**Introduction**

Design a database schema for an online bookstore, "Book Haven," which aims to manage a wide variety of books, authors, customers, orders, and reviews. The goal is to create a database that efficiently supports the operations of the bookstore, including inventory management, customer order processing, and handling customer reviews and ratings.

**Requirements**

**1. Books:**

* Each book has a unique ISBN, title, publication year, price, and genre.
* A book can have one or more authors.
* Books need to track their stock quantity.

**2. Authors:**

* Each author has a unique ID, name, biography, and a list of books they have authored.

**3. Customers:**

* Customers have a unique ID, name, email, shipping address, and phone number.
* Customers can place orders and write reviews.

**4. Orders:**

* Each order has a unique order ID, customer ID, order date, and a list of books ordered with quantities.
* The order should also track the status (e.g., pending, shipped, delivered).

**5. Reviews:**

* Each review has a unique review ID, book ISBN, customer ID, rating (1-5), and comment.
* Reviews need to be timestamped.

**6. Publishers:**

* Each publisher has a unique ID, name, and contact information.
* Each book is published by one publisher.

**Database Schema**

**Tables and Columns:**

1. **Books**
   * ISBN (Primary Key)
   * Title
   * PublicationYear
   * Price
   * Genre
   * PublisherID (Foreign Key referencing Publishers)
   * StockQuantity
2. **Authors**
   * AuthorID (Primary Key)
   * Name
   * Biography
3. **Customers**
   * CustomerID (Primary Key)
   * Name
   * Email
   * ShippingAddress
   * PhoneNumber
4. **Orders**
   * OrderID (Primary Key)
   * CustomerID (Foreign Key referencing Customers)
   * OrderDate
   * Status
5. **OrderDetails** (junction table for many-to-many relationship between Orders and Books)
   * OrderID (Foreign Key referencing Orders)
   * ISBN (Foreign Key referencing Books)
   * Quantity
   * Price (to handle price at the time of order)
6. **Reviews**
   * ReviewID (Primary Key)
   * ISBN (Foreign Key referencing Books)
   * CustomerID (Foreign Key referencing Customers)
   * Rating
   * Comment
   * Timestamp
7. **Publishers**
   * PublisherID (Primary Key)
   * Name
   * ContactInfo
8. **BookAuthors** (junction table for many-to-many relationship between Books and Authors)
   * ISBN (Foreign Key referencing Books)
   * AuthorID (Foreign Key referencing Authors)

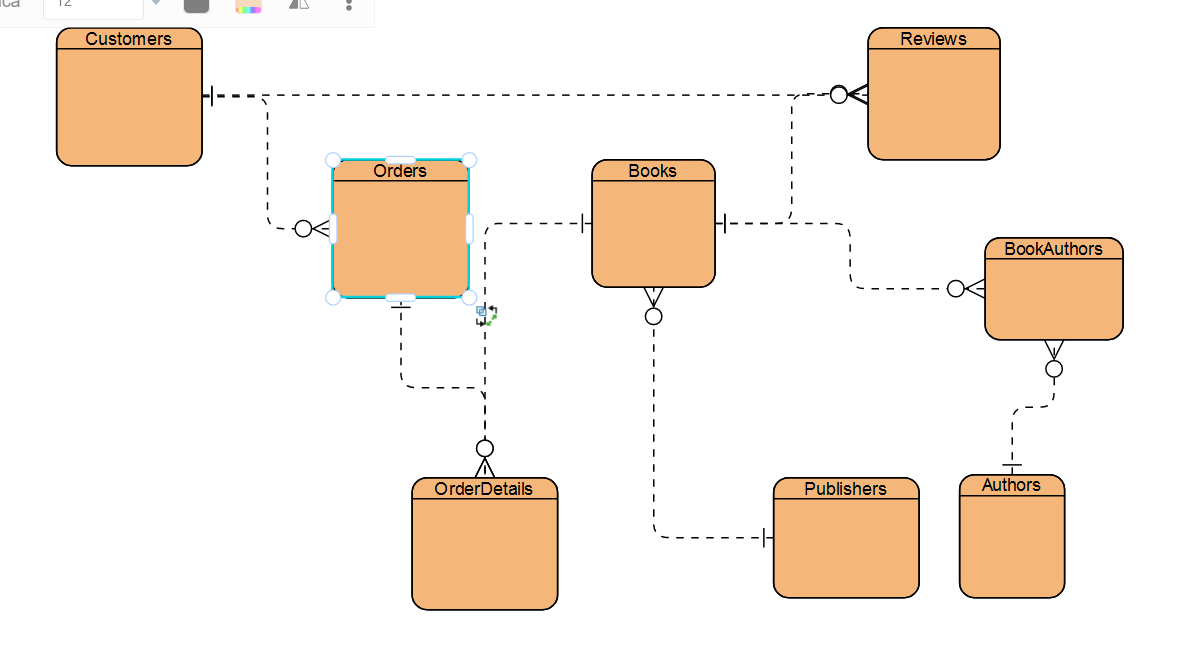
 **Books and Authors:** Books and authors have a many-to-many relationship. A junction table, BookAuthors, is used to manage this relationship.

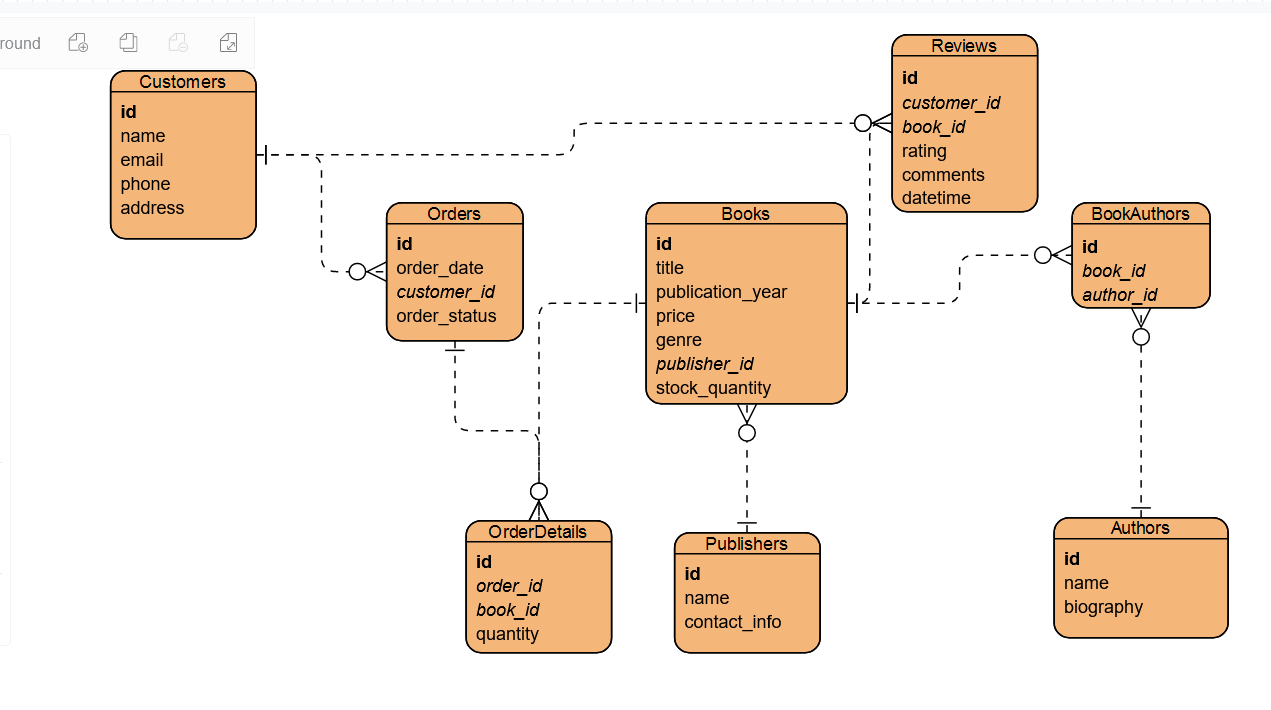
 **Orders and Books:** Orders can include multiple books, and each book can be part of multiple orders. The OrderDetails table manages this many-to-many relationship and also stores the quantity of each book in an order.

 **Reviews:** Reviews are linked to both customers and books, indicating that a customer can review multiple books and each book can have multiple reviews.

 **Publishers:** Each book is associated with one publisher, while a publisher can publish multiple books.

 **Customers and Orders:** A customer can place multiple orders, but each order is associated with only one customer.





CREATE TABLE customers(

id INT PRIMARY KEY AUTO\_INCREMENT,

NAME VARCHAR(200),

email VARCHAR(200),

phone VARCHAR (20),

address TEXT

);

CREATE TABLE orders(

id INT PRIMARY KEY AUTO\_INCREMENT,

order\_date DATE,

customer\_id INT,

order\_status VARCHAR(30)

);

CREATE TABLE books(

id INT PRIMARY KEY AUTO\_INCREMENT,

title VARCHAR(200),

publication\_year INT,

price INT,

genre VARCHAR(100),

publisher\_id INT,

stock\_quantity INT

);

CREATE TABLE `authors`( id INT PRIMARY KEY AUTO\_INCREMENT, `name` VARCHAR(200), biography TEXT );

CREATE TABLE `book\_authors`( id INT PRIMARY KEY AUTO\_INCREMENT, book\_id INT, author\_id INT );

CREATE TABLE `publishers`( id INT PRIMARY KEY AUTO\_INCREMENT, `name` VARCHAR(200), contact\_info TEXT );

CREATE TABLE `order\_details`(

id INT PRIMARY KEY AUTO\_INCREMENT,

order\_id INT,

book\_id INT,

quantity INT

);

CREATE TABLE `reviews`(

id INT PRIMARY KEY AUTO\_INCREMENT,

customer\_id INT,

book\_id INT,

rating INT,

comments TEXT,

date\_time DATETIME

);