

SQL – GROUP BY E HAVING

Group By

Agrupa as linhas de consulta com base nos valores de uma ou mais colunas.

O parâmetro pode ser o nome do campo ou seu número no resultado do comando **Select**.

A cláusula **Group By** é muito útil quando a utilizamos para trabalhar junto às funções do comando **Select**, como encontrar a soma de um campo numérico, dependendo de outro campo (sum), ou simplesmente contar o número de registro agrupados (count).

Exemplos:

- Quantos alunos existem em cada curso?

```
SELECT CURSO, COUNT(*) FROM ALUNOS  
GROUP BY CURSO
```

Quantos alunos estudam por turno e curso?

```
SELECT TURNO, CURSO, COUNT(*) FROM ALUNOS  
GROUP BY 1, CURSO
```

Cláusula Having

Especifica uma condição de filtragem que os grupos devem satisfazer para que sejam incluídos nos resultados da consulta. **HAVING** deve ser utilizada com **GROUP BY**. Ela pode incluir quantas condições de filtragem você desejar, conectadas com o operador **AND** ou **OR**. Você também pode utilizar **NOT** para inverter o valor de uma expressão lógica.

A maior importância da cláusula **Having** é ser utilizada como o **WHERE**, porém para o resultado do **GROUP BY**. No **where**, não há como utilizar as funções do **SELECT** dentro do Where (sum, count, min, max e avg) somente o **HAVING** aceita estas funções na filtragem.

Exemplos:

Quantos alunos existem em cada curso? Exiba apenas os que possuem 44 alunos.

```
SELECT CURSO, COUNT(*) FROM ALUNOS  
GROUP BY CURSO  
HAVING COUNT(*) = 44
```

PROFESSORA	LUCIANA BUHRER
AULA 4	LINGUAGEM SQL – GROUP BY E HAVING

Quantos alunos estudam por turno e curso, onde o menor débito é igual a R\$10,00?

```
SELECT TURNO, CURSO, COUNT(*) FROM ALUNOS  
GROUP BY 1, CURSO  
HAVING MIN(DEBITO) = 10
```

Exercícios para sala de aula – Parte 1

- EXECUTAR O SCRIPT DA AULA 4.

CREATE TABLE TRABALHADORES (

```
    COD          INT AUTO_INCREMENT PRIMARY KEY NOT NULL,  
    NOME         VARCHAR (30) NOT NULL,  
    SEXO         VARCHAR (1) NOT NULL,  
    IDADE        INT NOT NULL,  
    DEPTO        VARCHAR (30) NOT NULL,  
    SITUACAO     VARCHAR (1) NOT NULL
```

);

CREATE TABLE PAGTO (

```
    EMP_COD      INT AUTO_INCREMENT PRIMARY KEY NOT NULL,  
    NOME         VARCHAR (30) NOT NULL,  
    FONE         VARCHAR (5) NOT NULL,  
    SALARIO      NUMERIC(20,2) NOT NULL,  
    CHEFE        VARCHAR (30) NOT NULL,  
    DEPTO        VARCHAR (30) NOT NULL
```

);

PARTE 1 – Crie query e execute no computador:

1. Exibir a média de idade dos **trabalhadores** por sexo.

```
Select sexo, avg(idade) from trabalhadores group by sexo;
```

2. Exibir a média de idade dos **trabalhadores** por sexo, onde a média de idade seja superior a 34 anos.

```
Select sexo, avg(idade) from trabalhadores  
Group by sexo  
Having avg(idade) > 34;
```

3. Exibir a média de idade dos **trabalhadores** por departamento e por sexo, onde a média de idade seja superior a 34 anos.

```
Select depto, sexo, avg(idade) from trabalhadores  
Group by depto, sexo
```

Having avg(idade) > 34;

4. Exibir quantas pessoas trabalham em cada departamento – utilize a tabela **trabalhadores**

```
Select DEPTO, count(*) as quantidade from trabalhadores  
Group by DEPTO;
```

5. Exibir quantos homens e mulheres trabalham na empresa – utilize a tabela **trabalhadores**

```
Select sexo, count(*) as quantidade from trabalhadores  
Group by sexo;
```

6. Exibir quantos homens e mulheres trabalham em cada departamento – utilize a tabela **trabalhadores**

```
Select sexo, depto, count(*) as quantidade from trabalhadores  
Group sexo, depto;
```

7. Exibir a quantidade de **trabalhadores** ativos e inativos

```
Select situacao, count(*) from trabalhadores  
Group by situacao;
```

8. Exibir a quantidade de **trabalhadores** ativos e inativos maiores de 20 anos

```
Select situacao, count(*) from trabalhadores  
Where idade > 20  
Group by situacao;
```

9. Exibir a quantidade homens e mulheres de **trabalhadores** ATIVOS.

```
Select sexo, count(*) from trabalhadores  
Where situacao = 'A'  
Group by sexo;
```

10. Exibir a especialidade médica e a quantidade de profissionais da **tabela médicos**.

```
Select especialidade, count(*) as quantidade from medicos  
Group by especialidade;
```

PROFESSORA	LUCIANA BUHRER
AULA 4	LINGUAGEM SQL – GROUP BY E HAVING

11. Exibir a especialidade médica e a quantidade de profissionais, mas apenas as especialidades que possuam 2 ou mais médicos na área.

```
Select especialidade, count(*) as quantidade from medicos
Group by especialidade
Having count(*) >= 2;
```

PARTE 2

1) Crie a query para exibir a quantidade de cidades existente em cada país, para os países **com mais de 2 cidades**. Utilize a tabela **cidades**.

```
Select pais, count(*) as quantidade_cidades from cidades
Group by pais
Having count(*) > 2;
```

2) Explique cada query abaixo:

a) Select avg(salario) AS MEDIA from pagto;

Exibir a média de salários da tabela PAGTO.

b) Select chefe, avg(salario) AS media from pagto group by chefe;

Exibir os chefes e a média de salários da tabela PAGTO, agrupando por chefe.

c) Select depto, avg(salario) AS media from pagto group by depto;

Exibir o departamento e a média de salários da tabela PAGTO, agrupando por departamento.

d) Select depto, avg(salario) AS media from pagto where salario > 20000 group by depto;

Exibir departamento e média de salários da tabela PAGTO, apenas para salários maiores do que 20000, agrupando por departamento.

e) Select depto, avg(salario) AS media from pagto group by depto having avg(salario) > 22000;

Exibir departamento e média de salários da tabela PAGTO, agrupando por departamento, apenas para os que possuem média salarial acima de 22000.

f) Select depto, avg(salario) AS media from pagto where salario > 20000 group by depto having avg(salario) > 22000;

Exibir departamento e média de salário da tabela PAGTO, somente para salários acima de 20000, agrupados por departamento que contenham média salarial superior a 22000.

3) Crie as queries solicitadas abaixo, utilizando a tabela **Pagto**.

a) Obter, a partir da tabela **Pagto**, o salário mínimo, máximo e médio dos funcionários de cada departamento. O resultado deve possuir 4 colunas nomeadas: departamento, mínimo, máximo e média.

Select depto as departamento, min(salario) as mínimo, max(salario) as máximo,
avg(salario) as média from PAGTO
Group by depto;

b) Obter os códigos de departamento cuja média de salário seja maior que 2.000 reais.

Select dpto, avg(salario) from pagto group by depto having avg(salario) > 2000;

c) Exibir o total de salários de cada departamento da empresa que tenha este total > R\$ 5.000,00.

Select depto, sum(salario) from pagto
Group by depto
Having salario > 5000;

d) Exibir o nome do chefe e a quantidade de subordinados.

Select chefe, count(*) as quantidade_subordinados from PAGTO group by chefe;

PROFESSORA	LUCIANA BUHRER
AULA 4	LINGUAGEM SQL – GROUP BY E HAVING

e) Exibir o nome do chefe e a quantidade de subordinados. Trazer apenas os chefes que possuem **2 ou mais** subordinados.

```
Select chefe, count(*) as quantidade_subordinados from PAGTO
group by chefe
having count(*) >= 2;
```