

Banco de dados

Consultas SQL – Parte 2

Prof. Eldane Vieira

Consultas aninhadas

- Nem sempre conseguimos as informações que queremos com um consulta simples.
- Para obter informações mais elaboradas das tabelas, podemos aninhar consultas.
- A consulta aninhada é considerada uma consulta secundária ou sub consulta que vem embutida na primária ou principal.

Esquema de tabelas

- Os exemplos nesta aula serão feitos sobre os seguintes esquemas de tabela:
 - Cliente (ID_Cliente, nome, endereco, cidade, telefone, tipo)
 - Empresa (ID_Empresa, CNPJ, nome, endereco, cidade, telefone)
 - Cliente_Empresa (ID_Empresa, ID_Cliente)
 - Produto (ID_Produto, nome, qnt_vendida, valor, total_produzido)
 - Produto_Empresa(ID_Produto,ID_Empresa)
 - Produto_Comprado_Cliente(ID_Produto,ID_Cliente, quantidade)

Consultas aninhadas

- Consulta 1: Listar o(s) produto(s) com menor valor.
 - `select min(valor)`
`from produto;`
 - A consulta acima retorna o menor valor, mas e se quisermos o nome do produto?
 - `select min(valor), nome`
`from produto;`
 - A consulta acima está errada, pois apresenta uma associação errada entre valor e nome do produto.

Consultas aninhadas

- Para sabermos o menor valor de um produto e seu nome, devemos utilizar a consulta pelo menor valor do produto como uma condição:
 - select nome, valor
from produto
where valor = (select min(valor)
from produto);

Cliente (<u>ID_Cliente</u> , nome, endereco, cidade, telefone, tipo)
Empresa (<u>ID_Empresa</u> , CNPJ, nome, endereco, cidade, telefone)
Cliente_Empresa (<u>ID_Empresa</u> , <u>ID_Cliente</u>)
Produto (<u>ID_Produto</u> , nome, qnt_vendida, valor, total_produzido)
Produto_Empresa(<u>ID_Produto</u> , <u>ID_Empresa</u>)
Produto_Comprado_Cliente(<u>ID_Produto</u> , <u>ID_Cliente</u> , quantidade)

Consulta com mais de uma tabela

- A cláusula FROM do comando SELECT permite consultar simultaneamente mais de uma tabela.
- Mas essas tabelas devem estar relacionadas e devemos mostrar esse relacionamento.
- Por exemplo, a tabela Cliente_Empresa está relacionada a tabela Cliente pelo atributo ID_Cliente, e com a tabela Empresa pelo atributo ID_Empresa.

Consulta com mais de uma tabela

- Consulta 2: Listar os nomes dos clientes que já compraram um produto.
 - `select distinct cliente.nome`
`from cliente, produto_comprado_cliente`
`where cliente.ID_Cliente = produto_comprado_cliente.ID_Cliente;`

Cliente (<u>ID_Cliente</u> , nome, endereco, cidade, telefone, tipo)
Empresa (<u>ID_Empresa</u> , CNPJ, nome, endereco, cidade, telefone)
Cliente_Empresa (<u>ID_Empresa</u> , <u>ID_Cliente</u>)
Produto (<u>ID_Produto</u> , nome, qnt_vendida, valor, total_produzido)
Produto_Empresa(<u>ID_Produto</u> , <u>ID_Empresa</u>)
Produto_Comprado_Cliente(<u>ID_Produto</u> , <u>ID_Cliente</u> , quantidade)

Consulta com mais de uma tabela

- Consulta 3: Listar os clientes que nunca compraram nenhum produto.

- select distinct c.nome

- from cliente c

- where c.ID_Cliente not in (select distinct cliente.ID_Cliente

- from cliente, produto_comprado_cliente

- where cliente.ID_Cliente = produto_comprado_cliente.ID_Cliente);

Cliente (<u>ID_Cliente</u> , nome, endereco, cidade, telefone, tipo)
Empresa (<u>ID_Empresa</u> , CNPJ, nome, endereco, cidade, telefone)
Cliente_Empresa (<u>ID_Empresa</u> , <u>ID_Cliente</u>)
Produto (<u>ID_Produto</u> , nome, qnt_vendida, valor, total_produzido)
Produto_Empresa(<u>ID_Produto</u> , <u>ID_Empresa</u>)
Produto_Comprado_Cliente(<u>ID_Produto</u> , <u>ID_Cliente</u> , quantidade)

Consulta com mais de uma tabela

- Consulta 4: Listar os nomes dos produtos comprados por cada cliente.
 - select c.nome AS Cliente, p.nome AS Produto
from cliente c, produto p, produto_comprado_cliente pcc
where c.ID_Cliente = pcc.ID_Cliente AND
p.ID_Produto = pcc.ID_Produto
order by c.nome;

Cliente (<u>ID_Cliente</u> , nome, endereco, cidade, telefone, tipo)
Empresa (<u>ID_Empresa</u> , CNPJ, nome, endereco, cidade, telefone)
Cliente_Empresa (<u>ID_Empresa</u> , <u>ID_Cliente</u>)
Produto (<u>ID_Produto</u> , nome, qnt_vendida, valor, total_produzido)
Produto_Empresa(<u>ID_Produto</u> , <u>ID_Empresa</u>)
Produto_Comprado_Cliente(<u>ID_Produto</u> , <u>ID_Cliente</u> , quantidade)

Junção de tabelas

- A linguagem SQL oferece recursos para fazer a junção de tabelas, por exemplo, o JOIN na cláusula FROM, em que podemos indicar o tipo da junção:
 - CROSS JOIN: é feito um produto cartesiano entre as tabelas em questão, ou seja, os registros da tabela 1 são unidos com todos os registros da tabela 2.

Junção de tabelas

- Consulta 5: Junção das tabelas Cliente e Empresa através do CROSS JOIN.
 - select*
from cliente cross join empresa;

Cliente (<u>ID_Cliente</u> , nome, endereco, cidade, telefone, tipo)
Empresa (<u>ID_Empresa</u> , CNPJ, nome, endereco, cidade, telefone)
Cliente_Empresa (<u>ID_Empresa</u> , <u>ID_Cliente</u>)
Produto (<u>ID_Produto</u> , nome, qnt_vendida, valor, total_produzido)
Produto_Empresa(<u>ID_Produto</u> , <u>ID_Empresa</u>)
Produto_Comprado_Cliente(<u>ID_Produto</u> , <u>ID_Cliente</u> , quantidade)

ID_Cliente	nome	endereco	cidade	telefone	tipo	ID_Empresa	CNPJ	nome	endereco	cidade	telefone
1	Ana Ribeiro	Rua 1, 234	Uberlândia	3432221111	varejista	5	115	empresa5	Av. Três, 1283	Uberlândia	3416545678
1	Ana Ribeiro	Rua 1, 234	Uberlândia	3432221111	varejista	4	114	empresa4	Av. Dois, 1323	Uberlândia	3422345678
1	Ana Ribeiro	Rua 1, 234	Uberlândia	3432221111	varejista	3	113	empresa3	Av. Um, 1233	Uberlândia	3412315678
1	Ana Ribeiro	Rua 1, 234	Uberlândia	3432221111	varejista	2	112	empresa2	Av. Dois, 1123	Uberlândia	3412345658
1	Ana Ribeiro	Rua 1, 234	Uberlândia	3432221111	varejista	1	111	empresa1	Av. Um, 123	Uberlândia	3412345678
2	Leandro Silva	Rua 1, 1234	Uberlândia	3432223111	varejista	5	115	empresa5	Av. Três, 1283	Uberlândia	3416545678
2	Leandro Silva	Rua 1, 1234	Uberlândia	3432223111	varejista	4	114	empresa4	Av. Dois, 1323	Uberlândia	3422345678
2	Leandro Silva	Rua 1, 1234	Uberlândia	3432223111	varejista	3	113	empresa3	Av. Um, 1233	Uberlândia	3412315678
2	Leandro Silva	Rua 1, 1234	Uberlândia	3432223111	varejista	2	112	empresa2	Av. Dois, 1123	Uberlândia	3412345658
2	Leandro Silva	Rua 1, 1234	Uberlândia	3432223111	varejista	1	111	empresa1	Av. Um, 123	Uberlândia	3412345678
3	José Santos	Rua 10, 2234	Uberlândia	3432221431	atacadista	5	115	empresa5	Av. Três, 1283	Uberlândia	3416545678
3	José Santos	Rua 10, 2234	Uberlândia	3432221431	atacadista	4	114	empresa4	Av. Dois, 1323	Uberlândia	3422345678
3	José Santos	Rua 10, 2234	Uberlândia	3432221431	atacadista	3	113	empresa3	Av. Um, 1233	Uberlândia	3412315678
3	José Santos	Rua 10, 2234	Uberlândia	3432221431	atacadista	2	112	empresa2	Av. Dois, 1123	Uberlândia	3412345658
3	José Santos	Rua 10, 2234	Uberlândia	3432221431	atacadista	1	111	empresa1	Av. Um, 123	Uberlândia	3412345678
4	Maria Custódio	Rua 3, 734	Uberlândia	3432111111	varejista	5	115	empresa5	Av. Três, 1283	Uberlândia	3416545678
4	Maria Custódio	Rua 3, 734	Uberlândia	3432111111	varejista	4	114	empresa4	Av. Dois, 1323	Uberlândia	3422345678
4	Maria Custódio	Rua 3, 734	Uberlândia	3432111111	varejista	3	113	empresa3	Av. Um, 1233	Uberlândia	3412315678
4	Maria Custódio	Rua 3, 734	Uberlândia	3432111111	varejista	2	112	empresa2	Av. Dois, 1123	Uberlândia	3412345658
4	Maria Custódio	Rua 3, 734	Uberlândia	3432111111	varejista	1	111	empresa1	Av. Um, 123	Uberlândia	3412345678
5	Gabriel Mach...	Rua 7, 2334	Uberlândia	3432229811	varejista	5	115	empresa5	Av. Três, 1283	Uberlândia	3416545678
5	Gabriel Mach...	Rua 7, 2334	Uberlândia	3432229811	varejista	4	114	empresa4	Av. Dois, 1323	Uberlândia	3422345678
5	Gabriel Mach...	Rua 7, 2334	Uberlândia	3432229811	varejista	3	113	empresa3	Av. Um, 1233	Uberlândia	3412315678
5	Gabriel Mach...	Rua 7, 2334	Uberlândia	3432229811	varejista	2	112	empresa2	Av. Dois, 1123	Uberlândia	3412345658
5	Gabriel Mach...	Rua 7, 2334	Uberlândia	3432229811	varejista	1	111	empresa1	Av. Um, 123	Uberlândia	3412345678
6	Carol Oliveira	Rua 15, 2534	Uberlândia	3432223411	atacadista	5	115	empresa5	Av. Três, 1283	Uberlândia	3416545678
6	Carol Oliveira	Rua 15, 2534	Uberlândia	3432223411	atacadista	4	114	empresa4	Av. Dois, 1323	Uberlândia	3422345678
6	Carol Oliveira	Rua 15, 2534	Uberlândia	3432223411	atacadista	3	113	empresa3	Av. Um, 1233	Uberlândia	3412315678
6	Carol Oliveira	Rua 15, 2534	Uberlândia	3432223411	atacadista	2	112	empresa2	Av. Dois, 1123	Uberlândia	3412345658
6	Carol Oliveira	Rua 15, 2534	Uberlândia	3432223411	atacadista	1	111	empresa1	Av. Um, 123	Uberlândia	3412345678

Junção de tabelas

- INNER JOIN: retorna apenas linhas que satisfaçam a condição, ou seja, é feita uma junção apenas dos registros que têm um correspondente na outra tabela.
 - Registros sem correspondência não são apresentados na consulta.

Junção de tabelas

- Consulta 6: Uso da junção INNER JOIN.
 - select c.ID_Cliente, c.Nome, pcc.ID_Produto, pcc.quantidade AS QuantidadeComprada
from cliente c inner join
produto_comprado_cliente pcc
on c.ID_Cliente = pcc.ID_Cliente;
- Comando equivalente:
 - select c.ID_Cliente, c.Nome, pcc.ID_Produto, pcc.quantidade AS QuantidadeComprada
from cliente c, produto_comprado_cliente pcc
where c.ID_Cliente = pcc.ID_Cliente;

ID_Cliente	Nome	ID_Produto	QuantidadeComprada
1	Ana Ribeiro	1	5
2	Leandro Silva	1	5
3	José Santos	2	8
5	Gabriel Machado	2	7
4	Maria Custódio	3	30
5	Gabriel Machado	3	4
3	José Santos	4	32
5	Gabriel Machado	5	100
1	Ana Ribeiro	6	15
2	Leandro Silva	6	15

Cliente (ID_Cliente, nome, endereco, cidade, telefone, tipo)
Empresa (ID_Empresa, CNPJ, nome, endereco, cidade, telefone)
Cliente_Empresa (ID_Empresa, ID_Cliente)
Produto (ID_Produto, nome, qnt_vendida, valor, total_produzido)
Produto_Empresa(ID_Produto,ID_Empresa)
Produto_Comprado_Cliente(ID_Produto,ID_Cliente, quantidade)

Junção de tabelas

- LEFT JOIN: nesta junção são retornadas todas as linhas da tabela esquerda em uma junção.
 - Todas as linhas da tabela do lado esquerdo do comando LEFT JOIN são retornadas independentemente de terem correspondência ou não.
 - Se não houver correspondência o valor NULL é usado para preencher os campos.

Junção de tabelas

- Consulta 7: Uso da junção LEFT JOIN.
 - select c.ID_Cliente, c.Nome, pcc.ID_Produto, pcc.quantidade AS QuantidadeComprada
from cliente c left join
produto_comprado_cliente pcc
on c.ID_Cliente = pcc.ID_Cliente;
 - Mostra todos os registros da tabela à esquerda (cliente), ou seja, inclui os clientes que não compraram nenhum produto.
 - Se não atender as condições o valor NULL é retornado, preenchendo os campos do lado direito.

ID_Cliente	Nome	ID_Produto	QuantidadeComprada
1	Ana Ribeiro	1	5
1	Ana Ribeiro	6	15
2	Leandro Silva	1	5
2	Leandro Silva	6	15
3	José Santos	2	8
3	José Santos	4	32
4	Maria Custódio	3	30
5	Gabriel Machado	2	7
5	Gabriel Machado	3	4
5	Gabriel Machado	5	100
6	Carol Oliveira	NULL	NULL

Cliente (ID_Cliente, nome, endereco, cidade, telefone, tipo)
Empresa (ID_Empresa, CNPJ, nome, endereco, cidade, telefone)
Cliente_Empresa (ID_Empresa, ID_Cliente)
Produto (ID_Produto, nome, qnt_vendida, valor, total_produzido)
Produto_Empresa(ID_Produto,ID_Empresa)
Produto_Comprado_Cliente(ID_Produto,ID_Cliente, quantidade)

Junção de tabelas

- RIGHT JOIN: é a forma inversa do LEFT JOIN, ou seja, a referência fica à direita.

Junção de tabelas

- Consulta 8: Uso da junção RIGHT JOIN.

- select c.ID_Cliente, c.Nome, pcc.ID_Produto, pcc.quantidade AS QuantidadeComprada

from cliente c right join
produto_comprado_cliente pcc

on c.ID_Cliente = pcc.ID_Cliente
and c.tipo like 'atacadista';

- Mostra todos os registros da tabela à direita (produto_comprado_cliente).
 - Se não atender as condições o valor NULL é retornado, preenchendo os campos do lado esquerdo.

ID_Cliente	Nome	ID_Produto	QuantidadeComprada
NULL	NULL	1	5
NULL	NULL	1	5
3	José Santos	2	8
NULL	NULL	2	7
NULL	NULL	3	30
NULL	NULL	3	4
3	José Santos	4	32
NULL	NULL	5	100
NULL	NULL	6	15
NULL	NULL	6	15

Cliente (ID_Cliente, nome, endereco, cidade, telefone, tipo)

Empresa (ID_Empresa, CNPJ, nome, endereco, cidade, telefone)

Cliente_Empresa (ID_Empresa, ID_Cliente)

Produto (ID_Produto, nome, qnt_vendida, valor, total_produzido)


Produto_Empresa(ID_Produto,ID_Empresa)

Produto_Comprado_Cliente(ID_Produto,ID_Cliente, quantidade)

UNION

- Combina o resultado de duas ou mais consultas.
- Elimina registros duplicados.
- Para fazer a união, as tabelas devem ter a mesma quantidade de colunas e possuírem o mesmo escopo (tipo de dados).
- Consulta 9:

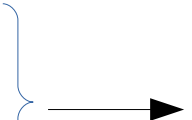
– select p.nome
from produto p
where p.valor >= 100



nome
p1
p4
p5
p6


union

select p.nome
from produto p
where p.valor <= 1000;



nome
p1
p2
p3
p4
p5
p6

Resultado do union




nome
p1
p4
p5
p6
p2
p3

UNION ALL

- Combina o resultado de duas ou mais consultas.
- Não elimina registros duplicados.
- Para fazer a união, as tabelas devem ter a mesma quantidade de colunas e possuírem o mesmo escopo (tipo de dados).
- Consulta 9:

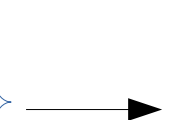
– select p.nome
from produto p
where p.valor >= 100



nome
p1
p4
p5
p6


union all

select p.nome
from produto p
where p.valor <= 1000;



nome
p1
p2
p3
p4
p5
p6

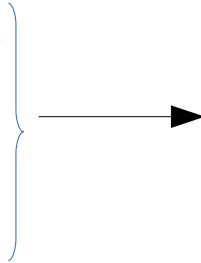
Resultado do union all



nome
p1
p4
p5
p6
p1
p2
p3
p4
p5
p6

GROUP BY

- Serve para obter valores totalizados, agrupando um conjunto de dados.
- Com o group by você terá só uma ocorrência de cada instância do atributo determinado para o agrupamento.
 - Se originalmente houver mais de uma ocorrência por instância, talvez seja preciso dizer o que fazer com certos valores, como por exemplo aplicar a função SUM().
- O uso do group by garante que uma determinada operação será feita para cada elemento presente neste agrupamento.
- Consulta 10: Apresente a quantidade de produtos produzidos por cada empresa.
 - ```
select e.nome as Empresa, count(p.ID_Produto) as QntProdutos
from empresa e, produto_empresa p
where e.ID_Empresa=p.ID_Empresa
group by e.ID_Empresa;
```



| Empresa  | QntProdutos |
|----------|-------------|
| empresa1 | 1           |
| empresa2 | 1           |
| empresa3 | 1           |
| empresa4 | 1           |
| empresa5 | 2           |

# Exercícios

# Utilizando a base de dados (BaseConsultas) disponível no Disco Virtual responda usando comandos SQL.

- 1) Dê o comando que liste o nome das empresas e de seus respectivos produtos.
- 2) Dê o comando que liste a quantidade de itens comprados por cada cliente.
- 3) Dê o comando que liste o valor gasto por cada cliente na compra de cada produto.
- 4) Dê o comando que liste o total gasto por cada cliente na compra de produtos.
- 5) Dê o comando que liste todos os clientes que já compraram ou não um produto, mas caso tenha comprado apresente o nome do produto comprado e sua quantidade.

Cliente (ID\_Cliente, nome, endereco, cidade, telefone, tipo)

Empresa (ID\_Empresa, CNPJ, nome, endereco, cidade, telefone)

Cliente\_Empresa (ID\_Empresa, ID\_Cliente)

Produto (ID\_Produto, nome, qnt\_vendida, valor, total\_produzido)

Produto\_Empresa(ID\_Produto,ID\_Empresa)

Produto\_Comprado\_Cliente(ID\_Produto,ID\_Cliente, quantidade)