

Q1 – CESGRANRIO - BNDES – Desenvolvimento – 2012

Qual consulta permite exibir a placa e o modelo dos veículos que **NÃO** foram alugados no mês de junho de 2012?

```
CREATE TABLE VEICULO (  
  PLACA CHAR(7) NOT NULL,  
  MODELO VARCHAR2(50) NOT NULL,  
  COD_CAT CHAR(2) NOT NULL,  
  CONSTRAINT VEICULO_PK PRIMARY KEY (PLACA),  
  CONSTRAINT VEICULO_FK FOREIGN KEY (COD_CAT) REFERENCES  
    CATEGORIA (COD_CAT))
```

```
CREATE TABLE CATEGORIA (  
  COD_CAT CHAR(2) NOT NULL,  
  DESCR VARCHAR2(80) NOT NULL,  
  VAL_DIARIA NUMBER(7,2),  
  CONSTRAINT CATEGORIA_PK PRIMARY KEY (COD_CAT))
```

```
CREATE TABLE ALUGUEL (  
  PLACA CHAR(7) NOT NULL,  
  DATA_DEV NUMBER(6),  
  DATA_ALG NUMBER(6) NOT NULL,  
  CONSTRAINT ALUGUEL_PK PRIMARY KEY (PLACA, DATA_ALG),  
  CONSTRAINT ALUGUEL_FK FOREIGN KEY (PLACA) REFERENCES  
    VEICULO (PLACA))
```

- (A) SELECT PLACA,MODELO
FROM VEICULO V
WHERE (SELECT COUNT(*) FROM ALUGUEL
WHERE PLACA=V.PLACA AND DATA_ALG>=120601 AND DATA_ALG<=120630)>0
- (B) SELECT V.PLACA,V.MODELO
FROM VEICULO V
MINUS
SELECT V.PLACA,V.MODELO
FROM VEICULO V,ALUGUEL A
WHERE V.PLACA=A.PLACA AND A.DATA_ALG>=120601 AND A.DATA_ALG<=120630
- (C) SELECT PLACA,MODELO
FROM VEICULO
WHERE PLACA IN
(SELECT PLACA FROM ALUGUEL
WHERE DATA_ALG>=120601 AND DATA_ALG<=120630)
- (D) SELECT V.PLACA,V.MODELO
FROM VEICULO V,ALUGUEL A
WHERE V.PLACA=A.PLACA AND (A.DATA_ALG<120601 OR A.DATA_ALG>120630)

```
(E) SELECT V.PLACA,V.MODELO
FROM VEICULO V,ALUGUEL A
WHERE V.PLACA=A.PLACA AND A.DATA_ALG>=120601 AND A.DATA_ALG<=120630
MINUS
SELECT V.PLACA,V.MODELO
FROM VEICULO V
```

Q2 - CESPE – EBC – 2011

- Ao se executar um comando SELECT com cláusula GROUP BY, serão agrupados NULLs, se constarem na coluna de agrupamento.

Q3 - CESPE – EBC - 2011

A respeito de sistema de gerência de banco de dados (SGBD), julgue os itens a seguir.

- O uso do comando SELECT com cláusula GROUP BY e sem qualquer função de agrupamento é semelhante à utilização da cláusula DISTINCT com seleção de apenas uma coluna

Q4 - CESPE – TCERO - 2013

```
CREATE TABLE sede
(
  codigo numeric(7,0),
  nome varchar(50),
  local numeric(12,0),
  apelido varchar(10),
  CONSTRAINT pk_sede PRIMARY KEY (codigo),
  CONSTRAINT uq_sede UNIQUE (apelido)
);

CREATE TABLE salas
(
  id numeric(7,0) NOT NULL,
  local varchar(10),
  descricao varchar(50),
  area numeric(10,2),
  CONSTRAINT pk_salas PRIMARY KEY (id),
  CONSTRAINT fk_sede_sala FOREIGN KEY (local)
REFERENCES sede (apelido)
);
```

Considerando as tabelas acima, criadas em SQL, julgue os itens que se seguem

- A expressão de consulta mostrada abaixo permite visualizar o nome da sede, com a área da sala e sua descrição. select nome, area, descricao from salas, sede where sede.local=salas.local

Q5 - CESPE – TCERO - 2013

```
CREATE TABLE sede
(
    codigo numeric(7,0),
    nome varchar(50),
    local numeric(12,0),
    apelido varchar(10),
    CONSTRAINT pk_sede PRIMARY KEY (codigo),
    CONSTRAINT uq_sede UNIQUE (apelido)
);

CREATE TABLE salas
(
    id numeric(7,0) NOT NULL,
    local varchar(10),
    descricao varchar(50),
    area numeric(10,2),
    CONSTRAINT pk_salas PRIMARY KEY (id),
    CONSTRAINT fk_sede_sala FOREIGN KEY (local)
    REFERENCES sede (apelido)
);
```

- A expressão de consulta mostrada a seguir apresenta como resultado o local da sede, a área da sala e a descrição da(s) sala(s) com a maior área.

```
select c1.local, c1.nome, c2.descricao
from sede as c1,
    ( select local, descricao, area from salas as c1
      where area = (select max(area) from salas as c2 where area>0) ) as c2
where c2.local=c1.apelido;
```

Q6 - FCC – TRT19 – 2011

Uma tabela A contém apenas o código (cod_pac) de todos os pacientes internados em um hospital. Uma tabela B, com a mesma estrutura, contém o código (cod_pac) de todos os pacientes deste hospital que estão internados há mais de 15 dias. Para listar o código de todos os pacientes deste hospital internados por um período menor ou igual a 15 dias, deve-se utilizar a seguinte instrução SQL:

- a) SELECT cod_pac FROM A DIFF SELECT cod_pac FROM B
- b) SELECT cod_pac FROM (A UNION B)
- c) SELECT cod_pac FROM A where cod_pac not in (SELECT cod_pac FROM B)
- d) SELECT cod_pac FROM A where cod_pac in (A MINUS B)
- e) SELECT cod_pac FROM (A DIFF B)

Q7 - FCC – MPE-AP – 2012

```
CREATE TABLE Livro (ISBN INT, Nome VARCHAR(40), Autor INT, Editora INT);
```

```
CREATE TABLE Autor (Codigo INT, NOME VARCHAR(40));
```

```
CREATE TABLE Editora (Codigo INT, Nome VARCHAR(40));
```

```
INSERT INTO Livro VALUES (12345, "Programas em C", 1, 1);
```

```
INSERT INTO Livro VALUES (67890, "Métodos Ágeis", 1, 2);
```

```
INSERT INTO Autor VALUES (1, "Manoel da Silva");
```

```
INSERT INTO Editora VALUES (1, "Editora Livros");
```

Note que os exemplos abaixo consideram que as linhas apresentadas acima já foram executadas. Para receber como resultado apenas a string Programas em C, é necessário executar o comando

- a) `SELECT Nome FROM Editora WHERE Editora.Codigo = Livro.Editora AND Autor.Codigo = Livro.Autor;`
- b) `SELECT b.Nome FROM Autor a, Livro c, Editora c WHERE a.Autor = b.Codigo AND a.Editora = c.Codigo;`
- c) `SELECT * FROM Livro a WHERE (SELECT Codigo FROM Autor WHERE Codigo = a.Autor) AND (SELECT Codigo FROM Editora WHERE Codigo = a.Editora);`
- d) `SELECT Nome FROM Livro WHERE Autor IN (SELECT Codigo FROM Autor) AND Editora IN (SELECT Codigo FROM Editora);`
- e) `SELECT * FROM Livro WHERE Livro.Autor = (SELECT Codigo FROM Autor) AND Livro.Editora = (SELECT Codigo FROM Editora);`

Q8 - FCC – MPE-AP – 2012

```
CREATE TABLE Livro (ISBN INT, Nome VARCHAR(40), Autor INT, Editora INT);
```

```
CREATE TABLE Autor (Codigo INT, NOME VARCHAR(40));
```

```
CREATE TABLE Editora (Codigo INT, Nome VARCHAR(40));
```

```
INSERT INTO Livro VALUES (12345, "Programas em C", 1, 1);
```

```
INSERT INTO Livro VALUES (67890, "Métodos Ágeis", 1, 2);
```

```
INSERT INTO Autor VALUES (1, "Manoel da Silva");
```

```
INSERT INTO Editora VALUES (1, "Editora Livros");
```

Para receber como resultado a quantidade de autores que possuem livros publicados por uma editora cadastrada é necessário executar o comando

- a) `SELECT COUNT(a.Codigo) FROM Autor a, Livro b, Editora c WHERE a.Codigo = b.Autor AND b.Editora = c.Codigo;`
- b) `SELECT * FROM Livro WHERE Livro.Autor = (SELECT Autor FROM Codigo) AND Editora.Codigo = (SELECT Codigo FROM Editora);`
- c) `SELECT COUNT(Nome) FROM Autor WHERE Autor IN (SELECT Codigo FROM Livro) AND Editora IN (SELECT Codigo FROM Editora);`
- d) `SELECT * FROM Autor WHERE Codigo IN Livro.Autor AND Livro.Editora IN Editora.Codigo;`
- e) `SELECT * FROM Autor, Livro, Editora WHERE COUNT(SELECT * FROM Livro) > 1 AND Editora.Codigo = 1;`

Q9 - FGV – TJ-BA - 2015

Analise as instâncias das tabelas R1 e R2 e o comando SQL, mostrados a seguir.

| R1 | | R2 | | |
|----|--|----|---|---|
| x | | a | b | |
| 1 | | 2 | 0 | <pre>update R2 set b=(select count(x) from R1 where x > a)</pre> |
| 3 | | 5 | 0 | |
| 4 | | 7 | 0 | |
| 6 | | 9 | 0 | |
| 7 | | 10 | 0 | |
| 8 | | | | |

Após a execução do comando, o conteúdo da coluna "b" da tabela R2 passa a ser, de cima para baixo:

- a) 5, 5, 5, 5, 5;
- b) NULL, NULL, NULL, NULL, NULL;
- c) 5, 3, 1, 0, 0;
- d) 0, 0, 0, 0, 0;
- e) 5, 3, 1, NULL, NULL.

Q10 - CESGRANRIO - EPE – Analista de Gestão Corporativa – 2012

A consulta "listar todos os cargos em ordem alfabética e a respectiva média salarial de cada um deles" é representada, em SQL ANSI, por

- (A) `select cargo, sum(salario)/count(*) from S order by cargo;`
- (B) `select cargo, sum(salario)/count(*) from S sort by cargo;`

- (C) select cargo from S having avg(cargo) order by cargo;
- (D) select cargo, avg(salario) from S group by cargo sort by cargo;
- (E) select cargo, avg(salario) from S group by cargo order by cargo;

Gabarito

Q1 – B

Q2 – C

Q3 – C

Q4 – E

Q5 – C

Q6 – C

Q7 – D

Q8 – A

Q9 – C

Q10 - E