Trabalho de Banco de Dados - Projeto – Parte I

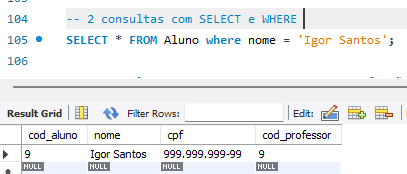
Aluno: 3° SGT F.Leite RG 93.177

Parte 2

1. 2 consultas com SELECT e WHERE

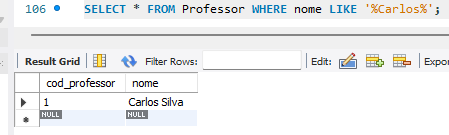
**Filtra todos os dados da tabela Aluno onde o campo nome é exatamente "Igor Santos".**

SELECT \* FROM Aluno where nome = 'Igor Santos';



**Mostra todos os professores cujo nome contém a palavra "Carlos".**

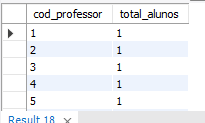
SELECT \* FROM Professor WHERE nome LIKE '%Carlos%';



1. 2 consultas com GROUP BY e ORDER BY com funções de agregação

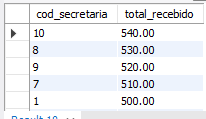
**Conta quantos alunos estão vinculados a cada cod\_professor.**

SELECT cod\_professor, COUNT(\*) AS total\_alunos FROM Aluno GROUP BY cod\_professor ORDER BY total\_alunos DESC;



**Soma todas as mensalidades recebidas por secretaria.**

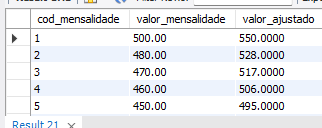
SELECT cod\_secretaria, SUM(valor\_mensalidade) AS total\_recebido FROM Mensalidade GROUP BY cod\_secretaria ORDER BY total\_recebido DESC;



1. 2 consultas com operadores aritméticos (+, -, \*, /)

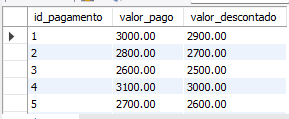
**Aplica um aumento de 10% na mensalidade (valor\_mensalidade \* 1.10).**

SELECT cod\_mensalidade, valor\_mensalidade, valor\_mensalidade \* 1.10 AS valor\_ajustado FROM Mensalidade;



**Subtrai R$ 100,00 de cada valor pago, como se fosse um desconto.**

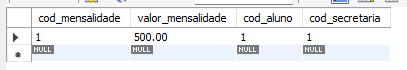
SELECT id\_pagamento, valor\_pago, valor\_pago - 100 AS valor\_descontado FROM Efetua\_Pagamento;



1. 3 consultas com operadores de comparação (=, !=, <, >, etc.)

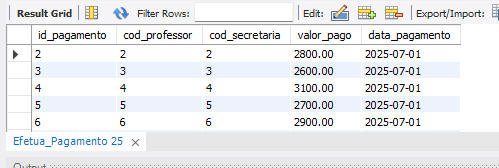
**Mostra mensalidades com valor exato de R$ 500,00.**

SELECT \* FROM Mensalidade WHERE valor\_mensalidade = 500;



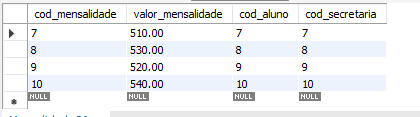
Mostra pagamentos que não são R$ 3000.

SELECT \* FROM Efetua\_Pagamento WHERE valor\_pago != 3000;



Filtra mensalidades com **valor maior** que R$ 500.

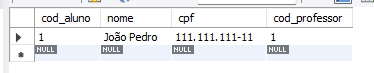
SELECT \* FROM Mensalidade WHERE valor\_mensalidade > 500;



1. 3 consultas com operadores lógicos (AND, OR)

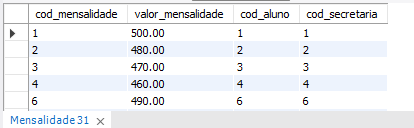
Mostra alunos que Têm a letra "a" no nome eEstão com o professor de cod\_professor = 1.

SELECT \* FROM Aluno WHERE nome LIKE '%a%' AND cod\_professor = 1;



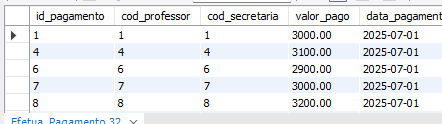
Mostra mensalidades com **valor acima de R$ 450** ou com secretaria 3.

SELECT \* FROM Mensalidade WHERE valor\_mensalidade > 450 OR cod\_secretaria = 3;



Filtra pagamentos **altos e feitos em 1º de julho de 2025**.

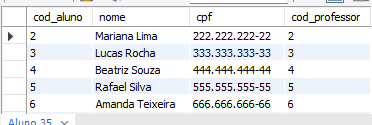
SELECT \* FROM Efetua\_Pagamento WHERE valor\_pago > 2800 AND data\_pagamento = '2025-07-01';



1. 2 consultas com operadores lógicos e negação (NOT)

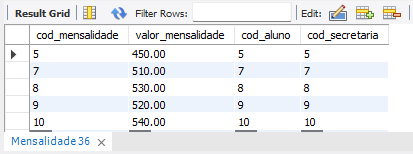
Mostra todos os alunos **exceto** "João Pedro".

SELECT \* FROM Aluno WHERE NOT nome LIKE '%João Pedro%';



**Exclui mensalidades entre R$ 460 e R$ 500.**

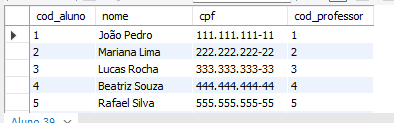
SELECT \* FROM Mensalidade WHERE NOT valor\_mensalidade BETWEEN 460 AND 500;



1. 3 consultas com operadores auxiliares (IS NULL, BETWEEN, LIKE, IN)

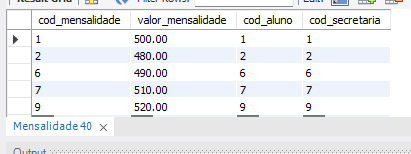
Mostra alunos que **têm CPF preenchido**.

SELECT \* FROM Aluno WHERE cpf IS NOT NULL;



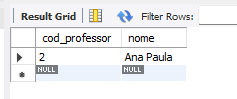
Mostra mensalidades entre **R$ 480 e R$ 520**.

SELECT \* FROM Mensalidade WHERE valor\_mensalidade BETWEEN 480 AND 520;



Mostra professores cujo **nome começa com "A"**.

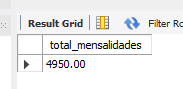
SELECT \* FROM Professor WHERE nome LIKE 'A%';



1. 3 consultas com funções de agregação (SUM(), AVG(), etc.)

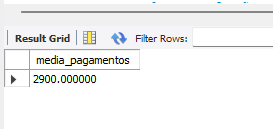
Soma todas as mensalidades do banco.

SELECT SUM(valor\_mensalidade) AS total\_mensalidades FROM Mensalidade;



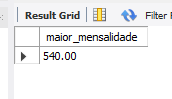
Mostra o valor **médio** dos pagamentos feitos.

SELECT AVG(valor\_pago) AS media\_pagamentos FROM Efetua\_Pagamento;



Mostra a **maior mensalidade** cadastrada.

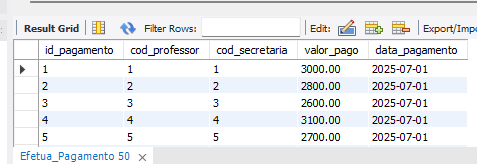
SELECT MAX(valor\_mensalidade) AS maior\_mensalidade FROM Mensalidade;



1. 2 consultas com funções de datas (NOW(), DATE(), YEAR(), etc.)

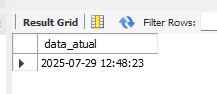
Filtra os pagamentos feitos **no ano de 2025**.

SELECT \* FROM Efetua\_Pagamento WHERE YEAR(data\_pagamento) = 2025; SELECT \* FROM Efetua\_Pagamento WHERE YEAR(data\_pagamento) = 2025;



Mostra a data e hora **atuais do sistema**.

SELECT NOW() AS data\_atual;



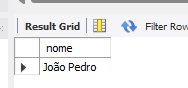
1. 3 sub-consultas com agrupamento e união de dados

Mostra nomes de alunos que têm professores com sobrenome **"Silva"**.

SELECT nome FROM Aluno WHERE cod\_professor IN (

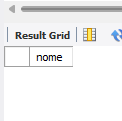
SELECT cod\_professor FROM Professor WHERE nome LIKE '%Silva%'

);



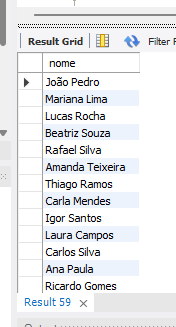
Mostra professores que têm mais de **1 aluno**.( Nenhum professor foi cadastrado com mais de um aluno)

SELECT nome FROM Professor WHERE cod\_professor IN ( SELECT cod\_professor FROM Aluno GROUP BY cod\_professor HAVING COUNT(\*) > 1 );



Junta os nomes dos alunos **e** professores (sem repetição).

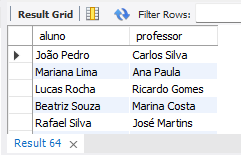
SELECT nome FROM Aluno UNION SELECT nome FROM Professor;



1. 3 consultas com JOIN e visualização de tabelas

Junta alunos e professores, mostrando quem está com quem.

SELECT a.nome AS aluno, p.nome AS professor FROM Aluno a JOIN Professor p ON a.cod\_professor = p.cod\_professor;



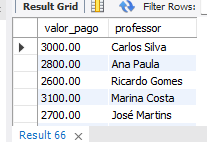
Mostra mensalidade com o nome do aluno correspondente.

SELECT m.valor\_mensalidade, a.nome AS aluno FROM Mensalidade m JOIN Aluno a ON m.cod\_aluno = a.cod\_aluno;



Mostra os pagamentos junto com o nome do professor envolvido.

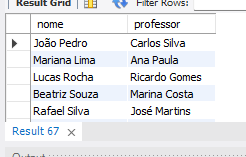
SELECT ep.valor\_pago, p.nome AS professor FROM Efetua\_Pagamento ep JOIN Professor p ON ep.cod\_professor = p.cod\_professor;



1. 4 consultas com tipos de JOIN: INNER, LEFT, RIGHT

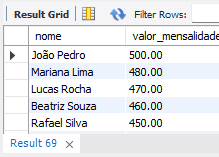
Só mostra quando há correspondência nas duas tabelas

SELECT a.nome, p.nome AS professor FROM Aluno a INNER JOIN Professor p ON a.cod\_professor = p.cod\_professor;



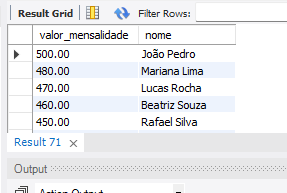
Mostra todos os alunos, mesmo sem mensalidade

SELECT a.nome, m.valor\_mensalidade FROM Aluno a LEFT JOIN Mensalidade m ON a.cod\_aluno = m.cod\_aluno;



Mostra todas as mensalidades, mesmo se aluno não existir

SELECT m.valor\_mensalidade, a.nome FROM Mensalidade m RIGHT JOIN Aluno a ON m.cod\_aluno = a.cod\_aluno;



Mostra todas as secretarias, mesmo se não tiverem pagamentos

SELECT s.cod\_secretaria, ep.valor\_pago FROM Secretaria s LEFT JOIN Efetua\_Pagamento ep ON s.cod\_secretaria = ep.cod\_secretaria;

