

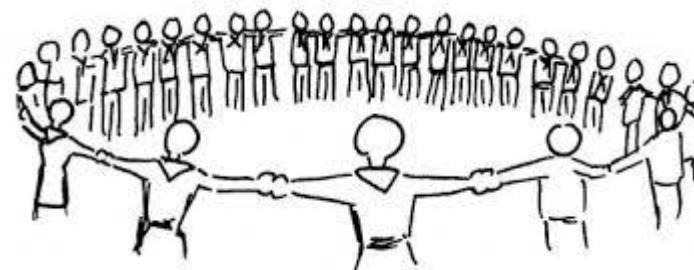


# Funções de agregação

## Prof. Ramon Souza

As **funções de agregação** são usadas para **resumir informações de várias tuplas em uma síntese de tupla única**. Existem diversas funções de agregação embutidas no SQL: **COUNT, SUM, MAX, MIN e AVG**.

· **SELECT FUNCAO(coluna1) FROM nome\_da\_tabela WHERE condição;** em que **FUNCAO** é qualquer uma das funções de agregação.



FUNÇÃO	RETORNO
MIN	Menor valor de uma coluna.
MAX	Maior valor de uma coluna.
COUNT	Número de linhas que atendem a um critério.
AVG	Média dos valores de uma coluna numérica.
SUM	Soma dos valores de uma coluna numérica.

# Count (exemplo)

ProdutoID	Nome_do_Produto	FornecedorID	CategoriaID	Unidade	Preco
1	Leite	1	1	Litros	3
2	Banana	1	1	Kilogramas	5
3	Melancia	1	2	Unidade	6
4	Pão	2	2	Pacote	4
5	Suco	2	2	Litros	8

```
SELECT COUNT(ProdutoID) FROM Produtos;
```

COUNT(ProdutoID)
5

ProdutoID	Nome_do_Produto	FornecedorID	CategoriaID	Unidade	Preco
1	Leite	1	1	Litros	3
2	Banana	1	1	Kilogramas	5
3	Melancia	1	2	Unidade	6
4	Pão	2	2	Pacote	4
5	Suco	2	2	Litros	8

```
SELECT MIN(Preco) FROM Produtos;
```

MIN(Preco)
3

ProdutoID	Nome_do_Produto	FornecedorID	CategoriaID	Unidade	Preco
1	Leite	1	1	Litros	3
2	Banana	1	1	Kilogramas	5
3	Melancia	1	2	Unidade	6
4	Pão	2	2	Pacote	4
5	Suco	2	2	Litros	8

```
SELECT MAX(Preco) FROM Produtos;
```

MAX (Preco)
8

ProdutoID	Nome_do_Produto	FornecedorID	CategoriaID	Unidade	Preco
1	Leite	1	1	Litros	3
2	Banana	1	1	Kilogramas	5
3	Melancia	1	2	Unidade	6
4	Pão	2	2	Pacote	4
5	Suco	2	2	Litros	8

```
SELECT AVG(ProdutoID) FROM Produtos;
```

AVG(ProdutoID)
5.2

ProdutoID	Nome_do_Produto	FornecedorID	CategoriaID	Unidade	Preco
1	Leite	1	1	Litros	3
2	Banana	1	1	Kilogramas	5
3	Melancia	1	2	Unidade	6
4	Pão	2	2	Pacote	4
5	Suco	2	2	Litros	8

**SELECT SUM(ProdutoID) FROM Produtos;**

SUM(ProdutoID)
76

A cláusula **COUNT** pode ser usada com o nome da coluna, **\*** ou com **1**:

- **COUNT(nome\_da\_coluna)**: retorna o número de linhas excluindo-se da contagem as linhas que possuem nulo para a coluna desejada.
- **COUNT(\*)** ou **COUNT(1)**: retorna o número total de linhas, independentemente de valores nulos registrados para qualquer campo.

DetalheVendasID	VendaID	ProdutoID	Quantidade	DetalheVendasID	VendaID
1	10248	11	12	1	10248
2	10248	42	10	2	10248
3	10248	72	5	3	10248
4	10249	14	9	4	10249
5	10250	20	NULL	5	10250

```
SELECT COUNT(Quantidade) FROM DetalheVendas;
```

**4**

```
SELECT COUNT(*) FROM DetalheVendas; ou SELECT COUNT(1) FROM DetalheVendas;
```

**5**

A cláusula **SUM** pode ser usada com o nome da coluna ou com um número indicativo da quantidade a ser somada:

- **SUM(nome\_da\_coluna)**: retorna o somatório dos valores presentes em nome\_da\_coluna.
- **SUM(1)**: retorna um somatório, sendo somado 1 para cada registro encontrado. Resultado similar a COUNT(\*) ou COUNT(1).
- **SUM(2)**: retorna um somatório, sendo somado 2 para cada registro encontrado.
- **SUM(N)**: retorna um somatório, sendo somado N para cada registro encontrado.

DetalheVendasID	VendaID	ProdutoID	Quantidade	DetalheVendasID	VendaID
1	10248	11	12	1	10248
2	10248	42	10	2	10248
3	10248	72	5	3	10248
4	10249	14	9	4	10249
5	10250	20	NULL	5	10250

```
SELECT SUM(Quantidade) FROM DetalheVendas;
```

36

```
SELECT SUM(1) FROM DetalheVendas;
```

5

```
SELECT SUM(3) FROM DetalheVendas;
```

15

## **(CESPE - 2018 - EBSERH - Analista de Tecnologia da Informação)**

Com relação a banco de dados, julgue o item seguinte.

Na SQL (structured query language), existem funções de agregação com diferentes capacidades; como, por exemplo, a função AVG, que é responsável pelo cálculo da média dos valores de determinado campo.

**(CESPE - 2015 - FUB - Analista de Tecnologia da Informação)**

Julgue o item seguinte, no que se refere à linguagem SQL.

A função max, utilizada conjuntamente com o comando select, retorna o maior valor em um determinado campo que tenha sido incluído na busca.

**(FCC - 2019 - TRF - 4<sup>a</sup> REGIÃO - Analista Judiciário - Sistemas de Tecnologia da Informação)** Em uma tabela chamada itemfatura há diversos registros em que constam o mesmo valor no campo idfatura. Para mostrar a quantidade de valores de idfatura diferentes que estão cadastrados na tabela, utiliza-se o comando

- a) SELECT DISTINCT (idfatura) FROM itemfatura;
- b) SELECT \* FROM itemfatura WHERE idfatura IS DIFFERENT;
- c) SELECT SUM(DISTINCT idfatura) FROM itemfatura;
- d) SELECT COUNT(DISTINCT idfatura) FROM itemfatura;
- e) SELECT COUNT(DIFFERENT idfatura) FROM itemfatura;

**(FCC - 2019 - TRF - 4<sup>a</sup> REGIÃO - Técnico Judiciário - Tecnologia da Informação)** Um Técnico Judiciário necessitou usar a linguagem padrão SQL para recuperar, de uma tabela do banco de dados relacional denominada tabela1,

- I. o menor valor em uma determinada coluna denominada coluna1.
- II. um padrão de valores denominado padrão\_desejado em uma outra coluna denominada coluna2.

Para tanto, em duas operações distintas, ele utilizou, respectivamente, as expressões

I e II são, correta e respectivamente,

- a) MINVALUE(coluna1) e padrão\_desejado %LIKE coluna2
- b) THIN (coluna1) e coluna2 = padrão\_desejado
- c) SMALL(coluna1) e padrão\_desejado = coluna2
- d) MIN(coluna1) e coluna2 LIKE padrão\_desejado
- e) GETSMLL(coluna1) e padrão\_desejado % coluna2

- 1. `SELECT I.....  
FROM tabelal  
WHERE condição;`
- 2. `SELECT coluna2  
FROM tabelal  
WHERE II.....;`

**(FCC - 2017 - TRT - 11ª Região (AM e RR) - Técnico Judiciário - Tecnologia da Informação)** Considere que no TRT exista, em um banco de dados, a tabela TRAB que possui como campos: nome, sexo, salario de vários trabalhadores. Um Técnico foi solicitado a escrever um comando SQL para obter a média salarial dos trabalhadores do sexo FEMININO. O comando correto é:

- a) SELECT sexo, AVG(salario) as MediaSalarial FROM TRAB GROUP BY sexo;
- b) SELECT AVG(salario) FROM TRAB WHERE sexo='FEMININO';
- c) SELECT sexo, AVG(salario) FROM TRAB GROUP BY sexo="FEMININO";
- d) SELECT \* FROM TRAB WHERE sexo='FEMININO' as AVG(salario);
- e) SELECT sexo="FEMININO" FROM TRAB WHERE AVG(salario);

**(FGV - 2015 - DPE-RO - Analista da Defensoria Pública - Analista Programador)** Analise o comando SQL a seguir.

```
select max(A1) X, count(*) Y, sum(A1) Z from T
```

Executado quando a instância da tabela T estiver vazia (com zero registros), esse comando produz como resultado:

a)

X	Y	Z
NULL	0	NULL

c)

X	Y	Z
0	0	0

e)

X	Y	Z
NULL	NULL	0

b)

X	Y	Z
NULL	NULL	NULL

d)

X	Y	Z
0	NULL	0

**(VUNESP - 2020 - FITO - Técnico em Gestão - Informática)** Considere a seguinte tabela de um banco de dados relacional:

Exame (ID, Tipo, Nome, Valor)

A consulta SQL que obtém o número de registros para os exames do tipo Laboratório é:

a) SELECT COUNT (Tipo = “Laboratório”)

FROM Exame

b) SELECT COUNT (Tipo)

FROM Exame

WHERE Tipo = “Laboratório”

c) SELECT SUM (Tipo = “Laboratório”)

FROM Exame

d) SELECT SUM (Tipo)

FROM Exame

WHERE Tipo = “Laboratório”

e) SELECT NUMBER (Tipo)

FROM Exame.Tipo = “Laboratório”

**(VUNESP - 2019 - Câmara de Piracicaba - SP - Analista de Sistemas)** Considere a seguinte tabela de um banco de dados relacional:

Coleção (ID, Tipo, Nome, Descrição)

O comando SQL para obter o número de registros da tabela Coleção é:

- a) SELECT AVG(ID) FROM Coleção
- b) SELECT SUM (Coleção)
- c) SELECT SUM OF Coleção
- d) SELECT COUNT (Coleção)
- e) SELECT COUNT(ID) FROM Coleção



# Funções de agregação

## Prof. Ramon Souza