



# Sistemas de Informação

Prof. Me. Alexandre Barcelos  
profalexandre.barcelos@fiap.com.br

# Subconsultas

## Objetivos

Ao concluir esta lição, você será capaz de:

- Definir subconsultas
- Descrever os tipos de problemas que as subconsultas podem solucionar
- Listar os tipos de subconsultas
- Criar subconsultas de uma única linha e de várias linhas

### Objetivos

Nesta lição, você conhecerá os recursos mais avançados da instrução `SELECT`. É possível criar subconsultas na cláusula `WHERE` de outra instrução SQL para obter valores baseados em um valor condicional desconhecido. Esta lição aborda subconsultas de uma única linha e de várias linhas.

## Usando uma Subconsulta para Solucionar um Problema

Quem tem um salário maior que o salário de Abel?

Consulta principal:



Quais funcionários têm um salário maior que o salário de Abel?

Subconsulta:



Qual é o salário de Abel?

1-5

### Usando uma Subconsulta para Solucionar um Problema

Suponha que você queira criar uma consulta para saber quem ganha um salário maior que o salário de Abel.

Para solucionar esse problema, são necessárias *duas* consultas: uma para saber quanto Abel ganha e outra para saber quem ganha uma quantia maior. Você pode solucionar esse problema com a combinação das duas consultas, *inserindo* uma na outra.

A consulta interna (ou *subconsulta*) retorna um valor usado pela consulta externa (ou *consulta principal*). Usar uma subconsulta é o mesmo que executar duas consultas em sequência e utilizar o resultado da primeira como o valor de pesquisa na segunda.

## + Sintaxe da Subconsulta

```
SELECT  select_list
FROM    table
WHERE   expr operator
        (SELECT  select_list
         FROM    table);
```

- A subconsulta (consulta interna) é executada uma vez antes da consulta principal (consulta externa).
- O resultado da subconsulta é usado pela consulta principal.

1-6

### Sintaxe da Subconsulta

Uma subconsulta é uma instrução `SELECT` incorporada a uma cláusula de outra instrução `SELECT`. É possível criar instruções complexas a partir de instruções simples usando subconsultas. Elas poderão ser muito úteis quando for necessário selecionar linhas de uma tabela com uma condição que dependa dos dados da própria tabela.

É possível inserir a subconsulta em várias cláusulas SQL, inclusive nestas:

Cláusula `WHERE`

Cláusula `HAVING`

Cláusula `FROM`

Na sintaxe:

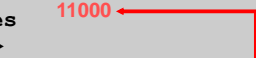
*operator* inclui uma condição de comparação, como `>`, `=` ou `IN`

**Observação:** As condições de comparação estão incluídas em duas classes: operadores de uma única linha (`>`, `=`, `>=`, `<`, `<>`, `<=`) e de várias linhas (`IN`, `ANY`, `ALL`).

Em geral, a subconsulta é denominada instrução `SELECT` interna, sub-`SELECT` ou `SELECT` aninhada. A subconsulta normalmente é executada primeiro e o seu resultado é usado para concluir a condição da consulta principal (ou externa).

## Usando uma Subconsulta

```
SELECT last_name  
FROM employees  
WHERE salary >  
      (SELECT salary  
       FROM employees  
       WHERE last_name = 'Abel');
```



	LAST_NAME
1	King
2	Kochhar
3	De Haan
4	Greenberg
5	Russell
6	Partners
7	Errazuriz
8	Ozer
9	Hartstein
10	Higgins

1-7

### Usando uma Subconsulta

No slide, a consulta interna determina o salário do funcionário Abel. A consulta externa recebe o resultado da consulta interna e o utiliza para exibir todos os funcionários que ganham mais que essa quantia.

## Diretrizes de Uso de Subconsultas

- Delimite subconsultas por parênteses.
- Posicione subconsultas à direita da condição de comparação.
- A cláusula `ORDER BY` não será necessária na subconsulta, a menos que uma análise Top-N seja executada.
- Use operadores de uma única linha com subconsultas de uma única linha e operadores de várias linhas com subconsultas de várias linhas.

1-8

### Diretrizes de Uso de Subconsultas

Uma subconsulta deve ser delimitada por parênteses.

Posicione a subconsulta à direita da condição de comparação para fins de legibilidade.

No Oracle8i e em releases mais recentes, uma cláusula `ORDER BY` pode ser usada, além de ser obrigatória na subconsulta para executar uma análise Top-N.

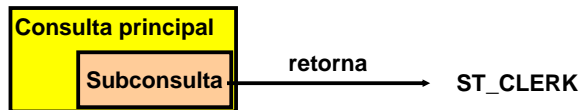
Entretanto, antes do Oracle8i, as subconsultas não podiam conter uma cláusula `ORDER BY`. Apenas uma cláusula `ORDER BY` era permitida em uma instrução `SELECT` e, se fosse especificada, deveria ser a última cláusula na instrução `SELECT` principal.

Duas classes de condições de comparação são usadas em subconsultas: operadores de uma única linha e operadores de várias linhas.

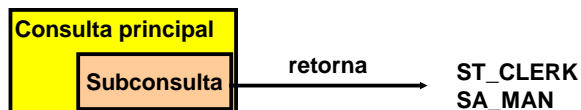


## Tipos de Subconsultas

- Subconsulta de uma única linha



- Subconsulta de várias linhas



1-9

### Tipos de Subconsultas

Subconsultas de uma única linha: consultas que retornam somente uma linha da instrução `SELECT` interna

Subconsultas de várias linhas: consultas que retornam mais de uma linha da instrução `SELECT` interna

**Observação:** Há também subconsultas de várias colunas, ou seja, consultas que retornam mais de uma coluna da instrução `SELECT` interna. Elas são abordadas no curso *Banco de Dados Oracle 10g: Fundamentos de SQL II*.

## Subconsultas de uma Única Linha

- Retornam somente uma linha
- Usam operadores de comparação de uma única linha

Operado	Significado
=	Igual a
>	Maior que
>=	Maior que ou igual a
<	Menor que
<=	Menor que ou igual a
<>	Diferente de

1-10

### Subconsultas de uma Única Linha

Uma subconsulta de uma única linha retorna uma linha da instrução `SELECT` interna. Esse tipo de subconsulta usa um operador de uma única linha. O slide fornece uma lista de operadores de uma única linha.

#### Exemplo

Exiba os funcionários cujo ID de cargo é igual ao do funcionário 141:

```
SELECT last_name, job_id
FROM employees
WHERE job_id =
      (SELECT job_id
       FROM employees
       WHERE employee_id = 141);
```

LAST_NAME	JOB_ID
Rajs	ST_CLERK
Davies	ST_CLERK
Matos	ST_CLERK
Vargas	ST_CLERK

## Executando Subconsultas de uma Única Linha

```
SELECT last_name, job_id, salary
FROM employees
WHERE job_id = (SELECT job_id
                FROM employees
                WHERE employee_id = 141)
AND salary > (SELECT salary
              FROM employees
              WHERE employee_id = 143);
```

	LAST_NAME	JOB_ID	SALARY
1	Naver	ST CLERK	3200
2	Mikkilineni	ST CLERK	2700
...			
10	Rajs	ST CLERK	3500
11	Davies	ST CLERK	3100

1-11

### Executando Subconsultas de uma Única Linha

Uma instrução **SELECT** pode ser considerada um bloco de consulta. O exemplo do slide mostra os funcionários cujo ID de cargo é igual ao do funcionário 141 e cujo salário é maior que o do funcionário 143.

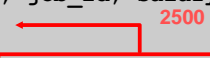
O exemplo consiste em três blocos de consulta: a consulta externa e duas consultas internas. Os blocos de consulta interna são executados primeiro, produzindo os resultados **ST\_CLERK** e **2600**, respectivamente. O bloco de consulta externa é processado depois e usa os valores retornados pelas consultas internas para concluir suas condições de pesquisa.

As duas consultas internas retornam valores únicos (**ST\_CLERK** e **2600**, respectivamente); por isso, essa instrução SQL é denominada subconsulta de uma única linha.

**Observação:** As consultas internas e externas podem obter dados de tabelas distintas.

## Usando Functions de Grupo em uma Subconsulta

```
SELECT last_name, job_id, salary
FROM   employees
WHERE  salary = (SELECT MIN(salary)
                 FROM   employees);
```



LAST_NAME	JOB_ID	SALARY
Olson	ST CLERK	2100

1-12

### Usando Functions de Grupo em uma Subconsulta

É possível exibir dados de uma consulta principal usando uma function de grupo em uma subconsulta para retornar uma única linha. A subconsulta é delimitada por parênteses e é posicionada após a condição de comparação. O exemplo do slide mostra o sobrenome, o ID do cargo e o salário de todos os funcionários cujo salário é igual ao salário mínimo. A function de grupo MIN retorna um único valor (2500) para a consulta externa.

## A Cláusula HAVING com Subconsultas

- O servidor Oracle executa primeiro as subconsultas.
- Ele retorna os resultados para a cláusula HAVING da consulta principal.

```
SELECT department_id, MIN(salary)
FROM employees
GROUP BY department_id
HAVING MIN(salary) > (SELECT MIN(salary)
FROM employees
WHERE department_id = 50);
```

DEPARTMENT_ID	MIN(SALARY)
1	90 17000
2	60 4200
...	
10	100 6900
11	40 6500

1-13

### A Cláusula HAVING com Subconsultas

É possível usar subconsultas tanto na cláusula WHERE como na cláusula HAVING. O servidor Oracle executa a subconsulta, e os resultados são retornados para a cláusula HAVING da consulta principal. A instrução SQL do slide mostra todos os departamentos cujo salário mínimo é maior que o do departamento 50.

DEPARTMENT_ID	MIN(SALARY)
10	4400
20	6000
...	
	7000

7 rows selected.

```
FROM employees
GROUP BY job_id
HAVING AVG(salary) = (SELECT MIN(AVG(salary))
FROM employees
GROUP BY job_id);
```

## O Que Está Errado Nesta Instrução?

```
SELECT employee_id, last_name
FROM employees
WHERE salary =
    (SELECT MIN(salary)
     FROM employees
     GROUP BY department_id);
```

```
ERROR at line 4:
ORA-01427: single-row subquery returns more than
one row
```

**Operador de uma única linha com  
uma subconsulta de várias linhas**

1-14

### Erros com Subconsultas

Um erro comum com subconsultas ocorre quando mais de uma linha é retornada para uma subconsulta de uma única linha.

Na instrução SQL do slide, a subconsulta contém uma cláusula `GROUP BY`, o que significa que ela retornará várias linhas, uma para cada grupo localizado. Nesse caso, os resultados da subconsulta são 4400, 6000, 2500, 4200, 7000, 17000 e 8300.

A consulta externa recebe esses resultados e os utiliza em sua cláusula `WHERE`. A cláusula `WHERE` contém um operador de igual a (`=`), que é um operador de comparação de uma única linha que espera apenas um valor. Esse operador não pode aceitar mais de um valor da subconsulta e, portanto, gera o erro.

Para corrigir esse erro, altere o operador `=` para `IN`.

## Esta Instrução Retornará Linhas?

```
SELECT last_name, job_id
FROM employees
WHERE job_id =
    (SELECT job_id
     FROM employees
     WHERE last name = 'Haas');
```

no rows selected

A subconsulta não retorna nenhum valor.

1-15

### Problemas com Subconsultas

Um problema comum com subconsultas ocorre quando nenhuma linha é retornada pela consulta interna.

Na instrução SQL do slide, a subconsulta contém uma cláusula `WHERE`.

Presumivelmente, o objetivo é localizar o funcionário cujo nome é Haas. A

instrução está correta, mas não seleciona nenhuma linha quando é executada.

Não há nenhum funcionário com esse nome. Por isso, a subconsulta não

retorna nenhuma linha. A consulta externa recebe o resultado da subconsulta

(nulo) e o utiliza em sua cláusula `WHERE`. Ela não encontra nenhum

funcionário com o ID de cargo igual a nulo e, portanto, não retorna nenhuma

linha. Se existisse um cargo com o valor nulo, a linha não seria retornada

porque a comparação de dois valores nulos resulta em um valor nulo; assim, a

condição `WHERE` não é verdadeira.

## Subconsultas de Várias Linhas

- Retornam mais de uma linha
- Usam operadores de comparação de várias linhas

Operador	Significado
IN	Igual a qualquer membro da lista
ANY	Compara o valor com cada valor retornado pela subconsulta
ALL	Compara o valor com todos os valores retornados pela subconsulta

1-16

### Subconsultas de Várias Linhas

As subconsultas que retornam mais de uma linha são denominadas subconsultas de várias linhas. Com uma subconsulta de várias linhas, você usa um operador de várias linhas, em vez de um operador de uma única linha. O operador de várias linhas espera um ou mais valores:

```
SELECT last_name, salary, department_id
FROM employees
WHERE salary IN (SELECT MIN(salary)
                 FROM employees
                 GROUP BY department_id);
```

#### Exemplo

Localize os funcionários cujo salário é igual ao salário mínimo em cada departamento.

A consulta interna é executada primeiro e produz um resultado. O bloco de consulta principal é processado depois e usa os valores retornados pela consulta interna para concluir sua condição de pesquisa. Na verdade, a consulta principal aparece para o servidor Oracle da seguinte forma:

```
SELECT last_name, salary, department_id
FROM employees
WHERE salary IN (2500, 4200, 4400, 6000, 7000, 8300,
                8600, 17000);
```



## Usando o Operador ANY em Subconsultas de Várias Linhas

```
SELECT employee_id, last_name, job_id, salary
FROM   employees
WHERE  salary < ANY
      (SELECT salary
       FROM   employees
       WHERE  job_id = 'IT_PROG')
AND    job_id <> 'IT_PROG';
```

	EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
1	132	Olson	ST CLERK	2100
2	136	Philtanker	ST CLERK	2200
3	128	Markle	ST CLERK	2200
4	135	Gee	ST CLERK	2400

...

1-17

### Subconsultas de Várias Linhas (continuação)

O operador **ANY** (e seu sinônimo, o operador **SOME**) compara um valor a *cada* valor retornado por uma subconsulta. O exemplo do slide mostra os funcionários que não são programadores de computação e cujo salário é menor que o de qualquer programador de computação. O salário máximo de um programador é US\$9.000.

<ANY significa menor que o máximo. >ANY significa maior que o mínimo.

=ANY equivale a IN.

## Usando o Operador ALL em Subconsultas de Várias Linhas

```
SELECT employee_id, last_name, job_id, salary
FROM   employees
WHERE  salary < ALL
      (SELECT salary
       FROM   employees
       WHERE  job_id = 'IT_PROG')
AND    job_id <> 'IT_PROG';
```

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
1	185 Bull	SH CLERK	4100
2	192 Bell	SH CLERK	4000
3	193 Everett	SH CLERK	3900
4	188 Chung	SH CLERK	3800

...

1-18

### Subconsultas de Várias Linhas (continuação)

O operador ALL compara um valor com *todos* os valores retornados por uma subconsulta. O exemplo do slide mostra os funcionários cujo salário é menor que o salário de todos os funcionários com o ID de cargo IT\_PROG e cujo cargo não é IT\_PROG.

>ALL significa maior que o máximo e <ALL significa menor que o mínimo. É possível usar o operador NOT com os operadores IN, ANY e ALL.

## Valores Nulos em uma Subconsulta

- A instrução SQL do slide tenta exibir todos os funcionários sem subordinados...

```
SELECT emp.last_name
FROM   employees emp
WHERE  emp.employee_id NOT IN
      (SELECT mgr.manager_id
       FROM   employees mgr);

no rows selected
```

- ... E nenhuma linha é retornada.

1-19

### Retornando Valores Nulos no Conjunto Resultante de uma Subconsulta

A instrução SQL do slide tenta exibir todos os funcionários sem subordinados. Logicamente, essa instrução deveria ter retornado 12 linhas. No entanto, ela não retorna nenhuma linha. Um dos valores retornados pela consulta interna é um valor nulo e, por isso, a consulta inteira não retorna nenhuma linha.

O motivo é que todas as condições que comparam um valor nulo resultam em um valor nulo. Dessa forma, sempre que houver a possibilidade de valores nulos integrarem o conjunto de resultados de uma subconsulta, não use o operador `NOT IN`. Esse operador corresponde a `<> ALL`.

Observe que o valor nulo como parte do conjunto de resultados de uma subconsulta não representa um problema se você usa o operador `IN`. Esse operador corresponde a `=ANY`. Por exemplo, para exibir os funcionários com subordinados, use a seguinte instrução SQL:

```
SELECT emp.last_name
FROM   employees emp
WHERE  emp.employee_id IN
      (SELECT mgr.manager_id
       FROM   employees mgr);
```

Retornando Valores Nulos no Conjunto Resultante de uma Subconsulta (continuação)

Como alternativa, é possível incluir uma cláusula `WHERE` na subconsulta para exibir todos os funcionários sem subordinados:

```
SELECT last_name FROM employees
WHERE employee_id NOT IN
    (SELECT manager_id
     FROM employees
     WHERE manager_id IS NOT NULL);
```

## Sumário

Nesta lição, você aprendeu a:

- Identificar quando uma subconsulta pode ajudar a solucionar um problema
- Criar subconsultas quando uma consulta se basear em valores desconhecidos

```
SELECT  select_list
FROM    table
WHERE   expr operator
        (SELECT select_list
         FROM    table);
```

1-21

### Sumário

Nesta lição, você aprendeu a usar subconsultas. Uma subconsulta é uma instrução `SELECT` incorporada a uma cláusula de outra instrução SQL. As subconsultas são úteis quando uma consulta se baseia em um critério de pesquisa com valores intermediários desconhecidos.

As subconsultas apresentam as seguintes características:

Podem passar uma linha de dados para uma instrução principal que contém um operador de uma única linha, como `=`, `<>`, `>`, `>=`, `<` ou `<=`

Podem passar várias linhas de dados para uma instrução principal que contém um operador de várias linhas, como `IN`

São processadas primeiro pelo servidor Oracle. Em seguida, a cláusula `WHERE` ou `HAVING` utiliza os resultados

Podem conter funções de grupo

## Exercício: Visão Geral

Este exercício aborda os seguintes tópicos:

- Criação de subconsultas para consultar valores baseados em critérios desconhecidos
- Utilização de subconsultas para descobrir os valores existentes em um conjunto de dados, e não em outro

1-22

### Exercício: Visão Geral

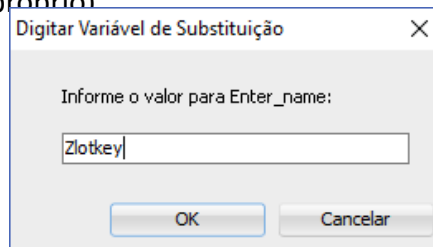
Neste exercício, você criará consultas complexas com instruções `SELECT` aninhadas.

#### Perguntas Impressas

Convém criar a consulta interna primeiro para estas perguntas. Certifique-se de que ela seja executada e gere os dados previstos antes de você codificar a consulta externa.

## Exercício

1. O departamento de recursos humanos precisa de uma consulta que solicite ao usuário o sobrenome de um funcionário. A consulta exibe o sobrenome e a data de admissão de todos os funcionários no mesmo departamento do funcionário cujo nome foi fornecido (excluindo esse funcionário). Por exemplo, se o usuário informar Zlotkey, serão exibidos todos os funcionários que trabalham com Zlotkey (excluindo ele próprio)



Digitar Variável de Substituição

Informe o valor para Enter\_name:

Zlotkey

OK Cancelar

	LAST_NAME	HIRE_DATE
1	Russell	01/10/96
2	Partners	05/01/97
3	Errazuriz	10/03/97
4	Cambrault	15/10/99
...		
29	Abel	11/05/96
30	Hutton	19/03/97
31	Taylor	24/03/98
32	Livingston	23/04/98
33	Johnson	04/01/00

2. Crie um relatório que exiba o número e o sobrenome de todos os funcionários cujo salário é maior que o salário médio. Classifique os resultados em ordem crescente de salário.

	EMPLOYEE_ID	LAST_NAME	SALARY
1	123	Vollman	6500
2	203	Mavris	6500
3	165	Lee	6800
4	113	Popp	6900
...			
48	145	Russell	14000
49	102	De Haan	17000
50	101	Kochhar	17000
51	100	King	24000

### Exercício (continuação)

3. Crie uma consulta que exiba o número e o sobrenome de todos os funcionários que trabalham em um departamento com funcionários cujos sobrenomes contêm a letra *u*. Inclua a instrução SQL no arquivo de texto `lab_06_03.sql`. Execute a consulta.

	EMPLOYEE_ID	LAST_NAME
1	107	Lorentz
2	106	Pataballa
3	105	Austin
4	104	Ernst
...		
87	117	Tobias
88	116	Baida
89	115	Khoo
90	114	Raphaely

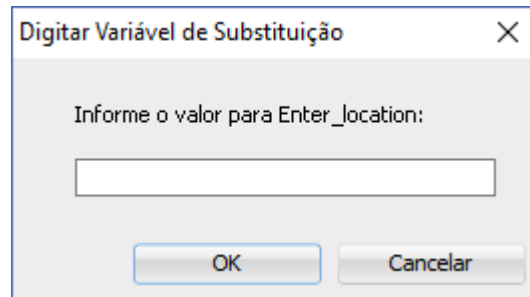
4. O departamento de recursos humanos precisa de um relatório que exiba o sobrenome, o número do departamento e o ID do cargo de todos os funcionários cujo ID de local do departamento é 1700.

	LAST_NAME	DEPARTMENT_ID	JOB_ID
1	King	90	AD_PRES
2	Kochhar	90	AD_VP
3	De Haan	90	AD_VP
4	Greenberg	100	FI_MGR
5	Faviet	100	FI_ACCOUNT
6	Chen	100	FI_ACCOUNT
7	Sciarra	100	FI_ACCOUNT
8	Urman	100	FI_ACCOUNT
9	Popp	100	FI_ACCOUNT
10	Raphaely	30	PU_MAN
11	Khoo	30	PU_CLERK
12	Baida	30	PU_CLERK
13	Tobias	30	PU_CLERK
14	Himuro	30	PU_CLERK
15	Colmenares	30	PU_CLERK
16	Whalen	10	AD_ASST
17	Higgins	110	AC_MGR
18	Gietz	110	AC_ACCOUNT



## Exercício (continuação)

Modifique a consulta para que um ID de local seja solicitado ao usuário. Salve-a no arquivo `lab_06_04.sql`.



5. Crie um relatório para o departamento de recursos humanos que exiba o sobrenome e o salário de todos os funcionários subordinados a Steven King.

	LAST_NAME	SALARY
1	Kochhar	17000
2	De Haan	17000
3	Raphaely	11000
4	Weiss	8000
5	Fripp	8200
6	Kaufling	7900
7	Vollman	6500
8	Mourgos	5800
9	Russell	14000
10	Partners	13500
11	Errazuriz	12000
12	Cambrault	11000
13	Zlotkey	10500
14	Hartstein	13000

6. Crie um relatório para o departamento de recursos humanos que exiba o número do departamento, o sobrenome e o ID do cargo de todos os funcionários no departamento executivo.

	DEPARTMENT_ID	LAST_NAME	JOB_ID
1	90	King	AD_PRES
2	90	Kochhar	AD_VP
3	90	De Haan	AD_VP



## Exercício (continuação)

7. Modifique a consulta em `lab_06_03.sql` para exibir o número, o sobrenome, bem como o salário de todos os funcionários que ganham mais que o salário médio e trabalham em um departamento com funcionários cujos sobrenomes contêm a letra *u*. Salve novamente `lab_06_03.sql` como `lab_06_07.sql`. Execute a instrução em `lab_06_07.sql`.

	EMPLOYEE_ID	LAST_NAME	SALARY
1	103	Hunold	9000
2	123	Vollman	6500
3	122	Kaufling	7900
4	121	Fripp	8200
...			
33	147	Errazuriz	12000
34	146	Partners	13500
35	145	Russell	14000
36	114	Raphaely	11000

## Bibliografia Utilizada

Database SQL Language Reference: <http://docs.oracle.com/database/121/SQLRF/toc.htm>

Manuais Oracle – Oracle Database 12c: SQL Workshop I/II

*Esta apresentação possui material de referência com propriedade da Oracle.  
Copyright © , Oracle. Todos os direitos reservados.*



OBRIGADO



profalexandre.barcelos@fiap.com.br



<https://www.linkedin.com/in/alexandrebarcelos>

FIAP

Copyright © 2022 | Professor Me. Alexandre Barcelos  
Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente  
proibido sem consentimento formal, por escrito, do professor/autor.

