



## Unidade 3

# Bancos de Dados Relacionais e Linguagem SQL

---

## 3.3 Subconsultas

---



# Subconsultas

- Alguém já perguntou a resposta de um problema de matemática, mas, antes de dar essa resposta, você teve que resolver o problema por conta própria?
- Na linguagem SQL, as subconsultas permitem encontrar as informações necessárias para obtermos estas informações desejadas.
- Aplica-se para atender a necessidade de gravar uma consulta onde precisamos descobrir dados que estão em outras tabelas para resolver a consulta principal.

# Subconsultas

- A possibilidade de combinar duas consultas em uma pode ser muito útil quando você precisa selecionar linhas de uma tabela com uma condição que depende dos dados da própria tabela.
- Você poderia resolver esse problema aninhando as consultas, ou seja, colocando uma consulta dentro da outra.
- **A consulta interna é chamada de subconsulta.**

# Subconsultas

- Uma subconsulta é uma instrução *SELECT* que está embutida em uma cláusula de outra instrução *SELECT* e é executada uma vez antes da consulta principal.
- Seu resultado da é usado pela consulta principal ou externa.
- As subconsultas podem ser colocadas em uma série de cláusulas SQL, incluindo *WHERE*, *HAVING* e *FROM*.

# Subconsultas

- A subconsulta deve estar entre parênteses e posicionada à direita da condição de comparação sendo que as consultas externa e interna podem obter dados de diferentes tabelas.
- Somente uma cláusula ORDER BY pode ser usada em uma instrução SELECT e se for usada, ela deve ser a última cláusula na consulta externa sendo que a subconsulta não pode ter uma cláusula ORDER BY própria
- A limitação do número de subconsultas é o tamanho do buffer utilizado por ela.

# Tipos de Subconsultas

- **Subconsultas de linha única:** utilizam operadores de linha única ( $>$ ,  $=$ ,  $>=$ ,  $<$ ,  $<>$ ,  $<=$ ) e retornam somente uma linha da consulta interna.
- **Subconsultas multilinha:** utilizam operadores multilinha (*IN*, *ANY*, *ALL*) e retornam mais de uma linha da consulta interna.
- 😊 Se uma subconsulta retornar um valor nulo ou nenhuma linha, a consulta externa pega os resultados da subconsulta(NULL) e usa-os em sua cláusula WHERE, logo a consulta retornará **NULL**.

# Tipos de Subconsultas

```
SELECT t.title, d.dept_name, e.first_name,e.last_name, s1.salary, de.from_date, de.to_date
FROM titles t INNER JOIN employees e USING (emp_no)
      INNER JOIN dept_emp de ON (e.emp_no=de.emp_no)
      INNER JOIN departments d ON (de.dept_no=d.dept_no)
      INNER JOIN salaries s1 ON(e.emp_no=s1.emp_no)
WHERE s1.salary > (SELECT ROUND((MAX(s2.salary)/12),2)
      FROM employees e2 INNER JOIN dept_manager dm ON (e2.emp_no=dm.emp_no)
      INNER JOIN salaries s2      ON (s2.emp_no=e2.emp_no)
      WHERE dm.to_date = '9999-01-01')
AND d.dept_name = 'Marketing'
AND de.to_date = '9999-01-01'
AND t.to_date = '9999-01-01'
AND s1.to_date = '9999-01-01';
```



# Tipos de Subconsultas

```
SELECT t.title, d.dept_name, e.first_name, e.last_name, s1.salary, de.from_date, de.to_date
FROM titles t INNER JOIN employees e USING (emp_no)
      INNER JOIN dept_emp de ON (e.emp_no=de.emp_no)
      INNER JOIN departments d ON (de.dept_no=d.dept_no)
      INNER JOIN salaries s1 ON(e.emp_no=s1.emp_no)
WHERE s1.salary > (SELECT ROUND((MAX(s2.salary)/12),2)
      FROM employees e2 INNER JOIN dept_manager dm ON (e2.emp_no=dm.emp_no)
      INNER JOIN salaries s2 ON (s2.emp_no=e2.emp_no)
      WHERE dm.to_date = '9999-01-01' AND s2.salary is NULL)
AND d.dept_name = 'Marketing'
AND de.to_date = '9999-01-01'
AND t.to_date = '9999-01-01'
AND s1.to_date = '9999-01-01';
```

---

*《 Now, let to advance a lot! 》*

---

# Subconsultas de linha única

```
SELECT c.ID, c.Name  
FROM city c  
WHERE c.CountryCode = (SELECT Code  
                        FROM country  
                        WHERE Name = 'Brazil')
```



Sempre à direita da condição  
de comparação

Tabelas diferentes



```
SELECT SUM(c.Population)  
FROM city c  
WHERE c.CountryCode = (SELECT Code  
                        FROM country  
                        WHERE Name = 'Brazil')
```



Tabelas diferentes



# Subconsultas de linha única

```
SELECT d.dept_name, MIN(s1.salary)
FROM titles t INNER JOIN employees e USING (emp_no)
            INNER JOIN dept_emp de ON (e.emp_no=de.emp_no)
            INNER JOIN departments d ON (de.dept_no=d.dept_no)
            INNER JOIN salaries s1 ON(e.emp_no=s1.emp_no)

WHERE
de.to_date = '9999-01-01'
AND t.to_date = '9999-01-01'
AND s1.to_date = '9999-01-01'
GROUP BY d.dept_name
HAVING MIN(s1.salary) > (SELECT ROUND((AVG(s2.salary))/12),2)
                        FROM employees e2 INNER JOIN dept_manager dm ON (e2.emp_no=dm.emp_no)
                        INNER JOIN salaries s2 ON (s2.emp_no=e2.emp_no)
                        WHERE dm.to_date = '9999-01-01')
ORDER BY d.dept_name;
```



# Subconsultas multilinha

```
SELECT *  
FROM employees e  
WHERE YEAR(e.hire_date) IN ( SELECT YEAR(t.from_date)  
                             FROM titles t  
                             WHERE t.title = 'Senior Staff'  
                             AND t.to_date = '9999-01-01');
```

```
SELECT *  
FROM employees e  
WHERE YEAR(e.hire_date) > ANY ( SELECT DISTINCT(YEAR(t.from_date))  
                                FROM titles t  
                                WHERE t.title = 'Staff'  
                                AND t.to_date = '9999-01-01');
```

```
SELECT *  
FROM employees e  
WHERE YEAR(e.hire_date) < ALL ( SELECT DISTINCT(YEAR(dm.from_date))  
                                FROM dept_manager dm  
                                WHERE  
                                dm.to_date = '9999-01-01')
```

# Subconsultas multilinha - Resultados NULL

- 😊 Supondo que um dos valores retornados por uma subconsulta multilinha seja nulo, mas os outros não sejam caso utilize-se *IN* ou *ANY* a consulta externa retornará linhas que correspondam aos valores não nulos.
- O retorno de um valor NULL na instrução *ALL*, a consulta externa não retornará linhas porque *ALL* compara a linha da consulta externa com todos os valores retornados pela subconsulta, incluindo o valor nulo e a comparação de qualquer elemento com resultados nulos é nula.

# Subconsultas multilinha

- As cláusulas *GROUP BY* e *HAVING* também podem ser usadas com subconsultas multilinha.
- Podem também utilizar mais de uma coluna e podem ser do tipo *pair-wise* ou *non-pair-wise*.

```
SELECT *  
FROM employees  
WHERE (emp_no,hire_date) IN (SELECT emp_no, from_date  
                             FROM dept_emp  
                             WHERE dept_no = 'd003')  
  
AND gender = 'F';
```

```
SELECT first_name, last_name, birth_date  
FROM employees  
WHERE birth_date IN ( SELECT from_date  
                      FROM   dept_emp);
```

# Subconsultas multilinha

- As cláusulas *GROUP BY* e *HAVING* também podem ser usadas com subconsultas multilinha.
- Podem também utilizar mais de uma coluna e podem ser do tipo *pair-wise* ou *non-pair-wise*.

```
SELECT *  
FROM employees  
WHERE (emp_no,hire_date) IN (SELECT emp_no, from_date  
                             FROM dept_emp  
                             WHERE dept_no = 'd003')  
  
AND gender = 'F';
```

```
SELECT first_name, last_name, birth_date  
FROM employees  
WHERE birth_date IN ( SELECT from_date  
                      FROM   dept_emp);
```



# Subconsultas multilinha

```
SELECT d.dept_name, MIN(s1.salary)
FROM titles t INNER JOIN employees e USING (emp_no)
            INNER JOIN dept_emp de ON (e.emp_no=de.emp_no)
            INNER JOIN departments d ON (de.dept_no=d.dept_no)
            INNER JOIN salaries s1 ON(e.emp_no=s1.emp_no)

WHERE
de.to_date = '9999-01-01'
AND t.to_date = '9999-01-01'
AND s1.to_date = '9999-01-01'
GROUP BY d.dept_name
HAVING MIN(s1.salary) > ANY (SELECT ((s2.salary)/12)
                             FROM employees e2 INNER JOIN dept_manager dm ON (e2.emp_no=dm.emp_no)
                             INNER JOIN salaries s2 ON (s2.emp_no=e2.emp_no)
                             WHERE dm.to_date = '9999-01-01'
                             AND dm.dept_no = 'd003')
ORDER BY d.dept_name;
```

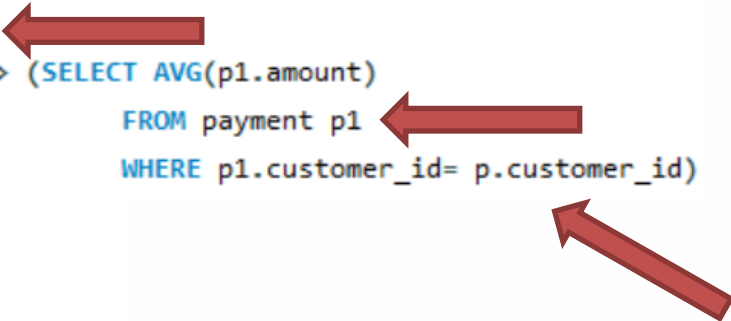
# Subconsultas correlacionadas

- As subconsultas correlacionadas são usadas para processamento linha a linha.
- Cada subconsulta é executada uma vez para cada linha da consulta externa ao contrário de uma subconsulta normal, onde a consulta *SELECT* interna é executada primeiro e uma única vez, retornando os valores a serem usados pela consulta externa.
- Podemos dizer que a consulta interna é impulsionada pela consulta externa quando elas estão correlacionadas.

# Subconsultas multilinha

```
use sakila;
```

```
SELECT p.customer_id, p.rental_id, p.amount  
FROM payment p  
WHERE p.amount > (SELECT AVG(p1.amount)  
                  FROM payment p1  
                  WHERE p1.customer_id= p.customer_id)
```



Atenção na dependência  
entre as consultas!

# EXISTS e NOT EXISTS em Subconsultas


- *EXISTS* e seu oposto, *NOT EXISTS*, são duas cláusulas que podem ser usadas para testar equivalências em subconsultas
- *EXISTS* faz o teste para encontrar um valor TRUE, ou um resultado equivalente na subconsulta.

```
use employees;
```

```
SELECT first_name, last_name  
FROM employees e  
WHERE EXISTS (SELECT *  
              FROM dept_manager dm  
              WHERE dm.emp_no=e.emp_no)
```

```
SELECT e.first_name, e.last_name  
FROM employees e  
WHERE NOT EXISTS (SELECT *  
                  FROM titles t  
                  WHERE t.emp_no=e.emp_no  
                  AND title = 'Staff')
```

# EXISTS e NOT EXISTS em Subconsultas

-  **TENHA CUIDADO** com valores nulos nas subconsultas quando estiver usando IN ou NOT IN.
- Se não tiver certeza se uma subconsulta incluirá ou não um valor nulo, elimine-o *IS NOT NULL* em uma cláusula WHERE ou use NOT EXISTS.

# Referências bibliográficas

**ELMASRI**, R., NAVATHE, S. B., Sistemas de Banco de Dados: Fundamentos e Aplicações. 3ª Ed., Editora LTC, 2002.

**MANNINO**, Michael V. Projeto, Desenvolvimento de Aplicações e Administração de Banco de Dados. 3ª. Ed. Porto Alegre. Bookman. 2008.