--1. Mostrar todos os clientes com saldo entre 10000 e 20000.

SELECT c.cliente\_cod, c.cliente\_nome, ct.conta\_numero, ct.saldo

FROM cliente c

JOIN conta ct ON c.cliente\_cod = ct.cliente\_cliente\_cod

WHERE ct.saldo BETWEEN 10000 AND 20000;

--2. Mostrar todos os clientes com emprestimos.

SELECT c.cliente\_cod, c.cliente\_nome, e.emprestimo\_numero, e.quantia

FROM cliente c

JOIN emprestimo e ON c.cliente\_cod = e.cliente\_cliente\_cod;

--3. Mostrar todos os clientes que moram em Niterói.

SELECT cliente\_cod, cliente\_nome, rua, cidade

FROM cliente

WHERE cidade = 'Niterói';

--4. Encontre os nomes e cidades de clientes que possuam empréstimos em alguma agência

SELECT cliente\_nome, cidade

FROM cliente

WHERE cliente\_cod IN (

SELECT cliente\_cliente\_cod

FROM emprestimo

);

--5. Mostrar o nome dos clientes que possuem conta, empréstimo ou ambos na agência de código 1

SELECT DISTINCT cliente.cliente\_nome

FROM cliente

WHERE cliente.cliente\_cod IN (

SELECT conta.cliente\_cliente\_cod

FROM conta

WHERE conta.agencia\_agencia\_cod = 1

UNION

SELECT emprestimo.cliente\_cliente\_cod

FROM emprestimo

WHERE emprestimo.agencia\_agencia\_cod = 1

);

--6. Achar todos os clientes que possuam uma conta e um empréstimo na agência de código 1

SELECT cliente.cliente\_nome

FROM cliente

WHERE cliente.cliente\_cod IN (

SELECT conta.cliente\_cliente\_cod

FROM conta

WHERE conta.agencia\_agencia\_cod = 1

)

AND cliente.cliente\_cod IN (

SELECT emprestimo.cliente\_cliente\_cod

FROM emprestimo

WHERE emprestimo.agencia\_agencia\_cod = 1

);

--7. Achar todos os clientes que possuem uma conta mas não possuem um empréstimo na agência de código 1

SELECT cliente.cliente\_nome

FROM cliente

WHERE cliente.cliente\_cod IN (

SELECT conta.cliente\_cliente\_cod

FROM conta

WHERE conta.agencia\_agencia\_cod = 1

)

AND cliente.cliente\_cod NOT IN (

SELECT emprestimo.cliente\_cliente\_cod

FROM emprestimo

WHERE emprestimo.agencia\_agencia\_cod = 1

);

--8. Liste todas as cidades onde há agências

SELECT DISTINCT agencia\_cidade

FROM agencia;

--9. Liste todas as agências existentes em ‘Rio de Janeiro’

SELECT agencia\_cod, agencia\_nome, fundos

FROM agencia

WHERE agencia\_cidade = 'Rio de Janeiro';

--10. Liste todas as agências que possuem conta com saldo maior que 100.000,00

SELECT DISTINCT a.agencia\_cod, a.agencia\_nome, a.agencia\_cidade, a.fundos

FROM agencia a

JOIN conta c ON a.agencia\_cod = c.agencia\_agencia\_cod

WHERE c.saldo > 100000;

--11. Mostre os dados de todos os clientes da agência ‘Niterói’ da cidade de ‘Niterói’ que têm saldo negativo

SELECT c.cliente\_cod, c.cliente\_nome, c.rua, c.cidade

FROM cliente c

JOIN conta ct ON c.cliente\_cod = ct.cliente\_cliente\_cod

JOIN agencia a ON ct.agencia\_agencia\_cod = a.agencia\_cod

WHERE a.agencia\_nome LIKE '%Niterói%' AND a.agencia\_cidade = 'Niterói' AND ct.saldo < 0;

--12. Liste os dados dos clientes que possuem conta mas não possuem empréstimo na agência de código 5

SELECT c.cliente\_cod, c.cliente\_nome, c.rua, c.cidade

FROM cliente c

WHERE c.cliente\_cod IN (

SELECT ct.cliente\_cliente\_cod

FROM conta ct

WHERE ct.agencia\_agencia\_cod = 5

)

AND c.cliente\_cod NOT IN (

SELECT e.cliente\_cliente\_cod

FROM emprestimo e

WHERE e.agencia\_agencia\_cod = 5

);

--13. Selecione todas as agências com código 1, 2 ou 3

SELECT agencia\_cod, agencia\_nome, agencia\_cidade, fundos

FROM agencia

WHERE agencia\_cod IN (1, 2, 3);

--14. Encontre todos os clientes que possuem uma conta e um empréstimo na agência ‘Nova Iguaçu’

SELECT c.cliente\_nome

FROM cliente c

WHERE c.cliente\_cod IN (

SELECT ct.cliente\_cliente\_cod

FROM conta ct

JOIN agencia a ON ct.agencia\_agencia\_cod = a.agencia\_cod

WHERE a.agencia\_nome LIKE '%Nova Iguaçu%'

)

AND c.cliente\_cod IN (

SELECT e.cliente\_cliente\_cod

FROM emprestimo e

JOIN agencia a ON e.agencia\_agencia\_cod = a.agencia\_cod

WHERE a.agencia\_nome LIKE '%Nova Iguaçu%'

);

--15. Encontre todas as agências que possuem ativos maiores que alguma agência de Angra dos Reis

SELECT agencia\_cod, agencia\_nome, agencia\_cidade, fundos

FROM agencia

WHERE fundos > (

SELECT MAX(fundos)

FROM agencia

WHERE agencia\_cidade = 'Angra dos Reis'

);

--16. Encontrar as agências que possuem ativos maiores do que todas as agências de Resende

SELECT agencia\_cod, agencia\_nome, agencia\_cidade, fundos

FROM agencia

WHERE fundos > (

SELECT MAX(fundos)

FROM agencia

WHERE agencia\_cidade = 'Resende'

);

--17. Usando a construção EXISTS, encontre todos os clientes que possuem uma conta e um empréstimo na agência ‘Petrópolis’

SELECT c.cliente\_nome

FROM cliente c

WHERE EXISTS (

SELECT 1

FROM conta ct

JOIN agencia a ON ct.agencia\_agencia\_cod = a.agencia\_cod

WHERE ct.cliente\_cliente\_cod = c.cliente\_cod AND a.agencia\_nome LIKE '%Petrópolis%'

)

AND EXISTS (

SELECT 1

FROM emprestimo e

JOIN agencia a ON e.agencia\_agencia\_cod = a.agencia\_cod

WHERE e.cliente\_cliente\_cod = c.cliente\_cod AND a.agencia\_nome LIKE '%Petrópolis%'

);

--18. Encontre o saldo médio de conta em cada agência

SELECT agencia\_agencia\_cod, AVG(saldo) AS saldo\_medio

FROM conta

GROUP BY agencia\_agencia\_cod;

--19. Encontre o número de depositantes de cada agência

SELECT agencia\_agencia\_cod, COUNT(DISTINCT cliente\_cliente\_cod) AS numero\_depositantes

FROM conta

GROUP BY agencia\_agencia\_cod;

--20. Encontre o maior saldo de cada agência

SELECT agencia\_agencia\_cod, MAX(saldo) AS maior\_saldo

FROM conta

GROUP BY agencia\_agencia\_cod;

--21. Encontre a média de saldos de todas as contas

SELECT AVG(saldo) AS media\_saldos

FROM conta;

--22. Remover todos os registros da conta de ‘João’

DELETE FROM conta

WHERE cliente\_cliente\_cod IN (

SELECT cliente\_cod

FROM cliente

WHERE cliente\_nome = 'João'

);

--23. Remova todas as contas de agências localizadas em ‘Vitória’

DELETE FROM conta

WHERE agencia\_agencia\_cod IN (

SELECT agencia\_cod

FROM agencia

WHERE agencia\_cidade = 'Vitória'

);

--24. Inserir uma nova conta para o cliente (código = 1), número 9000, na agência de código=2 cujo valor seja 1200

INSERT INTO conta (conta\_numero, saldo, cliente\_cliente\_cod, agencia\_agencia\_cod)

VALUES (9000, 1200, 1, 2);

--25. Inserir todos os clientes que possuam empréstimos na agência ‘Macaé’ na relação CONTA com um saldo de 200. O número da nova conta é o número do empréstimo \* 3

INSERT INTO conta (conta\_numero, saldo, cliente\_cliente\_cod, agencia\_agencia\_cod)

SELECT emprestimo\_numero \* 3, 200, cliente\_cliente\_cod, agencia\_agencia\_cod

FROM emprestimo

WHERE agencia\_agencia\_cod = (

SELECT agencia\_cod

FROM agencia

WHERE agencia\_nome LIKE '%Macaé%'

);

--26. Suponha que esteja sendo feito o pagamento de juros, e que em todos saldos sejam acrescentados em 5%

UPDATE conta

SET saldo = saldo \* 1.05;

--27. Suponha que todas as contas com saldo superiores a 10000 recebam aumento de 6% e as demais de 5%

UPDATE conta

SET saldo = CASE

WHEN saldo > 10000 THEN saldo \* 1.06

ELSE saldo \* 1.05

END;

--28. Considere, por exemplo, que todas as contas de pessoas que possuem empréstimos no banco terão acréscimo de 1%

UPDATE conta

SET saldo = saldo \* 1.01

WHERE cliente\_cliente\_cod IN (

SELECT cliente\_cliente\_cod

FROM emprestimo

);

--29. Atualize o valor dos ativos. Os ativos são os valores dos saldos das contas da agência

UPDATE agencia a

SET fundos = (

SELECT SUM(c.saldo)

FROM conta c

WHERE c.agencia\_agencia\_cod = a.agencia\_cod

);

--30. Considere a visão consistindo em nomes de agências e de clientes

CREATE VIEW agencia\_cliente\_view AS

SELECT a.agencia\_nome, c.cliente\_nome

FROM agencia a

JOIN conta ct ON a.agencia\_cod = ct.agencia\_agencia\_cod

JOIN cliente c ON ct.cliente\_cliente\_cod = c.cliente\_cod;

--31. Usando a visão agencia\_cliente\_view, podemos achar todos os clientes da agência ‘Rio de Janeiro’

SELECT cliente\_nome

FROM agencia\_cliente\_view

WHERE agencia\_nome LIKE '%Rio de Janeiro%';

--32. Selecione todos os clientes que possuem contas em agência(s) que possui(m) o maior ativo

SELECT DISTINCT c.cliente\_nome

FROM cliente c

JOIN conta ct ON c.cliente\_cod = ct.cliente\_cliente\_cod

JOIN agencia a ON ct.agencia\_agencia\_cod = a.agencia\_cod

WHERE a.fundos = (

SELECT MAX(fundos)

FROM agencia

);

--33. Selecione o total de agências por cidade, classificado por cidade

SELECT agencia\_cidade, COUNT(\*) AS total\_agencias

FROM agencia

GROUP BY agencia\_cidade

ORDER BY agencia\_cidade;

--34. Selecione o valor médio de empréstimos efetuados por cada agência em ordem crescente das cidades onde estas agências se situam

SELECT a.agencia\_cidade, AVG(e.quantia) AS media\_emprestimos

FROM emprestimo e

JOIN agencia a ON e.agencia\_agencia\_cod = a.agencia\_cod

GROUP BY a.agencia\_cidade

ORDER BY a.agencia\_cidade;

--35. Selecione a(s) agência(s) que possui(m) a maior média de quantia emprestada

SELECT agencia\_cod, agencia\_nome, AVG(quantia) AS media\_emprestada

FROM emprestimo

GROUP BY agencia\_cod, agencia\_nome

HAVING AVG(quantia) = (

SELECT MAX(AVG(quantia))

FROM emprestimo

GROUP BY agencia\_agencia\_cod

);

--36. Selecione todas as agências situadas fora de Rio de Janeiro que possuem a média de depósitos maior do que alguma agência localizada em Duque de Caxias

SELECT a.agencia\_cod, a.agencia\_nome, a.agencia\_cidade

FROM agencia a

WHERE a.agencia\_cidade <> 'Rio de Janeiro'

AND (

SELECT AVG(ct.saldo)

FROM conta ct

WHERE ct.agencia\_agencia\_cod = a.agencia\_cod

) > (

SELECT MIN(AVG(ct.saldo))

FROM conta ct

JOIN agencia ag ON ct.agencia\_agencia\_cod = ag.agencia\_cod

WHERE ag.agencia\_cidade = 'Duque de Caxias'

GROUP BY ag.agencia\_cod

);

--37. Selecione o menor saldo de clientes, por agências

SELECT agencia\_agencia\_cod, MIN(saldo) AS menor\_saldo

FROM conta

GROUP BY agencia\_agencia\_cod;

--38. Selecione o saldo de cada cliente, caso ele possua mais de uma conta no banco

SELECT cliente\_cliente\_cod, SUM(saldo) AS total\_saldo

FROM conta

GROUP BY cliente\_cliente\_cod

HAVING COUNT(\*) > 1;