

## Cláusula HAVING

Considere as tabelas PILOTO e ESCALA abaixo. Foi incluída uma coluna Nro em cada tabela, representando o registro, a fim de facilitar as explicações que seguem.

| PILOTO |      |         |         |           |      |
|--------|------|---------|---------|-----------|------|
| Nro    | Codp | Nomep   | Salario | Companhia | País |
| 1      | p1   | joao    | 1000    | Varig     | BR   |
| 2      | p2   | pedro   | 2000    | Varig     | BR   |
| 3      | p3   | maria   | 1000    | Varig     | BR   |
| 4      | p4   | jose    | 2000    | Tam       | BR   |
| 5      | p5   | antonio | 3000    | Tam       | BR   |
| 6      | p6   | hanz    | 5000    | KLM       | NL   |

| ESCALA |       |      |      |        |
|--------|-------|------|------|--------|
| Nro    | Codv  | data | codp | avião  |
| 7      | rg230 | 2/1  | p1   | MD11   |
| 8      | rg230 | 2/3  | p1   | Boeing |
| 9      | rg300 | 3/3  | p2   | MD11   |
| 10     | rg400 | 4/4  | p2   | MD11   |
| 11     | tm100 | 5/5  | p4   | MD11   |
|        |       |      |      |        |

Consultas com a estrutura SELECT-FROM-WHERE (join) podem responder consultas do tipo PELO MENOS UM. Isto porque a cláusula WHERE seleciona um a um os registros das tabelas listadas na cláusula FROM, e esta última, faz o produto cartesiano dos registros retidos pela cláusula WHERE. Por exemplo, considerando as tabelas PILOTO e ESCALA acima, e a consulta:

a) o nome dos pilotos da Varig com salário > 100  
select nomep  
from piloto  
where companhia = 'Varig' and salario >1000

toma cada registro de PILOTO, e perguntando se atende à restrição da cláusula WHERE.

O mesmo na consulta:

b) o nome dos pilotos da Varig que voam de MD11  
select nomep  
from piloto p, escala e  
where companhia = 'Varig' and avião = 'MD11' and e.codp = p.codp

Os pilotos que foram escalados para pelo menos 1 voo de MD11 estarão no resultado.

A cláusula WHERE seleciona portanto **1 registro de cada vez, aplicando os critérios de seleção nos valores daquele registro.**

Imagine agora as seguintes consultas

- c) As companhias brasileiras que pagam como maior salario a seus pilotos o valor de 2000.
- d) O nome dos pilotos que usam pelo menos dois tipos de aeronave..
- e) O código dos pilotos que voam sempre com uma aeronave diferente.
- f) O código dos pilotos que sempre voam com a mesma aeronave.

Para processar estas consultas, é necessário olhar um conjunto de registros simultaneamente:

c) o conjunto de todos os registros de pilotos de uma dada companhia para ver qual o maior salário  
registros 1, 2, 3 para companhia Varig -> Varig aparece no resultado  
registros 4,5 para companhia Tam -> Tam não aparece no resultado  
registro 6 para companhia KLM --> KLM não aparece no resultado



d) o conjunto de todos as escalas de um dado piloto para ver quantos tipos de aeronave usam registros 7,8 para piloto p1 -> o nome de p1 aparece no resultado  
registros 9,10 para piloto p2 -> o nome de p2 não aparece no resultado  
registro 11 para piloto p4 -> o nome de p4 não aparece no resultado

O mesmo para as consultas e) e f).

Quando o critério de seleção de uma tupla é baseado em um conjunto de outras tuplas, como neste caso, há duas soluções:

- uso da cláusula HAVING, em combinação com cláusula GROUP BY
- uso de subconsultas.

Neste material abordamos o uso da cláusula HAVING. Ela permite fazer consultas cujo critério de seleção envolva um **conjunto** de tuplas, desde que:

- o critério possa ser expresso de forma **quantitativa**, através de uma função de agregação aplicada aos conjuntos produzidos pela cláusula GROUP BY
- os atributos que se deseja visualizar (i.e projetar) sejam compatíveis com a cláusula GROUP BY

Vamos retomar às consultas acima:

c) As companhias brasileiras que pagam como maior salário a seus pilotos o valor de 2000.

c.1. resposta errada ( usando junção)

```
select companhia
from piloto
where salario = 2000 and pais = 'BR'
```

retorna a companhia brasileira que paga a PELO MENOS um de seus empregados 2000. No caso, retornaria a TAM, que paga 2000 a um de seus empregados, mas não é o maior salário que paga.

c.2. resposta errada ( usando função de agregação na cláusula WHERE)

```
select companhia
from piloto
where país = 'BR' and max(salario) = 2000
```

erro de **sintaxe**. A cláusula WHERE envolve uma única tupla por vez, logo, não pode ser usadas funções de agregação nela.

c.3. resposta certa ( usando HAVING)

```
select companhia
from piloto
where país = 'BR'
group by companhia
having max(salario) = 2000
```

Quando há uso de cláusula GROUP BY – HAVING, a consulta processa da seguinte forma:

- 1) produz “tabelão” resultante do processamento de FROM-WHERE
- 2) entrega o resultado ao GROUP BY, que divide tabelão em subgrupos
- 3) entrega subgrupos a HAVING, que aplica a condição ao GRUPO, eliminando ou mantendo o GRUPO
- 4) entrega os subgrupos selecionados ao SELECT
- 5) ordena a visualização destes resultados, se há cláusula ORDER BY

Vejamos no exemplo:



Passo 1. (removidas tuplas onde país não é BR)

| Nro | Codp | Nomep   | Salario | Companhia | País |
|-----|------|---------|---------|-----------|------|
| 1   | p1   | joao    | 1000    | Varig     | BR   |
| 2   | p2   | pedro   | 2000    | Varig     | BR   |
| 3   | p3   | maria   | 1000    | Varig     | BR   |
| 4   | p4   | jose    | 2000    | Tam       | BR   |
| 5   | p5   | antonio | 3000    | Tam       | BR   |

Passo 2. Divide em subgrupos onde o valor de companhia é igual (representados com cores distintas)

| Nro | Codp | Nomep   | Salario | Companhia | País |
|-----|------|---------|---------|-----------|------|
| 1   | p1   | joao    | 1000    | Varig     | BR   |
| 2   | p2   | pedro   | 2000    | Varig     | BR   |
| 3   | p3   | maria   | 1000    | Varig     | BR   |
| 4   | p4   | jose    | 2000    | Tam       | BR   |
| 5   | p5   | antonio | 3000    | Tam       | BR   |

Passo 3: seleciona grupo(s) que atendem critério de HAVING

| Nro | Codp | Nomep | Salario | Companhia | País |
|-----|------|-------|---------|-----------|------|
| 1   | p1   | joao  | 1000    | Varig     | BR   |
| 2   | p2   | pedro | 2000    | Varig     | BR   |
| 3   | p3   | maria | 1000    | Varig     | BR   |

Passo 4 : mostra resultado para cada grupo (uma linha por grupo)

| Companhia |
|-----------|
| Varig     |

d) O nome dos pilotos que usam pelo menos dois tipos de aeronave..

```
select nomep
from piloto p, escala e
where p.codp = e.codp
group by nomep
having count(distinct aviao) >= 2
```

1. Passo 1 (produz tabelao resultante da junção de cada piloto com sua respectiva escala)

| Nro | Codp | Nomep | Salario | Companhia | País | Nro | Codv  | data | codp | avião  |
|-----|------|-------|---------|-----------|------|-----|-------|------|------|--------|
| 1   | p1   | joao  | 1000    | Varig     | BR   | 7   | rg230 | 2/1  | p1   | MD11   |
| 1   | p1   | joao  | 1000    | Varig     | BR   | 8   | rg230 | 2/3  | p1   | Boeing |
| 2   | p2   | pedro | 2000    | Varig     | BR   | 9   | rg300 | 3/3  | p2   | MD11   |
| 2   | p2   | pedro | 2000    | Varig     | BR   | 10  | rg400 | 4/4  | p2   | MD11   |
| 4   | p4   | jose  | 2000    | Tam       | BR   | 11  | tm100 | 5/5  | p4   | MD11   |

2. Passo 2 : criar subgrupos onr nomep é o mesmo

| Nro | Codp | Nomep | Salario | Companhia | País | Nro | Codv  | data | codp | avião  |
|-----|------|-------|---------|-----------|------|-----|-------|------|------|--------|
| 1   | p1   | joao  | 1000    | Varig     | BR   | 7   | rg230 | 2/1  | p1   | MD11   |
| 1   | p1   | joao  | 1000    | Varig     | BR   | 8   | rg230 | 2/3  | p1   | Boeing |
| 2   | p2   | pedro | 2000    | Varig     | BR   | 9   | rg300 | 3/3  | p2   | MD11   |
| 2   | p2   | pedro | 2000    | Varig     | BR   | 10  | rg400 | 4/4  | p2   | MD11   |
| 4   | p4   | jose  | 2000    | Tam       | BR   | 11  | tm100 | 5/5  | p4   | MD11   |

3. Passo 3: elimina subgrupos que não atendem a condição de having

| Nro | Codp | Nomep | Salario | Companhia | País | Nro | Codv  | data | codp | avião  |
|-----|------|-------|---------|-----------|------|-----|-------|------|------|--------|
| 1   | p1   | joao  | 1000    | Varig     | BR   | 7   | rg230 | 2/1  | p1   | MD11   |
| 1   | p1   | joao  | 1000    | Varig     | BR   | 8   | rg230 | 2/3  | p1   | Boeing |



Passo 4: mostra resultado para cada grupo (uma linha por grupo)

| Nomep |
|-------|
| joao  |

e) O código dos pilotos que voam sempre com uma aeronave diferente.  
neste caso temos que contar quantas escalas cada piloto fez, e comparar com o número de tipos de avião

```
select codp
from escala
group by codp
having count(*) = count(distinct aviao)
```

f) O código dos pilotos que sempre voam com a mesma aeronave.  
neste caso temos que garantir que em todas as tuplas de um mesmo piloto, temos o mesmo valor no atributo avião

```
select codp
from escala
group by codp
having count(distinct aviao) = 1
```

Observe que os atributos a projetar devem ser compatíveis com a cláusula GROUP BY. Por exemplo, se a consulta f) fosse “o nome e o destino dos pilotos que sempre voam com a mesma aeronave”, o resultado não seria compatível, isto é, agrupa-se por código de piloto, mas deseja-se mostrar destinos de vôos, sendo que há possivelmente vários para cada piloto. Neste caso, é necessário usar subconsultas.

Agora faça os exercícios da aula de laboratório. BOM TRABALHO.