

Vamos imaginar que nós queremos armazenar no nosso banco de dados os produtos que utilizamos em nossa loja, os dados dos nossos vendedores e as vendas efetuadas por esses vendedores, como criaríamos esse banco de dados.

Para atendermos essa demanda seria necessária a criação de três tabelas. A primeira para armazenar os dados de cada produto, a segunda para armazenar os dados dos vendedores e a terceira para armazenar os dados da venda em si.

O primeiro CREATE TABLE cria justamente a tabela responsável por armazenar as informações de todos os produtos. Nesse script nós temos um atributo chamado de id. Esse atributo é do tipo integer (capaz de armazenar números inteiros). Esse atributo é AUTOINCREMENT. Ou seja ele será criado pelo próprio banco de dados começando pelo valor 1 e incrementando automaticamente em uma unidade a cada registro inserido. Além disso, esse atributo é uma PRIMARY KEY (chave primária). Isso significa dizer que não haverá nenhum registro nessa tabela produto com o mesmo id. Dessa forma, o atributo id serve para identificar unicamente cada produto dentro da tabela. Esse será o campo utilizado para referenciar os registros dessa tabela com registros de outras tabelas do banco. Percebam que todas as três tabelas possuirão esse atributo.

O segundo atributo da tabela produto é o atributo nome. Esse atributo é do tipo varchar. Isso significa dizer que ele é capaz de armazenar campos alfanuméricos com 255 caracteres.

O terceiro atributo é o atributo quantidade. Ele serve para armazenar a quantidade disponível no estoque de um determinado produto. Ele também é do tipo integer, assim como o atributo id.

O quarto e último atributo da tabela produto é o atributo valor. Ele será utilizado para armazenar quanto custa um determinado produto. Por isso, ele é do tipo float. Isso significa dizer que ele pode conter ponto flutuante, ou seja números com casas decimais.

```
CREATE TABLE produto
(
  id integer PRIMARY KEY AUTOINCREMENT,
  nome varchar(255),
  quantidade integer,
  valor float
);
```

O segundo CREATE TABLE será utilizado para criar a tabela responsável por armazenar os dados dos vendedores da loja. Não falaremos sobre o atributo id, pois será redundante, como já falamos na criação da tabela produto. O segundo atributo é atributo nome. Como o próprio nome já diz, ele será utilizado para armazenar os nomes dos vendedores da loja. Por isso ele é do tipo varchar.

O terceiro e o quarto atributo também são do tipo varchar e eles servem para armazenar respectivamente as matrículas e os e-mails dos vendedores da loja.

```
CREATE TABLE vendedor
(
    id integer PRIMARY KEY AUTOINCREMENT,
    nome varchar(255),
    matricula varchar(255),
    email varchar(255)
);
```

O terceiro e último CREATE TABLE será utilizado para criar a tabela responsável por armazenar as vendas propriamente ditas. Ela possui o atributo id, assim como nas demais tabelas. Além disso, ela possui o atributo quantidade, do tipo inteiro, que será utilizado para armazenar a quantidade daquele produto específico que está sendo vendido. O terceiro atributo é do tipo float (números com casas decimais) e ele serve para armazenar o valor pelo qual o produto em questão estão sendo vendido. Isso porque o valor da venda pode ser diferente do valor do produto em si, visto que algum desconto pode ter sido concedido. O quarto e o quinto atributos são do tipo integer (inteiro) e eles servem para armazenar respectivamente os ids (identificadores) do produto que está sendo vendido e do vendedor que está efetuando a venda. Por esse motivo, é utilizado nesse CREATE TABLE a criação das CONSTRAINTs responsável por fazer a ligação entre essa tabela (venda) e as demais tabelas (produto e vendedor).

Para tanto definimos da seguinte forma: CONSTRAINT seguida do nome da constraint que estamos criando (pode ser qualquer nome, mas é interessante utilizarmos um nome que identifique facilmente quais tabelas serão relacionadas. No nosso caso fk_venda_produto e fk_venda_vendedor), seguida da expressão FOREIGN KEY (porque se trata de uma chave estrangeira), seguida do atributo da tabela venda que será utilizado para relacionar essa tabela com outra entre parênteses (no nosso caso foram os atributos id_produto e id_vendedor), seguida da palavra REFERENCES, seguida do nome da tabela a qual essa chave estrangeira fará ligação, seguida do atributo utilizado para fazer a ligação entre parênteses.

```
CREATE TABLE venda
(
    ID integer PRIMARY KEY AUTOINCREMENT,
    quantidade integer,
    valor float,
    id_produto integer,
    id_vendedor integer,
    CONSTRAINT fk_venda_produto FOREIGN KEY (id_produto) REFERENCES produto(id),
    CONSTRAINT fk_venda_vendedor FOREIGN KEY (id_vendedor) REFERENCES
vendedor(id)
);
```

Essa expressão “CONSTRAINT fk_venda_produto FOREIGN KEY (id_produto) REFERENCES produto(id)” serve portanto, para informar ao banco de dados que o atributo id_produto da tabela venda será utilizado para relacionar o registro em questão com o registro da tabela produto através do atributo id.