



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina Banco de Dados

AP2 2º semestre de 2007.

Nome –

Assinatura –

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
 2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
 3. Você pode usar lápis para responder as questões.
 4. Ao final da prova devolva as folhas de questões e as de respostas.
 5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
-

Questão 1 [2,5 pontos]

Considere o esquema relacional abaixo (usado na AP1). As chaves primárias estão sublinhadas.

Aluno (<u>CPF</u> , Nome, Curso, DataI)
Disciplina (<u>NumDiscipl</u> , Dnome, Depto)
Matricula (<u>CPF</u> , NumDiscipl, <u>Semestre</u> , Nota)
CPF referencia Aluno
NumDiscipl referencia Disciplina
LivroAdotado (<u>NumDiscipl</u> , <u>Semestre</u> , ISBNLivro)
NumDiscipl referencia Disciplina
ISBNLivro referencia Texto
Texto (<u>ISBNLivro</u> , TituloLivro, Editora, Autor)

- (a) Escreva os comandos SQL para criar as tabelas *Aluno*, *Disciplina* e *Matricula*, incluindo as restrições de integridade que se aplicam. [1,5 ponto]

Assuma que:

1. Quando um aluno é excluído, todas as suas matrículas são excluídas automaticamente.
2. Uma disciplina não pode ser excluída se houver alguma informação sobre matrícula associada a ela.
3. Ao alterar o número de uma disciplina, todas as referências na tabela Matrícula devem ser atualizadas.

```
CREATE TABLE ALUNO (  
    CPF VARCHAR(9) NOT NULL,  
    NOME VARCHAR(30),  
    CURSO VARCHAR(30),  
    DATA DATE,  
    PRIMARY KEY (CPF)  
)
```

```
CREATE TABLE DISCIPLINA (  
    NumDiscipl INTEGER NOT NULL,  
    Dnome VARCHAR(30),  
    Depto VARCHAR(30),  
    PRIMARY KEY (NumDiscipl)  
)
```

```
CREATE TABLE MATRICULA (  
    CPF VARCHAR(9) NOT NULL,  
    NumDiscipl INTEGER,  
    Semestre VARCHAR(7) NOT NULL,  
    Nota DOUBLE,  
    PRIMARY KEY (CPF, SEMESTRE)  
    FOREIGN KEY CPF references ALUNO (CPF) ON DELETE CASCADE,  
    FOREIGN KEY NumDiscipl references DISCIPLINA (NumDiscipl) ON  
DELETE RESTRICT, ON UPDATE CASCADE  
)
```

- (b) Escreva o comando SQL necessário para incluir um novo aluno no banco de dados, com os seguintes dados: CPF 123456789-01, nome “Alexandre Silva”, curso “Computação” e Data de Início 01/03/2007. [0,5 ponto]

```
INSERT INTO ALUNO  
VALUES (“12345678901”, “Alexandre Silva”, “Computação”, 01/03/2007);
```

- (c) Escreva um comando SQL para excluir todas as adoções de livros da editora “XYZ”. [1 ponto]

```
DELETE FROM LIVROADOTADO  
WHERE ISBNLivro IN (SELECT ISBNLivro  
    FROM TEXTO  
    WHERE Editora=“XYZ”)
```

Questão 2 [2 pontos]

Considere o esquema relacional da questão 1. Apresente o código SQL para as seguintes consultas:

(a) Crie uma visão que contenha o nome, o CPF do aluno e o número disciplinas que ele se matriculou no semestre de 02/2007. [0,5 ponto]

```
CREATE VIEW V (CPF, Nome, NumDisciplinas) AS
SELECT a.CPF, a.Nome, COUNT(*)
FROM Aluno a, Matricula m
WHERE a.CPF = m.CPF
AND m.Semestre="02/2007"
GROUP BY a.CPF, a.Nome
```

(b) Escreva uma instrução SQL que dado o aluno de CPF 99999999999 retorna o nome das disciplinas do semestre = "02/2007" em que ele está matriculado e o nome dos livros-texto de cada disciplina. [0,5 ponto]

```
SELECT d.Dnome, t.TituloLivro
FROM Matricula m, Disciplina d, LivroAdotado l, Texto t
WHERE m.NumDiscipl = d.NumDiscipl
AND l.NumDiscipl = d.NumDiscipl
AND t.ISBNLivro = l.ISBNLivro
AND m.Semestre = l.Semestre
AND m.CPF = 99999999999
AND m.Semestre="02/2007"
```

(c) Escreva uma instrução SQL que retorna o número de disciplinas do semestre = "02/2007" com mais de 2 livros-texto. No resultado deve aparecer o nome e número de cada disciplina e a quantidade de livros a ela alocados. [0,5 ponto]

```
SELECT d.Dnome, d.NumDiscipl, COUNT(*)
FROM Disciplina d, LivroAdotado l
WHERE d.NumDiscipl = l.NumDiscipl
AND l.Semestre="02/2007"
GROUP BY l.NumDiscipl, d.Dnome
HAVING COUNT(*) > 2
```

(d) Escreva uma instrução SQL aninhada correlacionada que mostre o nome dos alunos que não estão matriculados no semestre = "02/2007". [0,5 ponto]

```
SELECT a.Nome
FROM Aluno a
WHERE a.CPF NOT IN (
    SELECT CPF
    FROM Matricula m
    WHERE a.CPF = m.CPF
    AND m.Semestre="02/2007"
)
```

Questão 3 [2 pontos]

Considere a seguinte tabela, não necessariamente normalizada, referente a uma base de dados que modela empregados, máquinas e departamentos.

Tab (EmpNumDepto, NumEmp, MaqNumDepto, NumMaq, NomeEmp, NomeMaq, NomeDepto)

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

(NumDepto) \rightarrow NomeDepto

(EmpNumDepto, NumEmp) \rightarrow NomeEmp

(MaqNumDepto, NumMaq) \rightarrow NomeMaq

(a). 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.

(b). 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. [1,5 ponto]

2FN

EmpMaq (EmpNumDepto, NumEmp, MaqNumDepto, NumMaq)

Departamento (NumDepto, NomeDepto)

Empregado (NumDepto, NumEmp, NomeEmp)

Maquina (NumDepto, NumMaq, NomeMaq)

3FN=2FN

Questão 4 [1,5 ponto]

Explique o que é e diga qual é o principal problema envolvido em atualização através de visões.

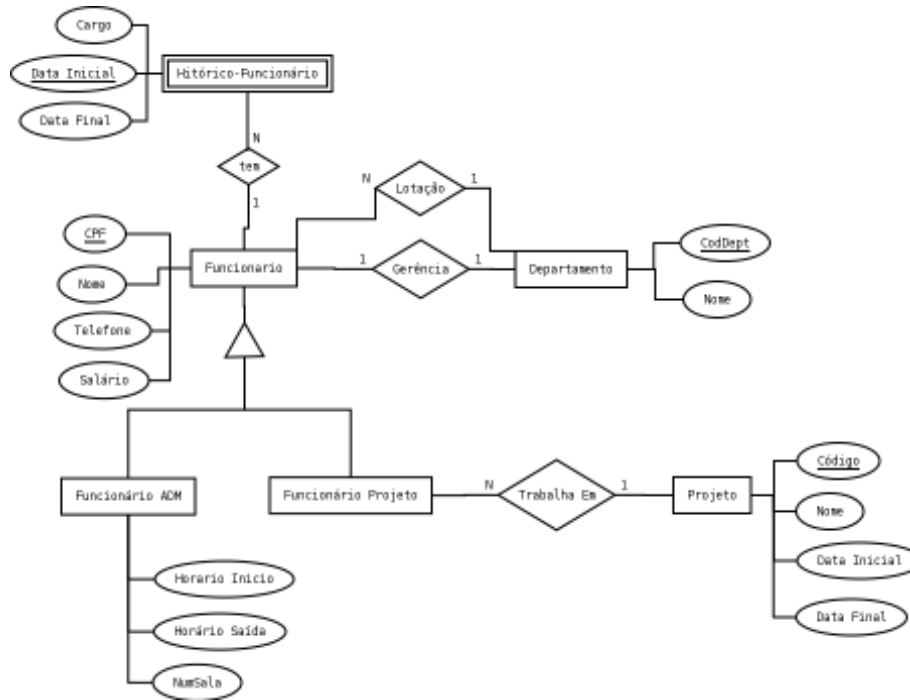
Atualização através de visões é uma técnica que permite atualizar (inserir, excluir ou modificar) tuplas de uma visão, de modo que as tabelas base sobre as quais a visão foi construída sejam automaticamente atualizadas.

Problema:

- pode existir mais de um mapeamento de uma atualização sobre a visão para atualizações nas tabelas base (ambigüidade).

Questão 5 [2 pontos]

Considere o diagrama ER mostrado abaixo (retirado da AP1). Construa um esquema relacional equivalente a este diagrama ER. O diagrama encontra-se na notação do DIA (ferramenta que usamos em sala de aula para construir modelos ER). A entidade mostrada com linha dupla (Histórico-Funcionário) representa uma entidade fraca. Use uma tabela para cada entidade especializada para mapear a generalização/especialização.



Departamento (CodDept, Nome, Gerente)
 Gerente referencia Funcionário
 Funcionário (CPF, Nome, Telefone, Salário, Tipo, CodDept)
 CodDept referencia Departamento
 FuncionárioADM (CPF, HorárioInício, HorárioSaída, NumSala)
 CPF referencia Funcionário
 FuncionárioProjeto (CPF, CodProjeto)
 CPF referencia Funcionário
 CodProjeto referencia Projeto
 Projeto (Código, Nome, DataInicial, DataFinal)
 Histórico-Funcionário (CPF, Cargo, DataInicial, DataFinal)
 CPF referencia Funcionário