

### Entidades:

1. Produtos
2. Clientes
3. Pedidos
4. Fornecedores

### Relacionamentos:

- Um Cliente pode fazer vários Pedidos.
- Um Pedido contém vários Produtos.
- Cada Produto pode estar em vários Pedidos.
- Cada Produto é fornecido por um Fornecedor.

Agora, vou modelar as tabelas em 3FN.

### Tabelas:

1. Produtos (ProdutoID, Nome, Preço, FornecedorID)
2. Clientes (ClienteID, Nome, Email)
3. Pedidos (PedidoID, ClienteID, DataPedido)
4. Fornecedores (FornecedorID, Nome, Contato)

### Informações de teste:

- Produtos: (1, 'Camiseta', 50.00, 1), (2, 'Calça Jeans', 100.00, 2)
- Clientes: (1, 'João Silva', 'joao@email.com'), (2, 'Maria Souza', 'maria@email.com')
- Pedidos: (1, 1, '2024-06-19'), (2, 2, '2024-06-20')
- Fornecedores: (1, 'Fornecedor A', 'contato@fornecedorA.com'), (2, 'Fornecedor B', 'contato@fornecedorB.com')

**Consultas SQL simples:** Para juntar tabelas e ver quais produtos um cliente pediu:

```
SELECT Clientes.Nome AS ClienteNome, Produtos.Nome AS ProdutoNome
FROM Clientes
JOIN Pedidos ON Clientes.ClienteID = Pedidos.ClienteID
JOIN Produtos ON
```

Terceira Forma Normal (3FN):

1. **Produtos** (ProdutoID, Nome, Preço, FornecedorID)
  - Cada coluna depende apenas da chave primária (ProdutoID).
  - Não há dependências parciais ou transitivas.
2. **Clientes** (ClienteID, Nome, Email)
  - Cada coluna depende apenas da chave primária (ClienteID).
  - Não há dependências parciais ou transitivas.
3. **Pedidos** (PedidoID, ClienteID, DataPedido)
  - Cada coluna depende apenas da chave primária (PedidoID).
  - ClienteID é uma chave estrangeira que cria um relacionamento com a tabela Clientes, mas não cria dependências transitivas.
4. **Fornecedores** (FornecedorID, Nome, Contato)
  - Cada coluna depende apenas da chave primária (FornecedorID).
  - Não há dependências parciais ou transitivas.