



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina Banco de Dados

AP2 2º semestre de 2006.

Nome –

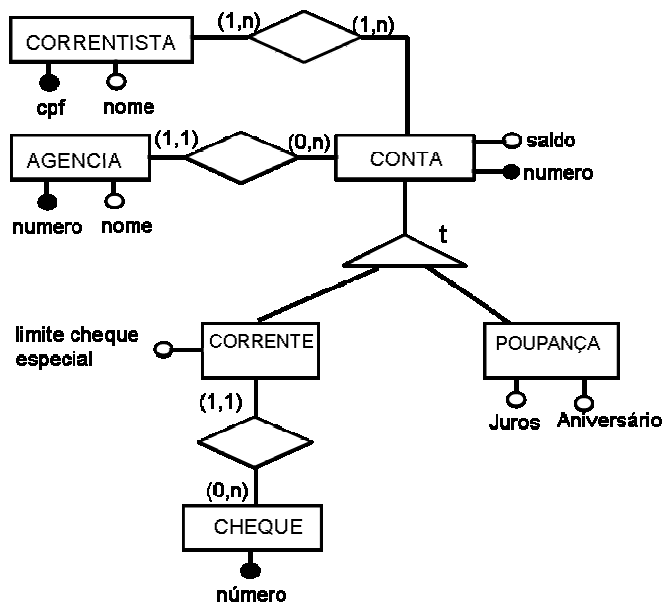
Assinatura –

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 diagrama ER mostrado abaixo. Construa um esquema relacional equivalente a este diagrama ER. [2,5 pontos]



Correntista (CPF, Nome)
 ContaCorrentista (CPF, NumConta)
 CPF referencia Correntista
 NumConta referencia Conta
 Agencia (NumAgencia, Nome)
 Conta (NumConta, NumAgencia, Saldo, **Tipo**)
 NumAgencia referencia Agencia
 ContaCorrente (NumConta, Limite)
 NumConta referencia Conta
 ContaPoupanca (NumConta, Juros, Aniversario)
 NumConta referencia Conta
 Cheque (NumConta, Numero)
 NumConta referencia ContaCorrente

Apesar de na prática fazer sentido duas contas terem o mesmo número em duas agências diferentes, na realidade modelada neste diagrama ER isso não acontece, pois a entidade conta não foi modelada como entidade fraca. Portanto, a chave primária de Conta é somente NumConta, e não (NumConta, NumAgencia).

Resposta alternativa: hierarquia de especialização/generalização em tabela única.

Correntista (CPF, Nome)
 ContaCorrentista (CPF, NumConta)
 CPF referencia Correntista
 NumConta referencia Conta
 Agencia (NumAgencia, Nome)
 Conta (NumConta, NumAgencia, Saldo, **Tipo**, Limite, Juros, Aniversario)
 NumAgencia referencia Agencia
 Cheque (NumConta, Numero)
 NumConta referencia Conta

Questão 2

Considere o esquema relacional abaixo. As chaves primárias estão sublinhadas.

Cliente (<u>CodCli</u> , nome, idade) Produto (<u>CodProd</u> , descrição) Pedido (<u>CodCli</u> , <u>numPedido</u> , data) CodCli referencia Cliente ItemPedido (<u>numPedido</u> , <u>numItem</u> , CodProd, Quant, PrecoUnit) numPedido referencia Pedido CodProd referencia Produto

- (a) Escreva os comandos SQL para criar as tabelas *ItemPedido*, *Produto* e *Pedido*, incluindo as restrições de integridade que se aplicam. Assuma que, ao excluir um pedido, todos os itens de pedido relacionados devem ser excluídos automaticamente. Assuma também que um produto não pode ser excluído caso

exista algum item de pedido relacionado a ele. Além disso, ao alterar o código de um produto, todos os itens de pedido relacionados devem ser alterados automaticamente. Em um pedido, assumo que a quantidade não pode ser nula. Se, por algum motivo, não for possível definir alguma restrição de integridade, justifique. [1 ponto]

```
CREATE TABLE PRODUTO (  
    CODPROD INT NOT NULL,  
    DESCRICAO VARCHAR(30),  
    PRIMARY KEY (CODPROD)  
)
```

```
CREATE TABLE PEDIDO (  
    CODCLI INT,  
    NUMPEDIDO INT NOT NULL,  
    DATA DATE,  
    PRIMARY KEY (NUMPEDIDO),  
)
```

```
CREATE TABLE ITEMPEDIDO (  
    NUMPEDIDO INT NOT NULL,  
    NUMITEM INT NOT NULL,  
    CODPROD INT,  
    QUANT FLOAT NOT NULL,  
    PRECUNIT FLOAT,  
    PRIMARY KEY (NUMPEDIDO, NUMITEM),  
    FOREIGN KEY (NUMPEDIDO) REFERENCES PEDIDO (NUMPEDIDO) ON  
DELETE CASCADE,  
    FOREIGN KEY (CODPROD) REFERENCES PRODUTO (CODPROD) ON  
DELETE RESTRICT ON UPDATE CASCADE  
)
```

É possível definir todas as restrições de integridade.

- (b) Escreva comandos SQL para incluir um novo pedido no banco de dados, com os seguintes dados: Pedido número 123, do cliente 02 (assumo que o cliente já está cadastrado). Os itens do pedido são: [0,5 ponto]
- 2 unidades do produto de código 02, custando 1,00 cada unidade
 - 3 unidades do produto de código 03, custando 2,30 cada unidade

```
INSERT INTO PEDIDO (CODCLI, NUMPEDIDO) VALUES (02, 123);
```

```
INSERT INTO ITEMPEDIDO (NUMPEDIDO, NUMITEM, CODPROD, QUANT,  
PRECUNIT) VALUES (123, 1, 02, 2, 1);
```

```
INSERT INTO ITEMPEDIDO (NUMPEDIDO, NUMITEM, CODPROD, QUANT,
PRECOUNT) VALUES (123, 2, 03, 3, 2.30);
```

- (c) Escreva um comando SQL para excluir todos os pedidos do cliente João da Silva.
[1 ponto]

```
DELETE FROM PEDIDO
WHERE CODCLI IN (SELECT CODCLI
                  FROM CLIENTE
                  WHERE NOME="João da Silva")
```

Não é necessário excluir de ItemPedido, devido à DELETE CASCADE.

Questão 3

Considere o esquema relacional da questão 2. Apresente o código SQL para as seguintes consultas:

- (a) Faça uma consulta que seleciona os nomes dos clientes com mais de 30 anos que fizeram algum pedido entre 01/01/2000 e 01/01/2002. [0,5 ponto]

```
SELECT NOME
FROM CLIENTE C, PEDIDO P
WHERE C.CODCLI = P.CODCLI
AND    IDADE > 30
AND    DATA >= 01/01/2000
AND    DATA <= 01/01/2002
```

- (b) Faça uma consulta que retorna o valor total do pedido número 111. [0,5 ponto]

```
SELECT SUM (QUANT * PRECUNIT)
FROM ITEMPEDIDO
WHERE NUMPEDIDO = 111
```

- (c) Crie uma visão que tenha o nome do cliente e o número de pedidos que aquele cliente já fez. [0,5 ponto]

```
CREATE VIEW CLIENTE_PEDIDO (NOME, TOTALP)
AS
    SELECT NOME, COUNT (*)
    FROM CLIENTE C, PEDIDO P
    WHERE C.CODCLI = P.CODCLI
    GROUP BY NOME
```

- (d) Faça uma consulta que retorna o nome do cliente e a descrição de produtos que o cliente pediu em quantidade superior a 10 unidades. [0,5 ponto]

```
SELECT NOME, DESCRICAO
FROM CLIENTE C, PEDIDO P, ITEMPEDIDO I, PRODUTO D
```

```

WHERE C.CODCLI = P.CODCLI
AND    P. NUMPEDIDO = I. NUMPEDIDO
AND    I. CODPROD = D. CODPROD
AND    QUANT > 10

```

(e) Mostre o número do pedido, o código do produto e a quantidade solicitada de produtos cuja descrição contenha a palavra “novidade”. [0,5 ponto]

```

SELECT NUMPEDIDO, CODPROD, QUANT
FROM   ITEMPEDIDO I, PRODUTO D
WHERE  P. NUMPEDIDO = I. NUMPEDIDO
AND    I. CODPROD = D. CODPROD
AND    DESCRICAO LIKE '%novidade%'

```

Questão 4

Considere o esquema relacional da questão 2. Apresente a expressão em álgebra relacional para as seguintes consultas:

(a) Faça duas expressões algébricas equivalentes que selecionam os nomes dos clientes com mais de 30 anos que fizeram algum pedido entre 01/01/2000 e 01/01/2002. [1 ponto]

$\pi_{nome} (\sigma_{idade > 30} \text{ Cliente}) \bowtie (\sigma_{DATA \geq 01/01/2000 \wedge DATA \leq 01/01/2002} \text{ Pedido})$
 $\pi_{nome} (\sigma_{DATA \geq 01/01/2000 \wedge DATA \leq 01/01/2002} \text{ Pedido}) \bowtie (\sigma_{idade > 30} \text{ Cliente})$
 $\pi_{nome} \sigma_{idade > 30} (\text{ Cliente } \bowtie \sigma_{DATA \geq 01/01/2000 \wedge DATA \leq 01/01/2002} \text{ Pedido})$

(b) Apresente a expressão algébrica da consulta que mostra os nomes dos clientes que não possuem pedidos registrados. [1 ponto]

$\rho(R1, \pi_{nome}(\text{Cliente}))$
 $\rho(R2, \pi_{nome}(\text{Cliente} \bowtie \text{Pedido}))$
 $R1 - R2$

(c) Mostre a expressão que apresenta a descrição de produtos que constam de algum pedido. [0,5 ponto]

$\pi_{descricao} (\text{Produto} \bowtie \text{Itempedido})$