

Utilizando Subquery

Subconsulta (*Subquery*)

- Uma *Subquery* é uma consulta aninhada dentro de uma query
- Normalmente usada para verifica se irá exibir uma determinada linha de acordo com o resultado da *query* aninhada
- Pode existir ligação (correlação) de colunas do SELECT mais externo com o SELECT mais interno

Subconsulta (*Subquery*)

Listar o nome do empregado com o maior salário

```
SELECT e.nom_empregado  
FROM empregado AS e  
WHERE e.val_salario =  
      (SELECT MAX(val_salario) FROM empregado)
```

	nom_empregado
1	José da Silva

Subconsulta (*Subquery*)

Listar o nome dos empregados que ganham o **maior** salário do seu respectivo departamento

```
SELECT e.nom_empregado,  
       e.val_salario,  
       e.cod_depto  
FROM empregado e  
WHERE e.val_salario =  
      (SELECT MAX(val_salario)  
       FROM empregado  
       WHERE cod_depto = e.cod_depto)
```

	nom_empregado	val_salario	cod_depto
1	Marcílio Costa	2000.00	4
2	Antônio Soares	1500.00	3
3	Maria Batista	2800.00	2
4	José da Silva	4500.00	1

Subconsulta (*Subquery*)

Listar o nome dos empregados que ganham mais do que a **média** de salário do seu departamento

```
SELECT e.nom_empregado, e.val_salario, e.cod_depto
FROM empregado e
WHERE e.val_salario >
      (SELECT AVG(val_salario)
       FROM empregado
       WHERE cod_depto = e.cod_depto)
```

Subconsulta (*Subquery*)

Listar o nome dos empregados que ganham mais do que a média de salário do seu departamento

	cod_depto	media
1	NULL	500.000000
2	1	2200.000000
3	2	2250.000000
4	3	1500.000000
5	4	2000.000000



	nom_empregado	val_salario	cod_depto
1	José da Silva	4500.00	1
2	João Oliveira	2500.00	1
3	Maria Batista	2800.00	2

Subconsulta (*Subquery*)

- Outros operadores

- **IN**

- Verificar se o valor de uma ou mais colunas estão contidas no resultado de uma *subquery*

- **EXISTS**

- Verificar se existe pelo menos uma linha retornada pela *subquery* com campos correlacionados

IN / NOT IN

Listar o nome dos empregados que possuem dependentes

Empregado

	num_matricula	nom_empregado
1	1	José da Silva
2	2	João Oliveira
3	3	Maria Batista
4	4	Rodrigo Moreira
5	5	Renata Carvalho
6	6	Alice Borboleta
7	7	Antônio Soares
8	8	Marcílio Costa
9	9	Novo empregado

Dependente

	num_matricula	nom_dependente
1	1	Huguinho
2	1	Luizinho
3	1	Margarida
4	1	Zezinho
5	3	Kelly Key
6	4	Luciana Melo
7	4	Sandra de Sá

IN / NOT IN

Listar o nome dos empregados que possuem dependentes

```
SELECT e.nom_empregado
```

```
FROM empregado AS e
```

```
WHERE e.num_matricula IN
```

```
(SELECT num_matricula FROM dependente)
```

Listar o nome dos empregados que possuem dependentes

Empregado

	num_matricula	nom_empregado
1	1	José da Silva
2	2	João Oliveira
3	3	Maria Batista
4	4	Rodrigo Moreira
5	5	Renata Carvalho
6	6	Alice Borboleta
7	7	Antônio Soares
8	8	Marcílio Costa
9	9	Novo empregado

IN

Dependente

	num_matricula
1	1
2	1
3	1
4	1
5	3
6	4
7	4

=

	nom_empregado
1	José da Silva
2	Maria Batista
3	Rodrigo Moreira

IN / NOT IN

Listar o nome dos empregados que não possuem dependentes

```
SELECT e.nom_empregado  
FROM empregado AS e  
WHERE e.num_matricula NOT IN  
      (SELECT num_matricula FROM dependente)
```

Listar o nome dos empregados que não possuem dependentes

Empregado

	num_matricula	nom_empregado
1	1	José da Silva
2	2	João Oliveira
3	3	Maria Batista
4	4	Rodrigo Moreira
5	5	Renata Carvalho
6	6	Alice Borboleta
7	7	Antônio Soares
8	8	Marcílio Costa
9	9	Novo empregado

NOT IN

Dependente

	num_matricula
1	1
2	1
3	1
4	1
5	3
6	4
7	4

=

	nom_empregado
1	Alice Borboleta
2	Antônio Soares
3	João Oliveira
4	Marcílio Costa
5	Novo empregado
6	Renata Carvalho

EXISTS / NOT EXISTS

Testa a existência de linhas na subquery

- EXISTS retorna
 - Falso, se não encontrou registros
 - Verdadeiro, se encontrou um ou mais registros
- Pode substituir o IN
- Os campos selecionados no SELECT interno não são relevantes quando se usa EXISTS
- Geralmente usado com campos correlacionados

EXISTS / NOT EXISTS

Listar o nome dos empregados que possuem dependente

- SELECT e.nom_empregado
FROM empregado as e
WHERE e.num_matricula **IN** (
SELECT num_matricula FROM dependente)
- SELECT e.nom_empregado
FROM empregado AS e
WHERE **EXISTS** (
SELECT * FROM dependente AS d
WHERE d.num_matricula = e.num_matricula)
- SELECT **DISTINCT** e.nom_empregado
FROM empregado as e
JOIN dependente as d ON e.num_matricula = d.num_matricula

EXISTS / NOT EXISTS

Listar o nome dos empregados que não possuem dependente

- SELECT e.nom_empregado
FROM empregado as e
WHERE e.num_matricula **NOT IN** (
 SELECT num_matricula FROM dependente)
- SELECT e.nom_empregado
FROM empregado AS e
WHERE **NOT EXISTS** (
 SELECT * FROM dependente AS d
 WHERE d.num_matricula = e.num_matricula)
- SELECT **DISTINCT** e.nom_empregado
FROM empregado as e
LEFT JOIN dependente as d ON e.num_matricula = d.num_matricula
WHERE d.nom_dependente IS NULL

EXISTS / NOT EXISTS

Listar o nome dos empregados que não possuem dependente

- SELECT e.nom_empregado
FROM empregado as e
WHERE e.num_matricula **NOT IN** (
SELECT num_matricula FROM dependente)
- SELECT e.nom_empregado
FROM empregado AS e
WHERE **NOT EXISTS** (
SELECT * FROM dependente AS d
WHERE d.num_matricula = e.num_matricula)
- SELECT **DISTINCT** e.nom_empregado
FROM empregado as e
LEFT JOIN dependente as d ON e.num_matricula = d.num_matricula
WHERE d.nom_dependente IS NULL

- Somente 1 campo na comparação
- Não deixa trazer campos no select
- Cuidado com comparativos com NULL

- Pode usar vários campos na comparação
- Não permite trazer campo no select

- Pode usar vários campos na comparação
- Deixa trazer campo no select
- Cuidado com possíveis duplicidades

Subquery como tabela

Listar a matrícula e o nome dos empregados

```
SELECT x.num_matricula,  
       x.nom_empregado  
FROM (SELECT *  
      FROM empregado) as x
```



No lugar da tabela pode ser usada uma outra consulta!

	num_matricula	nom_empregado
1	1	José da Silva
2	2	João Oliveira
3	3	Maria Batista
4	4	Rodrigo Moreira
5	5	Renata Carvalho
6	6	Alice Borboleta
7	7	Antônio Soares
8	8	Marcílio Costa
9	9	Novo empregado

Subquery como tabela

Listar o nome dos empregados que ganham o maior salário do seu respectivo departamento

```
SELECT e.nom_empregado,  
       e.val_salario,  
       e.cod_depto  
FROM empregado AS e  
JOIN (SELECT cod_depto,  
             MAX(val_salario) AS maior  
      FROM empregado  
      GROUP BY cod_depto) as M  
ON e.cod_depto = M.cod_depto  
WHERE e.val_salario = M.maior
```

	nom_empregado	val_salario	cod_depto
1	Marcílio Costa	2000.00	4
2	Antônio Soares	1500.00	3
3	Maria Batista	2800.00	2
4	José da Silva	4500.00	1

Subquery como coluna

Listar o nome do empregado e o total de dependentes de cada um deles (exibir todos os empregado)

```
SELECT e.nom_empregado,  
       (SELECT COUNT(*)  
        FROM dependente d  
        WHERE d.num_matricula = e.num_matricula)  
       AS total  
FROM empregado e
```

	nom_empregado	total
1	José da Silva	4
2	João Oliveira	0
3	Maria Batista	1
4	Rodrigo Moreira	2
5	Renata Carvalho	0
6	Alice Borboleta	0
7	Antônio Soares	0
8	Marcílio Costa	0
9	Novo empregado	0



PUC Minas
Virtual