

```

-- 1. Selecione título de filmes que comecem com 'batman'.
SELECT titulo FROM filme WHERE titulo LIKE 'batman%';

-- 2. Exibir o título e ano de filmes lançados entre 2010 e 2015.
-- Renomeie a coluna ano para 'lançamento'.
-- Coloque os resultados em ordem ascendente de ano.
SELECT titulo, ano AS lancamento FROM filme WHERE ano BETWEEN
    2010 AND 2015 ORDER BY ano ASC;

-- 3. Selecionar os anos de lançamento de filmes (sem repeti-
    los).
-- Retornar por ordem ascendente de ano.
SELECT DISTINCT ano FROM filme ORDER BY ano ASC;

-- 4. Selecionar a quantidade de anos distintos em que foram
    lançados filmes.
SELECT COUNT(DISTINCT ano) FROM filme;

-- 5. Selecionar o número de filmes lançados por ano.
-- Ordene pela contagem.
SELECT ano, COUNT(*) AS contagem FROM filme GROUP BY ano ORDER BY
    contagem ASC;

-- 6. Exibir nome de diretores e o título de seus filmes.
SELECT diretor.nome, filme.titulo FROM diretor, filme WHERE
    diretor.idDiretor = filme.idDiretor;

-- 7. Selecionar título e ano de filmes dirigidos por Steven
    Spielberg.
-- Ordene pelo ano.
SELECT filme.titulo, filme.ano FROM diretor, filme WHERE
    diretor.idDiretor = filme.idDiretor AND diretor.nome =
    'Steven Spielberg' ORDER BY filme.ano;

-- 8. Retornar nomes de diretores juntamente com o número de
    filmes que cada um dirigiu.
SELECT diretor.nome, COUNT(*) FROM diretor, filme WHERE
    diretor.idDiretor = filme.idDiretor GROUP BY
    filme.idDiretor;

-- 9. Retornar nome de diretores juntamente com o número de
    filmes que cada um dirigiu.
-- Mostrar o diretor apenas caso o número de filmes dirigido seja
    maior do que 2.

```

```
SELECT diretor.nome, COUNT(*) FROM diretor, filme WHERE
    diretor.idDiretor = filme.idDiretor GROUP BY
    filme.idDiretor HAVING COUNT(*) > 2;
```

-- 10. Para cada diretor, retornar o ano em que teve o primeiro filme lançado.

```
SELECT diretor.nome, MIN(filme.ano) FROM diretor, filme WHERE
    diretor.idDiretor = filme.idDiretor GROUP BY
    filme.idDiretor;
```

-- Os próximos exercícios não serão avaliados, mas podem ser entregues junto com os anteriores

-- 11. Selecione a média de idade dos artistas.

```
SELECT AVG(IF(falec IS NOT NULL, TIMESTAMPDIFF(YEAR, nasc,
    CURDATE()), TIMESTAMPDIFF(YEAR, nasc, falec))) idade FROM
    ator;
```

-- 12. Para cada filme, exibir o título, o ano e o lucro, sendo que o lucro é a diferença entre a bilheteria e o custo.

-- Mostrar o resultado na ordem dos milhões.

-- Ordene do maior para o menor valor.

```
SELECT titulo, ano, (bilheteria - custo) / 1000000 lucro FROM
    filme ORDER BY lucro;
```

-- 13. Para cada diretor, mostrar a diferença entre a sua maior e menor bilheteria.

-- Mostrar a resposta na ordem dos milhões.

```
SELECT diretor.nome, (MAX(filme.bilheteria) -
    MIN(filme.bilheteria)) / 1000000 diferenca FROM diretor,
    filme WHERE diretor.idDiretor = filme.idDiretor GROUP BY
    filme.idDiretor;
```

-- 14. Exibir o nome dos artistas e seu país de origem.

-- Países do Reino Unido (Irlanda do Norte, Escócia, Inglaterra, País de Gales) devem ser exibidos como Reino Unido.

```
SELECT nome, CASE
    WHEN pais = 'Irlanda do Norte' OR pais = 'Inglaterra' OR pais
        = 'País de Gales' THEN 'Reino Unido'
    ELSE pais
    END paisDeOrigem
FROM ator;
```

-- 15. Exibir o país e a quantidade de artistas de cada país.

```
-- Países do Reino Unido (Irlanda do Norte,Escocia,  
    Inglaterra,Pais de Gales) devem ser agrupados como uma  
    única entrada, referente ao Reino Unido.
```

```
SELECT CASE
```

```
    WHEN pais = 'Irlanda do Norte' OR pais = 'Inglaterra' OR pais  
        = 'Pais de Gales' THEN 'Reino Unido'
```

```
    ELSE pais
```

```
    END paisDeOrigem,
```

```
    COUNT(*)
```

```
FROM ator GROUP BY paisDeOrigem;
```

```
-- 16. Exiba títulos de filmes cuja primeira letra seja 'A'.
```

```
SELECT titulo FROM filme WHERE titulo LIKE 'A%';
```