



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
Disciplina Banco de Dados
AD2 2º semestre de 2015.

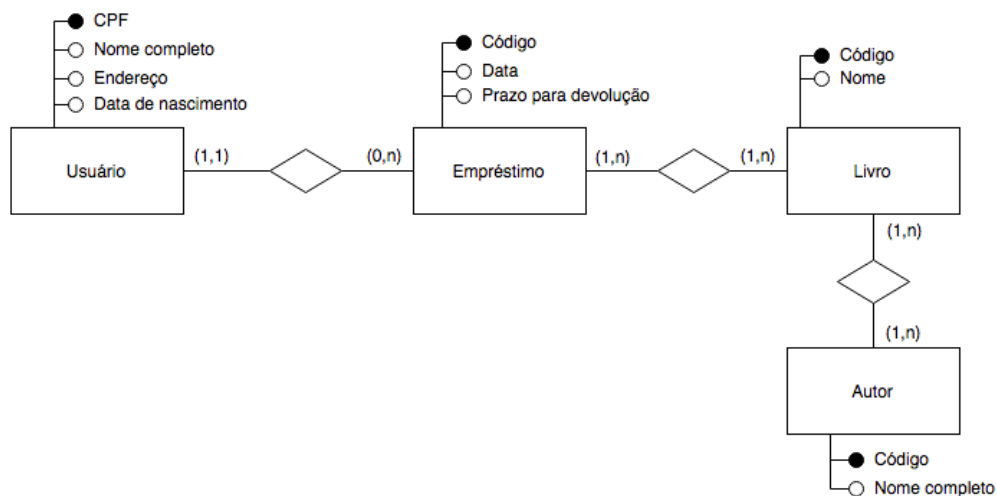
Nome: _____

Observações:

1. Prova COM consulta.
2. As ADs deverão ser postadas na plataforma antes do prazo final de entrega estabelecido no calendário de entrega de ADs.
3. Lembre-se de enviar as ADs para avaliação. Cuidado para não deixar a AD como “Rascunho” na plataforma!
4. ADs em forma de “Rascunho” não serão corrigidas!
5. As ADs devem ser enviadas no formato de arquivo PDF ou DOC. No caso de arquivos no formato DOC, eles devem conter apenas o link para o Google Drive.
6. ADs entregues em outros formatos não serão corrigidas!

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. [2,5 pontos] Considere o diagrama ER mostrado abaixo.



Construa um esquema relacional equivalente a este diagrama ER.

As chaves primárias estão sublinhadas.

Usuario (CPF, NomeCompleto, Endereco, DataNascimento)

Emprestimo (Codigo, Data, PrazoDevolucao, CPF)

CPF REFERENCIA Usuario

EmprestimoLivro (CodEmprestimo, CodLivro)

CodEmprestimo REFERENCIA Emprestimo

CodLivro REFERENCIA Livro

Livro (Codigo, Nome)

LivroAutor (CodLivro, CodAutor)

CodLivro REFERENCIA Livro

CodAutor REFERENCIA Autor

Autor (Codigo, NomeCompleto)

Questão 2. [1,0 ponto] Quais as vantagens de se usar visões em bancos de dados?

As visões simplificam a especificação de consultas frequentes e auxilia os mecanismos de autorização e segurança.

Questão 3. [5,0 pontos] Considere o esquema relacional a seguir, onde as chaves primárias estão sublinhadas.

Cliente(<u>cid</u> : integer, nome: string, endereco: string)
Compra (<u>cid</u> : integer, <u>pid</u> : integer, quantidade: integer)
cid REFERENCIA Cliente
pid REFERENCIA Produto
Produto (<u>pid</u> : integer, nome: string, tid: integer, preco: real)
tid REFERENCIA Tipo
Tipo (<u>tid</u> : integer, nome: string)

Sobre esta base de dados, resolver as consultas utilizando SQL. Não usar mais tabelas que o estritamente necessário.

a) Escreva uma instrução SQL que cria a tabela Cliente com as colunas especificadas no enunciado. [0,5 ponto]

```
CREATE TABLE Cliente (  
    cid integer,  
    nome varchar(255),  
    endereco varchar(255),  
    PRIMARY KEY (cid)  
);
```

b) Escreva uma instrução SQL para inserir uma tupla na tabela Cliente com identificador 10, nome “José Antunes” e endereço “Rua Óbidos”. [0,5 ponto]

```
INSERT INTO Cliente (cid, nome, endereco) VALUES (10, “José Antunes”, “Rua Óbidos”);
```

c) Faça uma consulta que retorna os identificadores dos clientes que tiveram compras com mais de 30 itens de quantidade. [0,5 ponto]

```
SELECT cid  
FROM Compra  
WHERE quantidade > 30;
```

d) Faça uma consulta que retorna os identificadores dos clientes que possuem compras com um produto com valor acima de 50 reais. [0,5 ponto]

```
SELECT c.cid  
FROM Produto as p, Compra as c  
WHERE p.pid = c.pid  
AND preco > 50;
```

e) Faça uma consulta que retorna os nomes dos clientes que não fizeram compras de mais de 10 itens de produtos que custem até 230 reais. [0,5 ponto]

```
SELECT c.nome  
FROM Cliente as c  
WHERE c.cid NOT IN  
  (SELECT cp.cid  
   FROM Compra as cp, Produto as p  
   WHERE cp.pid = p.pid  
   AND cp.quantidade >= 10  
   AND p.preco >= 230);
```

f) Faça uma consulta que retorna o número de clientes que compraram cada tipo de produto. A consulta deve retornar o nome do tipo e o número de clientes que compraram produtos daquele tipo. [0,5 ponto]

```
SELECT t.nome, COUNT(c.cid)  
FROM Cliente as c, Compra as cp, Produto as p, Tipo as t  
WHERE c.cid = cp.cid  
AND cp.pid = p.pid  
AND p.tid = t.tid  
GROUP BY t.nome;
```

g) Escreva uma instrução SQL para excluir a tabela Tipo. [0,5 ponto]

```
DROP TABLE Tipo;
```

h) Escreva uma instrução SQL para excluir todos os produtos que são do tipo “esporte”. [0,5 ponto]

```
DELETE FROM Produto  
WHERE pid IN (SELECT pid FROM Produto as p, Tipo as t  
WHERE p.tid = t.tid AND t.nome = “esporte”);
```

i) Faça uma consulta que retorna os nomes dos produtos do tipo “esporte” que nunca foram comprados.
[0,5 ponto]

```
SELECT p.nome  
FROM Produto as p, Tipo as t  
WHERE p.tid = t.tid  
AND t.nome = “esporte”  
AND p.pid NOT IN  
(SELECT c.pid  
FROM Compra as c);
```

j) Faça uma consulta que retorna os nomes dos produtos que foram comprados por menos de 10 clientes.
[0,5 ponto]

```
SELECT p.nome  
FROM Compra as cp, Produto as p  
WHERE cp.pid = p.pid  
GROUP BY p.nome  
HAVING COUNT(cp.cid) < 10;
```

Questão 4. [1,5 ponto]

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

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

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

a) Diga em que forma normal cada uma destas tabelas se encontra. Justifique sua resposta. [1,0 ponto]

Todas as tabelas estão na Terceira Forma Normal, pois não possuem tabelas aninhadas, não possuem dependências parciais, nem dependências transitivas.

b) 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]

Já estão na 3FN.