da própria tabela. Por exemplo, a subconsulta normalmente é executada primeiro e sua saída é usada para concluir a condição de consulta da consulta principal (consulta externa).

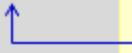
É possível colocar a subconsulta em várias cláusulas do comando SELECT: cláusula WHERE, cláusula HAVING e cláusula FROM. Alguns SGBDs relacionais admitem a utilização de subconsulta na próxima cláusula SELECT, mas isso não é padrão ANSI. A subconsulta, por sua vez, sempre deve estar entre parênteses e no lado direito do operador de comparação, o que é uma exigência do padrão ANSI. Na sintaxe, observe que temos “colunax” na cláusula WHERE e na subconsulta. Isto pode representar que é necessário que as colunas apresentem o mesmo nome, porém não é este o sentido. É necessário que as colunas apresentem características em comum, como o tipo de dados, por exemplo; já o operador pode ser de comparação, o que exigirá correlação de mesma natureza das colunas para haver comparação.

i Dica

Adiante vamos falar mais sobre os operadores e sua utilização. Isso porque, de acordo com a quantidade de colunas e linhas retomadas pela subconsulta, precisamos ajustar o operador utilizado.

A instrução SQL a seguir (Figura 67) corresponde à questão inicialmente apontada como exemplo (Figura 66), ou seja, identificar quem tem salário maior que o de Jones.

```
SELECT ename
FROM emp
WHERE sal > 2975
```



```
(SELECT sal
FROM emp
WHERE empno=7566)
```

Figura 67

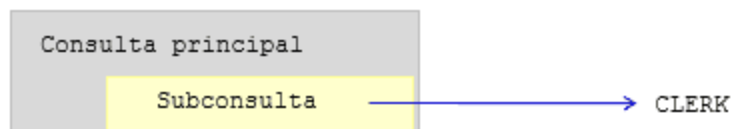
Agora observe com atenção a cláusula WHERE da subconsulta do exemplo acima. Você pode perceber que estamos utilizando a coluna correspondente à chave primária da tabela – no caso, a coluna EMPNO com o valor de 7566, que corresponde ao código do empregado JONES. Você imagina por quê? Não? Uma dica: e se houver dois funcionários com o nome JONES? E agora? Adiante vamos falar mais sobre isso! Como resultado da execução da sentença acima, teremos o exposto no Quadro 96.

Quadro 96 – Resultado da utilização de subconsulta

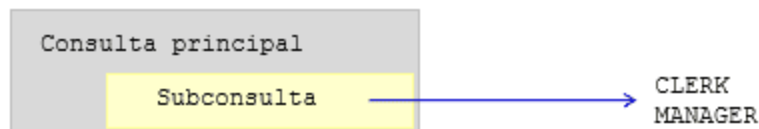
```
ENAME
-----
KING
FORD
SCOTT
```

Agora que você sabe o que são e para que servem, vamos conhecer os tipos de subconsultas, que são basicamente classificadas em três tipos, conforme ilustra a Figura 68.

Subconsulta de uma única linha



Subconsulta de várias linhas



Subconsulta de várias linhas e colunas



Figura 68

Na ilustração acima é possível observar que a classificação dos tipos de subconsulta se dá pelo retorno da sua execução (interna), onde temos: (i) subconsulta de uma única linha é quando o valor retornado por esta entrega apenas um valor para a consulta principal; (ii) a subconsulta de várias linhas é quando ocorre a entrega de mais de uma linha, porém de apenas um campo; (iii) subconsulta de várias linhas e colunas é quando a subconsulta entrega mais de uma linha com mais de uma coluna.

Quando um subconsulta retorna uma única linha (registro), utilizamos os seguintes operadores relacionais (Figura 69):

OPERADOR	SIGNIFICADO
=	igual a
>	maior que
>=	maior que ou igual a
<	menor que
<=	menor que ou igual a
<>	diferente de

Figura 69

Dica

Você se lembra que no último exemplo apresentado (aquele que solicitava recuperar os empregados que tinham salário maior que o de JONES) nós utilizamos a coluna que identificava unicamente cada registro? Pois bem, fizemos isso para garantir que a subconsulta não retornaria mais de uma linha. Portanto, atenção! Se a subconsulta retornar mais de uma linha, será gerado um erro, pois os operadores acima não suportam o retorno de mais de uma linha.

Como exemplo da utilização de subconsulta de única linha temos a sentença SQL da Figura 70, que exibe o nome (ENAME) e a atividade (JOB) dos empregados que exercem a mesma atividade do empregado "CLERK" e recebem mais do que o empregado "ADAMS".

```

SELECT  ename, job
FROM    emp
WHERE   job = CLERK ← (SELECT  job
                        FROM    emp
                        WHERE   empno = 7369)

AND     sal > 1100 ← (SELECT  sal
                        FROM    emp
                        WHERE   empno = 7876);

```

Figura 70

O exemplo apresentado exibe empregados cujo cargo é o mesmo que o empregado “7369” e cujo salário é maior que o empregado “7876”. Observe que em ambas as subconsultas estão sendo utilizadas colunas que correspondem ao identificador único da tabela (chave primária). O exemplo consiste em três blocos de consulta: a consulta principal (consulta externa) e duas subconsultas (consulta interna).

Os blocos de consulta interna são executados primeiro, produzindo os resultados da consulta principal “CLERK” e “1100”, respectivamente. O bloco de consulta externa é então processado e usa os valores retornados pelas consultas internas para completar suas condições de pesquisa. Ambas as consultas internas retornam valores únicos: “CLERK” e “1100” respectivamente. Portanto essa instrução SQL é chamada de subconsulta de uma única linha. É importante destacar que as consultas externas e internas podem obter dados de diferentes tabelas. Como resultado da execução do comando temos o resultado apresentado no Quadro 97.

Quadro 97 – Resultado da utilização de subconsulta de única linha

ENAME	JOB
-----	-----
MILLER	CLERK

Assim como utilizamos colunas que correspondem a um identificador único como critério de filtro em uma subconsulta, podemos também utilizar função de grupo. Isto porque é sabido que uma função de grupo retorna apenas um valor, quando não agrupada por coluna, é claro. A sentença SQL abaixo (Figura 71) exibe o nome, a atividade e o salário do(s) empregado(s) que possui(em) o menor salário. A utilização de subconsulta é fundamental, pois inicialmente (consulta interna) é apurado qual o menor salário, e em seguida quem possui esse salário.


```

SELECT  ename, job, sal
FROM    emp
WHERE   sal = 800 ← (SELECT MIN(sal)
                     FROM emp);

```

Figura 71

Como resultado da execução da sentença SQL acima teremos o que é apresentado no Quadro 98. É importante destacar que, no exemplo, o menor salário é “800” e há apenas um empregado (“SMITH”) que recebe o mesmo. Caso houvesse mais de um empregado recebendo o valor “800”, todos seriam listados.

Quadro 98 – Resultado da utilização de subconsulta de única linha com função de grupo

ENAME	JOB	SAL
-----	-----	-----
SMITH	CLERK	800

Poderá haver cenário em que será necessário utilizar o resultado de uma função de grupo como expressão da consulta principal (consulta externa). Neste caso, você pode utilizar subconsultas na cláusula HAVING. No exemplo da Figura 72 temos um cenário hipotético onde se deseja exibir o(s) departamento(s) onde o salário mínimo pago a um empregado é maior do que o menor salário pago no departamento 20.

```

SELECT deptno, MIN(sal)
FROM emp
GROUP BY deptno
HAVING MIN(sal) > 800 ← (SELECT MIN(sal)
FROM emp
WHERE deptno = 20);

```

Figura 72

Mas fique atento, pois não mudam as regras de utilização dos operadores de comparação, ou seja, se a subconsulta retornar mais de uma linha (registro), os operadores relacionais não podem ser utilizados.

Dica

Sugerimos a você executar isoladamente a instrução SQL que corresponde à consulta para validar o resultado e somente a partir daí adicioná-la à consulta principal. Lembre-se que os erros podem não necessariamente ser sintáticos, e sim semânticos, ou seja, de uso inadequado de operador.

Quando uma subconsulta retorna mais de uma linha, é necessário utilizar operadores de comparação de várias linhas. Isso é o que classificamos como subconsulta de várias linhas. Neste contexto temos os seguintes operadores (Figura 73).

OPERADOR	SIGNIFICADO
IN	igual a qualquer membro na lista
ANY	compare o valor a cada valor retornado pela subconsulta
ALL	compare o valor a todo valor retornado pela subconsulta

Figura 73

A seguir vamos ilustrar a utilização destes operadores através de exemplos. Vamos começar pelo operador IN, sobre o qual já vimos algo relacionado na seção em que tratamos de consulta básica. Considere a sentença SQL apresentada na Figura 74.

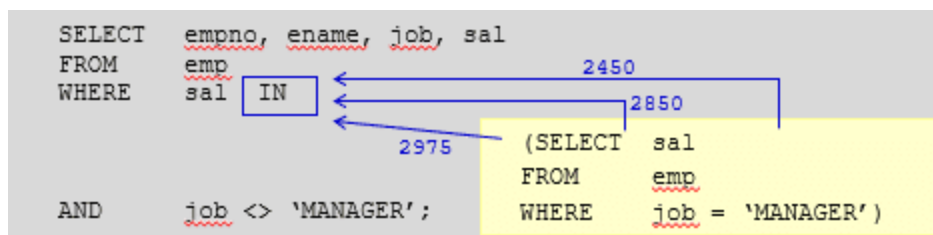


Figura 74

O operador IN compara um valor a cada valor retornado por uma subconsulta. No exemplo, a sentença busca exibir empregados onde o salário seja igual ao de qualquer gerente, mas que não seja um gerente (job <> "MANAGER"). Neste caso não houve linha recuperada, pois não há empregados, exceto os gerentes, que ganham a mesma coisa que qualquer um deles.

A seguir vamos conhecer o operador ANY. Considere a sentença SQL ilustrada por meio da Figura 75.

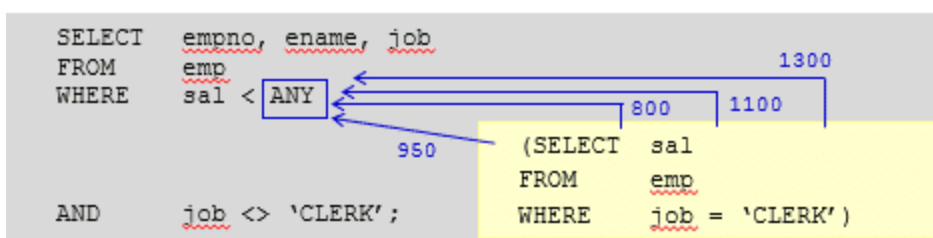


Figura 75

O operador ANY compara um valor a cada valor retornado por uma subconsulta. O exemplo exibe empregados cujo salário é menor do que qualquer escriturário ("CLERK") mas que não seja escriturário,

assim como o salário máximo que um escriturário recebe seja "1300". Ou seja, a instrução SQL exibe todos os empregados que não são escriturários mas recebem menos do que 1300. Como resultado temos o conteúdo apresentado no Quadro 99.

Quadro 99 – Resultado da utilização de subconsulta de única linha com operador ANY

EMPNO	ENAME	JOB
-----	-----	-----
7654	MARTIN	SALESMAN
7521	WARD	SALESMAN

Dica

Observe que o operador ANY é sempre utilizado em conjunto com um operador relacional. Logo, podemos afirmar que quando utilizamos o operador ANY associado ao operador de igual (=), é a mesma coisa que utilizar o operador IN.

O terceiro e último operador de subconsulta de várias linhas é ALL. Para entendermos a utilização do operador ALL, considere a seguinte sentença SQL (Figura 76).

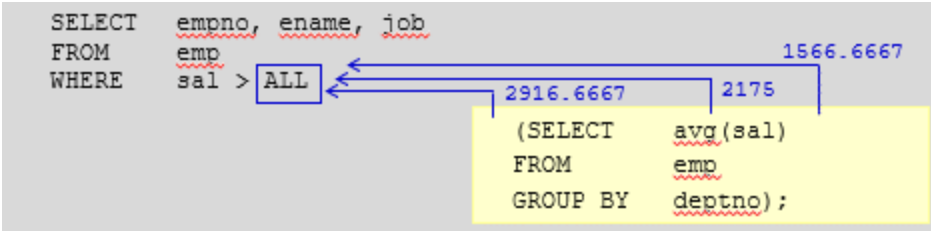


Figura 76

O operador ALL compara um valor a todo valor retornado por uma subconsulta. O exemplo apresentado exibe empregados cujo salário seja maior que os salários médios de todos os departamentos. O salário médio mais alto de um departamento é “2916.66”, portanto a consulta retorna os empregados cujo salário seja maior que “2916.66”. Isso pode ser observado no Quadro 100.

Quadro 100 – Resultado da utilização de subconsulta de única linha com operador ALL

EMPNO	ENAME	JOB
-----	-----	-----
7839	KING	PRESIDENT
7566	JONES	MANAGER
7902	FORD	ANALYST
7788	SCOTT	ANALYST

Dica

Quando apresentamos a você o operador IN (consulta básica), comentamos também sobre a negação a ele, ou seja, o NOT IN. Pois bem, o NOT, assim como foi utilizado com o operador IN, também pode ser utilizado com os operadores ANY e ALL.

Agora vamos tratar de subconsultas de várias colunas e várias linhas. Para isso, considere o cenário ilustrado por meio da Figura 77. Imagine que você precise comparar mais de uma coluna do resultado da subconsulta com a consulta principal. No cenário temos como resultado dois campos: JOB e DEPTNO. Ambos, simultaneamente, serão comparados a cada conjunto de dados da consulta principal.

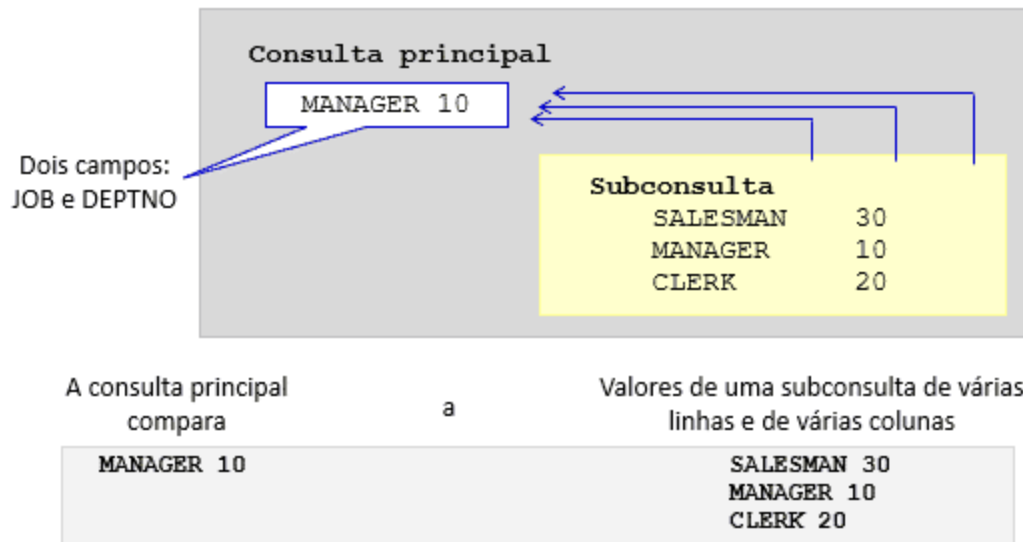


Figura 77

Com os dados apresentados inicialmente para a tabela EMP podemos observar que apenas o conjunto "MANAGER 10" possui o equivalente na subconsulta. A sentença SQL abaixo (Figura 78) exibe o nome e o salário dos empregados que trabalham no mesmo departamento e exercem a mesma atividade do empregado ALLEN.

```
SELECT  ename, sal
FROM    emp
WHERE   (job, deptno) in      (SELECT job, deptno
                                  FROM emp
                                  WHERE ename = 'ALLEN')
```

Figura 78

Como resultado da execução da sentença SQL acima teremos o conteúdo do Quadro 101.

Quadro 101 – Resultado da utilização de subconsulta de várias colunas e linhas

ENAME	SAL
-----	-----
MARTIN	1250
ALLEN	1600
TURNER	1500
WARD	1250

Destacamos o que provavelmente você já deve ter constatado. Deve haver uma correlação entre o número de colunas da consulta principal com a consulta interna, ou seja, se houver duas colunas na consulta principal, duas colunas deverão existir na consulta interna. Também não esqueça das características delas para obter sucesso na execução.

Encerramento

Nesta seção descobrimos que podemos utilizar subconsultas quando uma consulta se baseia em valores desconhecidos. Neste caso, podemos utilizar uma consulta interna (que denominamos “subconsulta”) para em seguida relacionarmos o resultado desta à consulta principal. Também conhecemos os operadores IN, ANY e ALL, assim como sua composição com o operador NOT. Devemos atentar para a utilização adequada dos operadores para evitar erros semânticos que nem sempre são identificados durante a construção das sentenças SQL, em função do conjunto de dados aos quais submetemos a sentença específica.

Dica

Avalie com muita atenção o resultado obtido com a sentença SQL construída quando for utilizar a subconsulta. Há situações em que a massa de dados de teste pode

sinalizar um resultado que provocará erro na execução da sentença, à medida que novos dados forem inseridos. Um exemplo comum é a utilização de um operador de subconsulta de linha única, mas que em dado momento retorna mais de uma linha.

CONTINUE

Referências

Referências

ORACLE. Documentação de utilização do sistema (Versão 11g). Califórnia: Oracle Corp., 2017.

CARDOSO, Vírínia M. **Linguagem SQL**: fundamentos e práticas. São Paulo: Saraiva, 2009.

Referências de Imagens

Divisão de Modalidades de ensino (DME), Fundação Universidade Regional de Blumenau (FURB), 2019.