



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina Banco de Dados

AD2 1º semestre de 2008.

Nome –

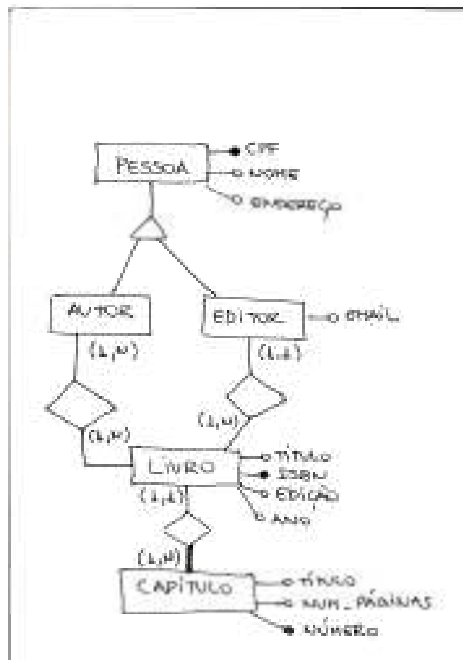
Observações:

1. Prova com consulta.

Atenção: Como a avaliação à distância é individual, caso seja constatado que provas de alunos distintos são cópias umas 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 da AD1).



Construa o esquema relacional equivalente a este diagrama ER. Explícite qual foi a forma escolhida para mapear a herança: uma tabela por hierarquia, ou uma tabela para cada entidade especializada.

Respostas (são alternativas):

Opção de mapeamento utilizada: uma tabela por hierarquia

Pessoa (CPF, Nome, Endereço, Email, Tipo)
Livro (ISBN, Título, Edição, Ano, CPF)
CPF referencia Pessoa /* Editor do Livro */
AutorLivro (CPF, ISBN)
CPF referencia Pessoa
ISBN referencia Livro
Capitulo (ISBN, Número, Título, Num_Páginas)
ISBN referencia Livro

Opção de mapeamento utilizada: uma tabela por entidade especializada

Pessoa (CPF, Nome, Endereço)
Editor (CPF, Email)
CPF referencia Pessoa
Livro (ISBN, Título, Edição, Ano, CPF)
CPF referencia Editor
AutorLivro (CPF, ISBN)
CPF referencia Pessoa
ISBN referencia Livro
Capitulo (ISBN, Número, Título, Num_Páginas)
ISBN referencia Livro

Questão 2 [3,0 pontos]

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

Emp (<u>eid</u> : integer, <i>ename</i> : string, <i>idade</i> : integer, <i>salario</i> : real)
Trabalha (<u>eid</u> : integer, <u>did</u> : integer, <i>cargahoraria</i> : integer)
Dept (<u>did</u> : integer, <i>dnome</i> : string, <i>orçamento</i> : real, <i>gerenteid</i> : integer)

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

1. Ao excluir um Departamento, todas as tuplas relacionadas em *Trabalha* devem ser excluídas.
2. Um empregado não pode ser excluído se ele for gerente de algum departamento.

```
CREATE TABLE Trabalha (  
    eid INTEGER NOT NULL,  
    did INTEGER NOT NULL,
```

```
cargahoraria INTEGER,  
PRIMARY KEY (eid, did),  
FOREIGN KEY (eid) references Emp (eid)  
FOREIGN KEY (did) references Dept (did) ON DELETE CASCADE  
)
```

```
CREATE TABLE Dept (  
    did INTEGER NOT NULL,  
    dnome VARCHAR(50),  
    orçamento: REAL,  
    gerenteid INTEGER,  
    PRIMARY KEY (did),  
    FOREIGN KEY (gerenteid) REFERENCES Emp (eid) ON DELETE RESTRICT  
)
```

(b) Escreva a instrução SQL necessária para incluir um novo empregado de nome *João Silva*, eid 1234 e salário 1200. [0,5 ponto]

```
INSERT INTO Emp VALUES (1234, "João Silva", 1200)
```

(c) Escreva a instrução SQL necessária para excluir todos os empregados que trabalham no departamento cujo did é igual a 122. [0,5 ponto]

```
DELETE FROM Emp  
WHERE eid IN (SELECT eid FROM Trabalha WHERE did = 122)
```

(d) Escreva uma instrução SQL para modificar o salário de todos os empregados que trabalham no departamento 122. O aumento no salário deve ser de 20%. [0,5 ponto]

```
UPDATE Emp  
SET salario=salario*1,2  
WHERE eid IN (SELECT eid FROM Trabalha WHERE did = 122)
```

(e) Escreva uma instrução SQL que retorna a média dos salários do departamento cujo did é 122 [0,5 ponto]

```
SELECT AVG(salario)  
FROM Emp e, Trabalha t  
WHERE e.eid=t.eid  
AND t.did=122
```

(f) Escreva uma instrução SQL que retorna o número de empregados de cada departamento. No resultado deve aparecer o nome do departamento, seu did e o número de empregados que nele trabalham. [0,5 ponto]

```
SELECT d.dnome, d.eid, COUNT(t.eid) AS NUMEMP  
FROM Dept d, Trabalha t  
WHERE t.did=d.did  
GROUP BY d.dnome, d.eid
```

(g) Escreva uma consulta SQL que retorne, para cada departamento, o nome do departamento e maior salário dentre seus empregados.

```

SELECT d.dnome, MAX(SALARIO)
FROM Emp e, Dept d, Trabalha t
WHERE t.did=d.did
AND e.eid=t.eid
GROUP BY d.dnome, d.eid

```

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 e.eid, e.nome
FROM Emp e
WHERE (
    SELECT COUNT(*), t.eid
    FROM Trabalha t
    WHERE t.eid = e.eid
    GROUP BY t.eid
) > 2

```

Consulta aninhada não correlacionada :

```

SELECT e.eid, e.nome
FROM Emp e
WHERE e.eid IN (
    SELECT t.eid
    FROM Trabalha t
    WHERE cargahoraria > 20
)

```

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

QUESTÃO ANULADA, pois referenciava tabelas que não existem no esquema da questão 2.

Questão 4 [1,5 ponto]

- (d) O que é uma Forma Normal? [0,5 ponto]

Uma Forma Normal é um conjunto de regras que uma tabela deve obedecer para ser considerada “bem projetada”.

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

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

Tab2(SiglaCia,NomeCia)

As dependências funcionais (podendo incluir dependências transitivas) que existem nestas tabelas 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 cada uma destas tabelas se encontra. Justifique sua resposta. [0,5 ponto]

Tab1: a tabela encontra-se na 2FN pois não contém tabelas aninhadas, e não contém dependências parciais, mas contém dependências transitivas.

Tab2: a tabela encontra-se na 3FN pois não contém tabelas aninhadas, não contém dependências parciais, e nem dependências transitivas.

2. Caso alguma das tabelas (ou ambas) não se encontre na terceira forma normal, mostre a transformação da para a terceira forma normal. Mostre cada forma normal intermediária, entre aquela em que as tabelas se encontram e a terceira forma normal. [0,5 ponto]

3FN:

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

Tab2(SiglaCia, NomeCia)

Tab3(SiglaAeropSaida, NomeAeropSaida)

Questão 5 [1,5 ponto]

Esta questão refere-se ao conceito de VISÃO em bancos de dados relacionais.

- (a) Explique o conceito de VISÃO em bancos de dados relacionais.

Visões são tabelas virtuais, derivadas de outras tabelas ou de outras visões.

- (b) Cite duas situações em que visões podem ser utilizadas.

Normalmente são usadas para simplificar a especificação de consultas freqüentes, como por exemplo, consultas que envolvem várias junções, e para auxiliar os mecanismos de segurança do SGBD, já que com visões é possível “esconder” informações dos usuários.

- (c) Dê um exemplo de visão, usando SQL.

```
CREATE VIEW EMP_PROJ AS (  
  SELECT PNO, UNOME, PJNO, HORAS  
  FROM EMPREGADO, PROJETO, TRABALHA_EM  
  WHERE SSN=ESSN AND PNO=PNUMERO  
)
```