

## 1. Junções (JOINS)

### 1.1

```
SELECT  
  
u.nome AS Nome_Professor,  
  
d.nome AS Disciplina,  
  
t.semestre  
  
FROM professor p  
  
JOIN usuario_aluno_professor u ON p.id_usuario_professor = u.id_usuario  
  
JOIN turma t ON p.id_professor = t.id_professor  
  
JOIN disciplina d ON t.id_disciplina = d.id_disciplina  
  
WHERE t.semestre = '2025/1';
```

Descrição: Lista o nome do professor, o nome da disciplina e o semestre de todas as turmas cadastradas no semestre '2025/1'. Envolve 4 tabelas.

### 1.2

```
SELECT  
  
u.nome AS Nome_Aluno,  
  
c.nome AS Curso,  
  
m.status AS Status_Matricula  
  
FROM aluno a  
  
JOIN usuario_aluno_professor u ON a.id_usuario_aluno = u.id_usuario  
  
JOIN matricula m ON a.id_aluno = m.id_aluno  
  
JOIN turma t ON m.id_turma = t.id_turma  
  
JOIN disciplina d ON t.id_disciplina = d.id_disciplina  
  
JOIN curso c ON d.id_curso = c.id_curso;
```

Descrição: Apresenta o nome completo do aluno, o nome do curso em que a disciplina está alocada e o status da sua matrícula em cada turma/disciplina.

## 2. Conjuntos (UNION, INTERSECT, EXCEPT/MINUS)

### 2.1

```
SELECT
    u.nome,
    u.email
FROM usuario_aluno_professor u
JOIN aluno a ON u.id_usuario = a.id_usuario_aluno
JOIN matricula m ON a.id_aluno = m.id_aluno
```

UNION

```
SELECT
    u.nome,
    u.email
FROM usuario_aluno_professor u
JOIN professor p ON u.id_usuario = p.id_usuario_professor
JOIN turma t ON p.id_professor = t.id_professor;
```

Descrição: UNIÃO (UNION): Lista nomes e emails de usuários que são alunos matriculados ou professores que possuem turmas.

### 2.2

```
SELECT
    d.nome
FROM disciplina d
JOIN turma t ON d.id_disciplina = t.id_disciplina
JOIN curso c ON d.id_curso = c.id_curso
WHERE d.carga_horaria > 60
```

INTERSECT

```
SELECT
    d.nome
FROM disciplina d
```

JOIN turma t ON d.id\_disciplina = t.id\_disciplina

JOIN matricula m ON t.id\_turma = m.id\_turma;

Descrição: INTERSEÇÃO (INTERSECT): Lista disciplinas com carga horária > 60 que também possuem alunos matriculados.

## 2.3

SELECT

d.nome

FROM disciplina d

EXCEPT

SELECT

d.nome

FROM disciplina d

JOIN turma t ON d.id\_disciplina = t.id\_disciplina

JOIN avaliacao a ON t.id\_turma = a.id\_turma;

Descrição: DIFERENÇA (EXCEPT/MINUS): Lista disciplinas sem nenhuma avaliação em turmas.

### 3. Agregações (GROUP BY e HAVING)

#### 3.1

```
SELECT  
  
d.nome AS Disciplina,  
  
COUNT(m.id_matricula) AS Total_Matriculas  
  
FROM disciplina d  
  
JOIN turma t ON d.id_disciplina = t.id_disciplina  
  
JOIN matricula m ON t.id_turma = m.id_turma  
  
GROUP BY d.nome  
  
HAVING COUNT(m.id_matricula) > 10;
```

Descrição: Conta e lista o total de matrículas por disciplina, mas apenas para as disciplinas que tiveram mais de 10 matrículas registradas no total (em todas as suas turmas).

#### 3.2

```
SELECT  
  
c.nome AS Curso,  
  
AVG(d.carga_horaria) AS Media_Carga_Horaria  
  
FROM curso c  
  
JOIN disciplina d ON c.id_curso = d.id_curso  
  
JOIN turma t ON d.id_disciplina = t.id_disciplina  
  
GROUP BY c.nome;
```

Descrição: Calcula e lista a média da carga horária de todas as disciplinas agrupadas por curso.

#### 3.3

```
SELECT  
  
u.nome AS Professor,  
  
COUNT(t.id_turma) AS Total_Turmas  
  
FROM professor p  
  
JOIN usuario_aluno_professor u ON p.id_usuario_professor = u.id_usuario
```

JOIN turma t ON p.id\_professor = t.id\_professor

GROUP BY u.nome

HAVING COUNT(t.id\_turma) = 0;

Descrição: Conta e lista o nome dos professores que não têm nenhuma turma associada a eles.

### 3.4

SELECT

m.id\_matricula,

AVG(i.nota) AS Media\_Geral\_Notas

FROM matricula m

JOIN item\_nota i ON m.id\_matricula = i.id\_matricula

GROUP BY m.id\_matricula

HAVING AVG(i.nota) < 6.0;

Descrição: Calcula a média das notas por matrícula e lista o ID da matrícula para os alunos com média geral inferior a 6.0.

## 4. LIKE, BETWEEN e IN

### 4.1

```
SELECT
    u.nome,
    d.nome AS Disciplina,
    t.semestre
FROM usuario_aluno_professor u
JOIN aluno a ON u.id_usuario = a.id_usuario_aluno
JOIN matricula m ON a.id_aluno = m.id_aluno
JOIN turma t ON m.id_turma = t.id_turma
JOIN disciplina d ON t.id_disciplina = d.id_disciplina
WHERE u.nome LIKE 'Maria %';
```

Descrição: LIKE: Lista o nome de todos os usuários (alunos ou professores) cujo nome começa com 'Maria' seguido por um espaço e qualquer outra coisa.

### 4.2

```
SELECT
    d.nome AS Disciplina,
    d.carga_horaria,
    c.nome AS Curso,
    t.semestre
FROM disciplina d
JOIN curso c ON d.id_curso = c.id_curso
JOIN turma t ON d.id_disciplina = t.id_disciplina
WHERE d.carga_horaria BETWEEN 40 AND 80;
```

Descrição: BETWEEN: Lista o nome e a carga horária das disciplinas cuja carga horária está entre 40 e 80 horas (inclusive).

### 4.3

SELECT

t.id\_turma,

t.semestre,

d.nome AS Disciplina,

u.nome AS Professor

FROM turma t

JOIN disciplina d ON t.id\_disciplina = d.id\_disciplina

JOIN professor p ON t.id\_professor = p.id\_professor

JOIN usuario\_aluno\_professor u ON p.id\_usuario\_professor = u.id\_usuario

WHERE t.semestre IN ('2025/1', '2024/2');

Descrição: IN: Lista o ID e o semestre das turmas que foram abertas nos semestres '2025/1' OU '2024/2'.

## 5. Visões (VIEWS) Envolvendo 3+ Tabelas

### 5.1

CREATE VIEW

vw\_aluno\_turma\_disciplina AS

SELECT

u.nome AS Nome\_Aluno,

d.nome AS Disciplina,

t.semestre AS Semestre\_Turma

FROM aluno a

JOIN usuario\_aluno\_professor u ON a.id\_usuario\_aluno = u.id\_usuario

JOIN matricula m ON a.id\_aluno = m.id\_aluno

JOIN turma t ON m.id\_turma = t.id\_turma

JOIN disciplina d ON t.id\_disciplina = d.id\_disciplina;

Descrição: Cria uma visão que associa o Nome do Aluno (via usuario\_aluno\_professor), a Disciplina e o Semestre em que ele está matriculado, simplificando a consulta a essas informações. (5 tabelas)

### 5.2

CREATE VIEW

vw\_professores\_cursos AS

SELECT

u.nome AS Nome\_Professor,

d.nome AS Disciplina\_Lecionada,

c.nome AS Curso\_Associado

FROM professor p

JOIN usuario\_aluno\_professor u ON p.id\_usuario\_professor = u.id\_usuario

JOIN turma t ON p.id\_professor = t.id\_professor

JOIN disciplina d ON t.id\_disciplina = d.id\_disciplina

JOIN curso c ON d.id\_curso = c.id\_curso

GROUP BY u.nome, d.nome, c.nome;

Descrição: Cria uma visão que lista o Nome do Professor, as Disciplinas que ele leciona e o Curso ao qual cada disciplina pertence. (5 tabelas)