



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**

**Disciplina Banco de Dados**

**AP2 1º semestre de 2016**

**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**

Considere o esquema de um banco de dados que modela publicações de pessoas de uma instituição (parecido com o que foi usado na AP1). As chaves primárias estão sublinhadas. Sobre esta base de dados, resolver as questões a seguir usando SQL. Não usar mais tabelas que o estritamente necessário.

```
Instituicao (CodInst, NomeInst)
Pessoa(CodPess, Nome, DataNasc, CodInst)
    CodInst referencia Instituicao
Aluno(CodPess, Nivel)
    CodPess referencia Pessoa
Publicacao(CodPub, Titulo)
Autor(CodPess, CodPub)
    CodPess referencia Pessoa
    CodPub referencia Publicacao
```

(a) Escreva uma instrução SQL para inserir uma pessoa de código 121, nome “João Antunes”, nascida em 10/10/1980 e pertencente à instituição de código 12. [0,5 ponto]

```
INSERT INTO Pessoa (CodPes, Nome, DataNasc, CodInst)
VALUES (123, “João Antunes”, “10/10/1980”, 12)
```

(b) Escreva uma instrução SQL para incluir uma coluna chamada Ordem, do tipo INT, na tabela Autor. [0,5 ponto]

```
ALTER TABLE Autor  
ADD Ordem INT
```

(c) Faça uma consulta SQL que retorna o nome e data de nascimento das pessoas que não possuem publicação. [1 ponto]

```
SELECT p.Nome, p.DataNasc  
FROM Pessoa p, Autor a  
WHERE p.codPess NOT IN (SELECT CodPess FROM Autor)
```

ATENÇÃO: a tabela publicação não é necessária para responder essa questão

(d) Faça uma consulta SQL que retorna o nome de todos os alunos que estão no nível de Doutorado na instituição de nome UFF. [1 ponto]

```
SELECT p.Nome  
FROM Pessoa p, Aluno a, Instituição i  
WHERE p.CodPess = a.CodPess  
AND p.CodInst = i.CodInst  
AND i.NomeInst = "UFF"  
AND a.Nivel = "Doutorado"
```

(e) Faça uma consulta SQL que retorna o número de publicações realizadas por “João Felipe Silva” na instituição de nome UFF. [1 ponto]

```
SELECT COUNT(*)  
FROM Pessoa p, Instituição i, Autor a  
WHERE p.CodPess = a.CodPess  
AND p.CodInst = i.CodInst  
AND i.NomeInst = "UFF"  
AND p.Nome = "João Felipe Silva"
```

ATENÇÃO: a tabela publicação não é necessária para responder essa questão

(f) Faça uma consulta SQL que retorna o número de publicações realizadas por autores de cada instituição. A consulta deve retornar o nome da instituição e o número de publicações de autores daquela instituição, ordenado pelo nome da instituição. [1 ponto]

```
SELECT i.NomeInst, COUNT(*)  
FROM Instituição i, Autor a, Pessoa p, Publicação pub  
WHERE p.CodInst = i.CodInst  
AND p.CodPess = a.CodPess  
AND pub.CodPub = a.CodPub  
GROUP BY i.NomeInst
```

ORDER BY NomeInst

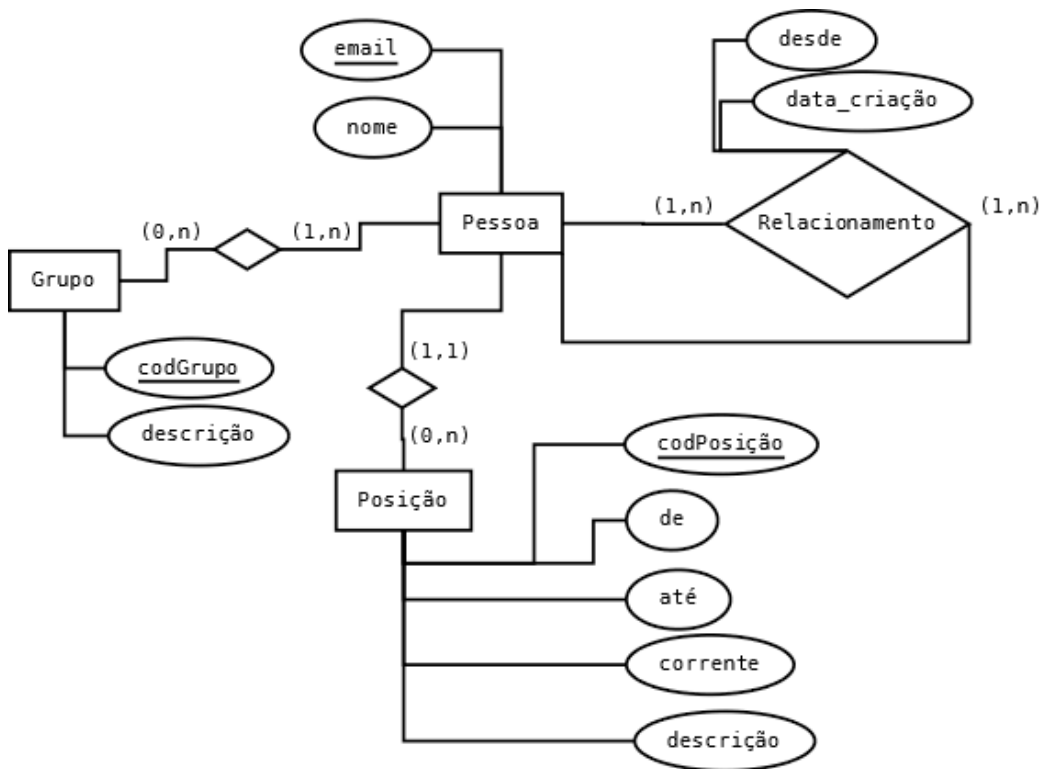
(g) Escreva uma instrução SQL para excluir todas as pessoas que são alunas da UFF no nível de Mestrado. [1 ponto]

```
DELETE FROM Pessoa
WHERE codPess IN (SELEC CodPess
                  FROM Aluno a, Instituicao i, Pessoa p
                  WHERE a.CodPess = p.CodPess
                  AND p.CodInst = i.CodInst
                  AND i.NomeInst = "UFF"
                  AND a.Nivel = "Mestrado")
```

**Questão 2** [2 pontos]

Considere o diagrama ER mostrado abaixo. Ele modela um sistema que armazena dados sobre pessoas. Para cada pessoa, o banco de dados armazena um identificador interno, um endereço de correio eletrônico e o nome da pessoa, bem como os relacionamentos da pessoa. Cada relacionamento é feito com uma outra pessoa cadastrada no banco de dados. Para um relacionamento, além da pessoa relacionada, armazena-se a data na qual o relacionamento foi cadastrado e uma descrição do relacionamento (algo como "colega no curso de Computação do CEDERJ").

As pessoas podem participar de grupos. Cada grupo é identificado por uma sigla e tem um nome. Um grupo pode reunir várias pessoas e uma pessoa pode participar de vários grupos. Grupos vazios são apagados da base de dados, isto é, todo grupo deve ter ao menos um participante. Além disso, o banco de dados armazena um histórico profissional da pessoa. Para cada posição ocupada pela pessoa, ele armazena o ano de início e de fim, bem como uma descrição da posição, em formato livre. O banco de dados guarda também a posição corrente da pessoa (se houver).



Deve ser projetado o esquema de uma base de dados relacional para o modelo ER em questão. A base de dados deve refletir exatamente o especificado no modelo conceitual. O esquema da base de dados relacional deve conter os nomes das tabelas, os nomes dos atributos, atributos que formam a chave primária e as chaves estrangeiras. Quando houver alternativas de projeto dizer que alternativa foi usada. Pode ser usada a notação vista em aula para representar esquemas relacionais.

Posicao (codPosição, de, até, corrente, descrição)

Pessoa (email, nome, codPosição)

codPosicao referencia Posicao

Grupo (codGrupo, descrição)

PessoaGrupo (email, codGrupo)

email referencia Pessoa

codGrupo referencia Grupo

Relacionamento (emailp1, emailp2, desde, dataCriacao)

emailp1 referencia Pessoa

emailp2 referencia Pessoa

### Questão 3 [2 pontos]

Considere a seguinte tabela, não necessariamente normalizada, de uma base de dados referente a locadoras de veículos (as chaves primárias estão sublinhadas):

Locadora (CodLocadora, NomeLocadora, CodCidLocadora, NomeCidLocadora)  
Veículo (CodLocadora, CodVeiculo, Modelo, Placa, Km)

O significado de cada coluna é o seguinte:

CodLocadora: código da locadora

NomeLocadora: nome da locadora

CodCidLocadora: código da cidade onde a locadora se localiza

NomeCidLocadora: nome da cidade onde a locadora se localiza

CodVeiculo: código do veículo

Modelo: modelo do veículo

Placa: placa do veículo

Km: kilometragem do veículo

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

CodLocadora → NomeLocadora

CodCidLocadora → NomeCidLocadora

CodLocadora → CodCidLocadora

(CodLocadora, CodVeiculo) → Modelo, Placa, Km

(a) Diga em que forma normal encontra-se a tabela. [0,5 ponto]

**Segunda Forma Normal.**

(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]

**Passagem para a Terceira Forma Normal**

Locadora (CodLocadora, NomeLocadora, CodCidLocadora)

Cidade (CodCidLocadora, NomeCidLocadora)

Veiculo (CodVeiculo, Modelo, Placa, Km)