



AS33C

Banco de Dados 2

Professor: Eduardo Cotrin Teixeira



cotrin@utfpr.edu.br



SQL – Consulta com múltiplas tabelas

- Como já vimos, consultas podem usar **colunas diferentes** de uma **mesma tabela** em suas cláusulas:

Obtenha os nomes dos projetos com preço entre \$20.000 e \$30.000.

```
SELECT PNome  
FROM Projeto  
WHERE PCusto BETWEEN 20000 AND 30000;
```

PNome
Alfa
Sea



SQL – Consulta com múltiplas tabelas

- Porém, consultas frequentemente combinam **dados de mais de uma tabela.**
- Várias tabelas podem ser consideradas em uma consulta, listando-as na cláusula **FROM.**
- **Lembrando:** para distinguir colunas com mesmo nome em tabelas diferentes, usamos:
<nome-tabela>.<nome-coluna>



SQL – Consulta com múltiplas tabelas

- Operadores de Junção só foram introduzidos no SQL2, mas é possível obter o mesmo resultado com consultas simples.

Obtenha os nomes das peças das quais já foram fornecidas mais que 2 unidades.

```
SELECT PeNome
FROM Fornece_para, Peca
WHERE Fornece_para.PeNro = Peca.PeNro AND
      Fornece_para.Quant > 2;
```

PeNome
Cinto
Limpador

SELECT *PeNome*

FROM *Fornece_para, Peca*

WHERE *Fornece_para.PeNro = Peca.PeNro AND*
Fornece_para.Quant > 2;

PeNome
Cinto
Limpador

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

FROM (CROSS JOIN)

PeNro	FNro	PNro	Quant	PeNro	PeNome
PE1	F5	P4	5	PE1	Cinto
PE1	F5	P4	5	PE2	Volante
PE1	F5	P4	5	PE3	Lanterna
PE1	F5	P4	5	PE4	Limpador
PE1	F5	P4	5	PE5	Painel
PE2	F2	P2	1	PE1	Cinto
PE2	F2	P2	1	PE2	Volante
PE2	F2	P2	1	PE3	Lanterna
...
PE4	F4	P5	3	PE4	Limpador

Tabela Peça

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

Linhas eliminadas pelas
condições no WHERE



SQL – Junções

- Há vários tipos de expressões de junção (JOIN) em SQL.
- A sintaxe geral é:

(<tabela>) [TIPO] **JOIN** (<tabela>)
[**ON** tabela1.campo=tabela2.campo];

* No lugar das tabelas também podem ser usadas subconsultas.

SQL – Junções

- JOIN sem operador: resultado é definido por campos explicitamente indicados nas duas tabelas.

Exemplo de junção (Qtd. de peças nos projetos):

```
SELECT PeNome, PNro, Quant
FROM Fornece_para, Peca
WHERE Fornece_para.PeNro=Peca.PeNro;
```

PeNome	PNro	Quant
Cinto	P4	5
Volante	P2	1
Lanterna	P4	2
Limpador	P5	3
Painel	P1	1
Volante	P3	1
Limpador	P5	2

FROM (CROSS JOIN)

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

PeNro	FNro	PNro	Quant	PeNro	PeNome
PE1	F5	P4	5	PE1	Cinto
PE1	F5	P4	5	PE2	Volante
PE1	F5	P4	5	PE3	Lanterna
PE1	F5	P4	5	PE4	Limpador
PE1	F5	P4	5	PE5	Painel
PE2	F2	P2	1	PE1	Cinto
PE2	F2	P2	1	PE2	Volante
PE2	F2	P2	1	PE3	Lanterna
...

Tabela Peça

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

SQL – Junções

- Outras formas de fazer a mesma junção:

** o desempenho é o mesmo*

EQUI JOIN:

```
SELECT PeNome, PNro, Quant
FROM Fornece_para JOIN Peca ON
    Fornece_para.PeNro = Peca.PeNro;
```

INNER JOIN:

```
SELECT PeNome, PNro, Quant
FROM Fornece_para INNER JOIN Peca
    USING (PeNro);
```

* OBS.: Pode ser usado **ON** ou **USING** nos dois casos.

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

PeNome	PNro	Quant
Cinto	P4	5
Volante	P2	1
Lanterna	P4	2
Limpador	P5	3
Painel	P1	1
Volante	P3	1
Limpador	P5	2

Tabela Peca

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

SQL – Junções

- Se os nomes dos campos envolvidos na junção for o mesmo, pode ser usado o **NATURAL JOIN**, que **não precisa especificar os nomes dos campos**:

NATURAL JOIN:

```
SELECT PeNome, PNro, Quant  
FROM Fornece_para NATURAL JOIN Peca;
```

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

PeNome	PNro	Quant
Cinto	P4	5
Volante	P2	1
Lanterna	P4	2
Limpador	P5	3
Painel	P1	1
Volante	P3	1
Limpador	P5	2

Tabela Peça

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

SQL – Junções - Exemplos

Obtenha os códigos dos fornecedores que já forneceram Limpador.

```
SELECT FNro
FROM Fornece_para NATURAL JOIN Peca
WHERE PeNome = 'Limpador';
-- OU
SELECT FNro
FROM Fornece_para JOIN Peca USING(PeNro)
WHERE PeNome = 'Limpador';
-- OU
SELECT FNro
FROM Fornece_para, Peca
WHERE Fornece_para.PeNro = Peca.PeNro AND
      PeNome = 'Limpador';
```

FNro
F4
F3



SQL – Junções - Exemplos

Obtenha os nomes das peças fornecidas pelos fornecedores de categoria A.

```
SELECT DISTINCT PeNome
FROM Peca, Fornecedor, Fornece_para
WHERE Fornece_para.PeNro = Peca.PeNro
AND Fornece_para.FNro = Fornecedor.FNro
AND FCateg = 'A';
```

-- OU

```
SELECT DISTINCT PeNome
FROM (Fornecedor NATURAL JOIN Fornece_para)
     NATURAL JOIN Peca
WHERE FCateg = 'A';
```

PeNome
Lanterna
Limpador



SQL – Junções

- **OUTER JOIN:** inclui no resultado registros que estão fora do critério definido pelos campos de junção.

Exemplo:

```
CREATE TABLE Empregado (
  RG CHAR(11) PRIMARY KEY,
  Endereco VARCHAR(30),
  Nome VARCHAR(30),
  Salario FLOAT,
  DepLotacao INT NOT NULL);
```

```
CREATE TABLE Departamento(
  NumId INT PRIMARY KEY,
  Nome VARCHAR(30),
  RGGerente CHAR(11));
```

```
ALTER TABLE Empregado ADD FOREIGN KEY (DepLotacao)
REFERENCES Departamento(NumId);
```

```
ALTER TABLE Departamento ADD FOREIGN KEY(RGGerente)
REFERENCES Empregado(RG);
```



SQL – Junções

```
SELECT Departamento.nome, Empregado.nome, salario
FROM Empregado RIGHT OUTER JOIN Departamento ON
(RG=RGgerente);
```

Empregado

RG	Nome	Salario	...
12345678-12	Fernando Paiva	4000,00	...
87654321-21	Ana Moreira	4200,00	...
11223344-55	Paulo Costa	2500,00	...
56781234-11	Pedro Faria	5000,00	...
43218765-22	Maria Maia	3300,00	...

Departamento

NumID	Nome	RGGerente
1	Computação	12345678-12
2	Automação	87654321-21
3	Elétrica	56781234-11
4	Mecânica	null

RIGHT OUTER JOIN

Dep.Nome	Emp.Nome	Salario
Computação	Fernando Paiva	4000
Automação	Ana Moreira	4200
Elétrica	Pedro Faria	5000
Mecânica	NULL	NULL

SQL – Junções

```
SELECT Departamento.nome, Empregado.nome, salario
FROM Empregado LEFT OUTER JOIN Departamento ON
(RG=RGgerente);
```

Empregado

RG	Nome	Salario	...
12345678-12	Fernando Paiva	4000,00	...
87654321-21	Ana Moreira	4200,00	...
11223344-55	Paulo Costa	2500,00	...
56781234-11	Pedro Faria	5000,00	...
43218765-22	Maria Maia	3300,00	...

Departamento

NumID	Nome	RGGerente
1	Computação	12345678-12
2	Automação	87654321-21
3	Elétrica	56781234-11
4	Mecânica	null

LEFT OUTER JOIN

Dep.Nome	Emp.Nome	Salario
Computação	Fernando Paiva	4000
Automação	Ana Moreira	4200
Elétrica	Pedro Faria	5000
NULL	Maria Maia	3300
NULL	Paulo Costa	2500

SQL – Junções

```
SELECT Departamento.nome, Empregado.nome, salario
FROM Empregado FULL OUTER JOIN Departamento ON
(RG=RGgerente);
```

Empregado

RG	Nome	Salario	...
12345678-12	Fernando Paiva	4000,00	...
87654321-21	Ana Moreira	4200,00	...
11223344-55	Paulo Costa	2500,00	...
56781234-11	Pedro Faria	5000,00	...
43218765-22	Maria Maia	3300,00	...

Departamento

NumID	Nome	RGGerente
1	Computação	12345678-12
2	Automação	87654321-21
3	Elétrica	56781234-11
4	Mecânica	null

FULL OUTER JOIN

Dep.Nome	Emp.Nome	Salario
Computação	Fernando Paiva	4000
Automação	Ana Moreira	4200
Elétrica	Pedro Faria	5000
Mecânica	NULL	NULL
NULL	Maria Maia	3300
NULL	Paulo Costa	2500



Resumindo:

