

IMB003 – Arquitetura e Modelagem de Banco de Dados – Prova Bimestral

Instruções: 1) Prova individual; e 2) O valor total da prova é de 8,0 pontos.

Questão 1 (8 pontos)

Considere as seguintes tabelas:

IMO_IMOVEL		
IMO_CEP	IMO_NUMERO	IMO_CONTATO
13243010	12	1244445151
12372190	25	1244231234
12372190	7	1244114321

SEN_SENSOR			
SEN_ID	IMO_CEP	IMO_NUMERO	SEN_TIPO
1	13243010	12	Presença
2	12372190	25	Presença
3	12372190	25	Gás
4	12372190	7	Gás

MED_MEDICAO		
SEN_ID	MED_DT_HR	MED_LEITURA
2	03/04/2016 10:30	1
4	03/04/2016 10:30	20
2	03/04/2016 11:00	0
3	03/04/2016 11:00	30
4	03/04/2016 11:00	15
3	03/04/2016 11:30	45

Onde:

- IMO_CEP, IMO_NUMERO, IMO_CONTATO e SEN_ID são colunas numéricas inteiras (sem casas decimais);
- SEN_ID deve ser gerado automaticamente por uma sequência de nome SEN_SEQ;
- SEN_TIPO é uma coluna de texto variável com, no máximo, 50 caracteres;
- MED_DT_HR é uma coluna de data/hora;
- MED_LEITURA é uma coluna numérica real com, no máximo, 10 dígitos, dos quais 2 são casas decimais.
- SEN_ID, na tabela MED_MEDICAO, é chave estrangeira para a tabela SEN_SENSOR;
- IMO_CEP e IMO_NUMERO, na tabela SEN_SENSOR, são chave estrangeira para a tabela IMO_IMOVEL;
- Nenhuma coluna aceita valores nulos; e
- As tabelas foram criadas pelo usuário IMB003, cuja senha é 123.

Crie comandos SQL DDL para:

- a) Criar uma nova tabela VIS_VISITA para armazenar registros de visitas técnicas. Ela deve referenciar um imóvel e armazenar a data da visita e uma observação (opcional). Não definir a chave estrangeira. Mais de um imóvel podem ser visitados na mesma data. Não ocorrem mais de uma visita a um mesmo imóvel em uma determinada data. A definição da chave primária é obrigatória. (1,0 ponto)
- b) Alterar a tabela VIS_VISITA definindo a chave estrangeira. (0,5 ponto)

Crie comandos SQL DML para:

- c) Inserir dois (2) registros na tabela VIS_VISITA: o primeiro deve referenciar a residência de CEP 12372190 e número 25 na data 10/01/2016 e o segundo deve referenciar a residência de CEP 12372190 e número 7 na mesma data, mas com uma observação "Morador não encontrado". Inserir um registro na tabela SEN_SENSOR com qualquer valor (use sequência). (1,0 ponto)
- d) Alterar o tipo do sensor inserido (para recuperar o id gerado use SEN_SEQ.CURRVAL). (0,5 pontos)
- e) Apagar o sensor inserido (para recuperar o id gerado use SEN_SEQ.CURRVAL). (0,5 pontos)

Para os itens a seguir considere que podem existir milhares de registros além dos apresentados no exemplo.

- f) Retornar o CEP, número e contato dos imóveis que possuem sensores de gás. Não repetir valores. (1,0 ponto)
- g) Retornar o CEP, número e contato dos imóveis que possuem algum sensor sem nenhuma medição. (1,0 ponto)
- h) Retornar o CEP e o número dos imóveis cujo sensor de gás apresentou elevação na medição (o valor de uma leitura é 20% superior ao de uma leitura anterior). (1,0 ponto)

Crie comandos SQL DCL para:

- i) As tabelas apresentadas foram criadas pelo usuário IMB003. Crie um usuário SENSOR que somente possa inserir e alterar registros na tabela MED_MEDICAO. (1,0 pontos)

Crie uma *view* que:

- j) Apresente os CEPs e números dos imóveis acompanhados dos tipos de sensores instalados. Cada linha deve apresentar um trio: CEP, número e tipo de sensor. Se um imóvel possui múltiplos sensores, deve existir uma linha para cada. Esta *view* não deve apresentar registros duplicados. (0,5 pontos)