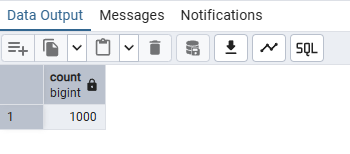
**Perguntas SQL**

**Quantos livros existem cadastrados?**

SELECT COUNT(\*) FROM livros;



**Listar todos os usuários ordenados pelo nome.**

SELECT distinct nome FROM usuarios;



**Mostrar todos os empréstimos realizados no mês atual.**

SELECT id\_livro, data\_emprestimo FROM emprestimos

where extract(month from data\_emprestimo) = 05

and extract(year from data\_emprestimo) = 2025

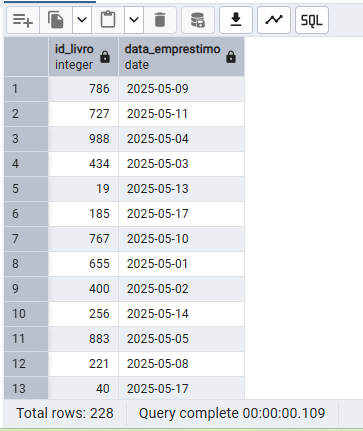
**ou**

SELECT id\_livro, data\_emprestimo

FROM emprestimos

WHERE data\_emprestimo >= '2025-05-01'

AND data\_emprestimo < '2025-06-01';

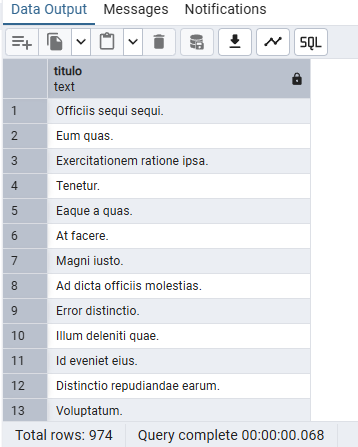


**Mostrar os títulos dos livros emprestados.**

SELECT DISTINCT l.titulo

FROM emprestimos AS e

JOIN livros AS l ON e.id\_livro = l.id\_livro;



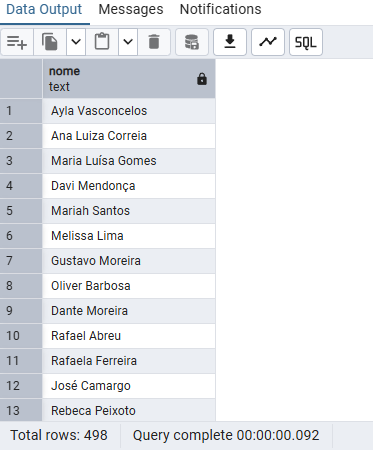
**Listar o nome dos usuários que pegaram livros emprestados.**

SELECT distinct U.nome

from emprestimos as e

join usuarios as U

on E.id\_usuario = U.id\_usuario



**🟡 MÉDIO**

**Quantos livros diferentes foram emprestados ao menos uma vez?**

SELECT COUNT(DISTINCT e.id\_livro) AS total\_livros\_emprestados

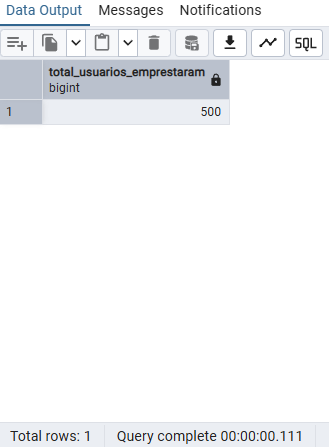
FROM emprestimos AS e;



**Quantos usuários distintos fizeram empréstimos?**

SELECT COUNT(DISTINCT e.id\_usuario) AS total\_usuarios\_emprestaram

FROM emprestimos AS e;



**Listar os 10 autores com mais livros cadastrados.**

SELECT autor, COUNT(\*) AS total\_livros

FROM livros

GROUP BY autor

ORDER BY total\_livros DESC

LIMIT 10;



**Mostrar os 5 livros mais emprestados.**

Select count(e.id\_livro) as Contagem, l.titulo

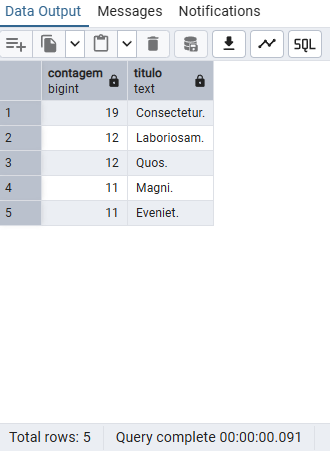
from emprestimos as e

JOIN livros as l on e.id\_livro = l.id\_livro

group by titulo

order by count(e.id\_livro) DESC

fetch first 5 rows only



**Listar os usuários que mais fizeram empréstimos.**

Select count(e.id\_usuario) as Contagem, U.nome

from emprestimos as e

JOIN usuarios as U on e.id\_usuario = U.id\_usuario

group by nome

order by count(e.id\_usuario) DESC



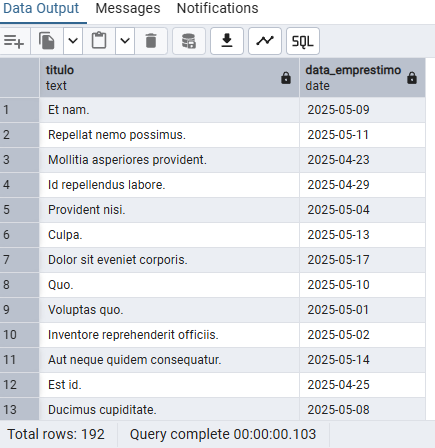
**Listar todos os empréstimos ativos hoje (data\_devolucao > CURRENT\_DATE).**

SELECT l.titulo, e.data\_emprestimo

FROM livros AS l

JOIN emprestimos AS e ON e.id\_livro = l.id\_livro

WHERE e.data\_devolucao > CURRENT\_DATE;



**Listar os livros que nunca foram emprestados.**

SELECT l.id\_livro, l.titulo

FROM livros AS l

LEFT JOIN emprestimos AS e ON l.id\_livro = e.id\_livro

WHERE e.id\_livro IS NULL;



**Mostrar os usuários que emprestaram mais de 5 livros.**

Select count(e.id\_usuario) as Contagem, U.nome

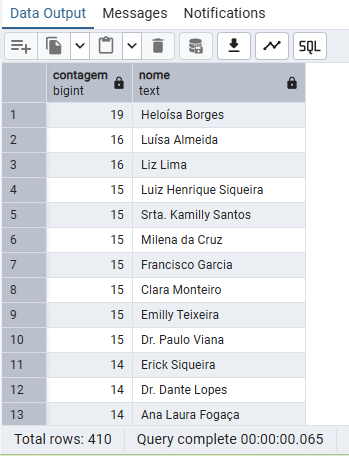
from emprestimos as e

JOIN usuarios as U on e.id\_usuario = U.id\_usuario

group by nome

having count(e.id\_usuario) > 5

order by count(e.id\_usuario) DESC



**Difícil:**

**Calcular a média de tempo de empréstimo (em dias) por usuário.**

SELECT U.nome, ROUND(AVG(e.data\_devolucao - e.data\_emprestimo),2) as dias\_emprestimos

FROM emprestimos as e

JOIN usuarios as U ON u.id\_usuario = e.id\_usuario

group by nome

order by dias\_emprestimos DESC



**Listar os livros mais emprestados por gênero.**

WITH livros\_por\_genero AS (

SELECT

l.genero,

l.titulo,

COUNT(\*) AS total\_emprestimos,

RANK() OVER (PARTITION BY l.genero ORDER BY COUNT(\*) DESC) AS posicao

FROM emprestimos AS e

JOIN livros AS l ON l.id\_livro = e.id\_livro

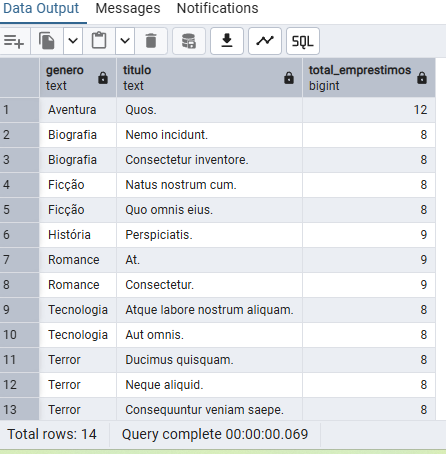
GROUP BY l.genero, l.titulo

)

SELECT genero, titulo, total\_emprestimos

FROM livros\_por\_genero

WHERE posicao = 1;



**Listar os 5 usuários que mais pegaram livros de “Ficção”.**

WITH contagem\_usuario AS (

SELECT

e.id\_usuario,

COUNT(\*) AS contagem

FROM livros AS l

JOIN emprestimos AS e ON e.id\_livro = l.id\_livro

WHERE l.genero = 'Ficção'

GROUP BY e.id\_usuario

)

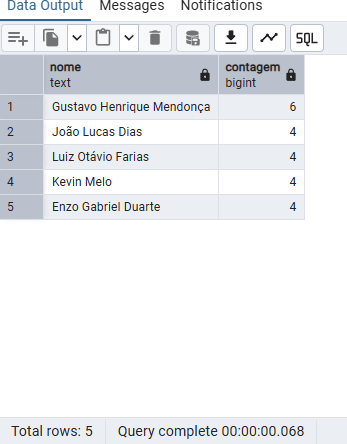
SELECT u.nome, cu.contagem

FROM contagem\_usuario AS cu

JOIN usuarios AS u ON u.id\_usuario = cu.id\_usuario

ORDER BY cu.contagem DESC

LIMIT 5;



**Encontrar o autor cujo livro teve o maior tempo total emprestado (em dias somados).**

SELECT l.autor, ROUND(SUM(e.data\_devolucao - e.data\_emprestimo),2) as dias\_emprestimos

FROM emprestimos as e

JOIN livros as l ON l.id\_livro = e.id\_livro

group by autor

order by dias\_emprestimos DESC

LIMIT 1;



**Para cada mês do último ano, mostrar quantos empréstimos foram feitos.**

SELECT

TO\_CHAR(data\_emprestimo, 'YYYY-MM') AS mes,

COUNT(\*) AS total\_emprestimos

FROM emprestimos

WHERE data\_emprestimo >= CURRENT\_DATE - INTERVAL '12 months'

GROUP BY mes

ORDER BY mes;



**Listar o livro com maior quantidade de empréstimos por mês.**

WITH contagem\_por\_mes AS (

SELECT

l.titulo,

EXTRACT(YEAR FROM e.data\_emprestimo) AS ano,

EXTRACT(MONTH FROM e.data\_emprestimo) AS mes,

COUNT(\*) AS total\_emprestimos,

RANK() OVER (

PARTITION BY EXTRACT(YEAR FROM e.data\_emprestimo), EXTRACT(MONTH FROM e.data\_emprestimo)

ORDER BY COUNT(\*) DESC

) AS posicao

FROM emprestimos AS e

JOIN livros AS l ON l.id\_livro = e.id\_livro

GROUP BY l.titulo, ano, mes

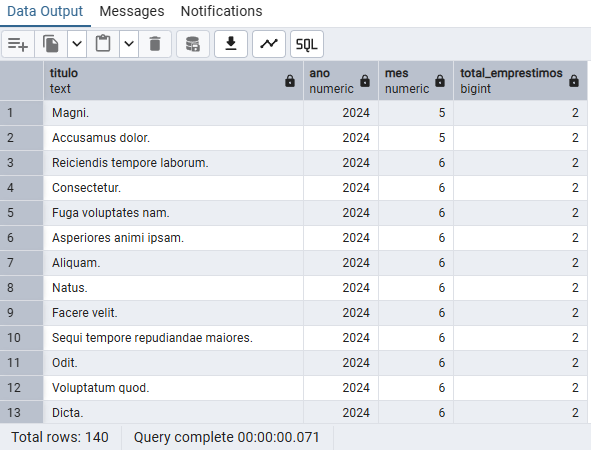
)

SELECT titulo, ano, mes, total\_emprestimos

FROM contagem\_por\_mes

WHERE posicao = 1

ORDER BY ano, mes;



**Listar os usuários que emprestaram livros consecutivamente (sem intervalo entre devolução e novo empréstimo).**

SELECT

u.nome,

e.id\_usuario,

e.id\_emprestimo,

e.data\_emprestimo,

e.data\_devolucao,

e.devolucao\_anterior

FROM (

SELECT \*,

LAG(data\_devolucao) OVER (PARTITION BY id\_usuario ORDER BY data\_emprestimo) AS devolucao\_anterior

FROM emprestimos

) AS e

JOIN usuarios AS u ON u.id\_usuario = e.id\_usuario

WHERE e.data\_emprestimo = e.devolucao\_anterior

ORDER BY e.id\_usuario, e.data\_emprestimo;

