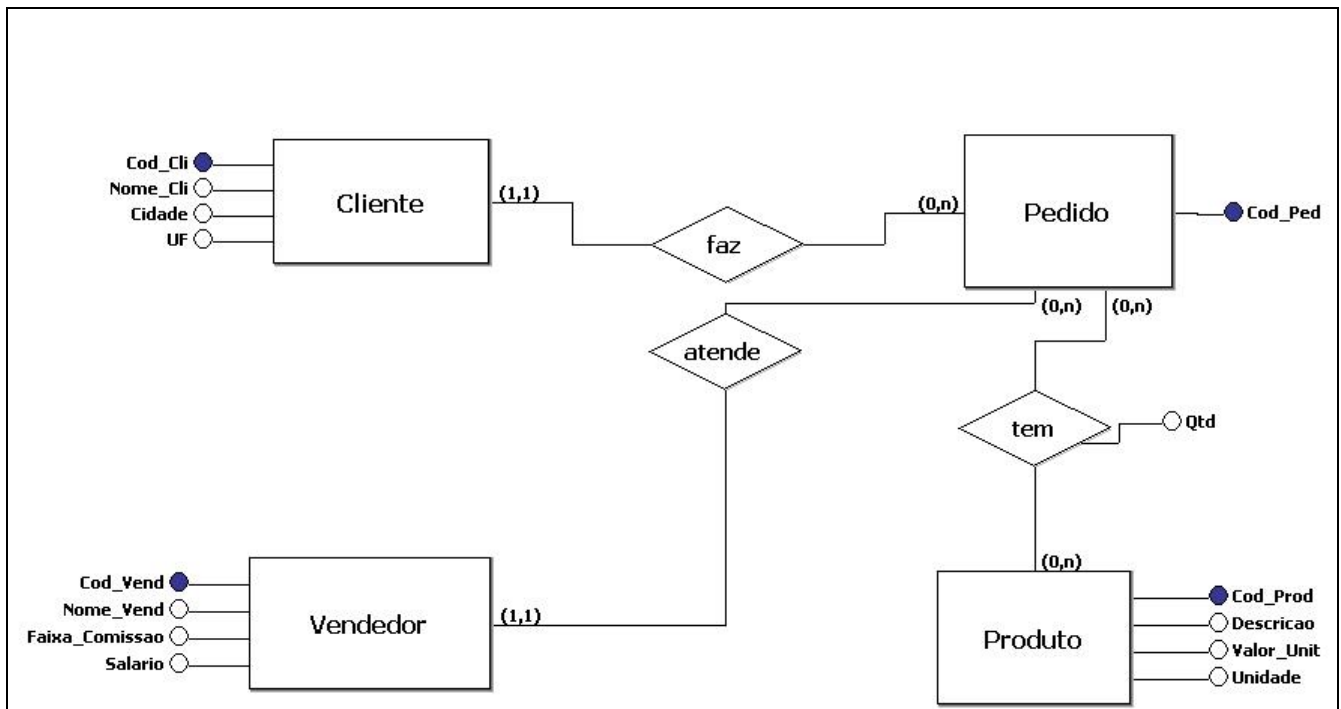


Exercícios da Aula 20 – *Stored Procedures*

Considere o seguinte Banco de Dados “Pedido” para responder as questões deste exercício, utilizando a Linguagem SQL (o script de criação deste banco de dados se encontra no final deste arquivo).

MER



Modelo Relacional

Cliente = {Cod_Cli, Nome_Cli, Cidade, UF}

Vendedor = {Cod_Vend, Nome_Vend, Salario, Faixa_Comissão}

Pedido = {Cod_Ped, Cod_Cli, Cod_Vend}

– Cod_Cli é chave estrangeira que referencia o atributo Cod_Cli da tabela Cliente

- Cod_Vend é chave estrangeira que referencia o atributo Cod_Vend da tabela Vendedor

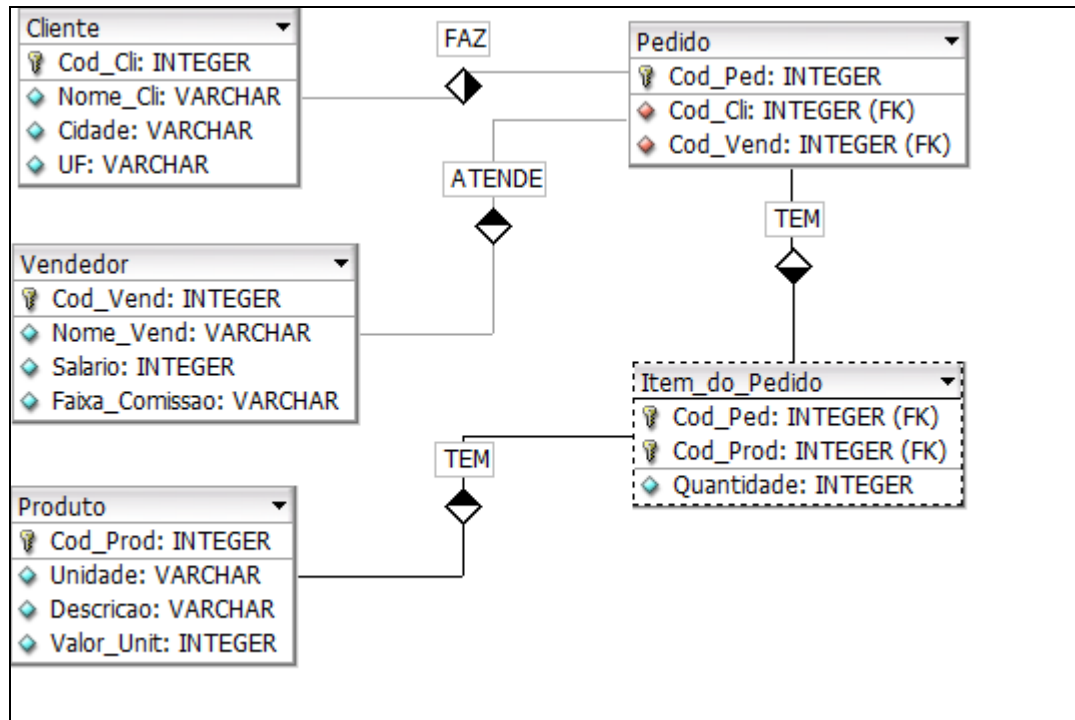
Produto = {Cod_Prod, Unidade, Descricao, Valor_Unit}

Item_do_Pedido = {Cod_Ped, Cod_Prod, Quant}

– Cod_Ped é chave estrangeira que referencia o atributo Cod_Ped da tabela Pedido

- Cod_Prod é chave estrangeira que referencia o atributo Cod_Prod da tabela Produto.

Modelo Relacional no DBDesigner



Tabelas populadas com dados

Cliente				Vendedor			
Cod_Cli	Nome_Cli	Cidade	UF	Cod_Vend	Nome_Vend	Salario	Faixa_Comissao
1	Ana	Niteroi	RJ	1000	Jose	1800	B
2	Flavio	Sao Paulo	SP	1001	Carlos	2500	A
3	Jorge	Belo Horizonte	MG	1002	Joao	2700	C
4	Lucia	Sorocaba	SP	1003	Antonio	4600	C
5	Mauro	Sao Paulo	SP	1004	Jonas	9500	A
				1005	Mateus	3000	C

Pedido			Produto			
Cod_Ped	Cod_Cli	Cod_Vend	Cod_Prod	Unidade	Descricao	Valor_Unit
100	5	1001	200	kg	queijo	10
101	1	1002	201	kg	chocolate	20
102	3	1004	202	l	vinho	30
103	2	1002	203	kg	acucar	2
104	1	1005	204	m	papel	2
105	5	1002				

Item_do_Pedido		
Cod_Ped	Cod_Prod	Quantidade
100	201	3
100	202	5
101	204	15
102	203	5
103	200	12
104	201	1
104	203	4
104	204	6
105	202	10

1) Crie uma *procedure* que **atualize a quantidade vendida de um determinado produto em um pedido específico**. Serão passados, como argumentos, a descrição do produto, o código do pedido deste produto, e, também, o valor da nova quantidade deste produto que deverá ser atualizada.

Depois da procedure criada, utilize-a para atualizar a quantidade vendida do produto 'acucar' do pedido 102 para 250 quilos.

2) Crie uma *procedure* que **atualize o salário de um determinado vendedor**, baseado no número de pedidos que eles atendeu e seguindo os seguintes critérios:

- se ele não atendeu nenhum pedido ainda, ele não terá aumento;
- se ele atendeu até 2 pedidos, ele terá 10% de aumento no seu salário;
- se ele atendeu mais de 2 pedidos, ele terá 20% de aumento no seu salário.

Será passado, como **argumento para esta procedure, somente o nome do vendedor** (que já deve existir na tabela Vendedor).

Depois da procedure criada, utilize-a para atualizar (se for o caso) o salário do vendedor 'Mateus'.

3) Crie uma *procedure* que **insira um novo cliente, um novo pedido para este cliente inserido**, sendo que **neste pedido deverá ter um único produto comprado em uma determinada quantidade**. Os argumentos para esta *procedure* serão:

- para os clientes deverão ser passados o nome, cidade e uf (você deverá criar um valor válido de chave primária para a tabela Cliente);
- para o pedido deverá ser passado o nome do vendedor (o vendedor já deve existir no banco de dados) que atendeu aquele cliente, a descrição do produto comprado neste pedido (o produto já deve existir no banco de dados) e a quantidade comprada deste produto (você deverá criar um valor válido para a chave primária da tabela Pedido).

Depois da procedure criada, utilize-a para inserir você como um novo cliente, sendo que no seu pedido você comprou 20 quilos de 'chocolate' e quem te atendeu foi o vendedor 'Carlos'.

Universidade Presbiteriana Mackenzie
Banco de Dados
Profa. Elisângela Botelho Gracias

-- SCRIPT COMPLETO DO BANCO

```
Drop table Item_do_Pedido;
Drop table Produto;
Drop table Pedido;
Drop table Vendedor;
Drop table Cliente;
```

```
Create table Cliente(
Cod_Cli integer,
Nome_Cli varchar(30),
Cidade varchar(30),
UF char(2),
Primary key(Cod_cli));
```

```
Create table Vendedor (
Cod_Vend integer,
Nome_Vend varchar(30),
Salario integer,
Faixa_comissao char(1),
Primary key (Cod_Vend));
```

```
Create table Pedido (
Cod_Ped integer,
Cod_Cli integer,
Cod_Vend integer,
Primary key(Cod_Ped),
Foreign key (Cod_Cli) References Cliente (Cod_Cli),
Foreign key (Cod_Vend) References Vendedor (Cod_Vend));
```

```
Create table Produto(
Cod_Prod integer,
Unidade varchar(10),
Descricao varchar(30),
Valor_Unit integer,
Primary key(Cod_Prod));
```

```
Create table Item_do_Pedido (
Cod_Ped integer,
Cod_Prod integer,
Quantidade integer,
Primary key(Cod_Ped, Cod_Prod),
Foreign key (Cod_Ped) References Pedido (Cod_Ped),
Foreign key (Cod_Prod) References Produto (Cod_Prod));
```

```
Insert into Cliente values (1, 'Ana', 'Niteroi', 'RJ');
Insert into Cliente values (2, 'Flavio', 'Sao Paulo', 'SP');
Insert into Cliente values (3, 'Jorge', 'Belo Horizonte', 'MG');
Insert into Cliente values (4, 'Lucia', 'Sorocaba', 'SP');
Insert into Cliente values (5, 'Mauro', 'Contagem', 'MG');
```

```
Insert into Vendedor values (1000, 'Jose', 1800, 'C');
Insert into Vendedor values (1001, 'Carlos', 2500, 'A');
Insert into Vendedor values (1002, 'Joao', 2700, 'C');
Insert into Vendedor values (1003, 'Antonio', 4600, 'C');
Insert into Vendedor values (1004, 'Jonas', 9500, 'A');
Insert into Vendedor values (1005, 'Mateus', 3000, 'C');
```

```
Insert into Pedido values (100, 5, 1001);
Insert into Pedido values (101, 1, 1002);
Insert into Pedido values (102, 3, 1004);
Insert into Pedido values (103, 2, 1002);
Insert into Pedido values (104, 1, 1005);
Insert into Pedido values (105, 5, 1002);
```

Universidade Presbiteriana Mackenzie
Banco de Dados
Profa. Elisângela Botelho Gracias

```
Insert into Produto values (200, 'kg','queijo', 10);  
Insert into Produto values (201, 'kg','chocolate', 20);  
Insert into Produto values (202, 'l','vinho', 30);  
Insert into Produto values (203, 'kg','acucar', 2);  
Insert into Produto values (204, 'm','papel', 2);
```

```
Insert into Item_do_Pedido values (100, 201,3);  
Insert into Item_do_Pedido values (100, 202,5);  
Insert into Item_do_Pedido values (101, 204,15);  
Insert into Item_do_Pedido values (102, 203,5);  
Insert into Item_do_Pedido values (103, 200,12);  
Insert into Item_do_Pedido values (104, 201,1);  
Insert into Item_do_Pedido values (104, 203,4);  
Insert into Item_do_Pedido values (104, 204,6);  
Insert into Item_do_Pedido values (105, 202,10);
```