



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina Banco de Dados

AD2 2º semestre de 2007.

Nome –

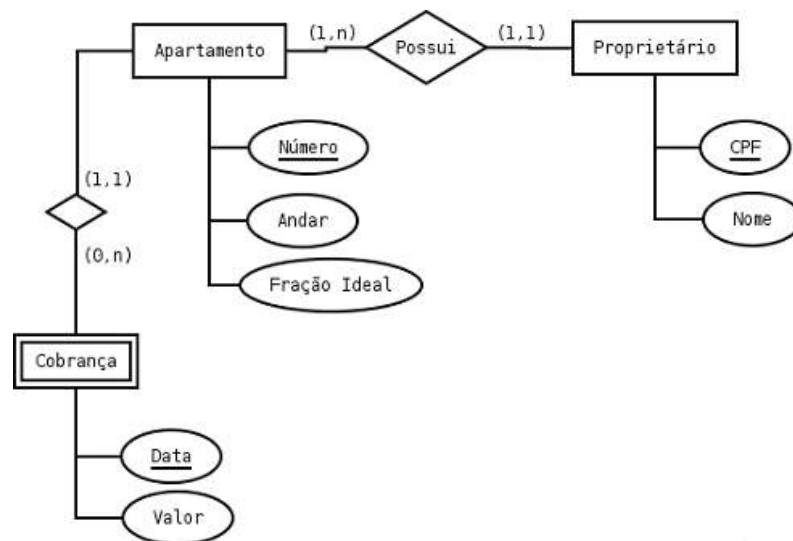
Observações:

1. Prova com consulta.

Atenção: Como a avaliação à distância é individual, caso seja constatado que provas de alunos distintos são cópias uma das outras, independentemente de qualquer motivo, a todas será atribuída a nota **ZERO**. As soluções para as questões podem sim, ser buscadas por grupos de alunos, mas a redação final de cada prova tem que ser individual.

Questão 1 [1 ponto]

Considere o diagrama ER mostrado abaixo (retirado em parte da AD1).



A notação utilizada é a da Ferramenta DIA. A entidade com bordas duplas (Cobrança) é uma entidade fraca. Os identificadores estão sublinhados.

Construa o esquema relacional equivalente a este diagrama ER.

Resposta:

Proprietário (Nome, CPF)

Apartamento (Número, Andar, FracaoIdeal, CPF)

CPF referencia Proprietário (CPF)

Cobrança (NumApartamento, Data, Valor)

NumApartamento referencia Apartamento (Numero)

Questão 2 [3,5 pontos]

Considere o seguinte esquema relacional (as chaves primárias estão sublinhadas):

Pessoa(<u>Cod</u> , Nome, DataNasc)
Publicacao(<u>Cod</u> , Titulo, CodArea)
CodArea referencia Area
Autor(<u>CodAutor</u> , <u>CodPublicacao</u>)
CodAutor referencia Pessoa
CodPublicacao referencia Publicacao
Avaliacao(<u>CodAvaliador</u> , <u>CodPublicacao</u> , Nota)
CodAvaliador referencia Pessoa,
CodPublicacao referencia Publicacao
Area(<u>Cod</u> , Nome, CodAreaGenerica)
CodAreaGenerica referencia Area

(a) Escreva os comandos SQL necessários para criar as relações *Área* e *Publicação* acima, incluindo as restrições de chave primária e chave estrangeira. As seguintes restrições de integridade devem ser garantidas: [0,5 ponto]

1. Ao excluir uma Área, todas as suas subáreas devem ser excluídas.
2. Uma área não pode ser excluída se houverem Publicações associadas a ela.

```
CREATE TABLE Area (  
    cod INTEGER NOT NULL,  
    nome VARCHAR(30),  
    codAreaGenerica INTEGER,  
    PRIMARY KEY (cod),  
    FOREIGN KEY (codAreaGenerica) references Area (cod) ON DELETE CASCADE  
)
```

```
CREATE TABLE Publicacao (  
    cod INTEGER NOT NULL,  
    titulo VARCHAR(50),  
    codArea INTEGER,  
    PRIMARY KEY (cod),  
    FOREIGN KEY (codArea) REFERENCES Area (cod) ON DELETE RESTRICT  
)
```

(b) Escreva a instrução SQL necessária para incluir uma nova Publicação com código 122, título “XML e Banco de Dados”, da área de código 11. [0,3 ponto]

```
INSERT INTO Publicação VALUES (122, “XML e Banco de Dados”, 11)
```

(c) Escreva a instrução SQL necessária para excluir todas as publicações da área de “Banco de Dados”. [0,3 ponto]

```
DELETE FROM Publicacao  
WHERE codArea IN (SELECT cod FROM AREA WHERE nome = "Banco de Dados")
```

(d) Escreva uma instrução SQL para modificar a nota de todas as avaliações do avaliador João para 10. [0,3 ponto]

```
UPDATE Avaliacao  
SET Nota=10  
WHERE codAvaliador IN (SELECT cod FROM Pessoa WHERE nome = "João")
```

(e) Escreva uma instrução SQL que retorna o número de publicações de cada área. No resultado deve aparecer o nome de cada área, seu código e a quantidade de publicações associada a ela. [0,5 ponto]

```
SELECT COUNT(p.cod) AS NumPublicacoes, a.nome, a.cod  
FROM Publicacao p, Area a  
WHERE p.CodArea=a.Cod  
GROUP BY a.cod, a.nome
```

(f) Escreva uma instrução SQL que retorna o código de todas as áreas que têm mais de 2 publicações. [0,5 ponto]

```
SELECT a.cod  
FROM Area a  
WHERE (  
    SELECT COUNT(*), p.cod  
    FROM Publicacao p  
    WHERE a.cod=p.codArea  
    GROUP BY a.codArea  
) > 2
```

(g) Escreva uma instrução SQL que retorna a média de notas obtidas pela publicação “XML e Banco de Dados”. [0,3 ponto]

```
SELECT AVG(nota)  
FROM Avaliacao v, Publicacao p  
WHERE p.cod=v.codPublicacao  
AND v.titulo="XML e Banco de Dados"
```

(h) Escreva uma instrução SQL que retorna o título da publicação que obteve a maior nota de avaliação, e o nome do Avaliador responsável por esta avaliação. [0,3 ponto]

```
SELECT MAX(v.nota), r.Nome, p.Titulo  
FROM Avaliacao v, Publicacao p, Pessoa r  
WHERE p.cod=v.codPublicacao  
AND v.codAvaliador=r.cod
```

(i) Escreva uma instrução SQL para mostrar o efeito de aumentar em 10% a nota de todas as avaliações do avaliador João. [0,5 ponto]

```
SELECT v.codAvaliador, v.codPublicacao, 1.1*v.nota
FROM Avaliacao v, Pessoa r
WHERE v.codAvaliador=r.cod
AND nome = "João"
```

Questão 3 [2,5 pontos]

(a) Qual a diferença entre consultas aninhadas correlacionadas e não correlacionadas ? [1 ponto]

Numa consulta aninhada existe a consulta externa e a consulta interna. A diferença é que na consulta aninhada correlacionada :

- a condição da cláusula WHERE da *consulta interna* referencia um atributo de uma relação declarada na *consulta externa*; e
- o resultado de uma consulta aninhada correlacionada é *diferente para cada tupla (ou combinação de tuplas) da relação(ões) da consulta externa*.

Enquanto que na consulta aninhada não correlacionada não há essa dependência ou correlação entre a consulta externa e a consulta interna.

(b) Exemplifique a diferença apresentando duas consultas sobre o esquema relacional da Questão 2, uma aninhada correlacionada e outra aninhada não correlacionada. [1 ponto]

Consulta aninhada correlacionada :

```
SELECT a.cod, a.nome
FROM Area a
WHERE (
    SELECT COUNT(*), p.codArea
    FROM Publicacao p
    WHERE p.codArea = a.cod
    GROUP BY p.codArea
) > 2
```

Consulta aninhada não correlacionada :

```
SELECT p.nome, p.codArea
FROM Publicacao p
WHERE p. codArea IN (
    SELECT cod
    FROM Area
    WHERE CodAreaGenerica =1
)
```

(c) Escreva uma consulta aninhada correlacionada que mostre o nome das pessoas que não são autores. [0,5 ponto]

```
SELECT p.nome
FROM Pessoa p
WHERE NOT EXISTS ( SELECT *
                    FROM Autor a
                    WHERE p.Cod=a.CodAutor )
```

Questão 4 [2 pontos]

(d) Quais são os objetivos do processo de Normalização? [0,5 ponto]

Reagrupar informações para eliminar redundâncias de dados e para eliminar estruturas inexistentes no modelo ER (atributos multivalorados).

(b) Analise o esquema relacional abaixo, referente ao domínio de vôos e aeroportos (não necessariamente normalizado).

Tab(SiglaCia, NumVoo, NomeCia, SiglaAeropSaida, NomeAeropSaida, HoraSaida, HoraChegada)

As dependências funcionais (podendo incluir dependências transitivas) que existem nesta tabela são as seguintes:

- (SiglaCia, NumVoo) → SiglaAeropSaida
- (SiglaCia, NumVoo) → NomeAeropSaida
- (SiglaCia, NumVoo) → HoraSaida
- (SiglaCia, NumVoo) → HoraChegada
- SiglaCia → NomeCia
- SiglaAeropSaida → NomeAeropSaida

1. Diga em que forma normal encontra-se a tabela. Justifique sua resposta. [0,5 ponto]

A tabela encontra-se na 1FN pois não contém tabelas aninhadas. Ela não está na 2FN por conter dependências parciais.

2. Caso a tabela não se encontre na terceira forma normal, mostre a transformação da tabela para a terceira forma normal. Mostre cada forma normal intermediária, entre aquela em que a tabela se encontra e a terceira forma normal. [1 ponto]

2FN:

Tab1(SiglaCia, NumVoo, SiglaAeropSaida, NomeAeropSaida, HoraSaida, HoraChegada)
Tab2(SiglaCia, NomeCia)

3FN:

Tab1(SiglaCia, NumVoo, NomeCia, SiglaAeropSaida, HoraSaida, HoraChegada)
Tab2(SiglaCia, NomeCia)
Tab3(SiglaAeropSaida, NomeAeropSaida)

Atenção TUTOR: a pontuação desta questão será atribuída da seguinte forma:

0,5 ponto para a 2FN

0,5 ponto para a 3FN

Questão 5 [1 ponto]

Suponha que você é o responsável por implementar o mecanismo de suporte a visões em um SGBD. Suponha também que este SGBD será usado em ambientes que têm uma grande demanda por consultas complexas, com várias junções e agregações, e ao mesmo tempo tem pouquíssima demanda por atualizações. Diante disso, qual das duas alternativas de

implementação de visões vistas em aula (materialização de visões ou reescrita de consultas) você usaria? Justifique sua resposta.

Neste caso, a melhor alternativa seria usar Materialização de visões. Nesta alternativa, o SGBD cria uma tabela temporária e insere nela todas as tuplas que fazem parte da visão, “materializando-as”. JUSTIFICATIVA: As consultas do usuário sobre a visão podem ser executadas de forma eficiente, uma vez que não é necessário nenhum pré-processamento da consulta para fazer a composição das consultas do usuário com a de definição da visão. Consultas sobre visões que envolvem operações caras, como junções, não precisam ser executadas, já que todas as tuplas da junção já estão pré-calculadas na visão materializada. Além disso, como a demanda por atualização é pequena, o custo de manter a visão materializada sempre atualizada será pequeno.