



# SQL (QUESTÕES)

Prof. Ramon Souza

(FGV - 2018 - Banestes - Analista em Tecnologia da Informação - Desenvolvimento de Sistemas)

Considere um banco de dados com duas tabelas, R e S, cujas instâncias são exibidas a seguir.

Na execução do comando SQL

```
select * from R left join S on a=b
```

UNION

```
select * from R right join S on b=a
```

o número de células contendo o valor nulo no resultado é:

- a) 0;
- b) 3;
- c) 6;
- d) 9;
- e) 12.

R	
a	b
1	2
2	3
4	5

S	
c	d
3	2
4	2
6	1

O left join gerará o seguinte resultado:

a	b	c	d
1	2	null	null
2	3	null	null
4	5	null	null

O right join gerará o seguinte resultado:

a	b	c	d
null	null	3	2
null	null	4	2
null	null	6	1

O UNION irá juntar os resultados, logo teremos:

a	b	c	d
1	2	null	null
2	3	null	null
4	5	null	null
null	null	3	2
null	null	4	2
null	null	6	1

**(FGV - 2017 - IBGE - Analista Censitário - Geoprocessamento)** A linguagem mais comum para elaboração de consultas em bancos de dados é a SQL. Ao elaborar uma consulta nessa linguagem, emprega-se a cláusula WHERE quando se deseja:

- a) especificar a tabela onde será realizada a consulta;
- b) especificar o diretório onde os dados estão armazenados;
- c) especificar o endereço IP onde os dados estão armazenados;
- d) especificar condições a que as instâncias selecionadas devem atender;
- e) extrair a geometria do objeto selecionado na consulta.

**(FGV - 2017 - SEPOG - RO - Analista em Tecnologia da Informação e Comunicação)** A consulta SQL a seguir retorna uma série de nomes da tabela usuarios:

```
select nome from usuários
```

Para obter a relação de nomes em ordem alfabética reversa você deve acrescentar ao final da consulta

- a) sort nome.
- b) sort reverse nome.
- c) order by nome desc.
- d) order by nome.
- e) order desc by nome.

(FGV - 2017 - SEPOG - RO - Analista em Tecnologia da Informação e Comunicação) Observe as tabelas a seguir:

Assinale a opção que indica o número de linhas retornadas pela consulta SQL a seguir.

```
SELECT DISTINCT nome FROM animais a, tipos t  
WHERE a.familia = t.família
```

- a) 0
- b) 1
- c) 3
- d) 4
- e) 5

Tabela: animais

familia	nome
mamifero	cachorro
mamifero	leao
peixe	linguado
passaro	aquia
reptil	cobra
passaro	pelicano
peixe	linguado

Tabela: tipos

familia	tipo_sangue
mamifero	quente
peixe	frio
passaro	quente

**(FGV - 2017 - MPE-BA - Analista Técnico - Tecnologia)** Atenção

Tabelas R e S referentes a um banco de dados relacional.

Considerando as tabelas R e S apresentadas anteriormente, o comando SQL

`select a FROM R UNION ALL select d FROM S`

produz um resultado que contém, além dos títulos:

- a) 1 linha;
- b) 3 linhas;
- c) 4 linhas;
- d) 5 linhas;
- e) 6 linhas.

R	
a	b
1	2
2	3
4	5

S	
c	d
3	2
4	2
6	1

**(FGV - 2017 - MPE-BA - Analista Técnico - Tecnologia)** Atenção

Tabelas R e S referentes a um banco de dados relacional.

Considere as tabelas R e S apresentadas anteriormente e o comando SQL a seguir.

```
update R set a = NULL
```

```
where b >= (select max(d) from S)
```

Após execução desse comando, os valores na coluna a da tabela R seriam, de cima para baixo:

a) NULL, NULL, NULL

b) 1, 2, 4

c) 1, 2, NULL

d) 1, NULL, NULL

e) NULL, 2, 4

R	
a	b
1	2
2	3
4	5

S	
c	d
3	2
4	2
6	1

**(FGV - 2017 - MPE-BA - Analista Técnico - Tecnologia)** Atenção

Tabelas R e S referentes a um banco de dados relacional.

Considerando as tabelas R e S apresentadas anteriormente, o comando SQL

```
select count(*)
```

```
from R, (select * from S where d=2) s
```

resulta no número:

a) 0

b) 1

c) 3

d) 6

e) 9

R	
a	b
1	2
2	3
4	5

S	
c	d
3	2
4	2
6	1

(FGV - 2017 - IBGE - Analista Censitário - Análise de Sistemas - Suporte à Produção) Suponha uma tabela criada com os seguintes comandos:

```
CREATE TABLE Tabela1 (`nome` varchar(8), `ano` int, `qtd` int);
```

```
INSERT INTO Tabela1 ( `nome`, `ano`, `qtd` )  
VALUES  
    ('fusca', 1976, '11'),  
    ('chevette', 1979, '2'),  
    ('corcel', 1981, '4'),  
    ('kombi', 1975, '3'),  
    ('brasilia', 1979, '8'),  
    ('mustang', 1975, '1');
```

Após a tabela ter sido criada, o comando SQL que retornará dois resultados é o:

- a) SELECT MAX(qtd) FROM Tabela1 WHERE ano=1975;
- b) SELECT sum(qtd) FROM Tabela1 WHERE ano = 1975 GROUP BY qtd HAVING qtd < 10;
- c) SELECT \* FROM Tabela1 WHERE qtd BETWEEN 1 AND 3;
- d) SELECT nome FROM Tabela1 WHERE ano not in (1979) and qtd > (select min(qtd) from Tabela1 where ano = 1975);
- e) SELECT sum(qtd) FROM Tabela1 WHERE ano in (1975,1977) GROUP BY qtd HAVING sum(qtd) >= 3;

(FGV - 2016 - Prefeitura de Paulínia - SP - Analista de Sistemas) Analise o comando SQL mostrado a seguir.

```
SELECT p.*,v.*  
FROM Produto p FULL OUTER JOIN Venda v  
ON p.ProdutoID = v.ProdutoID
```

Sabendo-se que na tabela Venda há registros que não possuem relacionamento com registros de Produto, e vice-versa, assinale a correta interpretação desse comando em relação aos registros de Produto e Venda que aparecem no resultado.

- a) Todos de Produto, e somente os de Venda que possuem relacionamento com registros de Produto.
- b) Todos de Venda, e somente os de Produto que possuem relacionamento com registros de Venda.
- c) Somente os de Venda que não possuem relacionamento com registros de Produto de Venda, e somente registros de Produto que não possuem relacionamento com registros de Venda.
- d) Somente os de Venda que possuem relacionamento com registros de Produto, e somente os de Produto que possuem relacionamento com registros de Venda.
- e) Todos de Produto e todos de Venda.

(FGV - 2016 - Prefeitura de Paulínia - SP - Analista de Sistemas) Analise o comando SQL a seguir.

```
delete from T where  
exists (select * from T t2 where T.a <- t2.a)
```

Sabendo-se que há quatro registros na instância da tabela T, o número de registros, após a execução desse comando, será

- a) zero.
- b) um.
- c) dois.
- d) três.
- e) quatro.



# SQL (QUESTÕES)

Prof. Ramon Souza