

SQL – DML

Funções agregação e SQL estendida

Carina F. Dorneles
dorneles@inf.ufsc.br

INE5423 – Banco de Dados I

Funções de Agregação de valores

■ AVG

- ✓ Calcula a **média** dos valores não nulos de uma coluna. O domínio da coluna deve ser numérico

■ MAX e MIN

- ✓ Encontra o valor **mais alto** e **mais baixo** (respectivamente) de uma coluna. O domínio da coluna deve ser numérico

■ SUM

- ✓ **Soma** os valores não nulos de uma coluna. O domínio da coluna deve ser numérico

■ COUNT

- ✓ **Conta** o número de linhas não nulas de uma coluna. **COUNT(*)** conta o número de **linhas** não nulas de uma tabela



Exemplo

SELECT avg(salario)
FROM pessoa



<i>avg(salario)</i>
516,666667

SELECT sum(salario)
FROM pessoa



<i>sum(salario)</i>
1550

SELECT min(salario)
FROM pessoa



<i>min(salario)</i>
500

SELECT max(salario)
FROM pessoa



<i>max(salario)</i>
550

SELECT count(salario)
FROM pessoa



<i>count(salario)</i>
3

Pessoa

<i>codigo</i>	<i>nome</i>	<i>salario</i>
1	Ana	500
2	Paula	550
3	Juca	NULL
4	Rui	500

SELECT count(**distinct** salario)
FROM pessoa



<i>count(salario)</i>
2



Funções de agregação - Exemplos

► Tabela: consulta (data, hora, codpac#, codmed#, codconv#)



Funções de agregação - Exemplos

- ▶ Tabela: consulta (data, hora, codpac#, codmed#, codconv#)
- ▶ Exemplo: “*Obter o número (quantidade) de consultas*”

```
SELECT COUNT (*)  
FROM consulta
```



Funções de agregação - Exemplos

- ▶ Tabela: consulta (data, hora, codpac#, codmed#, codconv#)
- ▶ Exemplo: *“Obter o número (quantidade) de consultas”*

```
SELECT COUNT (*)  
FROM consulta
```

- ▶ *“Obter a quantidade de consultas que tiveram algum convênio associado”*

```
SELECT COUNT (codConv)  
FROM consulta
```



Funções de agregação - Exemplos

- ▶ Exemplo: *“Obter a média de idade dos paciente”*

```
SELECT AVG(idade)  
FROM paciente
```

- ▶ *“Obter a idade mínima dos pacientes cadastrados”*

```
SELECT MIN(idade)  
FROM paciente
```

- ▶ *“Obter o valor total das consultas entre os dias `2005-10-21` e `2005-12-21`”*

```
SELECT SUM(valor)  
FROM consulta  
WHERE data BETWEEN `2005-10-21` AND `2005-12-21`
```



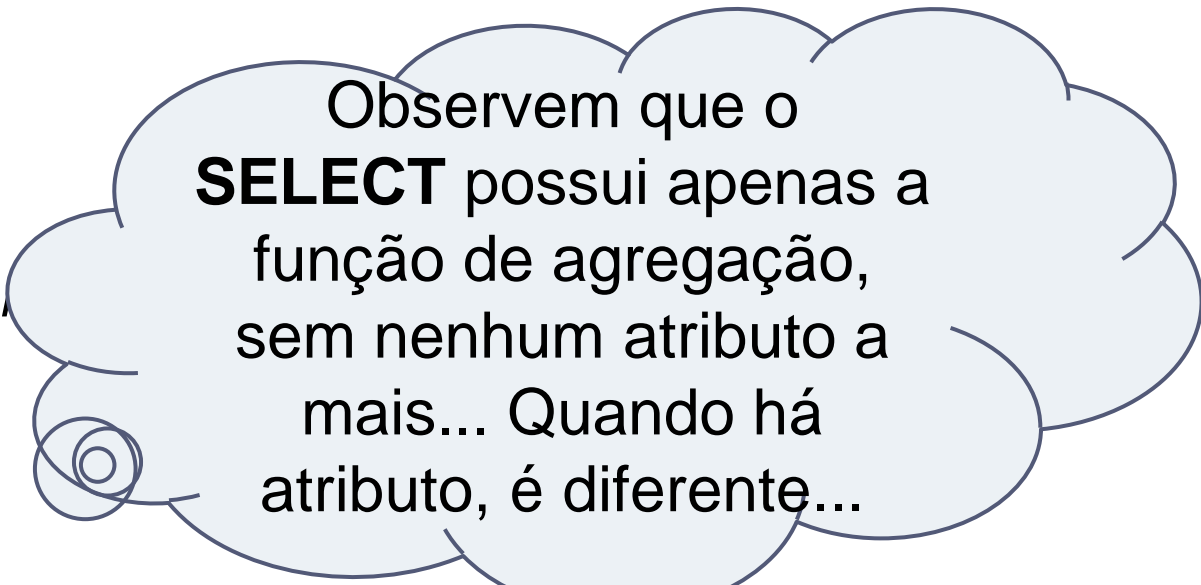
Funções de agregação - Exemplos

- ▶ Exemplo: *“Obter a média de idade dos paciente”*

```
SELECT AVG(idade)  
FROM paciente
```

- ▶ *“Obter a idade mínima dos*

```
SELECT MIN(idade)  
FROM paciente
```



Observem que o **SELECT** possui apenas a função de agregação, sem nenhum atributo a mais... Quando há atributo, é diferente...

- ▶ *“Obter o valor total das consultas entre os dias `2005-10-21` e `2005-12-21`”*

```
SELECT SUM(valor)  
FROM consulta  
WHERE data BETWEEN `2005-10-21` AND `2005-12-21`
```



SQL

order by, group by e having



Modelo estendido de consulta

► Sintaxe

```
SELECT <Lista_de_colunas>  
FROM <Lista_de_tabelas>  
[WHERE <Condição>]  
[GROUP BY <Lista_de_Colunas>]  
[HAVING <Critério>]  
[ORDER BY <Lista_de_Colunas>]
```



ORDER BY

```
SELECT <Lista_de_colunas>  
FROM <Lista_de_tabelas>  
[WHERE <Condição>]  
[ORDER BY <Lista_de_Colunas>]
```



ORDER BY

- ▶ Ordena as linhas da tabela resultante por ordem crescente

```
SELECT m.nome, m.email, m.CRM  
FROM medico m  
WHERE m.email LIKE '%gmail%'  
ORDER BY m.nome ASC
```

Default!

ORDER BY

- ▶ Ordena as linhas da tabela resultante por ordem **descrescente**

```
SELECT m.nome, m.email, m.CRM  
FROM medico m  
WHERE m.email LIKE '%eee%'  
ORDER BY m.nome DESC
```

```
SELECT m.nome, m.fone, m.CRM  
FROM medico m  
WHERE m.email LIKE '%eee%'  
ORDER BY m.dtanasc
```



GROUP BY

```
SELECT <Lista_de_colunas>  
FROM <Lista_de_tabelas>  
[WHERE <Condição>]  
[GROUP BY <Lista_de_Colunas>]
```



GROUP BY

- ▶ Usado **junto com** as funções de agregação para **agrupar o resultado** por uma ou mais colunas



GROUP BY

- ▶ Usado **junto com** as funções de agregação para **agrupar o resultado** por uma ou mais colunas
 - ▶ Exemplo: *Obter a quantidade de consultas*

```
SELECT COUNT (*)  
FROM consulta c
```



GROUP BY

- ▶ Usado **junto com** as funções de agregação para **agrupar o resultado** por uma ou mais colunas
 - ▶ Exemplo: *Obter a quantidade de consultas*

```
SELECT COUNT (*)  
FROM consulta c
```

- ▶ Exemplo: *obter a quantidade de consultas para cada data.*
Resultado: data e o número de consultas

```
SELECT c.data, COUNT (*)  
FROM consulta c  
GROUP BY c.data
```

GROUP BY

```
SELECT c.data, c.hora, count(*)  
FROM consulta c  
GROUP BY c.data, c.hora
```

***Atributo(s) do SELECT
deve (m) estar no GROUP BY***

- Não precisa ser na ordem
- Podem ser representados por 1, 2...
 - Indicando a posição



HAVING

```
SELECT <Lista_de_colunas>  
FROM <Lista_de_tabelas>  
[WHERE <Condição>]  
[GROUP BY <Lista_de_Colunas>]  
[HAVING <critério>]
```



HAVING

- ▶ Aplicada sobre um agrupamento
- ▶ Criada porque WHERE não pode ser usada com funções de agregação



HAVING

- ▶ Aplicada sobre um agrupamento
- ▶ Criada porque WHERE não pode ser usada com funções de agregação
- ▶ Exemplo: *Obter a quantidade de consultas para cada data*

```
SELECT c.data, COUNT (*)  
FROM consulta c  
GROUP BY c.data
```



HAVING

- ▶ Aplicada sobre um agrupamento
- ▶ Criada porque WHERE não pode ser usada com funções de agregação
- ▶ Exemplo: *Obter a quantidade de consultas para cada data*

```
SELECT c.data, COUNT (*)  
FROM consulta c  
GROUP BY c.data
```

- ▶ Exemplo: *Obter a quantidade de consultas para cada data, desde que esta quantidade seja maior do que 10*

```
SELECT c.data, COUNT (*)  
FROM consulta c  
GROUP BY c.data
```

```
HAVING count(*) > 10
```

Cuidar

- ▶ Sempre que houver necessidade de fazer um teste nos valores retornados pelas funções de agregação
 - ▶ usar HAVING!
 - ▶ nunca fazer no WHERE

✓
SELECT c.data, COUNT(*)
FROM consulta c
GROUP BY c.data
HAVING count(*) > 10

certo

~~**SELECT c.data, COUNT(*)**
FROM consulta c
WHERE count(*) > 10
GROUP BY c.data~~

errado

HAVING

- ▶ A função de agregação não precisa ser usada no SELECT

```
SELECT c.data  
FROM consulta c  
GROUP BY c.data  
HAVING count(*) > 10
```

Lembre-se: HAVING é aplicada sobre um agrupamento



Usando as 3 cláusulas juntas

- Manter a seguinte ordem:

```
SELECT <Lista_de_colunas>  
FROM <Lista_de_tabelas>  
[WHERE <Condição>]  
[GROUP BY <Lista_de_Colunas>]  
[HAVING <Critério>]  
[ORDER BY <Lista_de_Colunas>]
```



▶ Exercícios

