



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
Disciplina Banco de Dados
AD2 1º semestre de 2016.

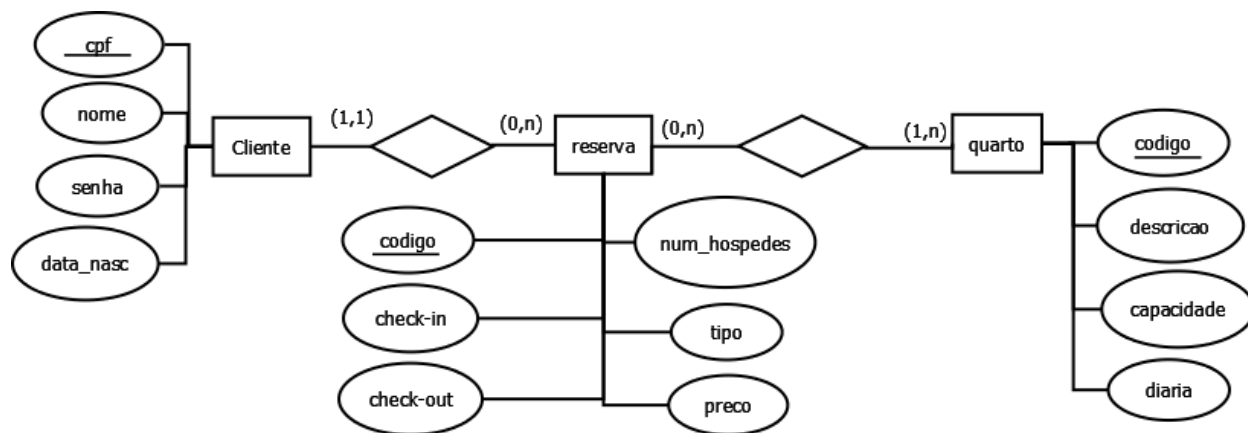
Nome: _____

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.
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 Entidade-Relacionamento (ER) mostrado abaixo.



Construa um esquema relacional equivalente a este diagrama ER, indicando chaves primárias e estrangeiras.

As chaves primárias estão sublinhadas.

Cliente(cpf,nome,senha,data_nasc)

Reserva(codigo,checkin,checkout,num_hospedes,tipo,preco,cpf)

cpf REFERENCIA Cliente

Quarto(codigo,descricao,capacidade,diaria)

ReservaQuarto(reservaid,quartoid)

reservaid REFERENCIA Reserva

quartoid REFERENCIA Quarto

Questão 2. [5,5 pontos] Considere o esquema relacional a seguir, onde as chaves primárias estão sublinhadas.

Usuario(uid:integer, login:string, email:string)

Relacionamento(uid1:integer, uid2:integer)

uid1 REFERENCIA Usuario

uid2 REFERENCIA Usuario

Evento(eid:integer, nome:string, criadorid:integer, data:date, local:string)

criadorid REFERENCIA Usuario

ParticipanteEvento(eid:integer, uid:integer, funcao:string)

eid REFERENCIA Evento

uid REFERENCIA Usuario

Postagem(pid:integer, uid:integer, data:date, assunto:string, mensagem:string)

uid REFERENCIA Usuario

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 Evento com as colunas especificadas no enunciado. Adicione uma restrição que garanta que, ao excluir um determinado usuário do banco de dados, todos os eventos criados por ele sejam excluídos automaticamente. [0,5 ponto]

CREATE TABLE Evento (

eid integer,

criadorid integer,

```
data date,  
local varchar(500),  
PRIMARY KEY (eid),  
FOREIGN KEY (criadorid) REFERENCES Usuario(uid) ON DELETE CASCADE  
);
```

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

```
DROP TABLE Relacionamento;
```

c) Escreva uma instrução SQL para inserir uma tupla na tabela Postagem com identificador 288, sendo que tal postagem pertence ao usuário com identificador 22 e foi publicada com a data de 10/02/2016 contendo o assunto “Plano de trabalho” e a mensagem “Como realizar um plano de trabalho?”. [0,5 ponto]

```
INSERT INTO Postagem (pid, uid, data, assunto, mensagem) VALUES (288, 22, “10/02/2016”,  
“Plano de trabalho”, “Como realizar um plano de trabalho?”);
```

d) Escreva uma instrução SQL para excluir todos os eventos que foram criados para a data de 29/02/2016 pelo usuário portador do login “pedro”. [0,5 ponto]

```
DELETE FROM Evento  
WHERE data = “29/02/2016”  
AND criadorid IN (SELECT uid FROM Usuario u  
WHERE u.login = “pedro”);
```

e) Faça uma consulta que retorna o identificador e o login do usuário criador das postagens com o assunto “Jogos Olímpicos”. [0,5 ponto]

```
SELECT p.pid, u.login  
FROM Postagem p, Usuario u  
WHERE p.criadorid = u.uid  
AND assunto = “Jogos Olímpicos”;
```

f) Escreva uma instrução SQL para modificar todas as postagens com assunto “Jogos Olímpicos” enviadas pelo usuário com login “pedro” para que suas datas sejam modificadas para o dia 15/08/2016. [0,5 ponto]

```
UPDATE Postagem  
SET data=“15/08/2016”  
WHERE pid IN (  
SELECT pid  
FROM Usuario
```

```
WHERE u.login = "pedro")  
AND assunto = "Jogos Olímpicos";
```

g) Faça uma consulta que retorna os logins dos usuários que participaram de mais de 20 eventos com a função de administrador. [0,5 ponto]

```
SELECT u.login  
FROM Usuario as u, Evento as e, ParticipanteEvento as pe  
WHERE u.uid = pe.uid  
AND e.eid = pe.eid  
AND pe.funcao = "administrador"  
GROUP BY u.login  
HAVING COUNT(e.eid) > 20;
```

h) Crie uma visão que contenha os logins dos usuários e os nomes dos eventos criados por eles. [0,5 ponto]

```
CREATE VIEW v AS  
SELECT u.login, e.nome  
FROM Usuario as u, Evento as e  
WHERE u.uid = e.criadorid;
```

i) Faça uma consulta que retorna os nomes e os locais dos eventos que não apresentem participantes com função de administrador. [0,5 ponto]

```
SELECT e.nome, e.local  
FROM Evento as e  
WHERE e.eid NOT IN  
(SELECT pe.eid  
FROM ParticipanteEvento as pe  
WHERE pe.funcao = "administrador");
```

j) Faça uma consulta que retorna os e-mails dos usuários que apresentam relacionamento com o usuário portador do login "pedro" e que nunca participaram de um evento localizado no Maracanã. [1,0 ponto]

```
SELECT u2.email  
FROM Usuario as u1, Usuario as u2, Relacionamento as r  
WHERE r.uid1 = u1.uid  
AND r.uid2 = u2.uid  
AND u1.login = "pedro"
```

```
AND u2.uid NOT IN (  
    SELECT u.uid  
    FROM Usuario as u, Evento as e, ParticipanteEvento as pe  
    WHERE u.uid = pe.uid  
    AND e.eid = pe.eid  
    AND e.local = "Maracanã"  
);
```

Questão 3. [2,0 pontos] Em virtude do crescimento da cadeia de concessionárias de veículos conhecida por Carros de Sucesso, ela precisa manter dados sobre os veículos que estão à venda, com seus respectivos modelos, nas concessionárias da rede. Além disso, é preciso manter informações das concessionárias e suas respectivas localidades. Cada veículo à venda tem um número de série único. Um projetista de banco de dados propôs a seguinte tabela como solução para armazenar os dados da rede de concessionárias de veículos, supondo que cada veículo tenha apenas um fabricante.

Concessionaria (CodConcessionaria, Nome, CodLocal, NomeLocal (no_serie, Modelo, Fabricante))

O significado de cada coluna é o seguinte:

- CodConcessionaria: código da concessionária
- Nome: nome da concessionária
- CodLocal: código da localidade onde a concessionária se localiza
- NomeLocal: nome da localidade onde a concessionária se localiza
- no_serie: Número de série do veículo
- Modelo: Modelo do veículo
- Fabricante: nome do fabricante

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

- CodConcessionaria → Nome
- CodLocal → NomeLocal
- CodConcessionaria → CodLocal
- (CodConcessionaria, no_serie) → Modelo, Fabricante
- Modelo → Fabricante

a) Assumindo que o profissional não conhece o conceito de normalização, explique para ele em que forma normal encontra-se a tabela.

Não se encontra normalizada.

b) Caso a tabela não se encontre na terceira forma normal, mostre a(s) transformação(ões) para a terceira forma normal. Mostre cada forma normal intermediária, entre aquela em que a tabela se encontra e a terceira forma normal.

Passagem para a Primeira Forma Normal:

Concessionaria (CodConcessionaria, Nome, CodLocal, NomeLocal)

Veiculo (CodConcessionaria, no_serie, Modelo, Fabricante)

Passagem para a Segunda Forma Normal:

Já está

Passagem para a Terceira Forma Normal:

Concessionaria (CodConcessionaria, Nome, CodLocal)

Localidade (CodLocal, NomeLocal)

Veiculo (no_serie, Modelo)

Modelo (Modelo, Fabricante)