



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**

**Disciplina Banco de Dados**

**AP2 1º semestre de 2011**

**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 a seguinte base de dados, usada por uma oficina de manutenção de automóveis (a mesma da AP1). As chaves primárias estão sublinhadas.

```
/* tabela de clientes cadastrados na oficina */
```

```
CLIENTE (cpf, nome)
```

```
/* tabela com dados dos automóveis dos clientes da oficina */
```

```
AUTOMOVEL (placa, no_chassis, modelo, cpf);
```

```
(cpf) references CLIENTE
```

```
/* tabela com as revisões periódicas programadas e feitas – para cada automóvel, a oficina cadastra todas revisões programadas:
```

```
– Km e data_programada são a quilometragem e a data em que deve ser feita a revisão
```

```
– data_ultim_telef serve para informar quando o pessoal da oficina ligou para o cliente lembrando da provável necessidade de fazer a revisão – caso o cliente não tenha sido chamado, este campo contém a string vazia (")
```

```
– data_executada e Km_executada informa a data e a quilometragem de uma revisão que já foi executada – caso a revisão não tenha sido executada ainda, estes campos contêm a string vazia (") */
```

```
REVISAO (placa, Km, data_programada, data_ultim_telef,  
data_executada, Km_executada)  
(placa) references AUTOMOVEL
```

```
/* tabela com as peças usadas em cada revisão */  
PEÇA_REVISAO (placa, Km, cod_peça, quantidade)  
(placa, Km) references REVISAO  
(cod_peça) references PEÇA
```

```
/* tabela com as descrições das peças */  
PEÇA (cod_peça, descricao_peça)
```

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

(a) Escreva uma instrução SQL para inserir um automóvel de placa KYK1234, no\_chassis igual a 999, modelo Honda Fit, e CPF igual a 1010. [1 ponto]

```
INSERT INTO AUTOMOVEL (placa, no_chassis, modelo, cpf)  
VALUES ("KYK1234", 999, "Honda Fit", 1010)
```

**ATENÇÃO:** na correção da questão também será aceito no\_chassis e CPF entre aspas.

(b) Escreva uma instrução SQL para excluir a tabela Cliente. [1 ponto]

```
DROP TABLE Cliente
```

(c) Faça uma consulta SQL que retorna o nome e cpf clientes que possuem algum automóvel cadastrado. O resultado deve estar ordenado pelo nome do cliente. [1 ponto]

```
SELECT c.nome, c.cpf  
FROM Cliente c, Automovel a  
WHERE c.cpf = a.cpf  
ORDER BY c.nome_cli
```

(d) Faça uma consulta SQL que retorna a descrição das peças que foram usadas na revisão do automóvel de placa KYK1234. [1 ponto]

```
SELECT p.descricao_peca  
FROM Peca p, Peca_Revisao pr  
WHERE p.cod_peca = pr.cod_peca  
AND pr.placa = "KYK1234"
```

(e) Faça uma consulta SQL que retorna a placa dos veículos que usaram mais de uma peça diferente na revisão de 10000Km (**Atenção!** O critério de seleção desta consulta não se refere ao campo quantidade da tabela peça\_revisao). [1 ponto]

```
SELECT pr.placa
FROM Peca_Revisao pr
WHERE pr.km = 10000
GROUP BY pr.placa
HAVING COUNT(*) > 1
```

(f) Faça uma consulta SQL que retorna o nome, o cpf do cliente e a quantidade de automóveis que fazem revisão na oficina. [1 ponto]

```
SELECT c.cpf, c.nome, COUNT(*)
FROM Cliente c, Automóvel a
WHERE c.cpf = a.cpf
GROUP BY c.cpf, c.nome
```

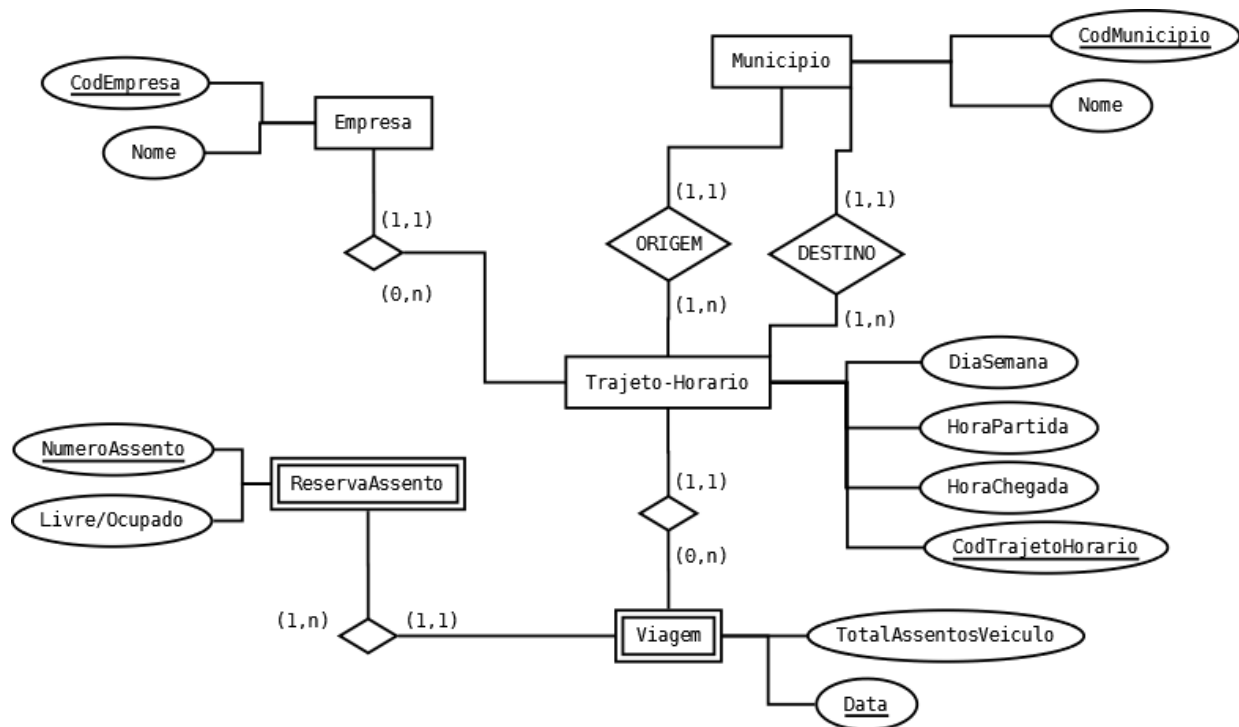
(g) Escreva uma instrução SQL para excluir todas as revisões dos automóveis do cliente “João Silva”. [1 ponto]

```
DELETE FROM Revisao
WHERE placa IN
    (SELECT placa
     FROM Automovel a, Cliente c
     WHERE c.cpf=a.cpf
     AND c.nome = “João Silva”)
```

**Questão 2** [3 pontos]

Considere o diagrama ER mostrado abaixo, relativo a um sistema de controle reservas de passagens rodoviárias. O sistema controla as reservas de várias empresas de ônibus.

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). As entidades representadas com linhas duplas (Viagem e ReservaAssento) são entidades fracas.



Empresa (CodEmpresa, Nome)

Municipio (CodMunicipio, Nome)

Trajeto-Horario(CodTrajetoHorario, CodEmpresa, CodMunicipioOrigem, CodMunicipioDestino, DiaSemana, HorarioPartida, HorarioChegada)

CodEmpresa referencia Empresa

CodMunicipioOrigem referencia Municipio

CodMunicipioDestino referencia Municipio

Viagem (CodTrajetoHorario, Data, TotalAssentosVeiculo)

CodTrajetoHorario referencia Trajeto-Horario

ReservaAssento (CodTrajetoHorario, Data, NumeroAssento, Livre/Ocupado)

(CodTrajetoHorario, Data) referencia Viagem