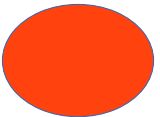


Modelo Relacional

SQL – Consultas aninhadas - FROM

Prof. André Britto



Consultas aninhadas (subselect)

- Até aqui já aprendemos:
 - Fazer consultas básicas;
 - Fazer junções;
 - Funções de agregação e group by;
 - Algumas funções básicas de tratamento de string.
- Porém, não conseguimos responder algumas perguntas somente com esses comandos.

Consultas aninhadas (subselect)

- Conseguimos calcular a média dos salários dos professores por departamento.
 - Mas, se quisermos o nome do professor, o nome do seu departamento e a média de salário?
 - Se quisermos a matrícula dos professores que recebem salário maior do que a média de salários do seu departamento?

Consultas aninhadas (subselect)

- Podemos utilizar consultas aninhadas e assim construir consultas mais complexas.
- De forma simples:
SELECT <campo>
FROM <tabela>
WHERE <condição>
- Podemos colocar uma consulta dentro das três cláusulas.

Subconsultas na cláusula FROM

- O resultado de uma consulta select-from-where é sempre um relação.
- Assim, podemos utilizar o retorno dessa consulta como uma tabela na cláusula FROM.

Subconsultas na cláusula FROM

- Para fazer isto, colocamos uma consulta mais interna na cláusula from:
 - É obrigatório dar um nome para a relação resultante.
 - É interessante dar nomes para os campos da nova relação.

Subconsultas na cláusula FROM

- Liste os departamentos que possuam média salarial maior do que 3500.
 - Primeiro temos que encontrar os departamentos e suas médias salariais.
 - Em seguida selecionar somente aqueles com média maior que 3500.
 - * Essa consulta pode ser feita usando HAVING

Subconsultas na cláusula FROM

- `SELECT p.departamento AS depto, avg(salario) AS
media_salario
FROM professor p JOIN cargo c ON (p.cargo = c.id_cargo)
GROUP BY p.departamento`
- Agora, vamos enxergar o resultado dessa consulta como uma tabela

Subconsultas na cláusula FROM

- **SELECT** depto, media_salario
FROM (SELECT p.departamento AS depto, avg(salario) AS media_salario
FROM professor p JOIN cargo c ON (p.cargo = c.id_cargo)
GROUP BY p.departamento) **AS d**
WHERE media_salario > 3500
 - * O postgres obriga a criação de um nome para o resultado da consulta

Cláusula WITH

- A cláusula WITH permite a criação de tabelas temporárias que são criadas somente para consulta.
- É utilizada para aumentar a legibilidade de uma consulta.
- Pode ser utilizado para reorganizar consultas aninhadas tanto da cláusula FROM, quanto da cláusula WHERE.
- As tabelas das consultas aninhadas são definidas antes da consulta.

Subconsultas na cláusula FROM

```
WITH salario_depto AS (SELECT p.departamento AS depto,  
avg(salario) AS media_salario
```

```
FROM professor p JOIN cargo c ON (p.cargo = c.id_cargo)
```

```
GROUP BY p.departamento)
```

```
SELECT depto, media_salario
```

```
FROM salario_depto
```

```
WHERE media_salario > 3500
```

Subconsultas na cláusula FROM

- Lista o nome do professor, o salario, o departamento e a media salarial por departamento:
- Consulta interna

```
SELECT departamento AS depto, avg(salario) AS media  
FROM professor p1 JOIN cargo c ON (p1.cargo = c.id_cargo)  
GROUP BY p1.departamento
```

Subconsultas na cláusula FROM

- SELECT p2.mat_professor, up.primeiro_nome, up.sobrenome,
c2.salario, ps.depto, ps.media
FROM (SELECT departamento AS depto, avg(salario) AS media
FROM professor p1 JOIN cargo c ON (p1.cargo = c.id_cargo)
GROUP BY p1.departamento) AS ps
JOIN professor p2 ON(ps.depto = p2.departamento)
JOIN usuario up USING (cpf)
JOIN cargo c2 ON (cargo = c2.id_cargo))

Leitura recomendada

ELMASRI, R; NAVATHE, S.B. **Sistemas de Banco de Dados**, Addison Wesley, 6ª Edição.

- Capítulo 4 e 5

Silberschatz, A; Korth H.F.; Sudarshan S. **Sistemas de Banco de Dados**, Editora Campus, 6ª Edição.

- Capítulo 3, 4 e 5

