



escola  
britânica de  
artes criativas  
& tecnologia

# SQL para análise de dados



# TÉCNICAS AVANÇADAS



## GUIA DA AULA 1



# Aplique subqueries – Teoria

● Introdução

● Sobre  
Subqueries



Acompanhe aqui  
os temas que  
serão tratados  
na videoaula



## Introdução

Nessa aula, usaremos a seguinte tabela:

```
CREATE TABLE transacoes (  
  id_cliente INT,  
  id_transacao INT,  
  data_compra DATE,  
  valor FLOAT,  
  id_loja varchar(25)  
);
```



Também temos os seguintes valores inseridos na tabela:

```

INSERT INTO transacoes VALUES (1,768805383,'2021-06-10',50.74,'magalu');
INSERT INTO transacoes VALUES (2,768805399,'2021-06-13',30.90,'giraffas');
INSERT INTO transacoes VALUES (3,818770008,'2021-06-05',110.00,'postoshell');
INSERT INTO transacoes VALUES (1,76856563,'2021-07-10',2000.90,'magalu');
INSERT INTO transacoes VALUES (1,767573759,'2021-06-20',15.70,'subway');
INSERT INTO transacoes VALUES (3,818575758,'2021-06-25',2.99,'seveneleven');
INSERT INTO transacoes VALUES (4,764545534,'2021-07-11',50.74,'extra');
INSERT INTO transacoes VALUES (5,76766789,'2021-08-02',10.00,'subway');
INSERT INTO transacoes VALUES (3,8154567758,'2021-08-15',1100.00,'shopee');
  
```



Como resultado da função SELECT, temos a seguinte tabela:

id_cliente	id_transacao	data_compra	valor	id_loja
1	768805383	2021-06-10	50.74	magalu
2	768805399	2021-06-13	30.90	giraffas
3	818770008	2021-06-05	110.00	postoshell
1	76856563	2021-07-10	2000.90	magalu
1	767573759	2021-06-20	15.70	subway
3	818575758	2021-06-25	2.99	seveneleven
4	764545534	2021-07-10	50.74	extra
5	76766789	2021-08-92	10.00	subway
3	8154567758	2021-08-15	1100.00	shopee



Além disso, também utilizaremos a tabela:

```

id_cliente INT,
nome varchar(25),
data_compra DATE,
valor_compra float,
loja_cadastro varchar(25)
);
  
```

Teremos os seguintes dados na tabela:

```

INSERT INTO cliente VALUES (5,'jose', '2020-07-01',500.43,'magalu');
INSERT INTO cliente VALUES (1,'maria','2019-03-02',150.70,'subway');
INSERT INTO cliente VALUES (2,'valentina','2020-01-09',210.99,'postoshell');
INSERT INTO cliente VALUES (4,'joana','2019-05-11',1300.50,'magalu');
INSERT INTO cliente VALUES (6,'fernando','2020-03-02',86.55,'seveneleven');
  
```



Como resultado, temos:

id_cliente	nome	data_compra	valor_compra	loja_cadastro
5	jose	2020-07-01	500.43	cea
1	maria	2019-03-02	150.70	riachuelo
2	valentina	2020-01-09	210.99	zara
4	joana	2019-05-11	1300.50	pontofrio
6	fernando	2020-03-02	86.55	pontofrio





## Sobre Subqueries

Podemos resumir o uso de subqueries quando utilizamos um comando dentro do outro.

Para utilizar, basta inserir um comando

- dentro de um SELECT, INSERT, UPDATE ou DELETE ;
- junto com um operador = , >, <=, >= e LIKE;
- aliado a um WHERE, HAVING e FROM.

A query externa é chamada de **main query** e a interna é chamada de **subquery**.

- a subquery deve ser utilizada com parênteses.
- a subquery em geral é executada primeiro.
- é como se a subquery fosse um comando condicional associado.
- subquery não pode ser associado com ORDER BY.



## Exemplo teórico:

```

SELECT <nome_coluna>
FROM <nome_tabela>
WHERE <nome_coluna> <expressao> <operador>
      ( SELECT <coluna> from <tabela> WHERE ... );
  
```

## Exemplo prático:

```

SELECT id_loja, id_cliente, id_transacao from transacoes
WHERE id_loja IN
(SELECT cliente.loja_cadastro from cliente where cliente.valor_compra > 160 )
  
```



## O que ele executa?

Primeiro a subquery é executada. O resultado de:

```
SELECT cliente.loja_cadastro from cliente where cliente.valor_compra > 160
```

### loja\_cadatro

magalu  
 postoshell  
 magalu



Ou seja, vamos considerar apenas essas lojas na seleção da main query. É como se traduzíssemos para:

```
SELECT id_loja, id_cliente, id_transacao from transacoes
WHERE id_loja IN
('magalu','postoshell')
```

Resultado:

id_loja	id_cliente	id_transacao
magalu	1	768805383
postoshell	3	818770008
magalu	1	76856563

Use a criatividade para gerar novas queries com subqueries!

