

1-

```
SELECT a.aluno_id, a.nome, a.email, m.data_matricula
FROM aluno a
JOIN matricula m ON a.aluno_id = m.aluno_id
JOIN curso c ON m.curso_id = c.curso_id
WHERE c.titulo = 'Banco de Dados'
ORDER BY a.nome;
```

2-

```
select titulo, carga_horaria from curso where carga_horaria > 40
```

3 -

```
SELECT a.aluno_id, a.nome, c.titulo AS curso, m.data_matricula
FROM aluno a
JOIN matricula m ON a.aluno_id = m.aluno_id
JOIN curso c ON m.curso_id = c.curso_id
WHERE m.nota IS NULL;
```

4-

```
SELECT
    a.nome AS aluno,
    c.titulo AS curso,
    m.data_matricula,
    m.nota
FROM
    matricula m
JOIN
    aluno a ON m.aluno_id = a.aluno_id
JOIN
    curso c ON m.curso_id = c.curso_id
WHERE
    m.data_matricula > TO_DATE('2024-01-01', 'YYYY-MM-DD')
```

5-

```
SELECT
    curso_id,
    titulo AS nome_curso,
    carga_horaria
FROM
    curso
WHERE
    carga_horaria BETWEEN 30 AND 60
```

6-

```
SELECT
    aluno_id,
    nome,
    email
FROM
    aluno
WHERE
    email LIKE '%@gmail.com'
```

7-

```
SELECT
    a.nome AS aluno,
    c.titulo AS curso,
    m.data_matricula
FROM
    aluno a
JOIN
    matricula m ON a.aluno_id = m.aluno_id
JOIN
    curso c ON m.curso_id = c.curso_id
ORDER BY
    a.nome, c.titulo;
```

8-

```
SELECT
    a.nome AS aluno,
    c.titulo AS curso,
    m.nota,
    CASE
        WHEN m.nota IS NULL THEN 'Sem nota'
        WHEN m.nota >= 7 THEN 'Aprovado'
        ELSE 'Reprovado'
    END AS status
FROM
    aluno a
JOIN
    matricula m ON a.aluno_id = m.aluno_id
JOIN
    curso c ON m.curso_id = c.curso_id
ORDER BY
    a.nome, c.titulo;
```

9-

```
SELECT
  c.curso_id,
  c.titulo AS nome_curso,
  c.carga_horaria,
  m.data_matricula,
  m.nota
FROM
  curso c
JOIN
  matricula m ON c.curso_id = m.curso_id
JOIN
  aluno a ON m.aluno_id = a.aluno_id
WHERE
  a.nome = 'João Silva'
```

10-

```
SELECT
  c.titulo AS curso,
  COUNT(m.aluno_id) AS total_alunos
FROM
  curso c
JOIN
  matricula m ON c.curso_id = m.curso_id
GROUP BY
  c.titulo
HAVING
  COUNT(m.aluno_id) > 1
ORDER BY total_alunos
```

11-

```
SELECT
  a.aluno_id,
  a.nome,
  a.email
FROM
  aluno a
LEFT JOIN
  matricula m ON a.aluno_id = m.aluno_id
WHERE
  m.aluno_id IS NULL;
```

12-

```
SELECT
    c.curso_id,
    c.titulo AS nome_curso,
    c.carga_horaria
FROM
    curso c
LEFT JOIN
    matricula m ON c.curso_id = m.curso_id
WHERE
    m.curso_id IS NULL;
```

13-

```
SELECT
    a.nome AS aluno,
    COUNT(m.curso_id) AS quantidade_cursos
FROM
    aluno a
LEFT JOIN
    matricula m ON a.aluno_id = m.aluno_id
GROUP BY
    a.nome
ORDER BY
    quantidade_cursos DESC, a.nome;
```

14-

```
SELECT
    a.nome AS aluno,
    ROUND(AVG(m.nota), 2) AS media_notas,
    COUNT(m.curso_id) AS cursos_com_nota
FROM
    aluno a
JOIN
    matricula m ON a.aluno_id = m.aluno_id
WHERE
    m.nota IS NOT NULL
GROUP BY
    a.nome
ORDER BY
    media_notas DESC;
```

15-

```
SELECT
    c.titulo AS curso,
    ROUND(AVG(m.nota), 2) AS media_notas,
    COUNT(m.aluno_id) AS alunos_avalidados
FROM
    curso c
LEFT JOIN
    matricula m ON c.curso_id = m.curso_id AND m.nota IS NOT NULL
GROUP BY
    c.titulo
ORDER BY
    media_notas DESC;
```

16-

```
SELECT
    a.nome AS aluno,
    c.titulo AS curso,
    m.nota AS maior_nota
FROM
    matricula m
JOIN
    aluno a ON m.aluno_id = a.aluno_id
JOIN
    curso c ON m.curso_id = c.curso_id
WHERE
    m.nota = (SELECT MAX(nota) FROM matricula WHERE nota IS NOT NULL);
```

17 -

```
SELECT
    a.nome AS aluno,
    c.titulo AS curso,
    m.nota AS menor_nota,
    m.data_matricula
FROM
    matricula m
JOIN
    aluno a ON m.aluno_id = a.aluno_id
JOIN
    curso c ON m.curso_id = c.curso_id
WHERE
    m.nota = (SELECT MIN(nota) FROM matricula WHERE nota IS NOT NULL);
```

18 -

```
SELECT
    c.titulo AS curso,
    COUNT(m.aluno_id) AS total_matriculas
FROM
    curso c
LEFT JOIN
    matricula m ON c.curso_id = m.curso_id
GROUP BY
    c.titulo
ORDER BY
    total_matriculas DESC
```

19 -

```
SELECT
    a.nome AS aluno,
    ROUND(AVG(m.nota), 2) AS media_notas,
    COUNT(m.curso_id) AS quantidade_cursos
FROM
    aluno a
JOIN
    matricula m ON a.aluno_id = m.aluno_id
WHERE
    m.nota IS NOT NULL
GROUP BY
    a.nome
HAVING
    AVG(m.nota) >= 8.0
ORDER BY
    media_notas DESC;
```

20 -

```
SELECT
    c.titulo AS curso,
    ROUND(AVG(m.nota), 2) AS media_notas,
    MIN(m.nota) AS menor_nota,
    MAX(m.nota) AS maior_nota,
    COUNT(m.aluno_id) AS alunos_avalidados
FROM
    curso c
LEFT JOIN
    matricula m ON c.curso_id = m.curso_id AND m.nota IS NOT NULL
GROUP BY
    c.titulo
ORDER BY
    media_notas DESC;
```