

# Grupo05

Lista de exercícios

Semana 01

Eberton Chaves Dias – 11621BSI215

Elisângela Rithiely Evaristo de Souza - 11811BSI251

Gabriel Miranda Silva - 11621BSI222

--01-Nomes das Agências com depósitos (nome pode aparecer repetido)

```
SELECT nome_agencia FROM deposito;
```

--02-Nomes das Agências com depósitos (nome sem repetição)

```
SELECT DISTINCT nome_agencia FROM deposito;
```

--03-Nomes de Clientes com depósitos e empréstimos ao mesmo tempo;

```
SELECT DISTINCT nome_cliente FROM deposito
```

```
intersect
```

```
SELECT DISTINCT nome_cliente FROM emprestimo;
```

--04-Nomes de Clientes com depósitos e empréstimos ao mesmo tempo na agência PUC;

```
SELECT DISTINCT nome_cliente FROM deposito WHERE nome_agencia='PUC'
```

```
intersect
```

```
SELECT DISTINCT nome_cliente FROM emprestimo WHERE nome_agencia='PUC';
```

--Alternativa2 para incluir a condição uma única vez

```
SELECT DISTINCT nome_cliente FROM
```

```
    (    SELECT DISTINCT nome_cliente, nome_agencia FROM deposito
```

```
        intersect
```

```
        select DISTINCT nome_cliente, nome_agencia FROM emprestimo
```

```
    ) AS relatorio
```

```
WHERE relatorio.nome_agencia = 'PUC';
```

--Alternativa3 para usar outro operador (IN)

```
SELECT DISTINCT nome_cliente FROM deposito WHERE nome_agencia='PUC'
AND nome_cliente IN (SELECT DISTINCT nome_cliente FROM emprestimo WHERE
nome_agencia='PUC');
```

--05-Nomes de Clientes com depósitos, mas sem empréstimos na agência PUC;

```
SELECT nome_cliente FROM deposito WHERE nome_agencia='PUC'
except
SELECT nome_cliente FROM emprestimo WHERE nome_agencia='PUC';
```

```
SELECT DISTINCT nome_cliente FROM deposito WHERE nome_agencia='PUC'
AND nome_cliente NOT IN (SELECT DISTINCT nome_cliente FROM emprestimo WHERE
nome_agencia='PUC');
```

--06-Clientes que possuem depósitos ou empréstimos na agencia da PUC

```
SELECT DISTINCT nome_cliente FROM deposito WHERE nome_agencia='PUC'
UNION
SELECT DISTINCT nome_cliente FROM emprestimo WHERE nome_agencia='PUC';
```

--07-Nomes de Clientes de Belo Horizonte

```
SELECT nome_cliente
FROM Cliente
WHERE cidade_cliente='Belo Horizonte';
```

--08-Nomes de clientes que tem 'Santos' no nome

```
SELECT nome_cliente FROM cliente
WHERE nome_cliente LIKE '%Santos%';
```

--09-Nomes de clientes que tem terminam com 'Souza' ou 'Sousa'

```
SELECT nome_cliente  
FROM cliente  
WHERE nome_cliente LIKE '%Sou_a';
```

--10-Total de depósitos para cada nome de cliente

```
SELECT nome_cliente, sum(saldo_deposito)  
FROM deposito GROUP BY nome_cliente;
```

--11-Nomes de clientes e depósitos que estão entre

--R\$ 3.000,00 e R\$ 4.000,00

```
SELECT nome_cliente, saldo_deposito  
FROM deposito  
WHERE saldo_deposito >= 3000  
AND saldo_deposito <= 4000;
```

```
SELECT nome_cliente, saldo_deposito  
FROM deposito  
WHERE saldo_deposito BETWEEN 3000 AND 4000;
```

--12-Nomes de clientes e a soma de depósitos de cada nome,

--quando a soma for maior que R\$ 5.000,00

```
SELECT nome_cliente, sum(saldo_deposito)  
FROM deposito GROUP BY nome_cliente  
HAVING sum(saldo_deposito) > 5000;
```

--13-Nomes de clientes e a soma de depósitos de cada nome,  
--quando a soma estiver entre R\$ 1.000,00 e R\$ 4.000,00,  
--em ordem pela soma

```
SELECT nome_cliente, sum(saldo_deposito)
FROM deposito GROUP BY nome_cliente
HAVING sum(saldo_deposito) BETWEEN 1000 AND 4000
ORDER BY sum(saldo_deposito);
```

--14-Nomes de clientes e a soma de depósitos de cada nome,  
--quando a soma estiver maior do que a média de depósitos,  
--em ordem pela soma

```
SELECT nome_cliente, sum(saldo_deposito)
FROM deposito GROUP BY nome_cliente
HAVING sum(saldo_deposito) > avg(saldo_deposito)
ORDER BY sum(saldo_deposito);
```

--15-Faça uma listagem da soma de depósitos feitos  
--por cada cliente em cada agência. Ordene o resultado  
--pelo nome do cliente e depois pelo nome da agência.

```
SELECT nome_agencia, nome_cliente, sum(saldo_deposito)
FROM deposito
GROUP BY nome_agencia, nome_cliente
ORDER BY nome_cliente, nome_agencia;
```

--16- Selecione os nomes dos clientes e a soma de depósitos  
--feitos por eles em cada agência que foram maiores que todos  
--os depósitos efetuados na agência 'Pampulha'.

--Conjunto1:

```
SELECT nome_cliente, nome_agencia, sum(saldo_deposito)
FROM deposito GROUP BY nome_cliente, nome_agencia;
```

--Conjunto2:

```
SELECT sum(saldo_deposito) FROM deposito
WHERE nome_agencia='Pampulha' GROUP BY nome_agencia;
```

--Conjunto1 HAVING sum > ALL Conjunto2

```
SELECT nome_cliente, nome_agencia, sum(saldo_deposito)
FROM deposito GROUP BY nome_cliente, nome_agencia
HAVING sum(saldo_deposito) > ALL
(SELECT sum(saldo_deposito) FROM deposito
WHERE nome_agencia='Pampulha' GROUP BY nome_agencia);
```

--Funções de agregação:

--Encontre a soma total de depósitos para cada cliente

```
select nome_cliente, sum(saldo_deposito)
from deposito
group by nome_cliente;
```

--Encontre a soma total de depósitos para cada cliente

--e ordene pela ordem descendente dos depósitos

```
select nome_cliente, sum(saldo_deposito)
from deposito
group by nome_cliente
order by sum(saldo_deposito) desc;
```

```
--Conte quantos depósitos realizou cada cliente
select nome_cliente, sum(1)
from deposito
group by nome_cliente;
```

```
--Conte quantos depósitos realizou cada cliente e ordene
--pelo ordem descendente da quantidade de depósitos
select nome_cliente, count(1)
from deposito
group by nome_cliente
order by count(1) desc;
```

```
--Encontre a soma total de depósitos para cada cliente,
--e também a quantidade de depósitos. Mostre o resultado
--ordenado primeiro pela ordem ascendente da quantidade de depósitos
--e depois pela soma total descendente de depósitos de cada cliente
select nome_cliente, sum(saldo_deposito), count(1)
from deposito
group by nome_cliente
order by count(1), sum(saldo_deposito) desc;
```

```
--Encontre o número de depositantes em cada agência
select nome_agencia, count(nome_cliente)
from deposito
group by nome_agencia;
```

```
--Mas a cláusula anterior está errada porque um cliente
--pode fazer mais de um depósito por agência. Veja:
select nome_agencia, nome_cliente, count(nome_cliente)
from deposito group by nome_agencia, nome_cliente
order by nome_agencia;
```

```
--Solução:
select nome_agencia, count( distinct nome_cliente)
from deposito
group by nome_agencia;
```

```
--Encontre o saldo médio de depósitos de cada agência
select nome_agencia, avg(saldo_deposito)
from deposito
group by nome_agencia;
```

```
--Encontre o saldo médio de depósitos de cada agência
--mas mostre apenas as agencias e saldo médio que forem
--maiores do que R$ 1.200,00
select nome_agencia, avg(saldo_deposito)
from deposito
group by nome_agencia
having avg(saldo_deposito) > 1200;
```

```
--Selecione o valor do maior depósito
select max(saldo_deposito)
from deposito;
```



```
--Selecione o valor do maior, da média e do menor depósito
select max(saldo_deposito), avg(saldo_deposito), min(saldo_deposito)
from deposito;
```

```
--Selecione o valor do maior, da média
--e do menor depósito, todos por agencia
select nome_agencia, max(saldo_deposito), avg(saldo_deposito),
min(saldo_deposito)
from deposito
group by nome_agencia;
```

```
--Selecione o nome do cliente que fez o maior deposito
select nome_cliente, saldo_deposito
from deposito
where saldo_deposito = (select max(saldo_deposito) from deposito)
```

--O CLIENTE "Thiago Andrade Fiuza", ANTES MORADOR DA CIDADE DE ITAMBACURI, MUDOU DE ENDEREÇO

--E SOLICITOU A ALTERÇÃO DE SEUS DADOS NO BANCO. O NOVO ENDEREÇO PASSA A SER NA CIDADE DE

--UBERLANDIA, À AVENIDA AFONSO PENA. ATUALIZE OS DADOS DO CLIENTE:

--ANTES

```
SELECT * FROM CLIENTE WHERE CIDADE_CLIENTE = 'Itambacuri';
```

```
UPDATE CLIENTE SET CIDADE_CLIENTE = 'Uberlandia',
```

```
        RUA_CLIENTE='Avenida Afonso Pena'
```

```
WHERE      NOME_CLIENTE = 'Thiago Andrade Fiuza'
```

```
AND CIDADE_CLIENTE = 'Itambacuri';
```

```
--DEPOIS
```

```
SELECT * FROM CLIENTE WHERE CIDADE_CLIENTE = 'Uberlandia';
```

```
--AO FINAL DO MÊS É NECESSÁRIO ATUALIZAR O SALDO DAS CONTAS DOS CLIENTES  
APLICANDO
```

```
--A CORREÇÃO DA INFLAÇÃO DE 1% SOBRE AS MESMAS. IMPLEMENTE ESTA ATUALIZAÇÃO  
VIA SQL.
```

```
--ANTES E DEPOIS
```

```
SELECT NOME_CLIENTE, SUM(SALDO_DEPOSITO) FROM DEPOSITO  
GROUP BY NOME_CLIENTE ORDER BY SUM(SALDO_DEPOSITO);
```

```
UPDATE DEPOSITO SET SALDO_DEPOSITO = SALDO_DEPOSITO * 1.01;
```

```
--AO FINAL DO MÊS É NECESSÁRIO ATUALIZAR O SALDO DAS CONTAS DOS CLIENTES  
APLICANDO
```

```
--OS JUROS DOS INVESTIMENTOS SOBRE AS MESMAS.
```

```
--CLIENTES COM SALDO ATÉ DEZ MIL RECEBEM 3% DE ACRÉSCIMO.
```

```
--CLIENTES COM SALDO MAIOR QUE DEZ MIL RECEBEM 5% DE ACRÉSCIMO.
```

```
--ANTES E DEPOIS
```

```
SELECT NOME_CLIENTE, SUM(SALDO_DEPOSITO) FROM DEPOSITO  
GROUP BY NOME_CLIENTE ORDER BY SUM(SALDO_DEPOSITO);
```

```
UPDATE DEPOSITO SET SALDO_DEPOSITO = SALDO_DEPOSITO * 1.03  
WHERE SALDO_DEPOSITO <= 10000;
```

```
UPDATE DEPOSITO SET SALDO_DEPOSITO = SALDO_DEPOSITO * 1.05
```

```
WHERE SALDO_DEPOSITO > 10000;
```

```
--A ORDEM DE EXECUÇÃO DAS CONSULTAS É IMPORTANTE PORQUE SENÃO UM  
--CLIENTE COM SALDO LIGEIRAMENTE INFERIOR A DEZ MIL PODE RECEBER  
--UM ACRÉSCIMO DE 8.15% = (1.03*1.05)
```

```
--INSERIR NOVOS DEPOSITOS NO BANCO DE DADOS
```

```
SELECT * FROM CONTA;  
SELECT * FROM DEPOSITO;  
SELECT * FROM CLIENTE;
```

```
INSERT INTO CLIENTE VALUES('Carlos Eduardo', 'Rua 0', 'Udão');  
INSERT INTO CONTA VALUES(20000, 'PUC', 'Carlos Eduardo', 0);
```

```
INSERT INTO DEPOSITO  
VALUES (10000, 20000, 'PUC', 'Carlos Eduardo', 1200);
```

```
--INCLUIR PARA TODOS OS CLIENTES COM EMPRESTIMOS NA AGENCIA PUC  
--UMA CONTA DE DEPÓSITO NO VALOR DE R$ 200,00.
```

```
INSERT INTO DEPOSITO  
SELECT NUMERO_EMPRESTIMO, NUMERO_CONTA, NOME_AGENCIA, NOME_CLIENTE, 200  
FROM EMPRESTIMO WHERE NOME_AGENCIA = 'PUC';
```

```
--01-Nome e cidade de clientes que possuem algum emprestimo, ordenado pelo  
nome do cliente
```

```
SELECT DISTINCT cliente.nome_cliente, cidade_cliente  
FROM emprestimo, cliente  
WHERE cliente.nome_cliente=emprestimo.nome_cliente  
ORDER BY cliente.nome_cliente;
```

--02-Nome e cidade de clientes que possuem emprestimo na PUC, ordenado pelo nome do cliente

```
SELECT DISTINCT cliente.nome_cliente, cliente.cidade_cliente
FROM cliente, emprestimo
WHERE cliente.nome_cliente=emprestimo.nome_cliente
and emprestimo.nome_agencia='PUC'
ORDER BY cliente.nome_cliente;
```

```
SELECT * FROM CONTA;
```

--03-Nomes de Clientes com saldo entre 100 e 900

```
SELECT cliente.nome_cliente, sum(deposito.saldo_deposito)
FROM cliente, deposito
WHERE cliente.nome_cliente=deposito.nome_cliente
GROUP BY cliente.nome_cliente
HAVING sum(deposito.saldo_deposito) between 100 and 900;
```

--04-Nomes de clientes, valores de depósitos e empréstimos na agencia PUC

```
SELECT D.nome_cliente, D.saldo_deposito, E.valor_emprestimo, E.nome_agencia
FROM deposito AS D, emprestimo AS E
WHERE D.nome_agencia='PUC' AND
D.nome_agencia=E.nome_agencia AND
D.nome_cliente=E.nome_cliente
```

--05-Seleciono os nomes dos clientes da cidade de Contagem com depósitos maiores do que R\$ 3.000,00

```
select cliente.nome_cliente, cliente.cidade_cliente,
deposito.saldo_deposito
```

```
from cliente, deposito
where cliente.nome_cliente=deposito.nome_cliente
and cliente.cidade_cliente = 'Contagem'
and deposito.saldo_deposito > 3000;
```

--06-Um gerente pretende criar uma lista de clientes que correm o risco  
--de se indviduar de maneira irreversível. Para tanto ele formulou  
--a seguinte pesquisa:

--Selecione os clientes da cidade de Santa Luzia com depósitos  
--menores do que R\$ 1.000,00 e empréstimos maiores que R\$ 1.000,00

```
select cliente.nome_cliente, cliente.cidade_cliente,
        deposito.saldo_deposito, emprestimo.valor_emprestimo
from cliente, emprestimo, deposito
where cliente.nome_cliente = deposito.nome_cliente
and deposito.nome_cliente = emprestimo.nome_cliente
and cliente.cidade_cliente = 'Santa Luzia'
and deposito.saldo_deposito < 1000
and emprestimo.valor_emprestimo > 1000
```

--07-Um gerente pretende criar uma lista de clientes que correm o risco  
--de se indviduar de maneira irreversível. Para tanto ele formulou  
--a seguinte pesquisa:

--Selecione os clientes da cidade de Santa Luzia com uma média de  
--depósitos menor do que a média de empréstimos

```
select cliente.nome_cliente, avg(deposito.saldo_deposito),
avg(emprestimo.valor_emprestimo)
from cliente, deposito, emprestimo
where cliente.nome_cliente = deposito.nome_cliente
and deposito.nome_cliente = emprestimo.nome_cliente
and cliente.cidade_cliente = 'Santa Luzia'
```

```
group by cliente.nome_cliente  
having avg(deposito.saldo_deposito) < avg(emprestimo.valor_emprestimo)
```

--08-É preciso atualizar a informação do saldo do cliente na tabela cliente.

--para este propósito devemos levar em conta o saldo dos depósitos menos os saldos de empréstimos. o cálculo final deve ser armazenado na tabela conta.

```
UPDATE CONTA SET SALDO_CONTA = 0;
```

```
SELECT NOME_CLIENTE, SALDO_CONTA FROM CONTA ORDER BY SALDO_CONTA DESC;
```

```
SELECT NUMERO_CONTA, NOME_AGENCIA, NOME_CLIENTE, SUM(SALDO_DEPOSITO)  
FROM DEPOSITO  
GROUP BY NUMERO_CONTA, NOME_AGENCIA, NOME_CLIENTE;
```

--Primeiro os clientes que possuem deposito e emprestimos (ambos)

```
SELECT      E.NOME_CLIENTE, E.NOME_AGENCIA, E.NUMERO_CONTA,  
            SUM(D.SALDO_DEPOSITO), SUM(E.VALOR_EMPRESTIMO),  
            SUM(D.SALDO_DEPOSITO)-SUM(E.VALOR_EMPRESTIMO) AS TOTAL  
FROM EMPRESTIMO AS E, DEPOSITO AS D  
WHERE E.NOME_CLIENTE = D.NOME_CLIENTE  
GROUP BY E.NOME_CLIENTE, E.NOME_AGENCIA, E.NUMERO_CONTA;
```

--Atualiza contas que possuem deposito e emprestimos (ambos)

```
UPDATE CONTA SET SALDO_CONTA = RELATORIO.TOTAL  
FROM(  
    SELECT      E.NOME_CLIENTE, E.NOME_AGENCIA, E.NUMERO_CONTA,
```

```

        SUM(D.SALDO_DEPOSITO), SUM(E.VALOR_EMPRESTIMO),
        SUM(D.SALDO_DEPOSITO)-SUM(E.VALOR_EMPRESTIMO) AS TOTAL
FROM EMPRESTIMO AS E, DEPOSITO AS D
WHERE E.NOME_CLIENTE = D.NOME_CLIENTE
GROUP BY E.NOME_CLIENTE, E.NOME_AGENCIA, E.NUMERO_CONTA
) AS RELATORIO
WHERE CONTA.NOME_CLIENTE = RELATORIO.NOME_CLIENTE
AND   CONTA.NOME_AGENCIA = RELATORIO.NOME_AGENCIA
AND   CONTA.NUMERO_CONTA = RELATORIO.NUMERO_CONTA;

```

```

-- SEGUNDO OS CLIENTES QUE POSSUEM APENAS DEPÓSITOS
SELECT  D.NOME_CLIENTE, D.NOME_AGENCIA, D.NUMERO_CONTA,
        SUM(D.SALDO_DEPOSITO) AS SALDO
FROM DEPOSITO AS D
WHERE D.NOME_CLIENTE NOT IN (
        SELECT DISTINCT E.NOME_CLIENTE
        FROM EMPRESTIMO AS E
)
GROUP BY D.NOME_CLIENTE, D.NOME_AGENCIA, D.NUMERO_CONTA;

```

```

-- ATUALIZA CONTAS QUE POSSUEM APENAS DEPOSITO
UPDATE CONTA SET SALDO_CONTA = R.SALDO
FROM (
        SELECT  D.NOME_CLIENTE, D.NOME_AGENCIA, D.NUMERO_CONTA,
                SUM(D.SALDO_DEPOSITO) AS SALDO
        FROM DEPOSITO AS D
        WHERE D.NOME_CLIENTE NOT IN (
                SELECT DISTINCT E.NOME_CLIENTE
                FROM EMPRESTIMO AS E
        )
)

```

```
        GROUP BY D.NOME_CLIENTE, D.NOME_AGENCIA, D.NUMERO_CONTA
    ) AS R
WHERE    CONTA.NOME_CLIENTE = R.NOME_CLIENTE
AND     CONTA.NOME_AGENCIA = R.NOME_AGENCIA
AND     CONTA.NUMERO_CONTA = R.NUMERO_CONTA;
```

```
-- TERCEIRO OS CLIENTE QUE POSSUEM APENAS EMPRESTIMOS
SELECT  E.NOME_CLIENTE, E.NOME_AGENCIA, E.NUMERO_CONTA,
        -SUM(E.VALOR_EMPRESTIMO) AS DIVIDA
FROM EMPRESTIMO AS E
WHERE E.NOME_CLIENTE NOT IN (
        SELECT DISTINCT D.NOME_CLIENTE
        FROM DEPOSITO AS D
    )
GROUP BY E.NOME_CLIENTE, E.NOME_AGENCIA, E.NUMERO_CONTA;
```

```
-- ATUALIZA CONTAS QUE POSSUEM APENAS EMPRESTIMO
UPDATE CONTA SET SALDO_CONTA = R.DIVIDA
FROM (
        SELECT  E.NOME_CLIENTE, E.NOME_AGENCIA, E.NUMERO_CONTA,
                -SUM(E.VALOR_EMPRESTIMO) AS DIVIDA
        FROM EMPRESTIMO AS E
        WHERE E.NOME_CLIENTE NOT IN (
                SELECT DISTINCT D.NOME_CLIENTE
                FROM DEPOSITO AS D
            )
        GROUP BY E.NOME_CLIENTE, E.NOME_AGENCIA, E.NUMERO_CONTA
    ) AS R
WHERE    CONTA.NOME_CLIENTE = R.NOME_CLIENTE
AND     CONTA.NOME_AGENCIA = R.NOME_AGENCIA
```



```
AND    CONTA.NUMERO_CONTA = R.NUMERO_CONTA;
```