



```
CREATE database publicacoes;
USE publicacoes;

CREATE TABLE assunto(
    codigo INT PRIMARY KEY AUTO_INCREMENT,
    descricao VARCHAR(45) NOT NULL
);

CREATE TABLE editora(
    codigo INT PRIMARY KEY AUTO_INCREMENT,
    cnpj CHAR(14) NOT NULL UNIQUE,
    nome VARCHAR(45) NOT NULL
);

CREATE TABLE autor(
    codigo INT PRIMARY KEY AUTO_INCREMENT,
    nome VARCHAR(100) NOT NULL,
    passaporte CHAR(6) NOT NULL UNIQUE,
    dataNascimento DATE NOT NULL,
    nacionalidade_codigo INT NOT NULL
);

CREATE TABLE livro(
    codigo INT PRIMARY KEY AUTO_INCREMENT,
    isbn CHAR(13) NOT NULL UNIQUE,
    titulo VARCHAR(100) NOT NULL,
    preco DECIMAL(10,2) NOT NULL,
    dataLancamento DATE NULL,
    assunto_codigo INT NOT NULL,
    editora_codigo INT NOT NULL
);

CREATE TABLE autor_livro(
    autor_codigo INT NOT NULL,
    livro_codigo INT NOT NULL,
    PRIMARY KEY (autor_codigo, livro_codigo)
);

CREATE TABLE nacionalidade(
    codigo INT PRIMARY KEY AUTO_INCREMENT,
    pais VARCHAR(45) NOT NULL
);

ALTER TABLE autor
    ADD CONSTRAINT fk_nacionalidade_codigo FOREIGN KEY (nacionalidade_codigo) REFERENCES nacionalidade(codigo);

ALTER TABLE livro
    ADD FOREIGN KEY (assunto_codigo) REFERENCES assunto(codigo),
    ADD FOREIGN KEY (editora_codigo) REFERENCES editora(codigo);

ALTER TABLE autor_livro
    ADD FOREIGN KEY (autor_codigo) REFERENCES autor(codigo),
    ADD FOREIGN KEY (livro_codigo) REFERENCES livro(codigo);

-- seed atÃ© a decima questÃ£o

-- Inserindo assuntos
INSERT INTO assunto (descricao) VALUES
('Banco de Dados'),
('Estruturas de Dados'),
('Algoritmos'),
('Engenharia de Software');
```

```
-- Inserindo editoras
INSERT INTO editora (cnpj, nome) VALUES
('12345678000101', 'Editora A'),
('98765432000102', 'Editora B');

-- Inserindo nacionalidades
INSERT INTO nacionalidade (pais) VALUES
('Brasil'),
('EUA');

-- Inserindo autores
INSERT INTO autor (nome, passaporte, dataNascimento, nacionalidade_codigo) VALUES
('João Silva', 'AB1234', '1980-05-20', 1),
('Mary Johnson', 'CD5678', '1975-08-15', 2);

-- Inserindo livros
INSERT INTO livro (isbn, titulo, preco, dataLancamento, assunto_codigo, editora_codigo) VALUES
('9780000000001', 'Banco de Dados Avançado', 120.00, '2015-06-10', 1, 1),
('9780000000002', 'Estruturas de Dados em Java', 80.00, '2018-09-01', 2, 1),
('9780000000003', 'Algoritmos para Iniciantes', 45.00, '2020-02-15', 3, 2),
('9780000000004', 'Banco de Dados', 150.00, '2020-02-14', 1, 1),
('9780000000005', 'Banco e Finanças', 60.00, '2022-01-01', 1, 2),
('9780000000006', 'Análise de Dados', 55.00, '2023-01-01', 4, 2),
('9780000000007', 'Banco Relacional de Dados', 200.00, '2010-05-10', 1, 2),
('9780000000008', 'Engenharia de Software Moderna', 75.00, '2024-05-20', 4, 1);

SELECT * FROM livro;

-- Relacionando autores e livros
INSERT INTO autor_livro (autor_codigo, livro_codigo) VALUES
(1, 1), (2, 2), (1, 3), (2, 4), (1, 5), (2, 6), (1, 7), (2, 8);
-- fim seed até a decima questão

-- Adicionando mais livros para enriquecer os dados estáticos
INSERT INTO livro (isbn, titulo, preco, dataLancamento, assunto_codigo, editora_codigo) VALUES
('9780000000009', 'Banco de Dados para Web', 95.00, NULL, 1, 2),
('9780000000010', 'Algoritmos Avançados', 220.00, '2017-10-01', 3, 1),
('9780000000011', 'Estruturas Lineares', 110.00, '2012-03-15', 2, 1),
('9780000000012', 'Engenharia de Software Avançada', 130.00, NULL, 4, 2),
('9780000000013', 'Banco de Dados NoSQL', 160.00, '2019-06-20', 1, 2),
('9780000000014', 'Lógica de Programação', 35.00, '2021-11-10', 3, 1),
('9780000000015', 'Dados Corporativos', 190.00, '2014-04-05', 4, 1),
('9780000000016', 'Estruturas de Dados com Python', 105.00, '2010-09-09', 2, 2);

-- fim até o 20
INSERT INTO autor (nome, passaporte, dataNascimento, nacionalidade_codigo)
VALUES ('Antônio Velho', 'EF7890', '1915-01-01', 1);
-- fim até 40

-- 1. Livros que possuam preços superiores a R$ 50,00.
SELECT *
FROM livro
WHERE preco > 50;

-- 2. Livros que possuam preços entre R$ 100,00 e R$ 200,00.
SELECT *
FROM livro
WHERE preco BETWEEN 100 AND 200;
```

```
SELECT *
FROM livro
WHERE preco >= 100 AND preco <= 200;
```

-- 3. Livros cujos tÃ-tulos possuam a palavra â\200\230Bancoâ\200\231.

```
SELECT titulo
FROM livro
WHERE LOWER(titulo) LIKE '%banco%';
```

```
-- 4. Livros cujos tÃ-tulos iniciam com a palavra â\200\230Bancoâ\200\231.
```

```
SELECT *
FROM livro
WHERE preco BETWEEN 100 AND 200;
```

```
SELECT *
FROM livro
WHERE preco >= 100 AND preco <= 200;
```

```
-- 3. Livros cujos tÃ-tulos possuam a palavra â\200\230Bancoâ\200\231.
```

```
SELECT titulo
FROM livro
WHERE LOWER(titulo) LIKE '%banco%';
```

```
-- 4. Livros cujos tÃ-tulos iniciam com a palavra â\200\230Bancoâ\200\231.
```

```
SELECT titulo
FROM livro
WHERE LOWER(titulo) LIKE 'banco%';
```

```
-- 5. Livros cujos tÃ-tulos terminam com a palavra â\200\230Dadosâ\200\231.
```

```
SELECT titulo
FROM livro
WHERE LOWER(titulo) LIKE '%dados';
```

-- 6. Livros cujos tÃ-tulos possuem a expressÃo "Banco de Dados" ou "Bancos de Dados".

```
SELECT titulo
FROM livro
WHERE LOWER(titulo) LIKE '%banco de dados%'
OR LOWER(titulo) LIKE '%bancos de dados%';
```

-- 7. Livros que foram lanados h; mais de 5 anos.

```
SELECT *
FROM livro
WHERE dataLancamento < DATE SUB(CURDATE(), INTERVAL 5 YEAR);
```

-- 8. Livros que ainda nÃ£o foram lanÃ§ados, ou seja, com a data de lanÃ§amento nula.

```
SELECT *
FROM livro
WHERE dataLancamento IS NULL;
```

-- 9. Livros cujo assunto seja "Estruturas de Dados".

```
SELECT *
FROM livro
JOIN assunto ON livro.assunto_codigo = assunto.codigo
WHERE LOWER(assunto.descricao) LIKE '%estruturas de dados%';
```

-- 10. Livros cujo assunto tenha código 1, 2 ou 3.

```
SELECT * FROM livro
```

```
WHERE assunto_codigo IN (1, 2, 3);

-- 11. Quantidade de livros
SELECT COUNT(*) AS quantidade_livros FROM livro;

-- 12. Quantidade de livros ainda não lançados (data nula)
SELECT COUNT(*) AS livros_nao_lancados
FROM livro
WHERE dataLancamento IS NULL;

-- 13. Soma dos preços dos livros
SELECT SUM(preco) AS soma_precos FROM livro;

-- 14. Média dos preços dos livros
SELECT AVG(preco) AS media_precos FROM livro;

-- 15. Maior preço dos livros
SELECT MAX(preco) AS maior_preco FROM livro;

-- 16. Menor preço dos livros
SELECT MIN(preco) AS menor_preco FROM livro;

-- 17. Preço médio dos livros para cada assunto
SELECT a.descricao AS assunto, ROUND(AVG(l.preco), 2) AS preco_medio
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
GROUP BY a.descricao;

-- 18. Quantidade de livros para cada assunto
SELECT a.descricao AS assunto, COUNT(*) AS quantidade
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
GROUP BY a.descricao;

-- 19. Preço do livro mais caro de cada assunto (somente os já lançados)
SELECT a.descricao AS assunto, MAX(l.preco) AS preco_maximo
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
WHERE l.dataLancamento IS NOT NULL
GROUP BY a.descricao;

-- 20. Quantidade de livros lançados por editora
SELECT e.nome AS editora, COUNT(*) AS livros_lancados
FROM livro l
JOIN editora e ON l.editora_codigo = e.codigo
WHERE l.dataLancamento IS NOT NULL
GROUP BY e.nome;

-- 21. Assuntos cujo preço médio dos livros ultrapassa R$ 50,00.
SELECT a.descricao AS assunto, AVG(l.preco) AS media_preco
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
GROUP BY a.descricao
HAVING AVG(l.preco) > 50.00;

-- 22. Assuntos que possuem pelo menos 2 livros.
SELECT a.descricao AS assunto, COUNT(*) AS quantidade
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
GROUP BY a.descricao
HAVING COUNT(*) >= 2;

-- 23. Assuntos que possuem pelo menos 2 livros já lançados.
```

```
SELECT a.descricao AS assunto, COUNT(*) AS quantidade_lancados
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
WHERE l.dataLancamento IS NOT NULL
GROUP BY a.descricao
HAVING COUNT(*) >= 2;
```

```
-- 24. Quantidade de livros lançados por assunto.
SELECT a.descricao AS assunto, COUNT(*) AS livros_lancados
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
WHERE l.dataLancamento IS NOT NULL
GROUP BY a.descricao;
```

```
-- 25. Nome e passaporte dos autores que possuem a palavra 'João' no nome.
SELECT nome, passaporte
FROM autor
WHERE nome LIKE '%João%';
```

```
-- 26. Nome e passaporte dos autores que nasceram após 1º de janeiro de 1970.
SELECT nome, passaporte
FROM autor
WHERE dataNascimento > '1970-01-01';
```

```
-- 27. Nome e passaporte dos autores que não são brasileiros.
SELECT a.nome, a.passaporte
FROM autor a
JOIN nacionalidade n ON a.nacionalidade_codigo = n.codigo
WHERE n.pais != 'Brasil';
```

```
-- 28. Quantidade de autores.
SELECT COUNT(*) AS quantidade_autores FROM autor;
```

```
-- 29. Quantidade média de autores dos livros.
SELECT AVG(qtd_autores) AS media_autores_por_livro
FROM (
    SELECT COUNT(*) AS qtd_autores
    FROM autor_livro
    GROUP BY livro_codigo
) AS subconsulta;
```

```
-- 30. Livros que possuem ao menos 2 autores.
SELECT l.titulo, COUNT(al.autor_codigo) AS qtd_autores
FROM livro l
JOIN autor_livro al ON l.codigo = al.livro_codigo
GROUP BY l.codigo, l.titulo
HAVING COUNT(al.autor_codigo) >= 2;
```

```
-- 31. Preço médio dos livros por editora.
SELECT e.nome AS editora, AVG(l.preco) AS media_preco
FROM livro l
JOIN editora e ON l.editora_codigo = e.codigo
GROUP BY e.nome;
```

```
-- 32. Preço máximo, mínimo e médio dos livros dos assuntos 1, 2 ou 3, por editora.
SELECT e.nome AS editora,
    MAX(l.preco) AS preco_maximo,
    MIN(l.preco) AS preco_minimo,
    AVG(l.preco) AS preco_medio
FROM livro l
JOIN editora e ON l.editora_codigo = e.codigo
WHERE l.assunto_codigo IN (1, 2, 3)
```

```
GROUP BY e.nome;
```

```
-- 33. Quantidade de autores para cada nacionalidade.
```

```
SELECT n.pais AS nacionalidade, COUNT(a.codigo) AS quantidade_autores
FROM autor a
JOIN nacionalidade n ON a.nacionalidade_codigo = n.codigo
GROUP BY n.pais;
```

```
-- 34. Quantidade de autores que nasceram antes de 1º de janeiro de 1920, por nacionalidade.
```

```
SELECT n.pais AS nacionalidade, COUNT(a.codigo) AS autores_antes_1920
FROM autor a
JOIN nacionalidade n ON a.nacionalidade_codigo = n.codigo
WHERE a.dataNascimento < '1920-01-01'
GROUP BY n.pais;
```

```
-- 35. A data de nascimento do autor mais velho.
```

```
SELECT MIN(dataNascimento) AS autor_mais_velho FROM autor;
```

```
-- 36. A data de nascimento do autor mais novo.
```

```
SELECT MAX(dataNascimento) AS autor_mais_novo FROM autor;
```

```
-- 37. Os novos preços dos livros se os valores fossem reajustados em 10%.
```

```
SELECT titulo, preco, preco * 1.10 AS preco_reajustado
FROM livro;
```

```
-- 38. O dia da publicação do livro de código 1.
```

```
SELECT DAY(dataLancamento) AS dia_publicacao
FROM livro
WHERE codigo = 1;
```

```
-- 39. O mês e o ano da publicação dos livros cujo assunto tem código 1.
```

```
SELECT MONTH(dataLancamento) AS mes, YEAR(dataLancamento) AS ano
FROM livro
WHERE assunto_codigo = 1 AND dataLancamento IS NOT NULL;
```

```
-- 40. Quantidade de autores distintos que estão associados a livros.
```

```
SELECT COUNT(DISTINCT autor_codigo) AS autores_distintos
FROM autor_livro;
```

```
-- 41. Título, assunto e preço, ordenado por assunto (crescente) e preço (decrescente).
```

```
SELECT l.titulo, a.descricao AS assunto, l.preco
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
ORDER BY a.descricao ASC, l.preco DESC;
```

```
-- 42. Editoras ordenadas alfabeticamente. A coluna de nomes deve ter o título e o editora.
```

```
SELECT nome AS "Editora"
FROM editora
ORDER BY nome ASC;
```

```
-- 43. Preços e os títulos dos livros, em ordem decrescente de preço.
```

```
SELECT preco, titulo
FROM livro
ORDER BY preco DESC;
```

```
-- 44. Editoras que já publicaram livros, sem repetições.
```

```
SELECT DISTINCT e.nome
FROM livro l
JOIN editora e ON l.editora_codigo = e.codigo;
```

```
-- 45. Autores brasileiros com mês e ano de nascimento, por ordem decrescente de idade e crescente de nome.
```

```
SELECT a.nome, MONTH(a.dataNascimento) AS mes_nascimento, YEAR(a.dataNascimento) AS ano_nascimento
FROM autor a
JOIN nacionalidade n ON a.nacionalidade_codigo = n.codigo
WHERE n.pais = 'Brasil'
ORDER BY a.dataNascimento ASC, a.nome ASC;
```

-- 46. Editora (nome), assunto (código) e quantidade de livros por editora e assunto, em ordem decrescente de quantidade.

```
SELECT e.nome AS editora, l.assunto_codigo AS assunto, COUNT(*) AS quantidade
FROM livro l
JOIN editora e ON l.editora_codigo = e.codigo
GROUP BY e.nome, l.assunto_codigo
ORDER BY quantidade DESC;
```

-- 46. Editora (nome), assunto (código) e quantidade de livros por editora e assunto, em ordem decrescente de quantidade.

```
SELECT e.nome AS editora, l.assunto_codigo AS assunto, COUNT(*) AS quantidade
FROM livro l
JOIN editora e ON l.editora_codigo = e.codigo
GROUP BY e.nome, l.assunto_codigo
ORDER BY quantidade DESC;
```

-- 47. Títulos cujo comprimento seja superior a 15 caracteres.

```
SELECT titulo
FROM livro
WHERE LENGTH(titulo) > 15;
```

-- 48. Títulos dos livros lançados e a descrição dos seus assuntos.

```
SELECT l.titulo, a.descricao AS assunto
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
WHERE l.dataLancamento IS NOT NULL;
```

-- 49. Título do livro, nome da editora que o publicou e a descrição do assunto.

```
SELECT l.titulo, e.nome AS editora, a.descricao AS assunto
FROM livro l
JOIN editora e ON l.editora_codigo = e.codigo
JOIN assunto a ON l.assunto_codigo = a.codigo;
```

-- 50. Editoras e títulos dos livros lançados pela editora, ordenado por nome da editora e título do livro.

```
SELECT e.nome AS editora, l.titulo
FROM livro l
JOIN editora e ON l.editora_codigo = e.codigo
WHERE l.dataLancamento IS NOT NULL
ORDER BY e.nome, l.titulo;
```

-- 51. Editoras cadastradas e, para as que têm livros, também o título, ordenado por nome da editora e título.

```
SELECT e.nome AS editora, l.titulo
FROM editora e
LEFT JOIN livro l ON l.editora_codigo = e.codigo
ORDER BY e.nome, l.titulo;
```

-- 52. Assuntos com os títulos dos livros, ordenados pela descrição do assunto.

```
SELECT a.descricao AS assunto, l.titulo
FROM assunto a
JOIN livro l ON l.assunto_codigo = a.codigo
ORDER BY a.descricao, l.titulo;
```

-- 53. Títulos e editoras, relacionando o livro à editora quando existir.

```
SELECT l.titulo, e.nome AS editora
```



```
FROM livro l
LEFT JOIN editora e ON l.editora_codigo = e.codigo;
```

```
-- 54. Descrição dos assuntos e títulos dos livros. Quando não houver livros, mostrar â\200\230Sem publicaçãoâ\200\231.
```

```
SELECT a.descricao AS assunto,
       COALESCE(l.titulo, 'Sem publicação') AS titulo
FROM assunto a
LEFT JOIN livro l ON l.assunto_codigo = a.codigo
ORDER BY a.descricao;
```

```
-- 55. Nomes dos autores e os livros de sua autoria, ordenado pelo nome do autor.
```

```
SELECT au.nome AS autor, l.titulo
FROM autor au
JOIN autor_livro al ON al.autor_codigo = au.codigo
JOIN livro l ON l.codigo = al.livro_codigo
ORDER BY au.nome, l.titulo;
```

```
-- 56. Editoras que publicaram livros do autor â\200\230Machado de Assisâ\200\231.
```

```
SELECT DISTINCT e.nome AS editora
FROM autor a
JOIN autor_livro al ON al.autor_codigo = a.codigo
JOIN livro l ON l.codigo = al.livro_codigo
JOIN editora e ON e.codigo = l.editora_codigo
WHERE a.nome = 'Machado de Assis';
```

```
-- 57. Quantidade de livros lançados por autores cujo nome contém â\200\230Luisâ\200\231.
```

```
SELECT COUNT(DISTINCT l.codigo) AS quantidade
FROM autor a
JOIN autor_livro al ON al.autor_codigo = a.codigo
JOIN livro l ON l.codigo = al.livro_codigo
WHERE a.nome ILIKE '%luis%' AND l.dataLancamento IS NOT NULL;
```

```
-- 58. Preço do livro mais caro da editora â\200\230Books Editoraâ\200\231 sobre banco de dados.
```

```
SELECT MAX(l.preco) AS preco_mais_caro
FROM livro l
JOIN editora e ON e.codigo = l.editora_codigo
JOIN assunto a ON a.codigo = l.assunto_codigo
WHERE e.nome = 'Books Editora' AND a.descricao ILIKE '%banco de dados%';
```

```
-- 59. Editoras que não publicaram nenhum livro.
```

```
SELECT e.nome
FROM editora e
LEFT JOIN livro l ON l.editora_codigo = e.codigo
WHERE l.codigo IS NULL;
```

```
-- 60. Título do livro e nome da editora para livros com preço inferior a R$ 50,00.
```

```
SELECT l.titulo, e.nome AS editora
FROM livro l
JOIN editora e ON e.codigo = l.editora_codigo
WHERE l.preco < 50.00;
```

```
-- 61. Nome e passaporte do autor brasileiro nascido antes de 1º de janeiro de 1950, com seus livros.
```

```
SELECT a.nome, a.passaporte, l.titulo
FROM autor a
JOIN autor_livro al ON al.autor_codigo = a.codigo
JOIN livro l ON l.codigo = al.livro_codigo
JOIN nacionalidade n ON a.nacionalidade_codigo = n.codigo
WHERE n.pais LIKE '%brasil%'
AND a.dataNascimento < '1950-01-01'
ORDER BY a.nome, l.titulo;
```

-- 62. Nome e passaporte do autor e o preço máximo dos livros de sua autoria.

```
SELECT a.nome, a.passaporte, MAX(l.preco) AS preco_maximo
FROM autor a
JOIN autor_livro al ON al.autor_codigo = a.codigo
JOIN livro l ON l.codigo = al.livro_codigo
GROUP BY a.nome, a.passaporte
ORDER BY a.nome;
```

-- 63. Nome do autor e nome da editora que já lançaram pelo menos 2 livros.

```
SELECT a.nome AS autor, e.nome AS editora
FROM autor a
JOIN autor_livro al ON al.autor_codigo = a.codigo
JOIN livro l ON l.codigo = al.livro_codigo
JOIN editora e ON e.codigo = l.editora_codigo
GROUP BY a.nome, e.nome
HAVING COUNT(DISTINCT l.codigo) >= 2
ORDER BY a.nome, e.nome;
```

-- 64. Descrição do assunto referenciado em pelo menos 10 livros.

```
SELECT a.descricao
FROM assunto a
JOIN livro l ON l.assunto_codigo = a.codigo
GROUP BY a.descricao
HAVING COUNT(l.codigo) >= 10;
```

-- 65. Nomes das editoras que possuem livros lançados.

```
SELECT DISTINCT e.nome
FROM editora e
JOIN livro l ON l.editora_codigo = e.codigo
WHERE l.dataLancamento IS NOT NULL;
```

-- 66. Assuntos que não tiveram livros lançados.

```
SELECT a.descricao
FROM assunto a
LEFT JOIN livro l ON l.assunto_codigo = a.codigo AND l.dataLancamento IS NOT NULL
WHERE l.codigo IS NULL;
```

-- 67. Descrição dos assuntos e quantidade de livros lançados de cada um.

```
SELECT a.descricao, COUNT(l.codigo) AS quantidade
FROM assunto a
LEFT JOIN livro l ON l.assunto_codigo = a.codigo AND l.dataLancamento IS NOT NULL
GROUP BY a.descricao
ORDER BY a.descricao;
```

-- 68. Nome das editoras e o preço médio dos livros de cada uma.

```
SELECT e.nome AS editora, ROUND(AVG(l.preco), 2) AS preco_medio
FROM editora e
JOIN livro l ON l.editora_codigo = e.codigo
GROUP BY e.nome
ORDER BY e.nome;
```

-- 69. Nome das editoras e os livros das editoras que lançaram ao menos 2 livros.

```
SELECT e.nome AS editora, l.titulo
FROM editora e
JOIN livro l ON l.editora_codigo = e.codigo
WHERE e.codigo IN (
```

```
SELECT editora_codigo
FROM livro
WHERE dataLancamento IS NOT NULL
GROUP BY editora_codigo
HAVING COUNT(*) >= 2
)
ORDER BY e.nome, l.titulo;

-- 70. Títulos dos livros dos assuntos com preço médio superior a R$ 40,00, com os respectivos assuntos.
SELECT l.titulo, a.descricao AS assunto
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
WHERE a.codigo IN (
    SELECT assunto_codigo
    FROM livro
    GROUP BY assunto_codigo
    HAVING AVG(preco) > 40
)
ORDER BY a.descricao, l.titulo;

-- 70. Títulos dos livros dos assuntos cujo preço médio é superior a R$ 40,00, com os respectivos assuntos.
SELECT l.titulo, a.descricao AS assunto
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
WHERE l.assunto_codigo IN (
    SELECT assunto_codigo
    FROM livro
    GROUP BY assunto_codigo
    HAVING AVG(preco) > 40
)
ORDER BY a.descricao, l.titulo;

-- 71. Títulos dos livros cujo assunto é "Banco de Dados" ou lançados por editoras que contenham "Books".
SELECT l.titulo
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
JOIN editora e ON l.editora_codigo = e.codigo
WHERE a.descricao = 'Banco de Dados'
    OR e.nome LIKE '%Books%';

-- 72. Títulos dos livros cujo assunto é "Banco de Dados" e lançados por editoras que contenham "Books".
SELECT l.titulo
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
JOIN editora e ON l.editora_codigo = e.codigo
WHERE a.descricao = 'Banco de Dados'
    AND e.nome LIKE '%Books%';

-- 73. Títulos dos livros cujo assunto é "Banco de Dados" e que não foram lançados por editoras com "Books".
SELECT l.titulo
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
JOIN editora e ON l.editora_codigo = e.codigo
WHERE a.descricao = 'Banco de Dados'
    AND e.nome NOT LIKE '%Books%';

-- **74. Títulos dos livros que não foram lançados por editoras com "Books" cujo assunto é "Banco de Dados".
```

```
SELECT l.titulo
FROM livro l
JOIN assunto a ON l.assunto_codigo = a.codigo
JOIN editora e ON l.editora_codigo = e.codigo
WHERE a.descricao = 'Banco de Dados'
      AND e.nome NOT LIKE '%Books%';

-- 75. Excluir editoras que não publicaram livros.
DELETE FROM editora
WHERE codigo NOT IN (
      SELECT DISTINCT editora_codigo FROM livro
);
```