EXERCÍCIOS:

1. Agrupamento de Dados

GROUP BY coluna; - Agrupa as linhas que têm os mesmos valores em colunas especificadas.

SELECT cliente\_id, COUNT(id) AS numero\_de\_pedidos FROM pedidos GROUP BY cliente\_id;

HAVING condição; - Filtra os grupos formados pelo GROUP BY baseado em uma condição.

SELECT cliente\_id, COUNT(id) AS numero\_de\_pedidos FROM pedidos GROUP BY cliente\_id HAVING COUNT(id) > 5;

2. Subconsultas

(SELECT coluna FROM tabela) AS subconsulta: Executa uma consulta dentro de outra consulta.

As subconsultas são uma ferramenta poderosa em SQL, permitindo que você realize consultas dentro de outras consultas para criar condições ou filtragens complexas. Vamos ver um exemplo prático usando novamente uma tabela genérica chamada clientes, que possui colunas para id, nome, e-mail, cidade, e data\_de\_cadastro, e uma tabela pedidos com colunas para id, cliente\_id, data\_do\_pedido, e valor.

Exemplo de Subconsulta

Suponha que você queira encontrar os nomes dos clientes que fizeram pedidos no ano de 2023. Você pode usar uma subconsulta para isso:

SELECT nome

FROM clientes

WHERE

id IN (

SELECT cliente\_id FROM pedidos WHERE YEAR(data\_do\_pedido) = 2023

) AS subconsulta;

Descrição: Esta consulta seleciona os nomes dos clientes da tabela clientes que têm um id presente no conjunto de resultados da subconsulta. A subconsulta seleciona os cliente\_id da tabela pedidos onde a data\_do\_pedido está no ano de 2023. A cláusula IN compara cada id de clientes com o conjunto de cliente\_id retornados pela subconsulta, efetivamente filtrando

3. Funções de Agregação

COUNT(), SUM(), AVG(), MIN(), MAX(): Funções que realizam cálculos sobre um conjunto de linhas.

SELECT COUNT(id) AS total\_clientes FROM clientes;

COUNT(): Conta o número de linhas.

SELECT SUM(valor) AS valor\_total FROM pedidos;

SUM(): Soma os valores.

SELECT SUM(valor) AS valor\_total FROM pedidos;

AVG(): Calcula a média.

SELECT AVG(valor) AS valor\_medio FROM pedidos;

MIN() e MAX(): Encontram o valor mínimo e máximo, respectivamente.

SELECT MIN(valor) AS valor\_minimo, MAX(valor) AS valor\_maximo FROM pedidos;

Banco de dados – biblioteca

1. Qual é o ID do autor do livro com o título ‘It’?

SELECT idAuthor

FROM books\_authors

WHERE idBook = (

SELECT idBook

FROM books

WHERE title = 'It'

);

1. Quais são os livros escritos por 'Matt Haig'?

SELECT title

FROM books

WHERE idBook IN (

SELECT idBook

FROM books\_authors

WHERE idAuthor = (

SELECT idAuthor

FROM authors

WHERE name = 'Matt Haig'

)

);

1. Contar quantos livros existem em cada categoria.

SELECT category, COUNT(\*) AS allBooks

FROM books

GROUP BY category;

1. Listar autores e a quantidade de livros publicados por ele.

SELECT

authors.name AS author,

(SELECT COUNT(\*)

FROM books\_authors

WHERE books\_authors.idAuthor = authors.idAuthor) AS booksPublished

FROM authors;

1. Listar categorias que têm mais de 5 livros.

SELECT

books.category AS category,

(SELECT COUNT(\*)

FROM books AS b

WHERE b.category = books.category) AS num\_books

FROM books

GROUP BY books.category

HAVING COUNT(\*) > 5;

1. Encontrar o livro mais antigo.

SELECT title, publication\_date

FROM books

WHERE publication\_date = (

SELECT MIN(publication\_date)

FROM books

);

1. Mostrar a quantidade de usuários registrados por mês no ano de 2023.

SELECT MONTH(register\_date) AS month,

COUNT(\*) AS users\_registered

FROM users

WHERE YEAR(registerDate) = 2023

GROUP BY MONTH(registerDate);

1. Identificar o livro mais e menos emprestado.

-- MAIS EMPRESTADO

SELECT books.title AS borrowedBook,

COUNT(\*) AS qtd\_loans

FROM books\_loans

JOIN books ON books\_loans.idBook = books.idBook

GROUP BY books.title

ORDER BY COUNT(\*) DESC

LIMIT 1;

-- MENOS EMPRESTADO

SELECT books.title AS borrowedBook,

COUNT(\*) AS qtd\_loans

FROM books\_loans

JOIN books ON books\_loans.idBook = books.idBook

GROUP BY books.title

ORDER BY COUNT(\*) ASC

LIMIT 1;

1. Calcular a idade média dos autores.

SELECT AVG(YEAR(CURRENT\_DATE) - YEAR(birthday)) AS mediumAge

FROM authors;

1. Listar os 3 livros mais recentes.

SELECT title, publication\_date

FROM books

ORDER BY publication\_date DESC

LIMIT 3;

1. Mostrar o total de empréstimos feitos por usuários premium.

SELECT COUNT(\*) AS allPremium

FROM loans

WHERE idUser IN (SELECT idUser FROM users WHERE premium = 'sim');

1. Determinar o número de livros por autor.

SELECT authors.name AS author, COUNT(\*) AS qtd\_books

FROM books\_authors

JOIN authors ON books\_authors.idAuthor = authors.idAuthor

GROUP BY authors.name;

1. Listar cada categoria e o número de livros emprestados para cada uma.

SELECT books.category AS category, COUNT(\*) AS borrowedBooks

FROM books\_loans

JOIN books ON books\_loans.idBook = books.idBook

GROUP BY books.category;

1. Encontrar o usuário que pegou emprestado o maior número de livros únicos.

SELECT idUser, COUNT(DISTINCT idBook) AS borrowedBooks

FROM loans

GROUP BY idUser

ORDER BY borrowedBooks DESC

LIMIT 1;

1. Obter a média de duração dos empréstimos.

SELECT AVG(DATEDIFF(deliveyDate, initialDate)) AS mediumLoans

FROM loans;

1. Listar os livros que nunca foram emprestados.

SELECT \*

FROM books

WHERE idBook NOT IN (SELECT DISTINCT idBook FROM books\_loans);

1. Determinar a categoria com o maior número de livros.

SELECT books.category AS category, COUNT(\*) AS numBooks

FROM books

GROUP BY books.category

ORDER BY COUNT(\*) DESC

LIMIT 1;

1. Identificar os meses com mais de 10 empréstimos.

SELECT MONTH(initialDate) AS month, COUNT() AS qtd\_loans

FROM loans

GROUP BY MONTH(initialDate)

HAVING COUNT() > 10;

1. Mostrar o autor mais jovem.

SELECT name AS autor, MIN(YEAR(CURRENT\_DATE) - YEAR(birthday)) AS age

FROM authors;

1. Listar os usuários que realizaram empréstimo no seu mês de registro.

SELECT idUser, MONTH(registerDate) AS registerMonth

FROM users

WHERE MONTH(registerDate) = MONTH(initialDate);