**EXERCÍCIOS DE BANCO DE DADOS**

1. O conjunto de operações cujo resultado seja uma nova relação e que envolve seleção, projeção, união e produto cartesiano é denominado.
2. mapeamento de cardinalidades.
3. álgebra relacional.
4. generalização.
5. chave primária
6. herança.
7. O modelo relacional estabeleceu-se como o primeiro modelo de dados para aplicações comerciais. Existe uma base teórica indispensável para os bancos de dados relacionais, que possibilita um processamento eficiente das necessidades de informações dos respectivos usuários. A respeito dessa base teórica, assinale a alternativa correta.
8. A álgebra relacional é uma linguagem de consultas procedural e possui operações definidas em termos das respectivas operações fundamentais.
9. As operações binárias da álgebra relacional são union, select e rename.
10. A operação binária project da álgebra relacional permite a seleção de tuplas que satisfaçam determinada relação.
11. A operação primária produto cartesiano da álgebra relacional permite associar informações de dois predicados quaisquer.
12. Uma relação do banco de dados, uma relação constante e uma relação de conjuntos são os três tipos existentes de expressões básicas na álgebra relacional.
13. Acerca de operação de projeção em álgebra relacional, analise as seguintes afirmativas:

I. A projeção em álgebra relacional é aplicada a uma relação simples.

II. A projeção em álgebra relacional remove implicitamente as tuplas duplicadas presentes na relação resultante.

III. A projeção em álgebra relacional é comutativa.

Podemos afirmar corretamente que:

1. Todas as afirmativas estão corretas.
2. Todas as afirmativas estão incorretas.
3. Apenas as afirmativas I e II estão corretas.
4. Apenas as afirmativas II e III estão corretas.
5. No âmbito da álgebra relacional, os símbolos π (Pi) e σ (Sigma) são utilizados, respectivamente, em operações de
6. reunião ou seleção; e intersecção ou projeção.
7. projeção ou particionamento horizontal; e seleção ou particionamento vertical.
8. projeção ou particionamento vertical; e seleção ou particionamento horizontal.
9. seleção ou particionamento horizontal; e projeção ou particionamento vertical.
10. seleção ou particionamento vertical; e projeção ou particionamento horizontal.
11. Analise a seguinte operação em álgebra relacional e assinale a alternativa correta.

nome\_cliente, endereco\_agencia, numero\_conta (*σ* data\_abertura = “01/07/2011” (Cliente |X| Conta |X| Agencia))

É uma projeção da data de abertura da conta por meio da seleção dos dados dos clientes (nome, endereço da agência e número de conta).

1. É uma projeção dos dados dos clientes (nome, endereço da agência e número de conta), por meio da seleção da data de abertura.
2. É uma seleção dos dados dos clientes (nome, endereço da agência e número de conta), por meio da projeção da data de abertura.
3. Existem três bancos de dados envolvidos na operação dados por “cliente", “conta" e “agencia".
4. Na operação da álgebra relacional dada o símbolo “|X|" representa a projeção dos atributos.
5. Observe as seguintes tabelas de um sistema:

**Pessoa** (nomePessoa, endereço)

**Filme** (nomeFilme, estúdio, verba)

**Trabalha** (*nomePessoa*(FK), *nomeFilme*(FK), atividade, salário)

Um administrador de dados deseja fazer o seguinte relatório: Todas as pessoas, os filmes nos quais já trabalharam e o total recebido por cada filme. Se uma pessoa nunca trabalhou num filme, ela deve aparecer no relatório. A atividade que a pessoa exerceu no filme (ex: ator, diretor, câmera etc.) não é importante para o relatório.

Qual consulta retorna o conjunto resposta correto?

a) SELECT P.nomePessoa, T.nomeFilme, sum(T.salario) total

FROM Pessoa P LEFT OUTER JOIN Trabalha T

ON P.nomePessoa = T.nomePessoa

GROUP BY P.nomePessoa, T.nomeFilme

b) SELECT T.nomePessoa, T.nomeFilme, sum(T.salario) total

FROM Pessoa P INNER JOIN Trabalha T

ON P.nomePessoa = T.nomePessoa

GROUP BY T.nomePessoa, T.nomeFilme

c) SELECT P.nomePessoa, T.nomeFilme, sum(T.salario) total

FROM Pessoa P LEFT OUTER JOIN Trabalha T

ON P.nomePessoa = T.nomePessoa

GROUP BY P.nomePessoa, T.nomeFilme HAVING sum(T.salario) > 0

1. Considere que um banco de dados relacional foi criado através da execução dos comandos em SQL a seguir.

CREATE TABLE Empregado (

empregadoMatricula char(5) PRIMARY KEY,

empregadoNome varchar(25),

departamentoID int

);

CREATE TABLE Departamento (

departamentoID int UNIQUE,

departamentoNome varchar(25),

);

ALTER TABLE Empregado ADD CONSTRAINT FK\_empregado\_depto

FOREIGN KEY (departamentoID)

REFERENCES Departamento(departamentoID);

INSERT INTO Departamento VALUES (1, ‘Vendas’);

INSERT INTO Departamento VALUES (3, ‘Engenharia’);

INSERT INTO Departamento VALUES (4, ‘Contabilidade’);

INSERT INTO Departamento VALUES (5, ‘Propaganda’);

INSERT INTO Empregado VALUES (‘10001’, ‘Edmeria’, 1);

INSERT INTO Empregado VALUES (‘10003’, ‘Suely’, 3);

INSERT INTO Empregado VALUES (‘10005’, ‘Marcelo’, 3);

INSERT INTO Empregado VALUES (‘10007, ‘Vagner’, 4);

INSERT INTO Empregado VALUES (‘10009’, ‘Adriana’, 4);

INSERT INTO Empregado VALUES (‘10011’, ‘Joao’, NULL);

Agora considere que a consulta de seleção fornecida a seguir foi executada sobre o banco de dados com esquema e estado resultantes da sequência de comandos acima.

SELECT \* FROM Empregado

LEFT OUTER JOIN Departamento

ON Empregado.departamentoID=Departamento.departamentoID;

* Qual a quantidade de registros retornados pela consulta acima? Por que?

1. Considere o seguinte grupo de tabelas de um sistema:

**fabricante** (idfabricante, nome, endereço)

**peca** (idmodelo, nome, descrição)

**constroi** (*idmodelo* (FK), *idfabricante* (FK), data, quantidade, cor)

Um programador monta a seguinte consulta SQL:

SELECT f.nome, COUNT(DISTINCT c.idmodelo) AS num

FROM fabricante f INNER JOIN constroi c

ON f.idfabricante = c.idfabricante

WHERE c.cor = ‘VERMELHO’

GROUP BY f.nome

HAVING COUNT(DISTINCT c.idmodelo) > 10

ORDER BY num DESC

Qual o retorno dessa consulta?

a) Os nomes dos fabricantes e a respectiva quantidade total de peças construídas na cor vermelha, desde que, em cada data, a quantidade construída seja maior que 10; o relatório estará ordenado de forma descendente pela quantidade de peças construídas.

b) Os nomes dos fabricantes que já construíram mais de 10 modelos diferentes de peças na cor vermelha e a quantidade de modelos diferentes, mostrando a lista ordenada de forma descendente pela quantidade.

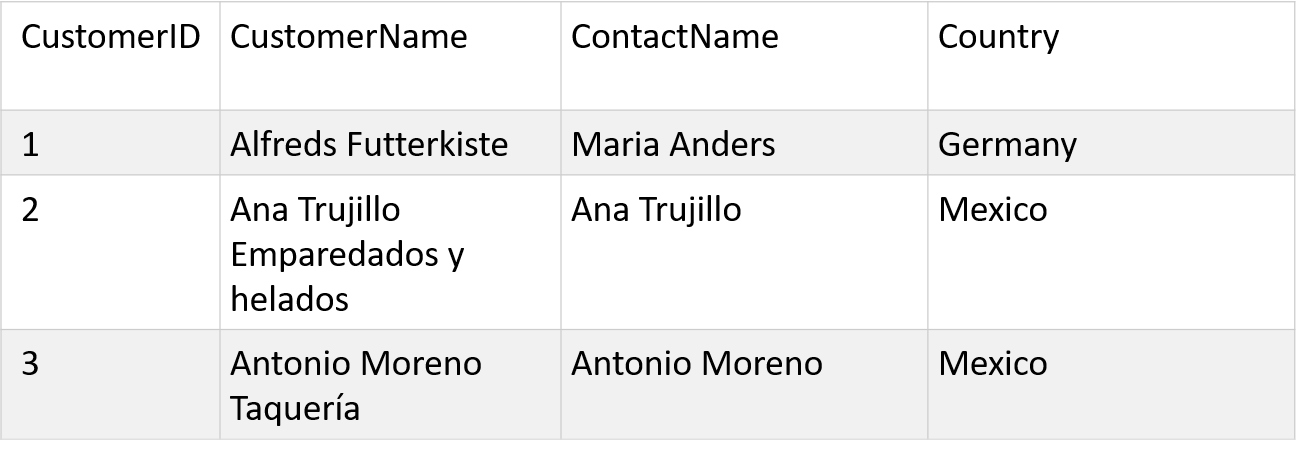
c) Os nomes dos fabricantes que já construíram pelo menos uma peça na cor vermelha; o relatório estará ordenado de forma descendente na quantidade de modelos diferentes construídos.

d) Todos os nomes dos fabricantes e a respectiva quantidade de modelos diferentes de peças vermelhas que já construíram; se um fabricante nunca construiu uma peça na cor vermelha, a contagem mostrará zero.

1. Construa as queries SQL para os dados de Orders e Customer utilizando as quatro modalidades de JOIN.

Orders





Customer

1. Desenhe o modelo conceitual e físico para o caso abaixo:

Uma atividade possui várias questões. Cada questão possui um gabarito. A atividade possui uma nota, que é atribuida a um aluno de uma disciplina. Uma disciplina pertence ao curso de uma universidade e é ministrada por um professor que faz parte de corpo docente da universidade.

Para o modelo elaborado construa os comandos em álgebra relacional:

1. Listar todas as atividades que estão cadastradas.
2. Listar todas as questões por atividade
3. Listar a quantidade de questões de uma atividade
4. Listar os professores que ministram pelo menos uma disciplina
5. Listar todas as disciplinas da universidade
6. Listar as notas dos alunos por disciplina

GABARITO

Item 1 - B  
Item 2 - A  
Item 3 - C  
Item 4 - C  
Item 5 - A  
Item 6 - A  
Item 7 - 6 itens  
Item 8 - B

Parte superior do formulário