

Atualização com Consulta

- Comandos de atualização podem incluir comandos de consulta
 - necessário toda vez que a atualização deve testar relacionamentos entre tabelas
- Exemplo 1

```
delete from Consultas
where hora > '17:00:00'
and codm in (select codm
             from Médicos
             where nome = 'Maria')
```

Atualização com Consulta

- Exemplo 2

```
update Médicos
set nroa = NULL
where not exists
      (select * from Médicos m
       where m.codm <> Médicos.codm
        and m.nroa = Médicos.nroa)
```

- Exemplo3

```
update Ambulatórios
set capacidade = capacidade +
      (select capacidade
       from Ambulatórios where nroa = 4)
where nroa = 2
```

Atualização com Consulta

- **Exemplo 4** (supondo `MedNovos(código, nome, especialidade)`)

```
insert into MedNovos
  select codm, nome, especialidade
  from Médicos
  where idade < 21;
```

- **Exemplo 5**

```
insert into Pacientes
  select p.codp+1, f.nome, f.idade, 'Fpolis',
        f.CPF, 'gripe'
  from Pacientes p join Funcionários f
  on f.codf = 1 and
  p.codp = (select max(codp) from Pacientes)
```

Ordenação de Resultados

- Cláusula ORDER BY

```
select lista_atributos
from lista_tabelas
[where condição]
[order by nome_atributo 1 [desc] {[,
  nome_atributo n [desc]]} ]
```

- Exemplos

```
select *
from Pacientes
order by nome
```

```
select salário, nome
from Funcionários
order by salário desc, nome
```

Ordenação de Resultados

- É possível determinar a quantidade de valores ordenados a retornar

```
select ...  
  limit valor1 [,valor2]
```

- Exemplos

retorna as 5 primeiras tuplas

```
select *  
from Pacientes  
order by nome  
limit 5
```

retorna tuplas 6 a 15

```
select salário, nome  
from Funcionários  
order by salário desc,  
       nome  
limit 10 offset 5
```

Definição de Grupos

- Cláusula GROUP BY

```
select lista_atributos  
from lista_tabelas  
[where condição]  
[group by lista_atributos_agrupamento  
  [having condição_para_agrupamento] ]
```

- GROUP BY

- define grupos para combinações de valores dos atributos definidos em *lista_atributos_agrupamento*
- um grupo mantém os atributos da tabela que não estão em *lista_atributos_agrupamento*
- apenas atributos definidos em *lista_atributos_agrupamento* podem aparecer no resultado da consulta
- geralmente o resultado da consulta possui uma **função de agregação**

Definição de Grupos

- Exemplo

```
select especialidade, count(*)  
from Médicos  
group by especialidade
```



especialidade	<i>Count</i>
ortopedia	2
pediatria	1
neurologia	1
traumatologia	3

especialidade	“grupos”					
ortopedia	codm	nome	idade	CPF	cidade	nroa
	1	João	40	1000010000	Fpolis	1
	4	Carlos	28	1100011000	Joinville	
pediatria	codm	nome	idade	CPF	cidade	nroa
	3	Pedro	51	1100010000	Fpolis	2
neurologia	codm	nome	idade	CPF	cidade	nroa
	5	Márcia	33	1100011100	Biguaçu	3
traumatologia	codm	nome	idade	CPF	cidade	nroa
	2	Maria	42	1000011000	Blumenau	2
	6	Joana	37	1111110000	Fpolis	3
	7	Mauro	53	1111000011	Blumenau	2

Definição de Grupos

- Cláusula HAVING

- define condições para que grupos sejam formados
 - condições só podem ser definidas sobre atributos do agrupamento ou serem funções de agregação
- existe somente associada à cláusula GROUP BY

- Exemplos

```
select especialidade, count(*)  
from Médicos  
group by especialidade  
having count(*) > 1
```


Exercícios

Buscar o que se pede utilizando ORDER BY ou GROUP BY:

- 1) os dados de todos os funcionários ordenados pelo salário (decrescente) e pela idade (crescente). Buscar apenas os três primeiros funcionários nesta ordem
- 2) o nome dos médicos e o número e andar do ambulatório onde eles atendem, ordenado pelo número do ambulatório
- 3) andares dos ambulatórios e a capacidade total por andar
- 4) andares dos ambulatórios cuja média de capacidade no andar seja ≥ 40
- 5) nome dos médicos que possuem mais de uma consulta marcada

Realizar as seguintes atualizações:

- 6) excluir os ambulatórios que não possuem médicos atendendo neles
- 7) o médico Pedro passa a residir na mesma cidade do paciente Paulo e sua idade passa a ser o dobro da idade da paciente Ana. Realizar essa alteração.
- 8) o funcionário Caio (codf = 3) tornou-se médico. Sua especialidade é a mesma da médica Maria (codm = 2) e ele vai atender no mesmo ambulatório dela. Inserir Caio na tabela Médicos, aproveitando os dados em comum.