



AS33C

Banco de Dados 2

Professor: Eduardo Cotrin Teixeira



cotrin@utfpr.edu.br



SQL – Subconsultas

- Um comando **SELECT-FROM-WHERE** entre **parênteses** pode ser usado como um valor em diferentes posições no Select, caracterizando uma **subconsulta**.
- Por exemplo, **no lugar de uma tabela na cláusula FROM**, pode ser usada uma subconsulta. Para isso, **faz-se necessário o uso de uma variável** para nomear os registros do resultado.



SQL – Subconsultas

Obtenha os nomes das peças utilizadas no projeto P4 mais caras que \$20.

```
SELECT Peca.PeNome
FROM Peca, (SELECT PeNro
            FROM Fornece_para
            WHERE Fornece_para.PNro = 'P4') PecasP4
WHERE Peca.PeNro = PecasP4.PeNro AND Peca.PePreco > 20;
```

PeNro
PE1
PE3

PeNome
Cinto



SQL – Subconsultas

- A subconsulta pode ser incluída tanto no **FROM** quanto no **WHERE**.
- Para uso no WHERE, o operador **IN** é útil.
Obtenha os nomes das peças com referência de cor vermelha ou amarela com preço de \$09, \$18, \$22, \$40 ou \$90.

```
SELECT PeNome
```

```
FROM Peca
```

```
WHERE ((PeCor = 'Vermelho' OR PeCor = 'Amarelo') AND  
       PePreco IN (09,18,22,40,90) );
```

PeNome
Volante
Limpador



SQL – Subconsultas

Obtenha os nomes das peças já utilizadas no projeto P5.

SELECT *PeNome*

FROM *Peca*

WHERE *PeNro* **IN** (

SELECT **DISTINCT** *PeNro*

FROM *Fornece_para*

WHERE *PNro* = 'P5');

PeNome
Limpador

PeNro
PE4

SQL – Subconsultas

Obtenha os nomes das peças já utilizadas nos projetos com duração maior que 3 meses.

```
SELECT PeNome
```

```
FROM Peca
```

```
WHERE PeNro IN (
```

```
    SELECT DISTINCT PeNro
```

```
    FROM Fornece_para
```

```
    WHERE PNro IN (
```

```
        SELECT PNro
```

```
        FROM Projeto
```

```
        WHERE PDuracao > 3) ) ;
```

PeNome
Painel

*Número das peças
utilizadas.*

PeNro
PE5

*Projetos com duração
maior que 3 meses.*

PNro
P1



SQL – Subconsultas

- A expressão:

EXISTS (<subconsulta>)

retorna verdadeiro se o resultado da subconsulta **não é vazio.**

```
SELECT *  
FROM TABELA1  
WHERE EXISTS (SELECT * FROM TABELA2)
```

**VAI RETORNAR A LINHA DA TABELA 1, SE A CONSULTA FEITA EM TABELA2
RETORNAR QUALQUER RESULTADO DIFERENTE DE NULO**

SQL – Subconsultas

- Retornar os **nomes** das peças que já foram fornecidas:
Percorrer a tabela Peca (onde estão os nomes), e verificar para cada linha se a peça EXISTE na tabela Fornece_para (onde estão os fornecimentos)

```
SELECT PeNome
FROM Peca
WHERE EXISTS (SELECT * FROM Fornece_para WHERE
Fornece_para.PeNro = Peca.PeNro)
```

PeNome
Cinto
Volante
Lanterna
Limpador
Painel

Tabela Peça

PeNro	...	PeNome
PE1	...	Cinto
PE2	...	Volante
PE3	...	Lanterna
PE4	...	Limpador
PE5	...	Painel

Tabela Fornece_para

PeNro	FNro	PNro	Quant
PE1	F5	P4	5
PE2	F2	P2	1
PE3	F3	P4	2
PE4	F4	P5	3
PE5	F1	P1	1
PE2	F2	P3	1
PE4	F3	P5	2



SQL – Subconsultas

Obtenha os nomes dos fornecedores cadastrados que ainda não realizaram nenhum fornecimento.

** Traduzindo: nomes dos fornecedores que não aparecem em Fornece_para.*

--Para testar:

```
insert into fornecedor values ('F6','Provider','Jaú','C');
```

```
SELECT FNome
```

```
FROM Fornecedor
```

```
WHERE NOT EXISTS (SELECT * FROM Fornece_para
```

```
WHERE Fornece_para.FNro=Fornecedor.FNro);
```

DÁ PRA RESOLVER COM “NOT IN”??



SQL – Subconsultas

Obtenha os nomes dos fornecedores cadastrados que ainda não realizaram nenhum fornecimento.

** Traduzindo: nomes dos fornecedores que não aparecem em Fornece_para.*

```
SELECT FNome  
FROM Fornecedor  
WHERE FNro NOT IN (SELECT FNro FROM Fornece_para)
```

SQL – Subconsultas

- O operador **ANY** compara com todo **o conjunto resultado da subconsulta.**

Obtenha os nomes dos fornecedores de outras cidades que são da mesma categoria dos fornecedores de Campinas.

```
SELECT FNome
FROM Fornecedor
WHERE FCidade <> 'Campinas' AND
      FCateg = ANY (
        SELECT FCateg
        FROM Fornecedor
        WHERE FCidade = 'Campinas');
```

FNome
Piloto's

*Categorias dos
fornecedores de
Campinas.*

FCateg
B
A

SQL – Subconsultas

- O operador **ALL** compara com todo **o conjunto resultado da subconsulta.**

Obtenha os nomes dos fornecedores de categoria pior ('A' é a melhor) que os fornecedores de Campinas.

```
SELECT FNome  
FROM Fornecedor
```

FNome
C&M
Equipament

```
WHERE FCateg > ALL (
```

```
    SELECT FCateg  
    FROM Fornecedor
```

```
    WHERE FCidade = 'Campinas');
```

FCateg
B
A

*Categorias dos
fornecedores de
Campinas.*



SQL – Consulta com múltiplas tabelas

- Ao invés de 2 tabelas diferentes, uma consulta pode precisar de **2 instâncias da mesma tabela**.
- Para diferenciar as instâncias, usamos **um nome de variável** na frente do nome da tabela na cláusula FROM.
- É sempre possível renomear uma tabela desta forma, mesmo quando isso não é indispensável para a consulta.



SQL – Consulta com múltiplas tabelas

Mostre os códigos das peças com referência de cor igual.

```
SELECT X.PeNro, Y.PeNro
FROM Peca AS X, Peca AS Y
WHERE (X.PeCor = Y.PeCor AND X.PeNro <> Y.PeNro);
```

X.PeNro	Y.PeNro
PE2	PE5
PE5	PE2

Mostre os códigos dos fornecedores concorrentes (que já forneceram os mesmos produtos).

```
SELECT X.FNro, Y.FNro
FROM Fornece_para AS X, Fornece_para AS Y
WHERE (X.PeNro = Y.PeNro AND X.FNro <> Y.FNro);
```

X.FNro	Y.FNro
F4	F3
F3	F4