

Introdução a SQL - Consultas Simples

Roteiro

- Consultas simples em SQL
- Consultas com mais de uma relação

Leitura:

Capítulo 3, “Simple Queries” do livro **SQL for Web Nerds**, de Philip Greenspun

<http://philip.greenspun.com/sql/>

Introdução a SQL

Linguagem padrão para consulta e manipulação de dados

Structured Query Language

Vários padrões existentes:

- ANSI SQL
- SQL92 (conhecido como SQL2)
- SQL99 (conhecido como SQL3)
- Fabricantes dão suporte a vários subconjuntos destes padrões
- O material visto no curso é comum a todos eles

SQL

- Data Definition Language (DDL)
 - Criação/modificação/remoção de tabelas e seus atributos
- Data Manipulation Language (DML)
 - Consulta a uma ou mais tabelas
 - Inserção/modificação/remoção de tuplas das tabelas
- Transact-SQL
 - Idéia: empacotar uma sequência de sentenças SQL
 - Não será visto

Consultas SQL

Forma Básica:

SELECT atributos
FROM relações (possivelmente mais de um)
WHERE condições (seleções)

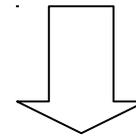
SELECT A1,...An Significado em Álgebra Relacional:
FROM R1, R2
WHERE cond $\mathcal{S}_{A1, \dots, An} (\mathcal{P}_{cond} (R1 \times R2))$

Uma Consulta SQL Simples

Produto

PNome	Preco	Categoria	Fabricante
lapis	\$19.99	papelaria	Farber
lapiseira	\$29.99	papelaria	Farber
camera	\$149.99	fotografia	Canon
televisao	\$203.99	eletronicos	Hitachi

```
SELECT *  
FROM Produto  
WHERE categoria='papelaria'
```



PNome	Preco	Categoria	Fabricante
lapis	\$19.99	papelaria	Farber
lapiseira	\$29.99	papelaria	Farber

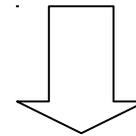
“seleção”

Outra Consulta SQL Simples

Produto

PNome	Preco	Categoria	Fabricante
lapis	\$19.99	papelaria	Farber
lapiseira	\$29.99	papelaria	Farber
camera	\$149.99	fotografia	Canon
televisao	\$203.99	eletronicos	Hitachi

```
SELECT PNome, Preco, Fabricante  
FROM Produto  
WHERE Preco > 100
```



“seleção” e
“projeção”

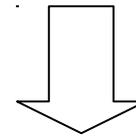
PNome	Preco	Fabricante
camera	\$149.99	Canon
televisao	\$203.99	Hitachi

Notação para Consultas SQL

Esquema de Entrada

Produto(PNome, Preço, Categoria, Fabricante)

```
SELECT PNome, Preço, Fabricante
FROM Produto
WHERE Preço > 100
```



Resultado(PNome, Preço, Fabricante)

Esquema de Saída

Seleções

A cláusula **WHERE** pode conter:

- $x = y$, $x < y$, $x \leq y$, etc
 - Para números: o significado usual
 - Para CHAR e VARCHAR: ordem lexicográfica
 - com conversão entre CHAR e VARCHAR
 - Para datas a hora: o significado esperado
- Casamento de padrão para strings...

O operador **LIKE**

- s **LIKE** p: casamento de padrão de strings
- p pode conter dois símbolos:
 - % = qualquer sequência de caracteres
 - _ = um único caracter

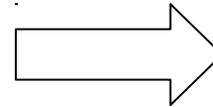
Produto(PNome, Preço, Categoria, Fabricante)

Encontre todos os produtos cujo nome contém 'lapis':

```
SELECT *  
FROM Produtos  
WHERE PNome LIKE '%lapis%'
```

Eliminação de Duplicações

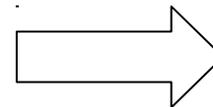
```
SELECT DISTINCT categoria  
FROM Produto
```



Categoria
papelaria
fotografia
eletronicos

Compare com:

```
SELECT categoria  
FROM Produto
```



Categoria
papelaria
papelaria
fotografia
eletronicos

Ordenação do Resultado

```
SELECT Pnome, preco, fabricante  
FROM Produto  
WHERE categoria='papeleria' AND preco > 50  
ORDER BY preco, pnome
```

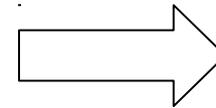
A ordenação é ascendente a não ser que a palavra DESC seja especificada.

No caso de igualdade no primeiro atributo da cláusula ORDER BY, o segundo atributo é utilizado e assim por diante.

Ordenação dos Resultados

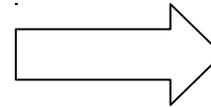
```
SELECT categoria  
FROM Produto  
ORDER BY pnome
```

PNome	Preco	Categoria	Fabricante
lapis	\$19.99	papelaria	Farber
lapiseira	\$29.99	papelaria	Farber
camera	\$149.99	fotografia	Canon
televisao	\$203.99	eletronicos	Hitachi



Ordenação dos Resultados

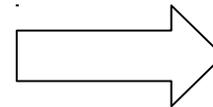
```
SELECT DISTINCT categoria  
FROM Produto  
ORDER BY categoria
```



Categoria
eletronicos
fotografia
papelaria

Compare to:

```
SELECT DISTINCT categoria  
FROM Produto  
ORDER BY pnome
```



?

Consultas com mais de uma tabela: Junções em SQL

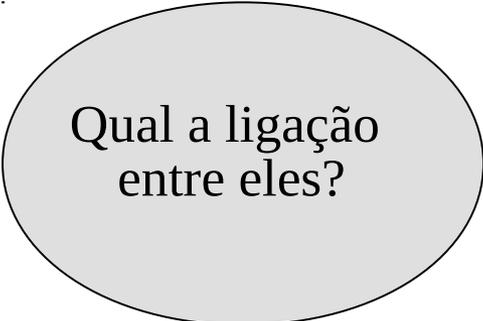
- Conectando duas ou mais tabelas:

Produto

PNome	Preco	Categoria	Fabricante
lapis	\$19.99	papelaria	Farber
lapiseira	\$29.99	papelaria	Farber
camera	\$149.99	fotografia	Canon
televisao	\$203.99	eletronicos	Hitachi

Companhia

CNome	ValorAcao	Pais
Farber	25	Brasil
Canon	65	Japao
Hitachi	15	Japao



Qual a ligação entre eles?

Junções

Produto (PNome, Preço, Categoria, Fabricante)

Companhia (CNome, ValorAcao, Pais)

Encontre o nome e preço de todos os produtos que custam menos de \$200 fabricados no Japão.

```
SELECT pnome, preco
FROM Produto, Companhia
WHERE fabricante=cnome AND pais='Japao'
AND preco < 200
```



Junção
entre Produto
e Companhia

Junções em SQL

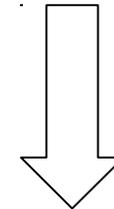
Produto

PNome	Preco	Categoria	Fabricante
lapis	\$19.99	papelaria	Farber
lapiseira	\$29.99	papelaria	Farber
camera	\$149.99	fotografia	Canon
televisao	\$203.99	eletronicos	Hitachi

Companhia

CNome	ValorAcao	Pais
Farber	25	Brasil
Canon	65	Japao
Hitachi	15	Japao

```
SELECT pnome, preco
FROM Produto, Companhia
WHERE fabricante=cnome AND pais='Japao'
AND preco < 200
```



PNome	Preco
camera	\$149.99

Junções

Produto (PNome, Preço, Categoria, Fabricante)

Companhia (CNome, ValorAcao, Pais)

Encontre todos os países que fabricam algum produto na categoria de ‘papeleria’.

```
SELECT pais
FROM Produto, Companhia
WHERE fabricante=cnome AND categoria='papeleria'
```

Junções em SQL

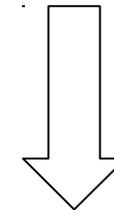
Produto

PNome	Preco	Categoria	Fabricante
lapis	\$19.99	papelaria	Farber
lapiseira	\$29.99	papelaria	Farber
camera	\$149.99	fotografia	Canon
televisao	\$203.99	eletronico	Hitachi

Companhia

CNome	ValorAcao	Pais
Farber	25	Brasil
Canon	65	Japao
Hitachi	15	Japao

```
SELECT pais
FROM Produto, Companhia
WHERE fabricante=cnome AND categoria='papelaria'
```



Pais
??
??

Qual o problema ?
Qual a solução ?

Junções

Produto (pnome, preco, categoria, fabricante)

Compra (comprador, vendedor, loja, produto)

Pessoa(nomePess, tel, cidade)

Encontre o nome das pessoas que moram em Curitiba, que compraram algum produto na categoria 'papelaria', e o nome da loja onde eles compraram o produto.

```
SELECT DISTINCT nomePess, loja
FROM Pessoa, Compra, Produto
WHERE nomePess=comprador AND produto = pnome AND
cidade='Curitiba' AND categoria='papelaria'
```

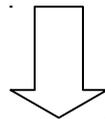
Como as tabelas são relacionadas?

- Você imagina que são
- Quando eu digo que são
- Chaves Estrangeiras (*Foreign keys*) são o método utilizado pelos projetista de esquema para explicitar o relacionamento.
 - Uma chave estrangeira define que uma coluna faz referência a chave de uma outra tabela.
ex: Produto.**Fabricante** é chave estrangeira em **Companhia**
 - Chaves estrangeiras definem o relacionamento e evitam que a restrição seja violada

Atributos com mesmo nome

- Pessoa(nomePess, end, funcDe)
Companhia(nomeCia, end)

```
SELECT DISTINCT nomePess, end  
FROM Pessoa, Companhia  
WHERE funcDe = nomeCia
```



```
SELECT DISTINCT Pessoa.nomePess, Companhia.end  
FROM Pessoa, Companhia  
WHERE Pessoa.funcDe = Companhia.nomeCia
```

Variáveis Tuplas

Produto (pnome, preco, categoria, fabricante)

Compra (comprador, vendedor, loja, produto)

Pessoa(nomePess, tel, cidade)

Encontre todas as lojas que vendem pelo menos um produto vendido também pela loja 'Ponto Frio':

```
SELECT DISTINCT x.loja
FROM   Compra AS x, Compra AS y
WHERE  x.produto = y.produto AND y.loja = 'Ponto Frio'
```

Resultado (loja)

Variáveis Tupla

Regra geral:

variáveis tupla são automaticamente definidas pelo sistema:

Produto (nome, preco, categoria, fabricante)

```
SELECT nome  
FROM Produto  
WHERE preco > 100
```

Se torna:

```
SELECT Produto.nome  
FROM Produto AS Produto  
WHERE Produto.preco > 100
```

Não funciona quando “Produto” ocorre mais de uma vez na consulta: neste caso é necessário definir as variáveis explicitamente.

Significado (Semântica) de Consultas SQL

SELECT a1, a2, ..., ak
FROM R1 AS x1, R2 AS x2, ..., Rn AS xn
WHERE Condições

1. Repetições encaixadas:

```
Answer = {}  
for x1 in R1 do  
    for x2 in R2 do  
        .....  
            for xn in Rn do  
                if Condições  
                    then Answer = Answer  $\cup$  {(a1,...,ak)}  
return Answer
```

Significado (Semântica) de Consultas SQL

```
SELECT a1, a2, ..., ak  
FROM R1 AS x1, R2 AS x2, ..., Rn AS xn  
WHERE Condições
```

2. Atribuição paralela

```
Answer = {}  
for all assignments x1 in R1, ..., xn in Rn do  
    if Condições then Answer = Answer  $\cup$  {(a1,...,ak)}  
return Answer
```

Não impõe nenhuma ordem!

Consulta não intuitiva

```
SELECT R.A  
FROM R, S, T  
WHERE R.A=S.A OR R.A=T.A
```

O que acontece se a tabela T estiver vazia?

Exercícios

Carregar o banco de dados “BDProduto”:

```
psql -h bd -U ci218 bdproduto
```

Produto (pnome, preco, categoria, fabricante)

Compra (comprador, vendedor, loja, produto)

Companhia (cnome, valorAcao, pais)

Pessoa (nomePess, tel, cidade)

Ex #1: Encontre todas as pessoas que compraram produtos eletrônicos.

Ex #2: Encontre o nome das pessoas que compraram produtos japoneses.

Ex #3: Encontre o nome das pessoas que compraram produtos japoneses e vivem em Curitiba.

Ex #4: Encontre as pessoas que compraram E venderam alguma coisa

Ex #5: Encontre as pessoas que compraram produtos de “Pedro” ou produtos fabricados por uma companhia com valor de ação maior que \$50.