

Trainee Data Science - Lapisco

Módulo 1 - Atividade 1



SQL

Para resolver os exercícios, utilize a descrição do modelo relacional abaixo e os comandos SQL aprendidos. Se houver a necessidade de utilizar mais de uma tabela em uma consulta, utilize o INNER JOIN, em vez do produto cartesiano.

Questão 1

Insira 3 registros para cada tabela criada.

Solução

```
1 INSERT INTO tbAtor
2 VALUES
3 (1,"Matheus Rocha",20,"Fortaleza",2000.0,"M"),
4 (2,"Roberto",40,"Fortaleza",15000.0,"M"),
5 (3,"Iagson",30,"Fortaleza",10000.0,"M");
6
7 INSERT INTO tbNovela
8 VALUES
9 (1,"Avenida Brasil",'2002-08-16',"2100-08-16",8),
10 (2,"Chocolate Com Pimenta",'2002-08-16',"2100-08-16",10),
11 (3,"O Rei do Gado",'2002-08-16',"2100-08-16",7);
12
13
14 INSERT INTO tbCapitulo
15 VALUES
16 (1,"Capitulo 1","2010-08-16",1),
17 (2,"Capitulo 1","2010-08-16",2),
18 (3,"Capitulo 1","2010-08-16",3);
19
20 INSERT INTO tbPersonagem
21 VALUES
22 (1,"Personagem A",15,"Rico",1),
23 (2,"Personagem B",20,"Rico Demais",2),
24 (3,"Personagem C",30,"Pobre",3);
25
26 INSERT INTO tbNovelaPersonagem
27 VALUES
28 (1,1),
29 (2,2),
30 (3,3);
```

Questão 2

Encontre todas as novelas que tenham o valor do horário de exibição vazio.

Solução

```
1 SELECT *
2 FROM tbNovela t
3 WHERE t.horario_exibicao IS NULL;
```

Questão 3

Selecione o nome de todos os atores que morem em cidades que comecem com a letra “M”.

Solução

```
1 SELECT *
2 FROM tbAtor t
3 WHERE t.cidade_ator LIKE "M%";
```

Questão 4

Selecione todos os campos da tabela tbPersonagem ordenados por nome em ordem crescente.

Solução

```
1 SELECT *
2 FROM tbPersonagem t
3 ORDER BY t.nome_personagem ASC;
```

Questão 5

Selecione quantos capítulos existem por novela, retorne o nome da novela e a quantidade de capítulos para a novela.

Solução

```
1 SELECT n.nome_novela, COUNT(*) as qtd_capitulos
2 FROM tbNovela n
3 INNER JOIN tbCapitulo c
4 ON n.codigo_novela = c.codigo_novela
5 GROUP BY c.codigo_novela;
```

Questão 6

Encontre o nome de todas as novelas que tem mais de 40 capítulos.

Solução

```
1 SELECT n.nome_novela
2 FROM tbNovela n
3 INNER JOIN tbCapitulo c
4 ON n.codigo_novela = c.codigo_novela
5 GROUP BY c.codigo_novela
6 HAVING COUNT(*) > 40;
```

Pandas

Nas questões seguintes, utilizar as seguintes bases de dados: titanic, iris.

Questão 1

Verifique a presença de outliers (e.g., valor maior que 3 vezes o desvio padrão da base) na base de dados iris ou outra base de dados. Caso não exista, insira uma ou mais linhas e selecione elas utilizando a regra mencionada.

Solução

```
1 INSERT INTO tbAtor
2 VALUES
3 (1,"Matheus Rocha",20,"Fortaleza",2000.0,"M"),
4 (2,"Roberto",40,"Fortaleza",15000.0,"M"),
5 (3,"Iagson",30,"Fortaleza",10000.0,"M");
6
7 INSERT INTO tbNovela
8 VALUES
9 (1,"Avenida Brasil",'2002-08-16',"2100-08-16",8),
10 (2,"Chocolate Com Pimenta",'2002-08-16',"2100-08-16",10),
11 (3,"O Rei do Gado",'2002-08-16',"2100-08-16",7);
12
13
14 INSERT INTO tbCapitulo
15 VALUES
16 (1,"Capitulo 1","2010-08-16",1),
17 (2,"Capitulo 1","2010-08-16",2),
18 (3,"Capitulo 1","2010-08-16",3);
19
20 INSERT INTO tbPersonagem
21 VALUES
22 (1,"Personagem A",15,"Rico",1),
23 (2,"Personagem B",20,"Rico Demais",2),
24 (3,"Personagem C",30,"Pobre",3);
25
26 INSERT INTO tbNovelaPersonagem
27 VALUES
28 (1,1),
29 (2,2),
30 (3,3);
```
