

niversidade do Sul de Santa Catarina – UNISUL  
 urso de Ciência da Computação  
 ampus Sul - Disciplina de Fundamentos de Banco de Dados  
 AVALIAÇÃO – 2016/1

LUNO: Tiago Beling

1) Construa um modelo E-R nível conceitual devidamente normalizado segundo as formas normais coradas na disciplina para o enunciado abaixo:

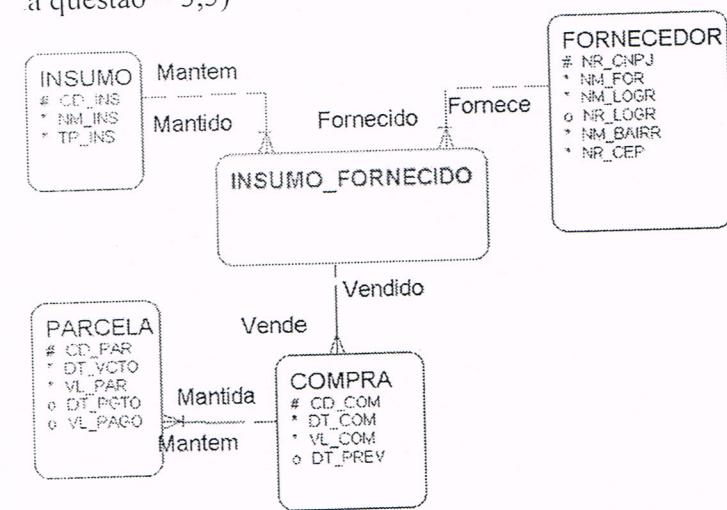
ma clínica médica necessita manter o cadastro de seus médicos, das especialidades médicas dos médicos da clínica, bem como um controle do número de consultas realizadas por cada médico no mês. Para os médicos, regista-se obrigatoriamente o CRM, nome e CPF. Opcionalmente pode-se registrar vários endereços (rua, cidade, estado, cep) e um telefone para este endereço, onde o médico pode ser localizado.

Para as especialidades médicas é registrado obrigatoriamente o nome da especialidade. Cada médico possui apenas uma especialidade, porém uma especialidade pode ser a qualificação de vários especialistas da clínica.

O controle de consultas realizadas mensalmente é bastante simples. Apenas são revisadas as consultas realizadas no mês e são informadas obrigatoriamente o ano, o mês, o médico e o número de consultas total que o médico realizou naquele mês. Opcionalmente pode-se registrar alguma observação para esclarecer os valores das consultas de um médico no mês.

Valor da questão = 2,5)

2) Dado o modelo de dados abaixo, descrito a nível conceitual, realize o devido mapeamento para o nível lógico. (Valor da questão = 3,5)

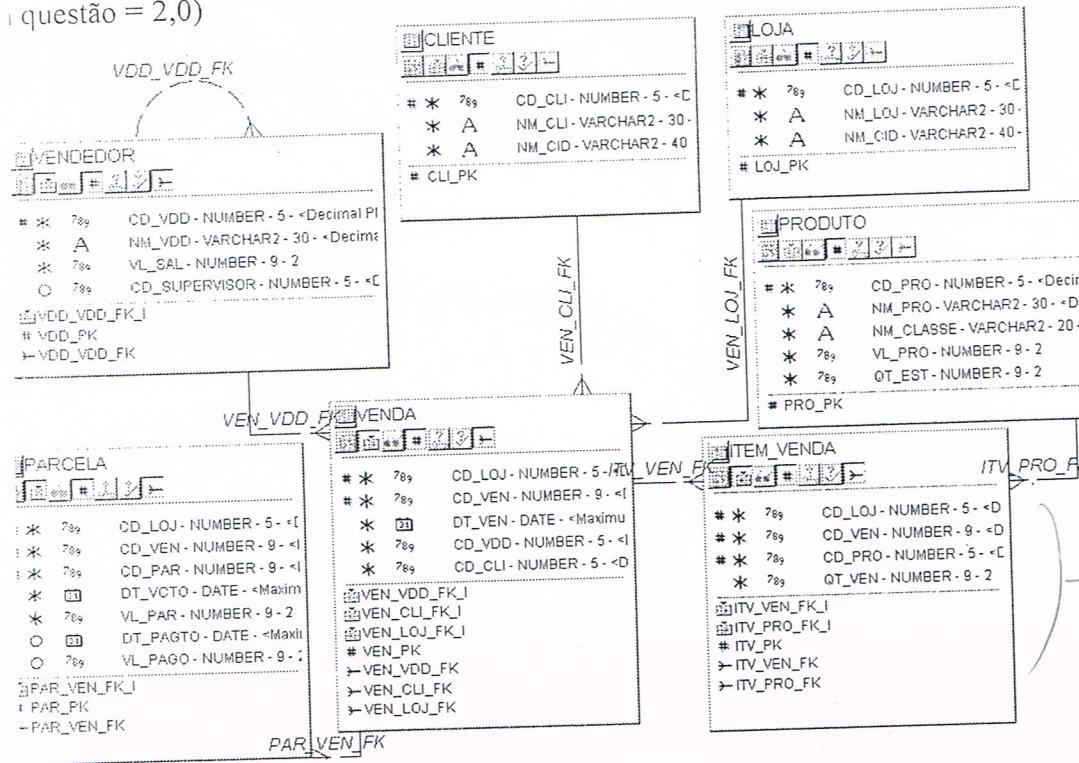


= chaves PK  
 vai junto e  
 compõem

DBS.: TP\_INS é um domínio com as opções “PRODUTO”, “SERVIÇO”. Mapear como constraint tipo CHECK.  
 As Colunas DT\_VCTO, DT\_COM e NM\_FOR são muito visadas para consultas e devem ser indexadas.

TP\_INS = CHECK

Dado o modelo de dados abaixo, descrito a nível lógico, descreva a sua implementação e das constraints que cabem no nível físico para as tabelas VENDA e ITEM\_VENDA. As demais tabelas não precisam ser implementadas. (Valor da questão = 2,0)



cd-loj  
cd-ven  
cd-pro  
qt-ven

4) Tendo como base o modelo de dados apresentado na questão 3, responda: Uma vez implementado o modelo de dados num SGBD, existem restrições que impeçam que a tabela VENDA seja EXCLUÍDA do projeto do banco de dados? Justifique sua resposta. (Valor da questão = 1,0)

5) Tendo como base o modelo de dados apresentado na questão 3, responda: (Valor da questão = 1,0)

- 40**
- Supondo que não existam dados cadastrados na base de dados, construa a(s) sentença(s) SQL de forma que se consiga inserir uma linha na tabela item\_venda.
  - Apresente a sentença SQL para que se consiga atualizar o nome do vendedor código 5 para 'JOAQUIM PEDROZA'.

UPDATE vendedor SET nm\_vdd = "JOAQUIM PEDROZA" WHERE cd\_vdd = 5;

4-Sim pois VENDA tem ITEM\_VENDA e PARCELA que dependem dela para operar, chaves estrangeiras impedem a exclusão

5-0) INSERT INTO vendedor (cd\_vdd, nm\_vdd, vl\_sal, cd\_supervisor) VALUES

5-a) INSERT INTO vendedor (cd\_vdd, nm\_vdd, vl\_sal, cd\_supervisor) VALUES (1, "Juca", 1200, 0);

INSERT INTO loja (cd\_loj, nm\_loj, nm\_cid) VALUES (1, "Loja", "Tubarão");

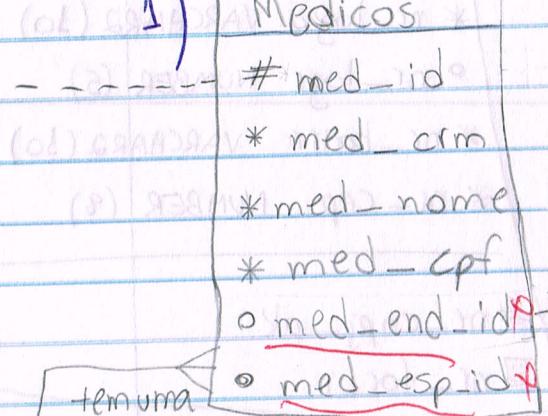
INSERT INTO cliente (cd\_cli, nm\_cli, nm\_cid) VALUES (1, "Tiago", "Armazém");

INSERT INTO produto (cd\_pro, nm\_pro, nm\_classe, vl\_pro, qt\_est) VALUES (1, "Tênis Verde", "Sapatos", 100, 41);

INSERT INTO venda (cd\_loj, cd\_ven, dt\_ven, cd\_vdd, cd\_cli) VALUES (1, 1, "2016-05-06", 1, 1);

INSERT INTO item\_venda (cd\_loj, cd\_ven, cd\_pro, qt\_ven) VALUES (1, 1, 1, 4);

INSERT INTO parcela (cd\_loj, cd\_ven, cd\_pro, dt\_vcto, vl\_par, vl\_pago, cd\_cli, cd\_vdd, dt\_pagto) VALUES (1, 1, 1, "2016-05-06", 100, 100, 1, 1, 1);



especialidades

# esp\_id

\* esp\_nome

realiza

2) INSUMO

```
# cd-ins INT(11) PK  
* nm-ins NUMBER(5)  
* tp-ins BOOLEAN("P","S")  
  
# cd-ins-pk CK?
```

fornecedor

```
# nr-cnpj NUMBER(15)  
* nr-for VARCHAR2(10)  
* nr-logr VARCHAR2(10)  
* nr-logr NUMBER(5)  
* nr-bairr VARCHAR2(10)  
* nr-cep NUMBER(8)
```

PARCELA

- # cd-par NUMBER (5)
- \* dt-vcto DATE
- \* v1-par NUMBER (10)
- o dt-pgto DATE
- o v1-pago NUMBER (10)
- #\* cd-com ? --

---

# cd-par-fk

→ par-com-fk

par-com-fk\_i

dt-vcto\_i

insumo\_fornecedor

# nc\_cnpj

# cd\_ins

\* qtd\_venda

---

# item\_pk

→ fornecedor\_fk

→ ins\_fornecedor\_fk

fornecedor\_fk\_i

ins\_fornecedor\_fk\_i

**COMPRA**

- # cd\_com NUMBER(5)
- \* dt\_com DATE
- \* vl\_com FLOAT(2,5) *(P-  
recurs)*
- o dt\_prev DATE

---

# cd\_com\_pk

→ ins\_fornecedor\_fk

ins\_fornecedor\_fk\_i

dt\_com\_i

3) CREATE TABLE VENDA ( cd\_loj NUMBER(5), cd\_ven NUMBER(5), dt\_ven DATE, NOT NULL, cd\_vdd NUMBER(5) NOT NULL, cd\_cli NUMBER(5) NOT NULL, PRIMARY KEY (cd\_ven));

CREATE TABLE item\_venda ( cd\_loj NUMBER(5) NOT NULL, cd\_ven NUMBER(9) NOT NULL, cd\_pro NUMBER(5) NOT NULL, qt\_ven NUMBER(9) NOT NULL );

```
ALTER TABLE venda ADD CONSTRAINT ven_vdd_fk FOREIGN KEY  
    (cd_vdd) REFERENCES vendedor (cd_vdd);  
  
ALTER TABLE venda ADD CONSTRAINT ven_cli_fk  
FOREIGN KEY (cd_cli) REFERENCES cliente (cd_cli);  
  
ALTER TABLE venda ADD CONSTRAINT ven_loj_fk  
FOREIGN KEY (cd_loj) REFERENCES loja (cd_loj);  
  
CREATE INDEX ven_vdd_fk_i ON venda (cd_vdd);  
CREATE INDEX ven_cli_fk_i ON venda (cd_cli);  
CREATE INDEX ven_loj_fk_i ON venda (cd_loj);
```

ALTER TABLE item-venda ADD CONSTRAINT itv-ven-fk  
FOREIGN KEY (cd-ven) REFERENCES venda (cd-ven);  
CREATE INDEX itv-ven-fk\_i ON item-venda (cd-ven);  
CREATE INDEX itv-pro-fk\_i ON item-venda (cd-pro);

ITV-8 ~~Ma~~ ?  
VGN-PK ?