

Seleção de dados no SQL da Oracle: GROUP BY, HAVING e NULL

Paulino Ng

2020-10-28

GROUP BY

- ▶ Exemplo:

```
select avg(preco)
from titulos;
```

```
select tipo, avg(preco), sum(preco)
from titulos
group by tipo;
```

- ▶ Observe a diferença do resultado da função de agregação avg()

grupos dentro de grupos

```
select subs_id, tipo, count(tipo)
from titulos
group by subs_id, tipo
```

NULL

- ▶ o NULL serve nos bancos relacionais para representar a ausência de valores
- ▶ o NULL apresenta tantos problemas que o Date propôs a sua eliminação de bancos de dados relacionais
- ▶ Existem duas situações distintas para o uso do NULL:
 - ▶ o dado não existe no momento da coleta
 - ▶ o dado é incompatível com os outros dados e não pode existir nesta situação

Problemas com o NULL

- ▶ como fazer operações com o NULL?
- ▶ podemos agrupar as linhas com um campo NULL
- ▶ como comparar um campo NULL? $=0$, $>$, $<$
- ▶ suponha que você seleciona todos os livros com preço maior do que 500, conta eles, todos livros com preço menor do que 500, conta eles e conta todos os livros com preço igual a 500. Você poderia pensar que a soma das 3 contagens dá a contagem total dos livros, mas . . . E se houver um ou mais livros sem preço?

Uso do HAVING

```
select tipo, count(*)  
from titulos  
group by tipo  
having count(*) > 1;
```

```
select tipos  
from titulos  
group by tipo  
having tipo like 'E%';
```

Mais exemplos

- ▶ A diferença entre o `where` e o `having` é quando a condição é calculada, no `where` a condição é calculada antes do agrupamento das linhas, o `having` é calculado depois.

```
select subs_id, tipo, count(avanco)
from titulos
where avanco > 1000
group by subs_id, tipo;
```

```
select subs_id, tipo, count(avanco)
from titulos
group by subs_id, tipo
having avanco > 1000;
```