

Unidade 5.2: DML – CONSULTAS ANINHADAS COM CORRELAÇÃO (SUBSELECTs)

Prof. Rodrigo Baroni

SUBSELECTs com Correlação

- É o caso mais complexo onde existe ligação (correlação) de colunas do SELECT mais externo com o SELECT mais interno.
- O Select interno depende do externo para ser executado.
- O Select externo seleciona um registro e o Select interno é executado.
- O Select externo seleciona o próximo registro e o Select interno é executado novamente. Isso se repete até o Select externo chegar ao fim dos registros.

Subselect com Correlação

- Quando condição na cláusula WHERE de uma consulta aninhada referencia algum atributo de uma relação declarada na consulta externa, as duas consultas são consideradas correlacionadas.
- A consulta aninhada é avaliada uma vez para cada linha na consulta externa.
- Muito usado na comparação de elementos em relação ao seu grupo
- Evita comparações distorcidas.
- Usualmente é mais caro em termos de performance, mas em determinados casos é a única solução possível

Subconsulta Correlacionada

```
SELECT Pnome, dnr  
FROM funcionario F  
WHERE salario > (SELECT AVG(salario)  
                  FROM funcionario  
                  WHERE dnr= F.dnr);
```

Retorna os funcionários que possuem salário maior que a media dos salários do seu departamento.

Execução

- 1) Pega a linha candidata (da consulta externa)
- 2) Executa a consulta interna usando o valor da linha candidata.
- 3) Usa o valor resultante da consulta interna para qualificar ou desqualificar a linha candidata.
- 4) Repetir enquanto houver linhas candidatas

Funcionario

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria

Depto

cod-depto	nome-depto
1	Informatica
2	Contabilidade

SUBSELECTs com Correlação

- Ex: Selecione os nomes dos funcionários que ganham o maior salário do seu **respectivo** departamento

Select nome, salario, cod_depto from Funcionario A

Where salario = (select max (salario) from funcionario B

Where **A.cod_depto=B.cod_depto**)

Pergunta: Quantas vezes o select interno foi executado ?

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria

Nome	salario	Cod_depto
Paulo	4000	1
Luiz	3500	2

SUBSELECTs com Correlação

- Ex: Selecione os nomes dos funcionários que ganham mais do que a média do seu respectivo departamento

Select nome, salario, cod_depto from Funcionario A

Where salario > (select avg (salario) from funcionario B Where **A.cod_depto=B.cod_depto**)

Média do depto 1 = R\$ 2875, média do depto 2 = R\$ 2450

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria

Nome	salario	Cod_depto
Paulo	4000	1
Luiz	3500	2
Joel	3200	1

Subselect com Correlação: EXISTS e NOT EXISTS

- Usados para verificar se o resultado de uma consulta aninhada correlacionada é vazio (não contém tuplas).

EXISTS

Resultado:

TRUE: quando a consulta aninhada retorna pelo menos uma tupla.

FALSE: quando a consulta aninhada não retorna tuplas.

NOT EXISTS

Resultado:

TRUE: quando a consulta aninhada não retorna tupla.

FALSE: quando a consulta aninhada retorna pelo menos uma tupla.

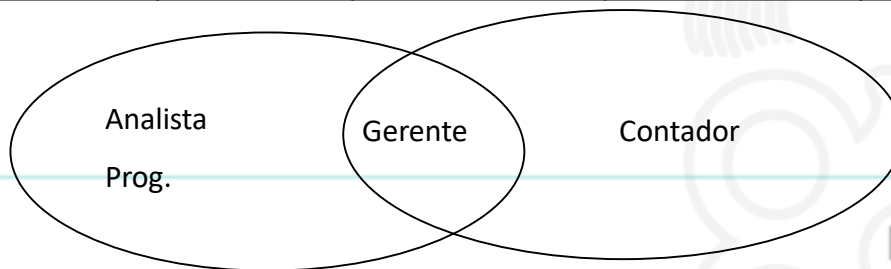
Subselect com Correlação: EXISTS e NOT EXISTS

- EXISTS: testa a existência de linhas que resultariam do subselect. Pode ser substituído pelo IN.
- EXISTS exige subselect com correlação pois testa a existência da linha inteira no subselect.
- O resultado de um subselect que sucede um EXISTS é tratado da seguinte forma:
 - Falso, se não encontrou registros
 - Verdadeiro, se encontrou um ou mais registros
- Os campos selecionados no select interno não são tão relevantes quando se usa EXISTS
- Usado para verificar se elementos de mesmo tipo pertencem ou não a conjuntos distintos

- Ex: Selecione os distintos cargos que existem no departamento 1, mas não existem no depto 2

Select distinct cargo from Funcionario A Where cod_depto = 1 and **not exists**
 (select distinct cargo from funcionario B
 Where cod_depto = 2 and **A.cargo=B.cargo**)

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria



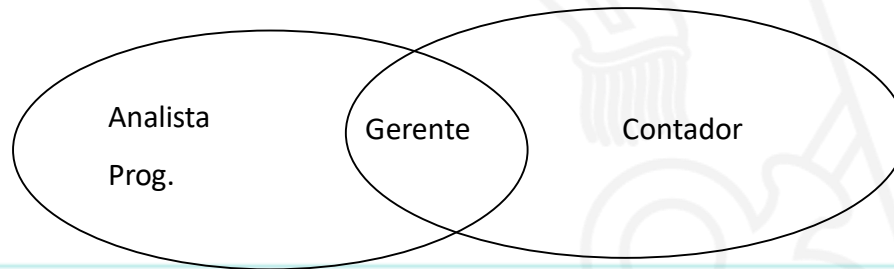
Cargo
analista
Prog.

- Ex: Selecione os distintos cargos que existem no departamento 1, mas não existem no depto 2
- EXISTS e NOT EXISTS podem ser substituídos por IN e NOT IN, dispensando a correlação na maioria das vezes

Select distinct cargo from Funcionario A
Where cod_depto = 1 and cargo NOT IN
(select distinct cargo from funcionario B
Where cod_depto = 2)

■ Resultado:

Cargo
analista
Prog.



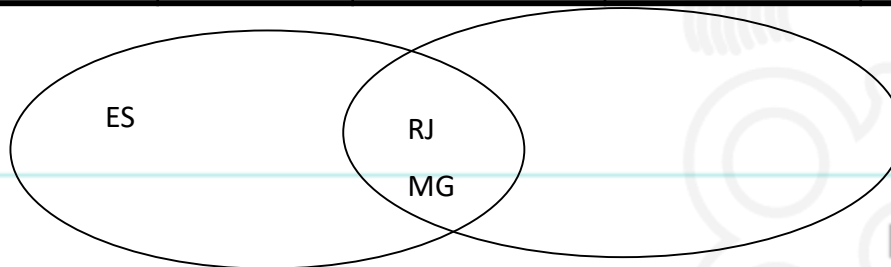
- Ex: Selecione os distintos estados que existem entre os funcionários do departamento de Informática e também existem no depto de Contabilidade

Select distinct estado from Funcionario A, Depto B Where A.Cod_depto = B.Cod_depto and nome_depto = 'Informatica' and exists

(Select distinct estado from Funcionario C, Depto D

Where C.Cod_depto = D.Cod_depto and nome_depto = 'Contabilidade' AND **C.Estado= A.Estado**)

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria



Estado
MG
RJ

- Ex: Selecione os distintos estados que existem entre os funcionários do departamento de Informática e também existem no depto de Contabilidade

Select distinct estado from Funcionario A, Depto B Where A.Cod_depto = B.Cod_depto and nome_depto = 'Informatica' and ESTADO IN

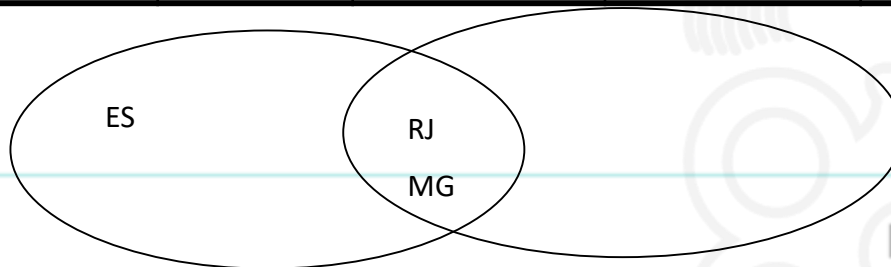
(Select distinct estado from Funcionario C, Depto D

Where C.Cod_depto = D.Cod_depto and nome_depto = 'Contabilidade')

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria

Resultado:

Estado
MG
RJ



ANY e IN

- ANY possui efeito bastante semelhante ao IN, diferenciando-se apenas na sintaxe
- Ex: Selecione os nomes dos departamentos que têm funcionários que trabalham no estado de MG

```
Select nome_depto from Depto Where cod_depto IN (select cod_depto from Funcionario  
where estado = 'MG')
```

```
Select nome_depto from Depto Where cod_depto = ANY (select cod_depto from Funcionario  
where estado = 'MG')
```

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria

Nome_depto
Informatica
Contabilidade

ANY e IN

- Ex: Selecione os nomes dos funcionários do depto 2 que ganham um salário maior do que qualquer funcionário do depto 1

Select nome, salario from Funcionario where cod_Depto = 2
and salario > ANY (select salario from Funcionario where cod_depto = 1)

Obs: Neste caso, o ANY não pode ser substituído por IN

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria

Nome	Salario
Luiz	3500

ALL

- ALL: compara um elemento contra todos de seu conjunto.
- Pode ser usado em substituição ao MAX em consultas analíticas
- Em consultas sintéticas (uso de Group By), deve-se usar o ALL quando se quer obter o maior ou menor de todos, pois não se aplica Max(sum(salario)), Max(count(*))

```
SELECT DEPTO, AVG (SALARIO)  
FROM FUNCIONARIOS  
GROUP BY DEPTO  
HAVING AVG(SALARIO) >= ALL  
(SELECT AVG(SALARIO)  
FROM FUNCIONARIOS  
GROUP BY DEPTO)
```

(Obs: departamento com a maior média salarial entre todos depts)

ALL

- Ex: Selecione o nome do funcionário que tem o maior salário dentre todos os funcionários.

Select nome, salario from Funcionario

Where salario >= ALL (Select salario from funcionario)

Também pode ser solucionado da seguinte forma

Select nome, salario from Funcionario

Where salario = (Select max(salario) from funcionario)

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria

Nome	Salario
Paulo	4000

- Ex: Selecione o código de departamento que têm o maior número de funcionários

Select cod_depto, Count(*) as QtdeFunc from Funcionario

Group by cod_depto Having Count (*) >= ALL

(Select count(*) from Funcionario Group by cod_depto)

Obs: Neste caso, são gerados dois grupos (select externo e interno) que são comparados entre si. Por isso, existe a necessidade do >= porque o elemento é comparado com os outros e com ele mesmo.

Obs: Reparar que o select interno tem apenas uma coluna para que seja possível a comparação. No select interno, é feito um group by pelo código do depto, que é exibido apenas no select externo.

Cod_Depto	QtdeFunc
1	4
2	2

QtdeFunc
4
2

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria

- Ex: Selecione o nome do departamento que têm a maior média de salários dentre todos departamentos

Select nome_depto, Avg(salario) as MediaSalario
 from Funcionario A, Depto B where A.cod_depto = B.cod_depto
 Group by nome_depto Having Avg (Salario) >= ALL
 (Select Avg(salario) as MediaSalario
 from Funcionario A, Depto B where A.cod_depto = B.cod_depto
 Group by nome_depto)

Nome_Depto	MediaSalario
1	2875
2	2450

MediaSalario
2875
2450

matricula	nome	cargo	salario	cod-depto	comissao	estado	cidade
10	Ana	analista	2800	1	100	null	null
20	Jose	prog.	1500	1	200	MG	BH
30	Paulo	gerente	4000	1	100	RJ	Rio
40	Lucas	contador	1400	2	300	RJ	Rio
50	Luiz	gerente	3500	2	200	MG	Betim
60	Joel	analista	3200	1	100	ES	Vitoria



PUC Minas
Virtual