

Q1 – Liquigás – Banco de Dados - 2012

Uma base de dados relativa a vendas efetuadas por uma empresa apresenta as seguintes tabelas:

VENDEDOR(IdVendedor, NomeVendedor)

CLIENTE(NomeCliente, CPFCliente, TelefoneCliente)

PRODUTO(IdItem, NomeItem, ValorItem)

VENDA(IdVenda, DataVenda, IdVendedor, CPFCliente)

ITEMVENDA(IdVenda, SequencialItemVenda, IdItem, QuantidadeItem)

Que comando SQL pode ser dado para, a partir das tabelas acima, ter como resultado duas colunas, a primeira contendo o IdVenda e a segunda o valor total da venda?

- (A) `SELECT VENDA.IdVenda, SUM(QuantidadeItem*ValorItem) as ValorTotal FROM VENDA, ITEMVENDA, PRODUTO WHERE VENDA.IdVenda=ITEMVENDA.IdVenda And ITEMVENDA.IdItem=PRODUTO.IdItem GROUP BY VENDA.IdVenda;`
- (B) `SELECT VENDA.IdVenda, SUM(QuantidadeItem*ValorItem) AS ValorTotal FROM VENDA, ITEMVENDA, PRODUTO WHERE VENDA.IdVenda=ITEMVENDA.IdVenda And ITEMVENDA.IdItem=PRODUTO.IdItem;`
- (C) `SELECT VENDA.IdVenda,QuantidadeItem*ValorItem AS ValorTotalFROM VENDA, ITEMVENDA, PRODUTO WHERE VENDA.IdVenda=ITEMVENDA.IdVenda And ITEMVENDA.IdItem=PRODUTO.IdItem ;`
- (D) `SELECT VENDA.IdVenda, SUM(QuantidadeItem*ValorItem) as ValorTotal FROM VENDA, ITEMVENDA, PRODUTO WHERE VENDA.IdVenda=ITEMVENDA.IdVenda And ITEMVENDA.IdItem=PRODUTO.IdItem ORDER BY VENDA.IdVenda;`
- (E) `SELECT VENDA.IdVenda, QuantidadeItem*ValorItem as ValorTotal FROM VENDA, ITEMVENDA, PRODUTO WHERE VENDA.IdVenda=ITEMVENDA.IdVenda And ITEMVENDA.IdItem=PRODUTO.IdItem GROUP BY VENDA.IdVenda;`

Q2 - CESGRANRIO - Liquigás – Análise de Sistemas – 2012

A consulta que permite exibir o nome de uma empresa cujo somatório das negociações (VALOR * QTD) de suas ações preferenciais (TIPO='PN') entre os dias 23-01-2012 (inclusive) e 25-01-2012 (inclusive) tenha sido superior a R\$ 1.000.000,00 é

- (A) SELECT E.NOME FROM EMPRESA E, ACAO A, NEGOCIACAO N
 WHERE E.CODIGO=A.COD_EMPRESA AND A.TIPO='PN' AND A.CODIGO=N.COD_ACAO
 AND N.DATA BETWEEN '23-01-2012' AND '25-01-2012' AND
 SUM(N.QTD*N.VALOR)>1000000
-
- (B) SELECT E.NOME FROM EMPRESA E, ACAO A, NEGOCIACAO N
 WHERE E.CODIGO=A.COD_EMPRESA AND A.TIPO='PN' AND A.CODIGO=N.COD_ACAO
 AND N.DATA BETWEEN '23-01-2012' AND '25-01-2012'
 GROUP BY E.CODIGO
 HAVING SUM(N.QTD*N.VALOR)>1000000
- (C) SELECT E.NOME FROM EMPRESA E, ACAO A, NEGOCIACAO N
 WHERE E.CODIGO=A.COD_EMPRESA AND A.TIPO='PN' AND A.CODIGO=N.COD_ACAO
 AND N.DATA >= '23-01-2012' AND <= '25-01-2012'
 GROUP BY E.NOME
 HAVING SUM(N.QTD*N.VALOR)>1000000
- (D) SELECT E.NOME FROM EMPRESA E, ACAO A, NEGOCIACAO N
 WHERE E.CODIGO=A.COD_EMPRESA AND A.TIPO='PN' AND A.CODIGO=N.COD_ACAO
 AND N.DATA BETWEEN '23-01-2012' AND '25-01-2012'
 ORDER BY E.NOME
 HAVING SUM(N.QTD*N.VALOR)>1000000
- (E) SELECT E.NOME FROM EMPRESA E, ACAO A, NEGOCIACAO N
 WHERE E.CODIGO=A.COD_EMPRESA AND A.TIPO='PN' AND A.CODIGO=N.COD_ACAO
 AND N.DATA > '22-01-2012' AND N.DATA < '26-01-2012'
 GROUP BY E.NOME
 HAVING SUM(N.QTD*N.VALOR)>1000000

Q3 - CESGRANRIO - LIQUIGAS – Analista de Sistemas – 2013

Seja x a média das marcas alcançadas por todos os saltos registrados na tabela PROVA. Qual consulta permite exibir o código dos atletas que realizaram mais de dois saltos cujas marcas são menores do que x ?

- (A) SELECT P.COD_ATL
 FROM PROVA P
 WHERE COUNT(*) > 2
 GROUP BY P.COD_ATL
 HAVING P.MARCA < AVG(P.MARCA)
- (B) SELECT P.COD_ATL
 FROM PROVA P
 WHERE P.MARCA < (SELECT AVG(MARCA) FROM PROVA)
 AND COUNT(*) > 2

- GROUP BY COD_ATL
- (C) SELECT P.COD_ATL
FROM PROVA P
WHERE P.MARCA < (SELECT AVG(MARCA) FROM PROVA)
GROUP BY COD_ATL
HAVING COUNT(*) > 2
- (D) SELECT P.COD_ATL FROM PROVA P MINUS
SELECT P.COD_ATL FROM PROVA P
WHERE P.MARCA < (SELECT AVG(MARCA) FROM PROVA)
GROUP BY COD_ATL
HAVING COUNT(*) > 2
- (E) SELECT P.COD_ATL
FROM PROVA P
WHERE COUNT(*) > 2 AND P.MARCA < AVG(P.MARCA)
GROUP BY P.COD_ATL

Q4 - CESGRANRIO - LIQUIGAS – Analista de Sistemas – 2013

Seja μ a média das marcas de todos os saltos realizados em uma prova. Qual consulta permite exibir o código da prova e a média dos saltos (μ) relativos às provas em que $7,55 \leq \mu \leq 7,60$?

- (A) SELECT COD_PROVA, AVG(MARCA)
FROM PROVA
GROUP BY COD_PROVA
HAVING AVG(MARCA) BETWEEN 7.55 AND 7.60
- (B) SELECT COD_PROVA, AVG(MARCA)
FROM PROVA
GROUP BY COD_PROVA, COD_ATL
HAVING AVG(MARCA) BETWEEN 7.55 AND 7.60
- (C) SELECT COD_PROVA, AVG(MARCA)
FROM PROVA

- GROUP BY COD_PROVA
- HAVING AVG(MARCA)>= 7.55 AND <= 7.60
- (D) SELECT COD_PROVA, AVG(MARCA)
- FROM PROVA
- WHERE AVG(MARCA)>= 7.55
- AND AVG(MARCA)<= 7.60
- (E) SELECT COD_PROVA, AVG(MARCA)
- FROM PROVA
- GROUP BY COD_PROVA, COD_ATL
- HAVING AVG(MARCA)>= 7.55
- AND AVG(MARCA)<= 7.60

Q5 - CESGRANRIO - Transpetro – Infraestrutura 2012

Tabela Lojas

nome_loja	vendas
Barra	1500
Ipanema	250
Copacabana	300
Tijuca	700

Tabela Regioes

nome_regiao	nome_loja
Sul	Ipanema
Sul	Copacabana
Oeste	Barra
Norte	Tijuca

Considerando as duas tabelas apresentadas, a consulta SQL

SELECT SUM(vendas) FROM Lojas WHERE EXISTS (SELECT * FROM Regioes WHERE nome_regiao = 'Sul')

apresentará o seguinte resultado:

- (A) 250
- (B) 350
- (C) 550
- (D) 2250
- (E) 2750

Q6 - CESGRANRIO - Banco da Amazônia – Banco de Dados – 2014

Considere um banco de dados relacional com as duas tabelas a seguir.

Empregado (emp_id, emp_nome, dno, salario)

Departamento (dep_id, dep_nome)

O campo Empregado.dno indica o dep_id do departamento onde o empregado trabalha, e os campos sublinhados são chave primária.

Nesse contexto, analise o seguinte comando SQL:

```
SELECT d.dep_nome, COUNT(*) AS x
FROM Departamento d, Empregado e
WHERE d.dep_id = e.dno AND e.salario > 5000 AND
e.dno IN (SELECT f.dno FROM Empregado f GROUP BY
f.dno HAVING COUNT(*) > 2)
GROUP BY d.dep_nome;
```

O que calcula o comando SQL apresentado acima?

- (A) Quantos empregados ganham mais de R\$ 5.000,00, listados por departamento.
- (B) Quantos empregados existem, listados por departamento, em departamentos com mais de duas pessoas que ganham mais de R\$ 5.000,00.
- (C) Quantos empregados existem, listados por departamento, em departamentos que possuem duas pessoas que ganham mais de R\$ 5.000,00.
- (D) Quantos empregados ganham mais de R\$ 5.000,00, listados por departamento, em departamentos com mais de duas pessoas.
- (E) Quantos departamentos existem com mais de duas pessoas que ganham R\$ 5.000,00.

Q7 - CHESF – Analista de Sistemas – 2012

Em um banco de dados de empregados, a tabela EMP contém os campos CPF, NOME, SALARIO e DEPTNO dos empregados, sendo o último uma chave estrangeira para o departamento.

Que consulta SQL fornece o CPF, NOME, SALARIO e DEPTNO de todos os funcionários que ganham mais do que qualquer empregado do departamento 5?

- (A) SELECT CPF, NOME, MAX(SALARIO), DEPTNO FROM EMP
WHERE SALARIO > MAX(SALARIO) AND DEPTNO = 5;
- (B) SELECT CPF, NOME, MAX(SALARIO), DEPTNO FROM EMP
WHERE SALARIO > (SELECT MAX (SALARIO) FROM EMP WHERE DEPTNO = 5);
- (C) SELECT CPF, NOME, SALARIO, DEPTNO FROM EMP
WHERE SALARIO > (SELECT MAX (SALARIO) FROM EMP WHERE DEPTNO = 5);
- (D) SELECT CPF, NOME, SALARIO, DEPTNO FROM EMP
WHERE SALARIO > (SELECT MAX (SALARIO) FROM EMP) AND DEPTNO = 5;
- (E) SELECT CPF, NOME, SALARIO, DEPTNO FROM EMP
WHERE SALARIO > MAX(SALARIO) AND DEPTNO = 5;

Q8 - CESGRANRIO IBGE – Desenvolvimento – 2013

Considere o seguinte código SQL:

```
CREATE TABLE EMPREG
(ID INTEGER PRIMARY KEY, NOME CHAR(20), SOBRENOME CHAR(60), SALARIO REAL);
INSERT INTO EMPREG VALUES (44, 'William', 'Simpson', 6387.01);
INSERT INTO EMPREG VALUES (11, 'Fulano', 'Brasil', 3045.78);
INSERT INTO EMPREG VALUES (22, 'Beltrano', 'da Silva', 4046.79);
INSERT INTO EMPREG VALUES (33, 'Carlos', 'da Silva', 13040.78);
CREATE TABLE COMISSAO
(ID INTEGER REFERENCES EMPREG(ID), MES INTEGER CHECK (MES BETWEEN 1 AND 12),
VALOR_COMISS REAL, PRIMARY KEY (ID, MES));
INSERT INTO COMISSAO VALUES (22,1,1001.67);
INSERT INTO COMISSAO VALUES (22,6,1001.67);
INSERT INTO COMISSAO VALUES (44,5,2338.67);
INSERT INTO COMISSAO VALUES (11,1,400.67);
INSERT INTO COMISSAO VALUES (33,9,2340.00);
INSERT INTO COMISSAO VALUES (44,12,2940.67);
```

O resultado da consulta será

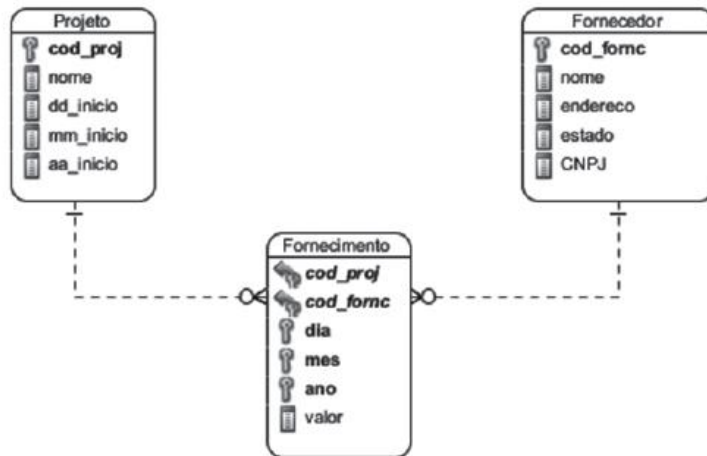
```
SELECT NOME FROM EMPREG WHERE 2340.00 < (SELECT AVG(VALOR_COMISS) FROM
COMISSAO WHERE EMPREG.ID = COMISSAO.ID);
```

será

- (A) William
- (B) Fulano
- (C) da Silva
- (D) Carlos
- (E) 67

Q9 - CESGRANRIO - SUAPE – Analista Infra Pleno – 2012

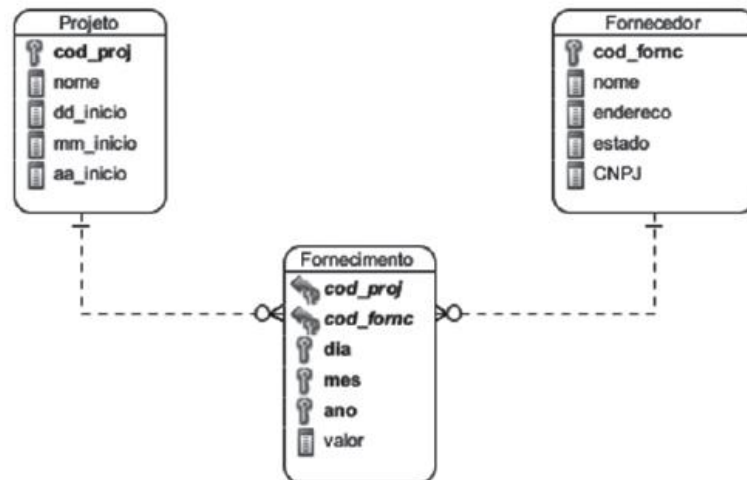
Que consulta permite exibir os nomes dos projetos cujos valores médios dos fornecimentos sejam maiores do que R\$ 5.000,00?



- (A) `SELECT P.NOME`
`FROM PROJETO P, FORNECIMENTO F`
`WHERE P.COD_PROJ=F.COD_PROJ AND AVG(F.VALOR)>5000`
`GROUP BY P.NOME`
- (B) `SELECT P.NOME`
`FROM PROJETO P, FORNECIMENTO F`
`WHERE P.COD_PROJ=F.COD_PROJ AND AVG(F.VALOR)>5000`
- (C) `SELECT P.NOME`
`FROM PROJETO P, FORNECIMENTO F`
`WHERE P.COD_PROJ=F.COD_PROJ`
`GROUP BY P.NOME`
`HAVING AVG(F.VALOR)>5000`
- (D) `SELECT F.COD_PROJ`
`FROM FORNECIMENTO F`
`WHERE AVG(F.VALOR)>5000`
`GROUP BY F.COD_PROJ`
- (E) `SELECT F.COD_PROJ`
`FROM FORNECIMENTO F`
`GROUP BY F.COD_PROJ`
`HAVING AVG(F.VALOR)>5000`

Q10 - CESGRANRIO - SUAPE – Analista Infra Pleno – 2012

Que consulta irá abater 10% dos valores dos fornecimentos relativos a fornecedores que realizaram mais de 2 fornecimentos, independente do projeto para os quais eles forneceram?



- (A) UPDATE FORNECIMENTO F
 SET VALOR=VALOR*0.9
 WHERE COUNT (F.COD_FORNC) >2
- (B) UPDATE FORNECIMENTO
 SET VALOR=VALOR*0.9
 WHERE COUNT (COD_FORNC) >2
 GROUP BY COD_FORNC
- (C) UPDATE FORNECIMENTO F
 VALOR=VALOR*0.9
 WHERE (SELECT COUNT (*) FROM FORNECIMENTO WHERE
 F.COD_FORNC=COD_FORNC) >2
- (D) UPDATE FORNECIMENTO
 SET VALOR=VALOR*0.9
 WHERE COD_FORNC IN (SELECT COD_FORNC FROM FORNECIMENTO WHERE
 COD_FORNC > 2)
- (E) UPDATE FORNECIMENTO F
 SET VALOR=VALOR*0.9
 WHERE (SELECT COUNT (*) FROM FORNECIMENTO WHERE
 F.COD_FORNC=COD_FORNC) >2

Gabarito

Q1 – A

Q2 – E

Q3 – C

Q4 – A

Q5 – E

Q6 – C

Q7 – C

Q8 – A

Q9 – C

Q10 - E