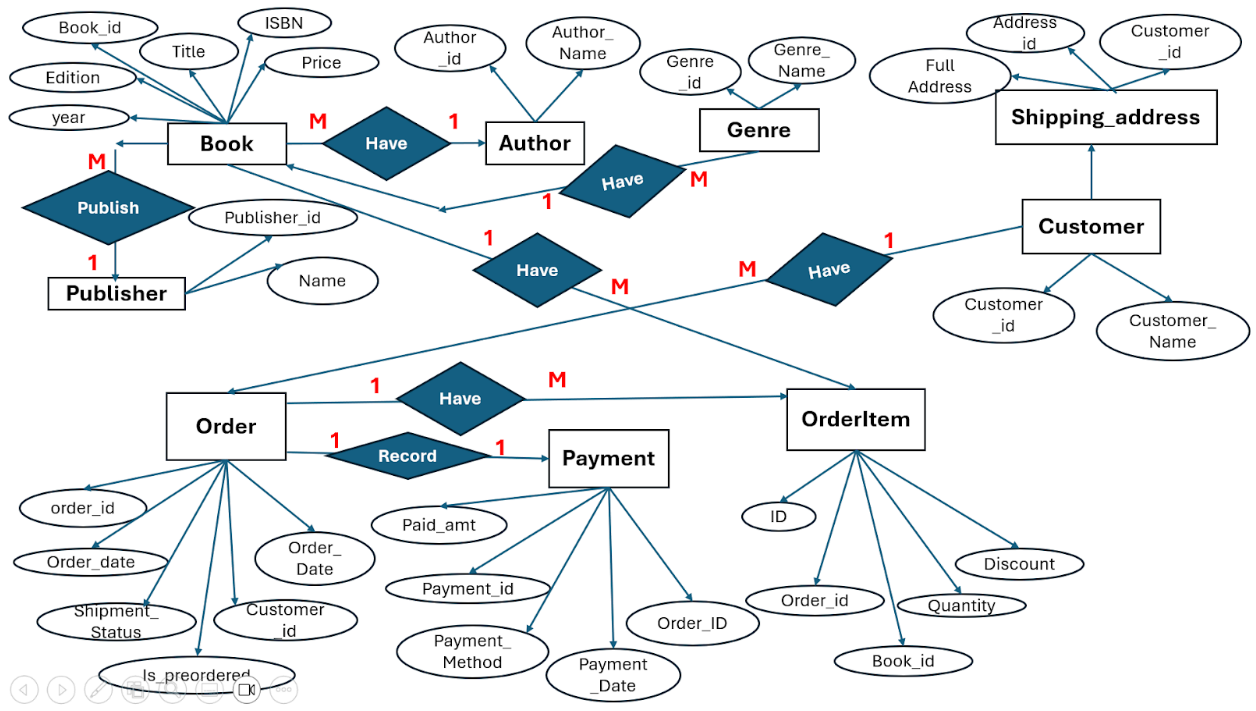
Problem 3: Online Book Publishing and Sales Platform

Design an Entity-Relationship schema for an online book publishing and sales platform. The database should contain information about books with title, ISBN, edition, publication year, publisher, genres, and price. Authors have ID, name, biography, and are associated with multiple books.
Customers have customer ID, name, purchase history, shipping addresses, and wishlist items. Orders have order number, order date, customer placing the order, list of books ordered with quantity and per item discounts, payment details, and shipment status.

Publishers have names, contact details, and the books they publish. Books can be written by multiple authors and can belong to multiple genres. Customers can place multiple orders, have multiple shipping addresses, and maintain a wishlist of books.
Each edition of a book is published by exactly one publisher, and books can have multiple editions sold in different years. Orders can contain multiple books with different quantities and item-specific discounts. Assume scenarios such as co-authored books, special editions, and pre-order capabilities.



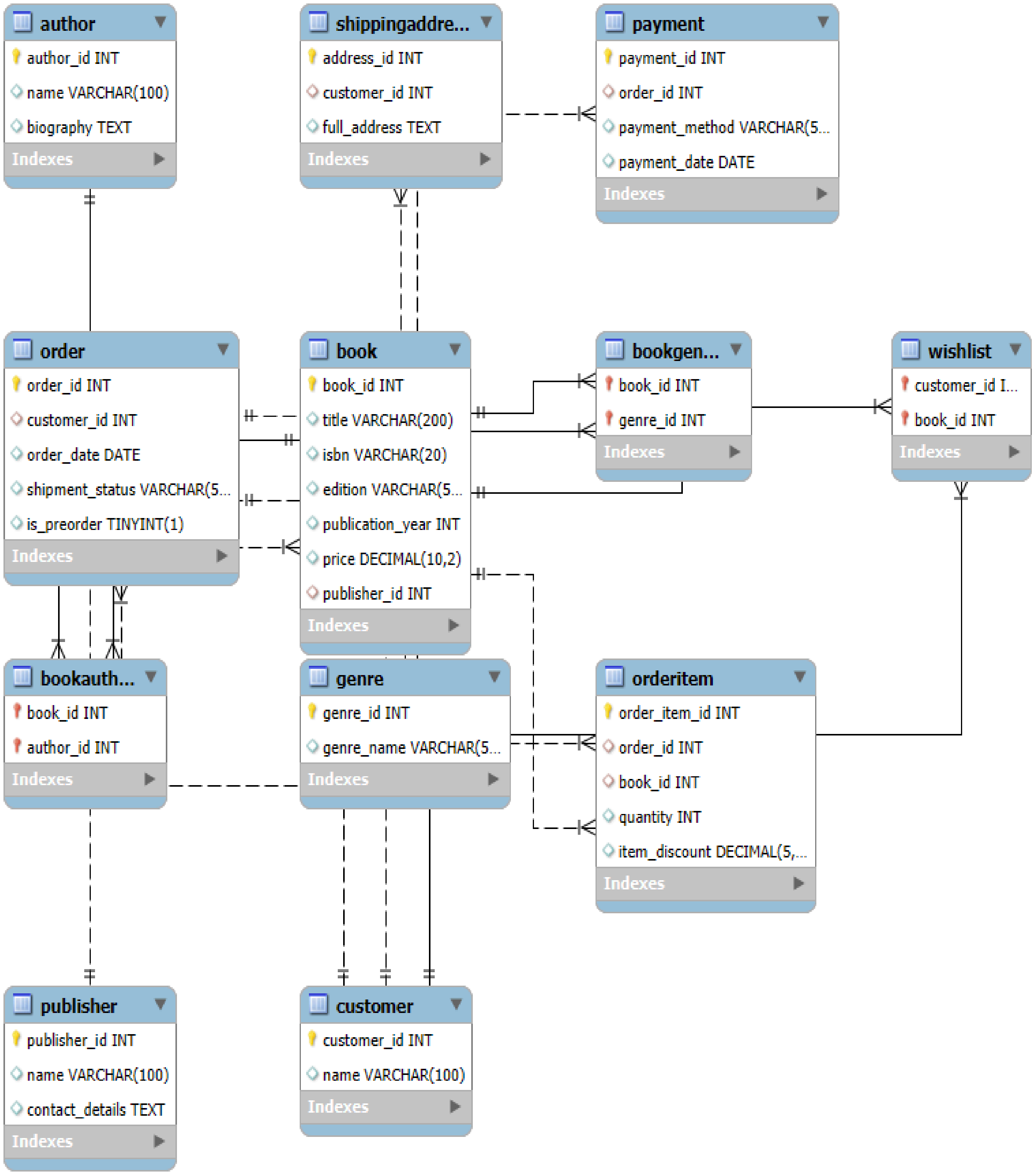
-- Author table

```
CREATE TABLE Author (
   author_id INT PRIMARY KEY AUTO_INCREMENT,
   name VARCHAR(100),
    biography TEXT
);

-- Publisher table
CREATE TABLE Publisher (
    publisher_id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(100),
    contact_details TEXT
);

-- Genre table
CREATE TABLE Genre (
    genre_id INT PRIMARY KEY AUTO_INCREMENT,
    genre_name VARCHAR(50) UNIQUE
);

-- Book Edition table (includes ISBN + Edition + Year combo as unique)
CREATE TABLE Book (
    book_id INT PRIMARY KEY AUTO_INCREMENT,
    title VARCHAR(200),
    isbn VARCHAR(20),
    edition VARCHAR(50),
    publication_year INT,
    price DECIMAL(10, 2),
    publisher_id INT,
    FOREIGN KEY (publisher_id) REFERENCES Publisher(publisher_id)
);

-- Book-Author (Many-to-Many)
CREATE TABLE BookAuthor (
    book_id INT,
    author_id INT,
    PRIMARY KEY (book_id, author_id),
    FOREIGN KEY (book_id) REFERENCES Book(book_id),
    FOREIGN KEY (author_id) REFERENCES Author(author_id)
);

-- Book-Genre (Many-to-Many)
CREATE TABLE BookGenre (
    book_id INT,
    genre_id INT,
    PRIMARY KEY (book_id, genre_id),
    FOREIGN KEY (book_id) REFERENCES Book(book_id),
    FOREIGN KEY (genre_id) REFERENCES Genre(genre_id)
```

```
);

-- Customer table
CREATE TABLE Customer (
    customer_id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(100)
);

-- Customer Shipping Addresses (Multiple Addresses)
CREATE TABLE ShippingAddress (
    address_id INT PRIMARY KEY AUTO_INCREMENT,
    customer_id INT,
    full_address TEXT,
    FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)
);

-- Wishlist (Many-to-Many between Customer and Book)
CREATE TABLE Wishlist (
    customer_id INT,
    book_id INT,
    PRIMARY KEY (customer_id, book_id),
    FOREIGN KEY (customer_id) REFERENCES Customer(customer_id),
    FOREIGN KEY (book_id) REFERENCES Book(book_id)
);

-- Order table
CREATE TABLE `Order` (
    order_id INT PRIMARY KEY AUTO_INCREMENT,
    customer_id INT,
    order_date DATE,
    shipment_status VARCHAR(50),
    is_preorder BOOLEAN DEFAULT FALSE,
    FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)
);



-- Payment details
CREATE TABLE Payment (
    payment_id INT PRIMARY KEY AUTO_INCREMENT,
    order_id INT,
    payment_method VARCHAR(50),
    payment_date DATE,
    FOREIGN KEY (order_id) REFERENCES `Order`(order_id)
);
```

-- Order details (books in each order, quantity, per item discount)

```
CREATE TABLE OrderItem (
    order_item_id INT PRIMARY KEY AUTO_INCREMENT,
    order_id INT,
    book_id INT,
    quantity INT,
    item_discount DECIMAL(5,2), -- e.g., 10.00 means 10% discount
    FOREIGN KEY (order_id) REFERENCES `Order`(order_id),
    FOREIGN KEY (book_id) REFERENCES Book(book_id)
```


author
author_id INT
name VARCHAR(100)
biography TEXT
Indexes
shippingaddre...
address_id INT
customer_id INT
full_address TEXT
Indexes
payment
payment_id INT
order_id INT
payment_method VARCHAR(5...
payment_date DATE
Indexes
order
order_id INT
customer_id INT
order_date DATE
shipment_status VARCHAR(5...
is_preorder TINYINT(1)
Indexes
book
book_id INT
title VARCHAR(200)
isbn VARCHAR(20)
edition VARCHAR(5...
publication_year INT
price DECIMAL(10,2)
publisher_id INT
Indexes
bookgen...
book_id INT
genre_id INT
Indexes
wishlist
customer_id I...
book_id INT
Indexes
bookauth...
book_id INT
author_id INT
Indexes
genre
genre_id INT
genre_name VARCHAR(5...
Indexes
orderitem
order_item_id INT
order_id INT
book_id INT
quantity INT
item_discount DECIMAL(5,...
Indexes
publisher
publisher_id INT
name VARCHAR(100)
contact_details TEXT
Indexes
customer
customer_id INT
name VARCHAR(100)
Indexes