

Colégio Politécnico da UFSM – Sistemas para Internet
Prova 2 – 2018/1 - Data: 29/06/2018 (90% da nota da segunda avaliação)

Nome: _____

1) Demonstre passo a passo a aplicação de cada uma das formas normais (1,5)

Projetos de Pesquisa

Cód. professor: 001 Nome professor: Fulano de Tal

Cód. projeto	Nome projeto	Número de bolsistas do projeto	Horas do professor no projeto
100	Projeto100	25	3h
200	Projeto200	50	1h

Cód. professor: 002 Nome professor: Beltrano de Tal

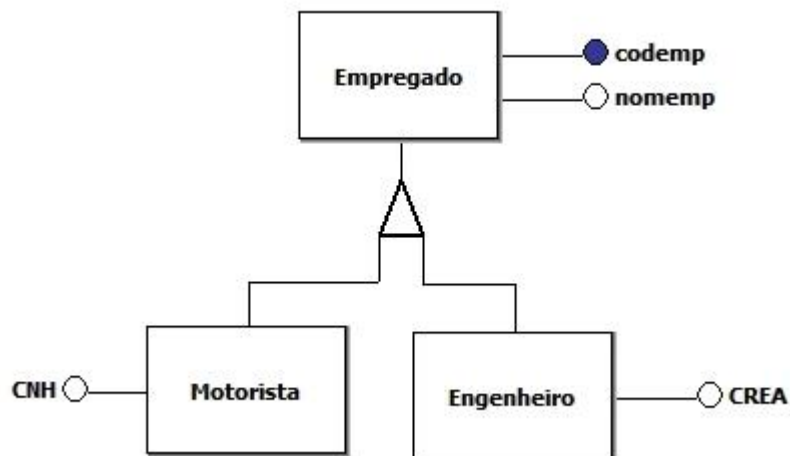
Cód. projeto	Nome projeto	Número de bolsistas do projeto	Horas do professor no projeto
100	Projeto100	25	1h
200	Projeto200	50	3h

Cód. professor: 003 Nome professor: Siclano de Tal

Cód. projeto	Nome projeto	Número de bolsistas do projeto	Horas do professor no projeto
100	Projeto100	25	2h
300	Projeto300	40	5h

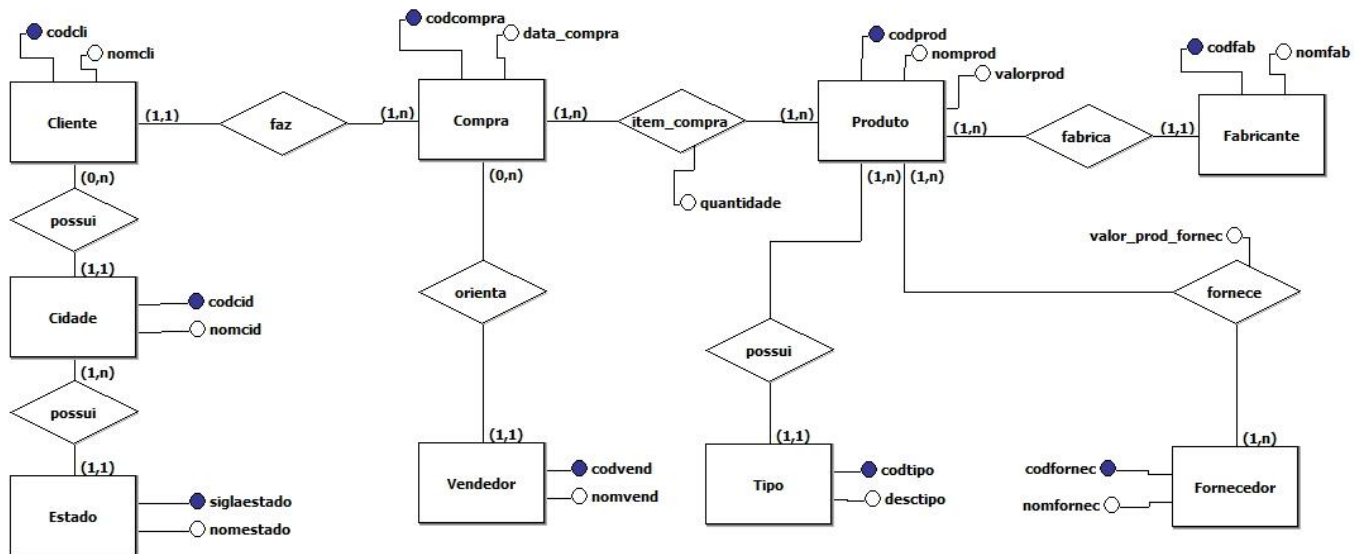
Nome: _____

- 2) Considere o modelo a seguir, que mostra uma estrutura de generalização/especialização. A partir dele, usando SQL, crie uma tabela para cada entidade que compõe a hierarquia (1,0).



Nome: _____

A partir do modelo E-R abaixo, resolva as questões 3 e 4.



```
CREATE TABLE Estado
(siglaestado char(2),
nomeestado varchar(50),
primary key (siglaestado));
```

```
CREATE TABLE Cidade
(codcid int,
nomcid varchar(50),
siglaestado char(2),
primary key(codcid),
foreign key (siglaestado) references Estado(siglaestado));
```

```
CREATE TABLE Cliente
(codcli int,
nomcli varchar(100),
codcid int,
primary key (codcli),
foreign key (codcid) references Cidade(codcid));
```

```
CREATE TABLE Tipo
(codtipo int,
nomtipo varchar(50),
primary key(codtipo));
```

```
CREATE TABLE Fabricante
(codfab int,
nomfab varchar(50),
primary key (codfab));
```

```
CREATE TABLE Vendedor
(codvend int,
nomvend varchar(100),
primary key(codvend));
```

```
CREATE TABLE Produto
(codprod int,
nomprod varchar(100),
valorprod numeric(12,2),
codfab int,
codtipo int,
primary key (codprod),
foreign key (codfab) references Fabricante (codfab),
foreign key (codtipo) references Tipo (codtipo));
```

```
CREATE TABLE Compra
(codcompra int,
data_compra date,
codcli int,
codvend int,
primary key(codcompra),
foreign key (codcli) references Cliente(codcli),
foreign key(codvend) references Vendedor(codvend));
```

```
CREATE TABLE Item_Compra
(codcompra int,
codprod int,
quantidade int,
primary key(codcompra,codprod),
foreign key (codprod) references Produto(codprod),
foreign key(codcompra) references Compra(codcompra));
```

```
CREATE TABLE Fornecedor
(codfornec int,
nomfornec varchar(100),
primary key(codfornec));
```

```
CREATE TABLE Fornece
(codfornec int,
codprod int,
valor_prod_fornec numeric(14,2),
primary key(codfornec,codprod),
foreign key (codprod) references Produto(codprod),
foreign key(codfornec) references Fornecedor(codfornec));
```

Nome: _____

3) Gatilhos/Funções

Como regra de negócio, ficou estabelecido que um produto não pode ser fornecido por mais de 4 fornecedores. A partir disto, escreva a(s) função(ões) e gatilho(s) para evitar que exista na base o registro de produtos fornecidos por mais de 4 fornecedores (1,0).

```
CREATE TRIGGER nome_gatilho AFTER update or insert or delete
ON tabela FOR EACH ROW
EXECUTE PROCEDURE procedimento_funcao ();

CREATE FUNCTION procedimento_função () returns tipo_retorno
AS $$
    DECLARE variável tipo;
BEGIN
    código...
    IF condição...
        THEN RAISE EXCEPTION 'mensagem';
    END IF;
    RETURN NEW;
END;
$$
LANGUAGE 'plpgsql';
```

Nome: _____

4) Use SQL ou Álgebra Relacional, conforme solicitado, para:

- a) Listar os nomes dos fornecedores que NÃO fornecem produtos do tipo denominado “BRINQUEDOS” (1,0) (SQL).

- b) Listar os nomes dos clientes com os nomes dos produtos que eles compraram. Ordenar o resultado pelo nome do cliente (1,0) (SQL).

- c) Listar os códigos e nomes dos vendedores que orientaram mais de 4 compras (1,5) (SQL).

- d) Listar os nomes dos tipos de produto com os nomes dos respectivos produtos. Aqui liste inclusive os nomes dos tipos que não possuam produtos cadastrados (1,5) (SQL).

- e) Criar uma view que contenha os nomes das cidades com os nomes dos clientes relacionados a estas cidades (1,0) (SQL)

- f) Listar os nomes dos fabricantes com os nomes dos produtos fabricados por eles (0,5) (Álgebra Relacional)