

Artigo

Invista em você! Saiba como a DevMedia pode ajudar sua carreira. 

SQL: Max, Min, Avg, Sum e Count

As funções agregadas são compostas por MAX, MIN, AVG, SUM e COUNT e cada uma delas executa uma tarefa específica.



13



Suporte ao aluno



Anotar



Marcar como concluído

Artigos > Banco de Dados > SQL: Max, Min, Avg, Sum e Count

As funções agregadas são compostas por MAX , MIN , AVG , SUM e COUNT e cada uma delas executa uma tarefa específica.

É possível utilizar as funções agregadoras em conjunto com o WHERE, lembrando que o resultado mudará de acordo com as sentenças definidas pelo filtro.

Visão geral

Funções agregadas são utilizadas a todo momento pelo desenvolvedor. Elas operam em um conjunto de linhas para fornecer um resultado por grupo, sendo esse conjunto de linhas uma tabela inteira ou a uma divisão em grupos.

Para este exemplo, considere a estrutura de tabelas abaixo, onde temos uma tabela chamada produto (Tabela 1), responsável por armazenar todos os produtos do estoque do cliente, a tabela categoria_produto (Tabela 2), responsável por armazenar todas as categorias existentes no banco de dados, e a tabela venda_produto (Tabela 3), que relaciona os produtos que foram vendidos.

id	nome	preco	Id_categoria
1	Bola	35.00	1
2	Patinete	120.00	1
3	Carrinho	15.00	1
4	Skate	296.00	1
5	Notebook	3500.00	2
6	Monitor LG 19	450.00	2
7	O Diário de Anne Frank	45.00	3
8	O dia do Curinga	65.00	3
9	O mundo de Sofia	48.00	3



id	nome
1	Infantil
2	Informatica
3	Educacional

Tabela 2. Tabela categoria_produto



13

id	id_produto	valor	data
1	1	35.00	2018-05-15
2	1	35.00	2018-06-15
3	1	35.00	2018-07-15
4	2	120.00	2018-07-15
5	2	120.00	2018-07-14
6	3	15.00	2018-07-15
7	7	45.00	2018-07-15
8	8	65.00	2018-07-15
9	8	65.00	2018-07-16
10	9	48.00	2018-07-16
11	5	3500.00	2018-07-16
12	5	3500.00	2018-07-16
13	6	450.00	2018-07-16

Tabela 3. Tabela venda_produto

Suponhamos que seja necessário saber exatamente qual o maior preço entre todos os produtos cadastrados na tabela produto. Para isso, podemos utilizar a função `MAX` , conforme a query abaixo:

```
1 | SELECT
2 |     MAX(preco) as MAIOR_PRECO
3 | FROM
4 |     produto
```

Observe que, conforme código acima e ilustrado na Tabela 4, trouxemos o valor máximo da coluna `preco` através da função `MAX` .

MAIOR_PRECO
3500.00

Tabela 4. Resultado da função MAX executada na coluna preco



```
1 | SELECT
2 |   [ funcao-agregadora ]( [ coluna ] )
3 | FROM
4 |   [ tabela1, tabela2, ... | * ]
```

Na pratica

Exemplo 1

Aproveitando a mesma estrutura da tabela, podemos trazer o maior preço por categoria , conforme o trecho de código abaixo:



```
1 | SELECT
2 |   C.nome as Categoria,
3 |   MAX(preco) as MAIOR_PRECO
4 | FROM
5 |   Produto P, categoria_produto C
6 | WHERE
7 |   P.id_categoria = C.id
8 | GROUP BY
9 |   C.id
```

O resultado da query acima nos trás o preço maior por cada categoria cadastrada no banco de dados. Nas **linhas 1 e 2** selecionamos o nome da categoria, nomeada como `Categoria` , e em seguida, na **linha 3** selecionamos o maior valor da coluna `preco` , através da função `MAX` , nomeando a coluna como `MAIOR_PRECO` .

Nas **linhas 4 e 5** informamos que a consulta irá se realizar nas tabelas `Produto` , nomeada de `P` , e `categoria_produto` , nomeada de `C` .

Em seguida, nas **linhas 6 e 7** definimos que a consulta deve atender ao filtro de o id do produto ser o mesmo nas duas tabelas, realizando assim o vínculo entre elas.

Por fim, nas **linhas 8 e 9** agrupamos o resultado pelo id da categoria.

O resultado pode ser visto na **Tabela 5**.

Categoria	MAIOR_PRECO
infantil	296.00
Informatica	3500.00
Educacional	65.00

Tabela 5. Resultado da função MAX agrupada por categoria

Exemplo 2



```
1 | SELECT
2 |     C.nome as Categoria,
3 |     MIN(preco) as MENOR_PRECO
4 | FROM
5 |     Produto P, categoria_produto C
6 | WHERE
7 |     P.id_categoria = C.id
8 | GROUP BY
9 |     C.id
```

O resultado da query pode ser visto na Tabela 6.

Categoria	MENOR_PRECO
infantil	15.00
Informatica	450.00
Educacional	38.00

Tabela 6. Resultado da função MIN agrupada por categoria

Exemplo 3

Outra função agregada seria `COUNT` , responsável por trazer o total de ocorrências da coluna informada.

Suponha que seja necessário saber exatamente qual a quantidade de produtos existentes no banco de dados. Para isso, podemos utilizar a função `COUNT` , conforme trecho de código abaixo:

```
1 | SELECT
2 |     COUNT(id) as TOTAL
3 | FROM
4 |     produto
```

Como podemos ver no trecho de código acima, e ilustrado na **Tabela 7**, conseguimos o total de produtos através da função `COUNT` .

TOTAL
10

Tabela 7. Resultado da consulta por total de produtos

Exemplo 4



podemos executar a query abaixo:

```
1 SELECT
2   C.nome as Categoria,
3   COUNT(P.id) as TOTAL_PRODUTOS
4 FROM
5   Produto P, categoria_produto C
6 WHERE
7   P.id_categoria = C.id
8 GROUP BY
9   C.id
```

Como podemos ver no trecho de código acima e na Tabela 8, a query é capaz de trazer o total de produtos separados por categoria.

Categoria	TOTAL
infantil	4
Informatica	2
Educacional	4

Tabela 8. Resultado da consulta do total de produtos separados por categoria

Exemplo 5

Suponhamos agora que seja necessário saber exatamente quanto ganhamos com cada produto até o momento, ou seja, olharemos na tabela vendas_produto o valor total recebido com cada produto.

Para isso podemos utilizar o seguinte trecho de código:

```
1 SELECT
2   P.nome,
3   SUM(V.valor) as TOTAL_RECEBIDO
4 FROM
5   produto P, venda_produto V
6 WHERE P.id = V.id_produto
7 GROUP BY P.id
```

No trecho de código acima e ilustrado na Tabela 9, utilizamos a função SUM para somar os valores de venda de um determinado produto e retornar na consulta.

Na linha 3 utilizamos a função SUM na coluna valor da tabela venda_produto e demos o nome de TOTAL_RECEBIDO.

Na linha 5 definimos que será feita a busca em duas tabelas, a tabela produto, nomeada de P, e na tabela venda_produto, nomeada de V.



nome	TOTAL_RECEBIDO
Bola	105.00
Patinete	240.00
Carrinho	15.00
Notebook	7000.00
Monitor LG 19	450.00
O Diário de Anne Frank	45.00
O dia do Curinga	130.00
O mundo de Sofia	48.00

Tabela 9. Resultado da soma de todos os valores recebidos por produto



Exemplo 6

Outra função agregada seria a `AVG` que traz a média de valores de uma determinada coluna informada na consulta. Suponhamos que seja necessário agora saber a média de preço dos produtos cadastrados.

```
1 | SELECT
2 |   AVG(preco) as PRECO_MEDIO
3 | FROM
4 |   produto
```

No trecho de código acima, selecionamos a média de preço de todos os produtos cadastrados no banco de dados, nomeando a coluna da média como `PRECO_MEDIO` . O resultado pode ser visto na Tabela 10.

PRECO_MEDIO
461.20

Tabela 10. Resultado da consulta para a média de preço da tabela produto

Exemplo 7

Agora um exemplo mais complexo, suponha que desejamos trazer a média de preço de todos os produtos, agrupados por categoria, trazendo também a quantidade de produtos em cada uma delas. Para isso temos o trecho de código abaixo:

```
1 | SELECT
2 |   C.nome as Categoria,
3 |   COUNT(P.ID) as TOTAL_DE_PRODUTOS,
4 |   AVG(P.preco) as PRECO_MEDIO
5 | FROM
```

```
9 | GROUP BY
10 | C.id
```

No trecho de código acima, nas **linhas 1 e 2** selecionamos o nome da categoria, nomeando a coluna de Categoria e, em seguida, na **linha 3** selecionamos o total de ID de produtos através da função COUNT , nomeando a coluna de TOTAL_DE_PRODUTOS .

Na **linha 4** selecionamos a média de valor da coluna preco da tabela produto, nomeando a coluna de PRECO_MEDIO e, em seguida, nas **linhas 5 e 6** informamos que a consulta será feita na tabela produto , nomeada de P , e na tabela categoria_produto , nomeada de C .

Nas **linhas 7 e 8** filramos a consulta a fim de trazer somente resultados onde o id de produto seja o mesmo nas duas tabelas, realizando assim o vínculo entre elas.


E por fim, nas **linhas 9 e 10** agrupamos o resultado pelo id da categoria.

O resultado da query pode ser vista na **Tabela 11**.

Categoria	TOTAL_DE_PRODUTOS	PRECO_MEDIO
infantil	4	116.50
Informatica	2	1975.00
Educacional	4	49.00

Tabela 11. Resultado da média de preços agrupados por categoria

Confira também

 Seleção em múltiplas tabelas

SQL: Seleção Múltiplas tabelas

Curso

 SQL

Curso de SQL

Curso

 SQL

Avançando no comando SQL Select

Curso