

Flutter – Banco de dados em aplicações Móveis

Professora: Angélica Karize

Aluno: David Luis Kim

Consultas SQL úteis para gerar relatórios com base na tabela mensalidade e as demais tabelas do seu banco escola.

Data de entrega: Aberto: 17/07/2025 Vencimento: 24/07/2025

Descrição da Atividade

“Consultas SQL úteis para gerar relatórios com base na tabela mensalidade e nas demais tabelas do banco de dados escola, como aluno, turma e curso. As consultas a seguir foram desenvolvidas para apoiar o controle financeiro e acadêmico, identificando inadimplência, arrecadações e análises por vencimento.”

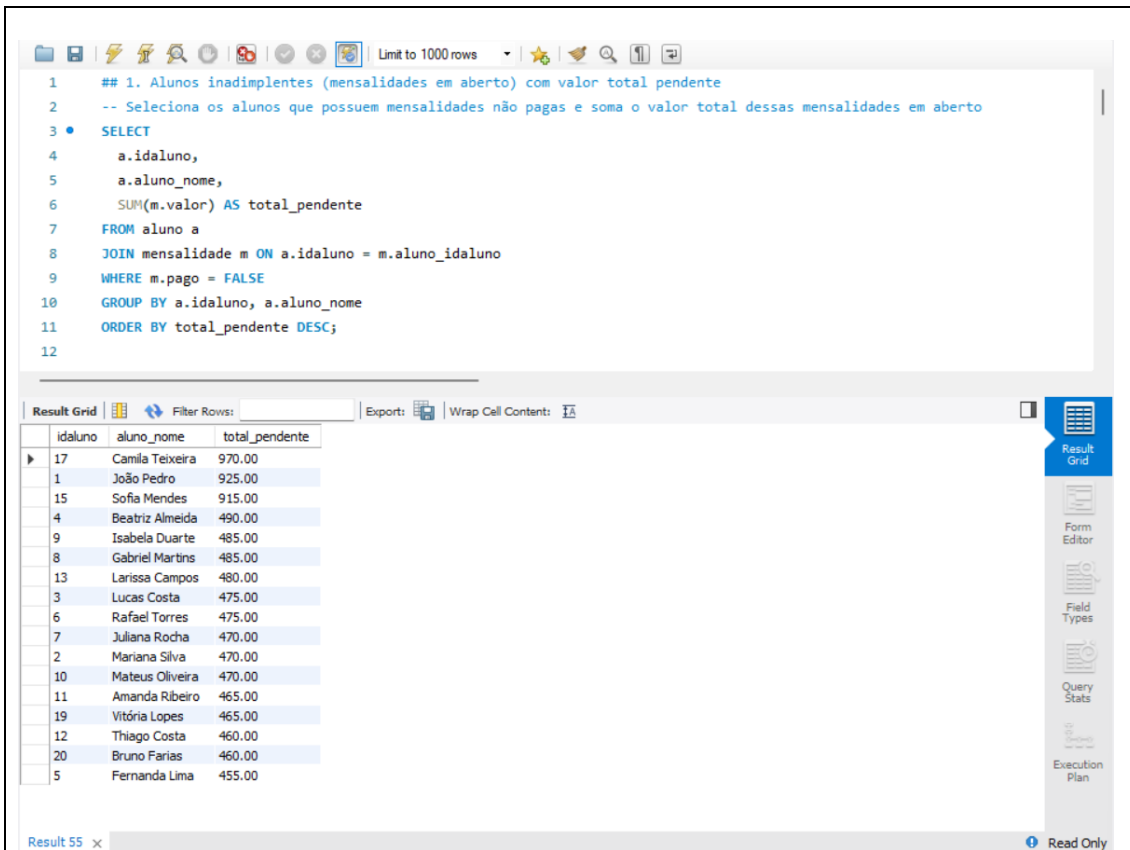
Preparação e População do Banco de Dados

Para viabilizar a execução das consultas SQL e gerar relatórios significativos, foi criado e estruturado um banco de dados chamado escola. As etapas realizadas foram:

- **Criação do schema e das tabelas:**
Foi criado o schema escola, dentro do qual foram definidas todas as tabelas necessárias, como aluno, departamento, professor, disciplina, turma, aproveitamento e mensalidade, respeitando os relacionamentos entre elas por meio de chaves estrangeiras.
- **Inserção de dados:**
As tabelas foram populadas com dados fictícios e coerentes com a realidade escolar, incluindo alunos com informações completas, professores vinculados a departamentos, disciplinas, turmas com horários definidos, registros de notas (aproveitamento) e mensalidades.
- **Atualização de mensalidades:**
As mensalidades previamente inseridas foram atualizadas para simular um cenário realista, com diferentes valores, datas de vencimento e status de

pagamento (pagas e em aberto), o que permite testar e validar as consultas de forma eficaz.

Essas ações garantiram um ambiente de dados consistente para a geração dos relatórios apresentados a seguir:



The screenshot shows a SQL query editor with a toolbar at the top. The query is as follows:

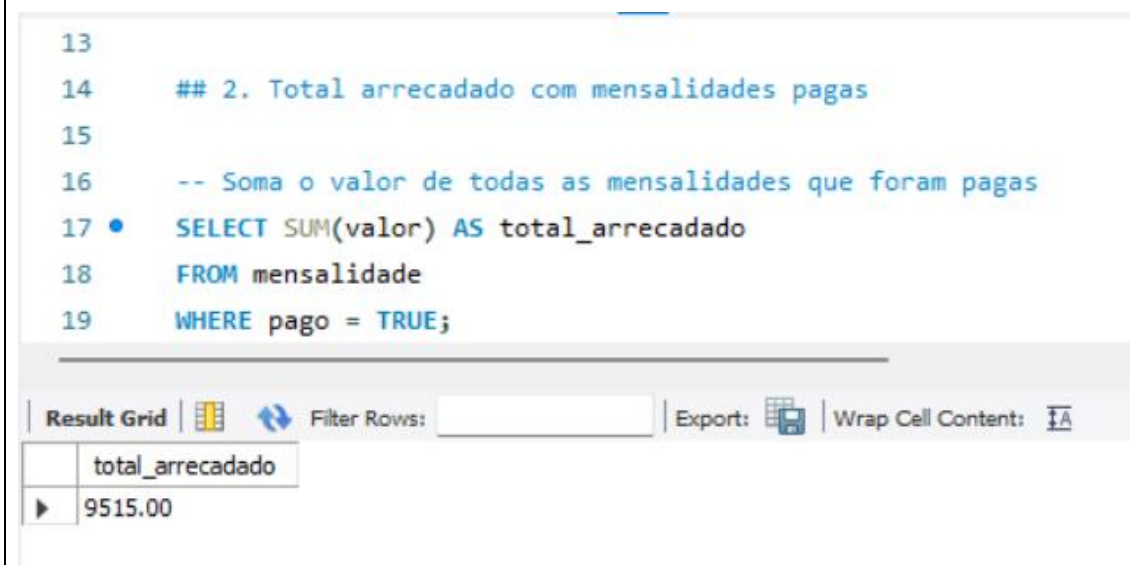
```
1  ## 1. Alunos inadimplentes (mensalidades em aberto) com valor total pendente
2  -- Selecciona os alunos que possuem mensalidades não pagas e soma o valor total dessas mensalidades em aberto
3  • SELECT
4      a.idaluno,
5      a.aluno_nome,
6      SUM(m.valor) AS total_pendente
7  FROM aluno a
8  JOIN mensalidade m ON a.idaluno = m.aluno_idaluno
9  WHERE m.pago = FALSE
10 GROUP BY a.idaluno, a.aluno_nome
11 ORDER BY total_pendente DESC;
12
```

Below the query, the 'Result Grid' is displayed, showing a table with 3 columns: 'idaluno', 'aluno_nome', and 'total_pendente'. The results are sorted by 'total_pendente' in descending order.

idaluno	aluno_nome	total_pendente
17	Camila Teixeira	970.00
1	João Pedro	925.00
15	Sofia Mendes	915.00
4	Beatriz Almeida	490.00
9	Isabela Duarte	485.00
8	Gabriel Martins	485.00
13	Larissa Campos	480.00
3	Lucas Costa	475.00
6	Rafael Torres	475.00
7	Juliana Rocha	470.00
2	Mariana Silva	470.00
10	Mateus Oliveira	470.00
11	Amanda Ribeiro	465.00
19	Vitória Lopes	465.00
12	Thiago Costa	460.00
20	Bruno Farias	460.00
5	Fernanda Lima	455.00

The interface includes a 'Filter Rows' field, an 'Export' button, and a 'Wrap Cell Content' checkbox. On the right side, there is a vertical toolbar with icons for 'Result Grid', 'Form Editor', 'Field Types', 'Query Stats', and 'Execution Plan'. The status bar at the bottom indicates 'Result 55' and 'Read Only'.

Print 01: Alunos inadimplentes (mensalidades em aberto) com valor total pendente.



The screenshot shows a SQL query editor with a toolbar at the top. The query is as follows:

```
13
14  ## 2. Total arrecadado com mensalidades pagas
15
16  -- Soma o valor de todas as mensalidades que foram pagas
17  • SELECT SUM(valor) AS total_arrecadado
18  FROM mensalidade
19  WHERE pago = TRUE;
```

Below the query, the 'Result Grid' is displayed, showing a table with 1 column: 'total_arrecadado'. The result is 9515.00.

total_arrecadado
9515.00

The interface includes a 'Filter Rows' field, an 'Export' button, and a 'Wrap Cell Content' checkbox. The status bar at the bottom indicates 'Result 55' and 'Read Only'.

Print 02: Total arrecadado com mensalidades pagas.

```

21  ## 3. Total pendente (mensalidades não pagas)
22
23  -- Soma o valor de todas as mensalidades que ainda não foram pagas
24  • SELECT SUM(valor) AS total_pendente
25      FROM mensalidade
26      WHERE pago = FALSE;
27

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
total_pendente			
9415.00			




Print 03: Total pendente (mensalidades não pagas).

Limit to 1000 rows

```

28 ## 4. Relatório de status de pagamento por aluno
29 -- Para cada aluno, mostra: total pago, total pendente, quantidade de mensalidades abertas, quantidade de mensalidades pagas
30 • SELECT
31     a.idaluno,
32     a.aluno_nome,
33     SUM(CASE WHEN m.pago = TRUE THEN m.valor ELSE 0 END) AS total_pago,
34     SUM(CASE WHEN m.pago = FALSE THEN m.valor ELSE 0 END) AS total_pendente,
35     COUNT(CASE WHEN m.pago = FALSE THEN 1 END) AS mensalidades_abertas,
36     COUNT(CASE WHEN m.pago = TRUE THEN 1 END) AS mensalidades_pagas
37 FROM aluno a
38 LEFT JOIN mensalidade m ON a.idaluno = m.aluno_idaluno
39 GROUP BY a.idaluno, a.aluno_nome

```

Result Grid  Filter Rows: Export:  Wrap Cell Content: 

idaluno	aluno_nome	total_pago	total_pendente	mensalidades_abertas	mensalidades_pagas
1	João Pedro	0.00	925.00	2	0
2	Mariana Silva	470.00	470.00	1	1
3	Lucas Costa	480.00	475.00	1	1
4	Beatriz Almeida	480.00	490.00	1	1
5	Fernanda Lima	455.00	455.00	1	1
6	Rafael Torres	460.00	475.00	1	1
7	Juliana Rocha	465.00	470.00	1	1
8	Gabriel Martins	475.00	485.00	1	1
9	Isabela Duarte	495.00	485.00	1	1
10	Mateus Oliveira	495.00	470.00	1	1
11	Amanda Ribeiro	480.00	465.00	1	1
12	Thiago Costa	470.00	460.00	1	1
13	Larissa Campos	470.00	480.00	1	1
14	Diego Freitas	945.00	0.00	0	2
15	Sofia Mendes	0.00	915.00	2	0
16	Leonardo Brito	940.00	0.00	0	2
17	Camila Teixeira	0.00	970.00	2	0
18	Eduardo Nunes	990.00	0.00	0	2
19	Vitória Lopes	475.00	465.00	1	1
20	Bruno Farias	470.00	460.00	1	1

Result 59 x

Result Grid

Form Editor

Field Types

Query Stats

Execution Plan

Read On

Print 04: Relatório de status de pagamento por aluno.

```

42  ## 5. Média das mensalidades pagas
43
44  -- Calcula a média do valor das mensalidades que foram pagas
45  • SELECT
46      CAST(AVG(m.valor) AS DECIMAL(10,2)) AS media_mensalidades_pagas
47  FROM mensalidade m
48  WHERE m.pago = TRUE;
49

```

media_mensalidades_pagas
475.75

Print 05: Média das mensalidades pagas.

```

51  ## 6. Mensalidades por vencimento (ordenado)
52  -- Lista todas as mensalidades com dados do aluno
53  • SELECT
54      a.idaluno,
55      a.aluno_nome,
56      m.valor,
57      m.vencimento,
58      CASE -- 0 e 1 Pago e não pago
59          WHEN m.pago = TRUE THEN 'PAGO'
60          ELSE 'ATRASADO'
61      END AS status_pagamento
62  FROM mensalidade m
63  JOIN aluno a ON m.aluno_idaluno = a.idaluno
64  ORDER BY m.vencimento ASC;

```

idaluno	aluno_nome	valor	vencimento	status_pagamento
1	João Pedro	465.00	2025-07-01	ATRASADO
2	Mariana Silva	470.00	2025-07-02	PAGO
3	Lucas Costa	475.00	2025-07-03	ATRASADO
4	Beatriz Almeida	480.00	2025-07-04	PAGO
5	Fernanda Lima	455.00	2025-07-06	ATRASADO
6	Rafael Torres	460.00	2025-07-07	PAGO
7	Juliana Rocha	470.00	2025-07-08	ATRASADO
8	Gabriel Martins	475.00	2025-07-09	PAGO
9	Isabela Duarte	485.00	2025-07-10	ATRASADO
10	Mateus Oliveira	495.00	2025-07-11	PAGO
11	Amanda Ribeiro	465.00	2025-07-12	ATRASADO
12	Thiago Costa	470.00	2025-07-13	PAGO
13	Larissa Campos	480.00	2025-07-14	ATRASADO
14	Diego Freitas	455.00	2025-07-15	PAGO
15	Sofia Mendes	460.00	2025-07-16	ATRASADO
16	Leonardo Brito	475.00	2025-07-17	PAGO

Print 06: Mensalidades por vencimento (ordenado) 1/2.

2	Mariana Silva	470.00	2025-09-10	ATRASADO
7	Juliana Rocha	465.00	2025-09-10	PAGO
12	Thiago Costa	460.00	2025-09-10	ATRASADO
17	Camila Teixeira	485.00	2025-09-10	ATRASADO
3	Lucas Costa	480.00	2025-09-15	PAGO
8	Gabriel Martins	485.00	2025-09-15	ATRASADO
13	Larissa Campos	470.00	2025-09-15	PAGO
18	Eduardo Nunes	495.00	2025-09-15	PAGO
4	Beatriz Almeida	490.00	2025-09-20	ATRASADO
9	Isabela Duarte	495.00	2025-09-20	PAGO
14	Diego Freitas	490.00	2025-09-20	PAGO
19	Vitória Lopes	475.00	2025-09-20	PAGO
5	Fernanda Lima	455.00	2025-09-25	PAGO

Print 07: Mensalidades por vencimento (ordenado) 2/2.

Conclusão

Com a realização das consultas SQL propostas, foi possível extrair informações relevantes a partir da base de dados da escola. Os relatórios gerados permitiram:

- Identificar **alunos inadimplentes**, facilitando o controle de cobranças;
- Calcular o **total arrecadado** e o **valor pendente**, oferecendo uma visão clara da situação financeira;
- Avaliar o **status de pagamento por aluno**, proporcionando um acompanhamento individualizado;
- Determinar a **média das mensalidades pagas**, o que pode auxiliar na previsão de receita;
- Obter uma listagem das **mensalidades ordenadas por vencimento**, permitindo melhor organização das cobranças futuras.

Essas consultas demonstram o poder do SQL na geração de relatórios precisos e estratégicos, fundamentais para a tomada de decisão no contexto educacional.