



Agrupamentos

Prof. Ramon Souza

A linguagem SQL tem uma cláusula **GROUP BY** para **aplicar agrupamentos**. Esta cláusula especifica os atributos de agrupamento, que também devem aparecer na cláusula **SELECT**, de modo que o valor resultante da aplicação de cada função de agregação a um grupo de tuplas apareça junto com o valor do(s) atributo(s) de agrupamento.

```
SELECT colunas FROM nome_da_tabela WHERE condição GROUP BY coluna;
```



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 FornecedorID, **COUNT**(ProdutoID) **FROM** Produtos **GROUP BY** FornecedorID;

FornecedorID	COUNT(ProdutoID)
1	3
2	2

A cláusula **HAVING** pode ser usada para **definir uma condição para um agrupamento** com GROUP BY.

```
SELECT colunas FROM nome_da_tabela WHERE condição GROUP BY coluna HAVING condição;
```

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 FornecedorID, COUNT(ProdutoID) FROM Produtos GROUP BY FornecedorID  
HAVING COUNT(ProdutoID) <= 2;
```

FornecedorID	COUNT(ProdutoID)
2	2

(CESPE - 2019 - TJ-AM - Assistente Judiciário - Programador)

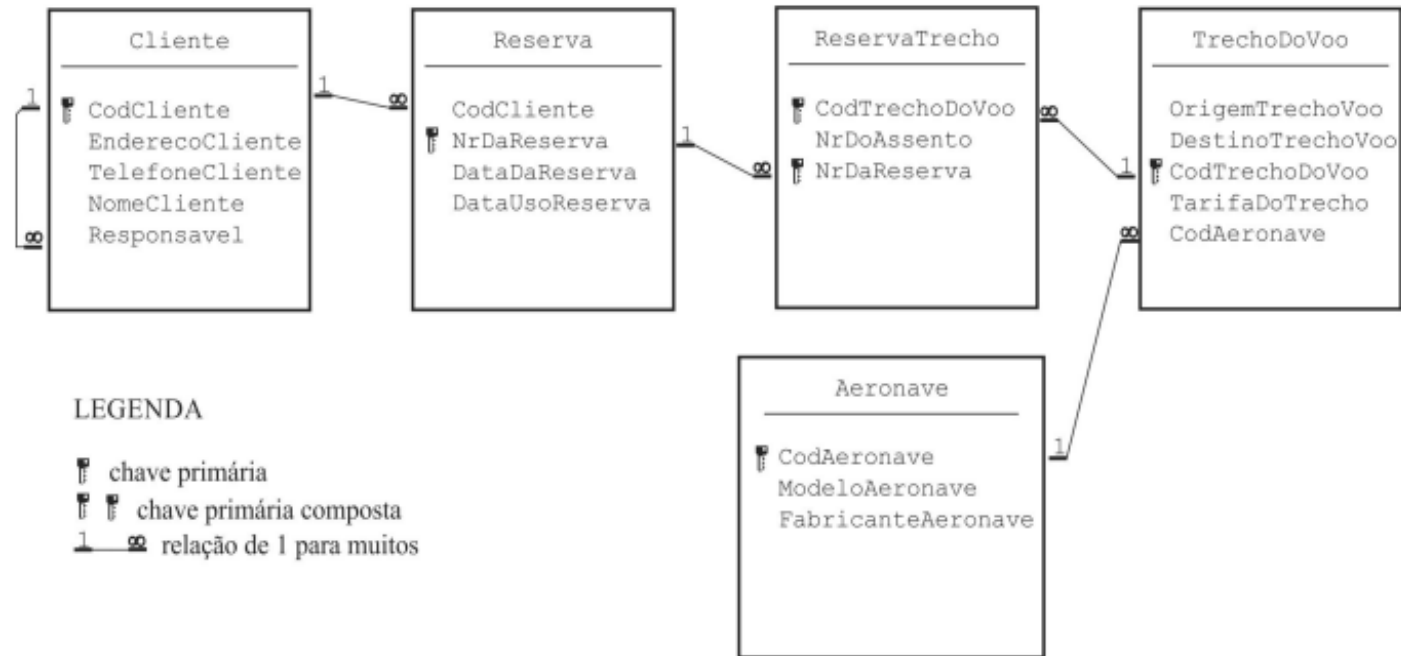
```
SELECT P.P#,  
       P.COR,  
       MAX (FP.QDE) AS QDEMAX  
FROM P, FP  
WHERE P.P# = FP.P#  
AND (P.COR = COR ('Azul') OR P.COR = COR ('Rosa'))  
AND FP.QDE > QDE (100)  
GROUP BY P.P#, P.COR  
HAVING SUM (FP.QDE) > QDE (250)
```

Considerando a formulação do algoritmo conceitual da consulta em SQL precedente, julgue o item a seguir.

O resultado da cláusula WHERE é agrupado pelos valores das colunas nomeadas na cláusula GROUP BY P.P#, P.COR. Além disso, o uso da cláusula HAVING elimina os grupos que não satisfazem à expressão booleana $SUM (FP.QDE) > QDE (250)$.

(CESPE - 2019 - MPC-PA - Assistente Ministerial de Informática)

As seguintes informações foram extraídas de um diagrama de entidades-relacionamento no contexto de um banco de dados relacional.



Considere que, no sistema modelado no diagrama a que se refere o texto 9A1-I, as tarifas de uma companhia aérea para os trechos de voos promocionais sejam de até R\$ 100. Considere, ainda, a aplicação do seguinte comando SQL nesse contexto.

```
select codAeronave, avg(TarifaDoTrecho)
  from TrechoDoVoo
 where TarifaDoTrecho >100
 group by CodAeronave
 having avg(TarifaDoTrecho) >250 ;
```


Nesse caso, o comando SQL em questão mostra apenas as médias de tarifas

- a) maiores que R\$ 250 de todas as aeronaves, de forma agrupada, sem os voos promocionais.
- b) maiores que R\$ 250 de todas as aeronaves, de forma agrupada, com os voos promocionais.
- c) maiores que R\$ 250 de todas as aeronaves, por trecho de voo, sem os voos promocionais.
- d) por aeronave, maiores que R\$ 250 nos trechos em que a companhia aérea participa, com os voos promocionais.
- e) por aeronave, maiores que R\$ 250 nos trechos em que a companhia aérea participa, sem os voos promocionais.

(CESPE - 2017 - TRE-PE - Analista Judiciário - Análise de Sistemas)

Tabela 3A6AAA

dados da tabela:

ID; nome; idtipo; preco

3125; creme; 3; 11,50

31; arroz; 4; 12,50

34; leite; 1; 14,00

42; sabão; 5; 11,00

46; carne; 1; 12,75

48; shampoo; 5; 12,30

58; azeite; 1; 13,25

Considerando-se os campos e dados contidos na tabela 3A6AAA, denominada tbproduto, é correto afirmar que o comando SQL.

a) `SELECT MAX(preco) FROM tbproduto WHERE
idtipo=5;`
retornará 14,00 como resultado.

b) `SELECT sum(preco) FROM tbproduto WHERE
idtipo=5
GROUP BY preco
HAVING preco < 14;`
retornará dois registros

c) `SELECT sum(preco) as total FROM tbproduto
WHERE idtipo in (1,5) and nome like '%e' group
by idtipo having sum(preco) > 13;`
retornará 26,00 como resultado.

d) `SELECT nome FROM tbproduto WHERE idtipo not in (5) and preco > (select min(preco) from tbproduto where idtipo = 1);`

retornará apenas leite como resultado

e) `SELECT * FROM tbproduto WHERE preco BETWEEN 10 AND 12;`

retornará cinco registros.

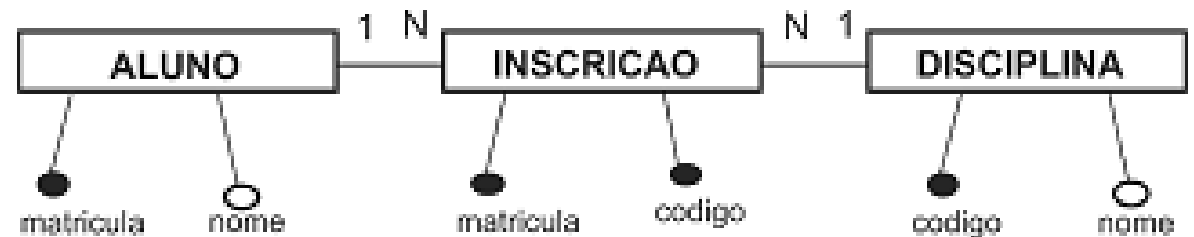
(CESPE - 2016 - FUB - Técnico de Tecnologia da Informação) A respeito das principais instruções da linguagem SQL, julgue o item subsecutivo.

Em uma consulta SQL que contém as cláusulas WHERE e HAVING, a cláusula WHERE é aplicada às linhas no conjunto de resultados.

(CESPE - 2016 - TRE-PI - Analista Judiciário - Análise de Sistemas)

O modelo lógico apresentado dá origem às tabelas ALUNO, INSCRICAO e DISCIPLINA. Considerando esse modelo e sabendo que não há nenhum procedimento armazenado no banco de dados, assinale a opção que apresenta código em SQL ANSI que resultará corretamente na listagem de matrícula e nome dos alunos que estão inscritos (INSCRICAO) em mais de duas disciplinas.

a) `SELECT aluno.matricula, aluno.nome`
`FROM inscricao, aluno, disciplina`
`WHERE inscricao.matricula=aluno.matricula`
`AND inscricao.codigo=disciplina.codigo`
`GROUP BY aluno.matricula, aluno.nome`
`HAVING COUNT(*) > 2`



b) SELECT aluno.matricula, aluno.nome
FROM inscricao, aluno, disciplina
WHERE inscricao.matricula=aluno.matricula
AND inscricao.codigo=disciplina.codigo
AND COUNT(*) > 2

c) SELECT aluno.matricula, aluno.nome
GROUP BY aluno.matricula, aluno.nome
FROM inscricao, aluno, disciplina
WHERE inscricao.matricula=aluno.matricula
AND inscricao.Codigo=diciplina. Codigo AND
COUNT

d) SELECT aluno.matricula, aluno.nome
FROM inscricao, aluno, disciplina
WHERE inscricao.matricula=aluno.matricula
AND quantidade > 2
GROUP BY inscricao, aluno, disciplina

e) SELECT aluno.matricula, aluno.nome, SUM() > 2
FROM inscricao, aluno, disciplina
WHERE inscricao.matricula=aluno.matricula
AND inscricao.codigo=disciplina.codigo
GROUP BY aluno.matricula, aluno.nome

(CESPE - 2015 - MEC - Desenvolvedor)

CANTOR	
CODIGO_CANTOR (PK)	NOME_CANTOR
A	JOAO
B	JOSE

CD			
CODIGO_CD (PK)	TITULO_CD	CODIGO_CANTOR (FK)	VALOR_CD
1	SERA	A	15
2	DESCOBRIMENTO	B	16
3	INDIOS	B	17
4	DOIS	A	15
5	FARDESTE	A	19

Considerando os dados apresentados nas tabelas, julgue o item subsequente, a respeito de conceitos e comandos SQL.

Assumindo que cada cantor vendeu apenas uma unidade de cada CD de sua autoria, o comando a seguir apresentará o ganho total de cada um deles.

```
SELECT CANTOR.NOME_CANTOR, SUM(CD.VALOR_CD)
```

```
FROM CANTOR, CD
```

```
WHERE CD.CODIGO_CANTOR = CANTOR.CODIGO_CANTOR
```

(CESPE - 2015 - MEC - Administrador de Dados) Julgue o item que se segue, com relação às definições e aos problemas de execução de comandos nas linguagens SQL.

Considerando a tabela aluno (id, curso, disciplina, notafinal, sexo) de um banco de dados relacional, o comando SQL que permite obter o nome da disciplina e a média das notas finais por disciplina apenas para os alunos do sexo igual a 'F' e para as disciplinas com média abaixo de 5 é o apresentado a seguir.

```
SELECT disciplina, AVG(notafinal) FROM aluno WHERE sexo= 'F'  
GROUP BY disciplina HAVING AVG (notafinal)<5
```

(FCC - 2019 - TRF - 3ª REGIÃO - Técnico Judiciário - Informática)

Em uma instrução SQL, a cláusula GROUP BY, se utilizada juntamente com funções de grupo, deve estar em uma posição específica como

a) SELECT coluna1 [, coluna2, ...] . função_de_grupo, [função_de_grupo, ...]
FROM tabela
ORDER BY coluna2
GROUP BY coluna4 [, coluna5, ...]
HAVING coluna5 > 10
WHERE condição;

b) SELECT coluna1 [, coluna2, ...] . função_de_grupo, [função_de_grupo, ...]
FROM tabela
WHERE condição
ORDER BY coluna2
GROUP BY coluna4 [, coluna5, ...]
HAVING coluna5 > 10;

c) SELECT coluna1 [, coluna2, ...] . função_de_grupo, [função_de_grupo, ...]
FROM tabela
WHERE condição
GROUP BY coluna4 [, coluna5, ...]
HAVING coluna5 > 10
ORDER BY coluna2;

d) SELECT coluna1 [, coluna2, ...] . função_de_grupo, [função_de_grupo, ...]
FROM tabela
HAVING coluna5 > 10
ORDER BY coluna2
WHERE condição
GROUP BY coluna4 [, coluna5, ...] ;

e) SELECT coluna1 [, coluna2, ...] . função_de_grupo, [função_de_grupo, ...]
FROM tabela
HAVING coluna5 > 10
WHERE condição
ORDER BY coluna2
GROUP BY coluna4 [, coluna5, ...];

(FCC - 2019 - Prefeitura de Manaus - AM - Técnico de Tecnologia da Informação da Fazenda Municipal)

Atribuiu-se uma tarefa a um técnico para, a partir da seguinte tabela de um banco de dados relacional: Item (ID, Nome, Tipo, Valor), gerar o comando SQL ANSI para obter o valor médio de cada tipo de item, apenas para aqueles tipos que comecem com a letra 'S'. O comando SQL solicitado é

- a) `SELECT Tipo, Σ Valor/Number(Tipo); FROM Item WHERE Tipo IS 'S%' COUNT Tipo;`
- b) `SELECT Tipo, Σ Valor/Number(Tipo) FROM Item ORDER BY Tipo ('S%');`
- c) `SELECT Tipo, AVG (Valor) FROM Item GROUP BY Tipo ('S%');`
- d) `SELECT Tipo, AVG (Valor) FROM Item WHERE Tipo LIKE 'S%' GROUP BY Tipo;`
- e) `SELECT Tipo, Total (Valor)/Number (Tipo) FROM Item WHERE Tipo START 'S%' HAVING Tipo IN GROUP;`

(FCC - 2018 - DPE-AM - Assistente Técnico de Defensoria - Programador)

Considere que exista uma tabela denominada TabProcessos que possui os seguintes campos: NroProcesso (tipo varchar), NroVara (tipo int), CPFAutor (tipo varchar). Para que seja feita uma consulta que resulte no total de processos em cada uma das varas, o comando SQL correto é

- a) `SELECT NroVara, COUNT (NroProcesso) FROM TabProcessos;`
- b) `SELECT NroVara, COUNT (NroProcesso) FROM TabProcessos GROUP BY NroVara;`
- c) `SELECT SUM (NroProcesso) FROM TabProcessos GROUP BY NroVara;`
- d) `SELECT COUNT (NroProcesso) FROM TabProcessos HAVING NroVara;`
- e) `SELECT NroVara, SUM (NroProcesso) FROM TabProcessos GROUP BY NroVara;`

(FCC - 2017 - ARTESP - Agente de Fiscalização à Regulação de Transporte - Tecnologia de Informação) Considere, por hipótese, que no banco de dados da ARTESP exista uma tabela denominada TabPSAD com os campos IdRodovia, Posto_Servico, Area_Descanso, Vagas_Estacionamento. Um Agente de Fiscalização à Regulação de Transporte utilizou os comandos SQL para fazer consultas aos dados desta tabela:

I. `SELECT IdRodovia FROM TabPSAD WHERE Vagas_Estacionamento > 300;`

II. `SELECT COUNT (DISTINCT IdRodovia) FROM TabPSAD;`

III. `SELECT IdRodovia, ADD (Vagas_Estacionamento) FROM TabPSAD GROUP BY Area_Descanso;`

Está correto o que consta em

- a) II, apenas.
- b) I e III, apenas.
- c) III, apenas.
- d) I e II, apenas.
- e) I, II e III.

(FCC - 2016 - Prefeitura de Teresina - PI - Analista Tecnológico - Analista de Suporte Técnico) Considere a seguinte tabela de um banco de dados relacional:

Item (ID, Nome, Valor, Tipo)

A consulta em linguagem SQL para obter o Tipo e o valor médio dos itens, considerando itens de mesmo tipo é

a) `SELECT Tipo, Valor_Médio`

`FROM Item`

`ORDER BY Tipo`

b) `SELECT Tipo, SUM(Valor)`

`FROM Item`

`HAVING SAME Tipo`

c) SELECT Tipo, AVG (Valor)

FROM Item

GROUP BY Tipo

d) SELECT Tipo, COUNT (Valor)

FROM Item

WHERE Tipo IN SET

e) SELECT Tipo, Média (Valor)

FROM Item

(FGV - 2015 - TJ-PI - Analista Judiciário - Analista de Sistemas / Banco de Dados) Considere a tabela F2 mostrada a seguir com a respectiva instância.

Piloto	Pontos	Equipe
Felipe Massa	117	Williams
Kimi Raikkonen	135	Williams
Lewis Hamilton	363	Mercedes
Nico Rosberg	297	Mercedes
Sebastian Vettel	266	Ferrari
Valtteri Bottas	136	Ferrari

Thiago precisa construir uma consulta SQL que retorne o resultado a seguir.

Equipe	Pilotos	Pontos
Mercedes	2	660
Ferrari	2	402
Williams	2	252

O comando que Thiago deve usar é:

- a) select equipe, sum(1) pilotos, SUM(pontos) pontos from F2 group by equipe order by 3 desc
- b) select equipe, sum(1) pilotos, SUM(pontos) pontos from F2 order by equipe
- c) select equipe, count(distinct equipe) pilotos, SUM(pontos) pontos from F2 group by equipe order by equipe desc
- d) select equipe, count(distinct 1) pilotos, SUM(pontos) pontos from F2 group by equipe order by equipe desc
- e) select equipe, pilotos as sum(1), SUM(pontos) pontos from F2 group by equipe order by 1

(FGV - 2015 - PGE-RO - Analista da Procuradoria - Analista de Sistemas (Negócios, Suporte e Microinformática)) Nos comandos SQL que empregam funções de agregação em conjunto com a cláusula group by, é possível filtrar linhas do resultado depois da aplicação das funções de agregação por meio da cláusula:

- a) Case;
- b) Distinctrow;
- c) Full outer join;
- d) Having;
- e) Where.

(VUNESP - 2018 - IPSM - Analista de Gestão Municipal - Informática TI) Considere a seguinte tabela de um banco de dados relacional:

Item (ID, Nome, Tipo, Peso)

O comando SQL para obter o peso médio por tipo de item, para os tipos que comecem com a letra 'C' e cujo peso médio seja superior a 200 unidades, é:

- a) `SELECT Tipo, AVG(Peso) FROM Item WHERE Tipo IN "C%" AND AVG (Peso) > 200`
- b) `SELECT Tipo, NUM(Peso) FROM Item WHERE Tipo LIKE "C%" AND NUM(Peso) > 200 ORDER BY Tipo`
- c) `SELECT Tipo, AVG(Peso) FROM Item WHERE Tipo LIKE "C%" GROUP BY Tipo HAVING AVG (Peso) > 200`
- d) `SELECT Tipo, MEDIA(Peso) FROM Item WHERE Tipo INSIDE "C%" ORDER BY Tipo HAVING AVG (Peso) > 200`
- e) `SELECT Tipo, COUNT(Peso) FROM Item WHERE Tipo = "C%" HAVING COUNT(Peso) > 200`

(VUNESP - 2016 - MPE-SP - Analista Técnico Científico - Engenheiro de Computação) Considere a seguinte tabela de um banco de dados relacional:

Funcionário (ID, Nome, Função, Salário)

O comando SQL para obter a média dos salários agrupados por Função, apenas para médias superiores a R\$ 2.000,00 é

SELECT Função, AVG (Salário)

FROM Funcionário

GROUP BY Função

X AVG(Salário) > 2000,00

Para que a consulta atenda ao especificado, o valor de X deve ser substituído por:

- a) CASCADE
- b) HAVING
- c) ORDER BY
- d) TOTAL
- e) WHERE

(VUNESP - 2013 - COREN-SP - Administrador de Banco de Dados) Considere a seguinte tabela de um banco de dados relacional:

Item (ID, Nome, Quantidade, Tipo)

O comando SQL para obter o Tipo e a Quantidade média de cada Item, por Tipo de Item, apenas para médias abaixo de 200, tem a seguinte composição:

```
SELECT Tipo, AVG (Quantidade)
```

```
FROM Item
```

Subcomando 1

Subcomando 2

De forma a atender a essa consulta solicitada, os Subcomandos 1 e 2 devem ser, respectivamente,

- a) GROUP BY Tipo e HAVING AVG(Quantidade) < 200.
- b) GROUP BY Tipo e WHERE AVG(Quantidade) < 200
- c) JOIN BY Tipo e HAVING AVG(Quantidade) < 200.
- d) JOIN BY Tipo e WHERE AVG(Quantidade) < 200.
- e) ORDER BY Tipo e WHERE AVG(Quantidade) < 200.

(VUNESP - 2013 - MPE-ES - Agente Especializado - Analista de Banco de Dados)

Considere o seguinte comando incompleto do SQL:

```
SELECT Loja, AVG(Vendas)
```

```
FROM Pedidos
```

```
XXX Loja
```

A cláusula a ser colocada no lugar de XXX, para que o resultado obtido seja o nome das Lojas e a média de Vendas em cada Loja é

- a) FIRST.
- b) GROUP BY.
- c) HAVING.
- d) ORDER BY.
- e) UPPER.



Agrupamentos

Prof. Ramon Souza