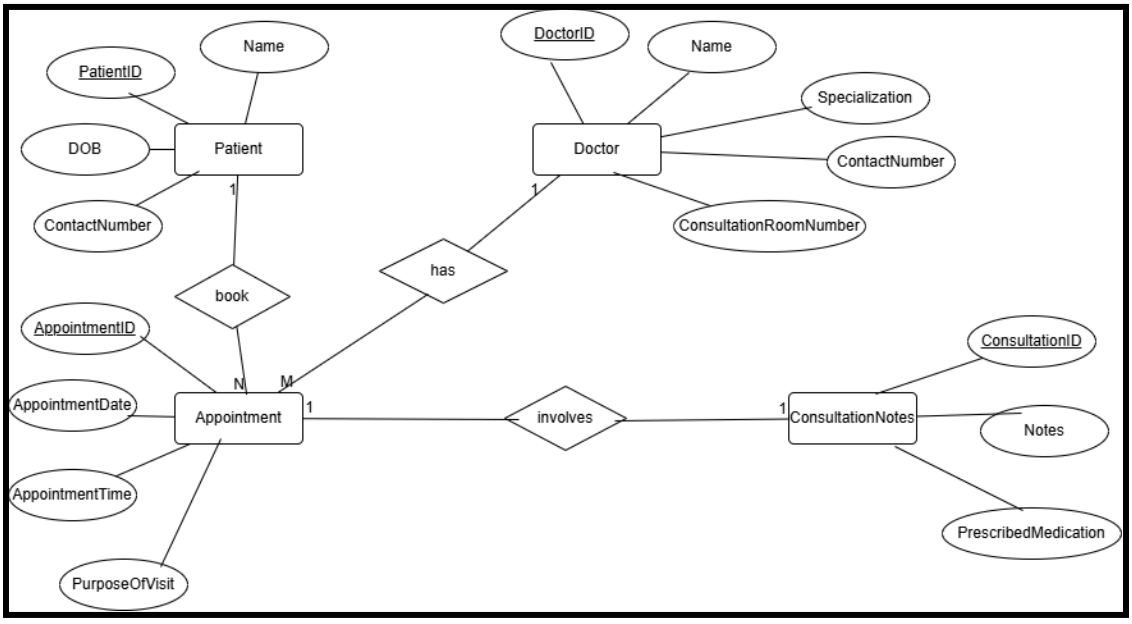
SCS1309/IS1210 Database Management

Practical - 03

Consider the below scenarios and the corresponding ER diagrams. Do the relational mapping.

Scenario 1
Hospital Appointment System

A hospital is implementing a system to manage appointments between patients and doctors. Each patient registers with their name, date of birth, contact number, and patient ID. Patients can book appointments with one or more doctors. Each doctor has a unique doctor ID, name, specialization, contact number, and consultation room number. An appointment is scheduled for a specific date and time and is linked to both the patient and the doctor. The system also tracks the purpose of the visit (e.g., routine check-up, follow-up, or emergency). After the consultation, the doctor can record notes about the patient's condition and prescribe medications if necessary.



Scenario 2

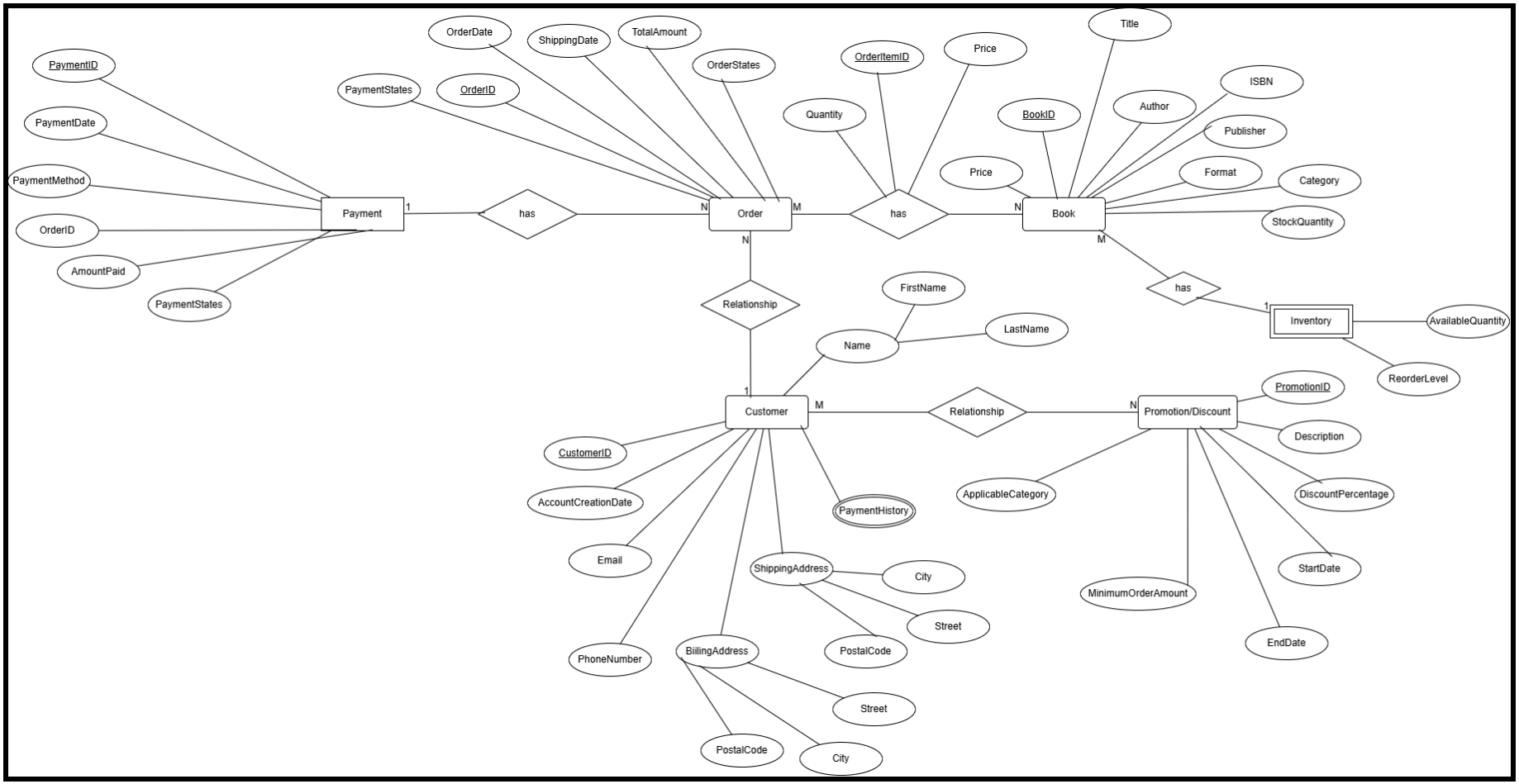
Designing a Database for "Book Haven" Online Bookstore. List the entities and attributes from the given scenario below.

"Book Haven" is a newly launched online bookstore that offers a diverse selection of books in genres ranging from fiction to nonfiction, science, history, and self-help. The website allows customers to browse books, add them to their shopping cart, and place orders. Each order can consist of multiple books, and customers can select from different formats, such as paperback or hardcover. Once an order is placed, the system tracks the status of the order and manages delivery details to ensure timely shipments.

Customers can create accounts on the website to streamline the ordering process. Each customer is required to provide essential contact information and a billing address. They can track their order history, save their preferred shipping addresses, and receive updates about their orders. The platform also offers various promotional discounts that customers can apply to their purchases. These promotions can be based on order totals or specific book categories, with limited-time validity.

The bookstore needs to maintain an up-to-date inventory system to reflect the available stock of each book. Whenever a book is ordered, the system must check stock availability and ensure that the quantity is updated accordingly. If a book is out of stock, the customer should be notified before they complete their purchase. Additionally, the system must handle different payment statuses, including processing payments and keeping a record of paid and pending transactions.

As the business grows, "Book Haven" plans to expand its product offerings and customer base, so the database must be designed to handle a large volume of orders, manage customer data securely, and efficiently track the availability of books and promotional offers. The design of the database will play a crucial role in maintaining the operational flow of the online bookstore and providing a seamless shopping experience for customers.


PaymentID
PaymentDate
PaymentMethod
OrderID
AmountPaid
PaymentStates
Payment
1
has
N
OrderDate
ShippingDate
TotalAmount
OrderStates
PaymentStates
OrderID
Order
M
has
N
Quantity
OrderItemID
Price
Book
Title
ISBN
Author
BookID
Publisher
Price
Format
Category
StockQuantity
N
Relationship
1
M
has
1
Inventory
AvailableQuantity
ReorderLevel
FirstName
LastName
Name
Customer
M
Relationship
N
Promotion/Discount
PromotionID
Description
DiscountPercentage
StartDate
EndDate
MinimumOrderAmount
ApplicableCategory
CustomerID
AccountCreationDate
Email
PhoneNumber
PaymentHistory
ShippingAddress
City
Street
PostalCode
BillingAddress
Street
City
PostalCode

Scenario 3

You are a database designer hired by a startup that operates an online food delivery platform. The platform aims to provide a seamless experience connecting customers, restaurants, and delivery agents, offering a robust infrastructure for managing orders, payments, and deliveries.

Customers are required to create an account on the platform by providing their name, email, and phone number. To ensure security, customers must set a password, which can later be reset using a verification process. Customers can save multiple delivery addresses categorized as "Home," "Office," or "Other," each consisting of details such as street name, apartment number, city, and postal code. They can update or delete these addresses at any time. Additionally, customers have a wallet feature to store credits or promotional discounts they receive from the platform. Each customer also has a purchase history that tracks all past orders with details such as order date, time, and amount.

Restaurants can register on the platform by providing their name, contact details, license number, and location. Each restaurant manages its own menu, which includes several items categorized under sections like "Starters," "Main Course," "Desserts," and "Beverages." Each menu item has a name, description, price, and availability status (e.g., available, out of stock). Restaurants can update their menu items as needed, including adding daily specials or seasonal items. In addition to offering dine-in, some restaurants provide discounts exclusively for online orders, which customers can apply during checkout. Restaurants are also rated by customers on the platform, with an aggregated rating displayed to help new customers decide where to order.

Customers place orders by selecting items from the menu of a specific restaurant. Each order includes the customer’s selected delivery address, the list of ordered items, their respective quantities, and the total price, which is automatically calculated based on the price of the selected items. Customers can choose from multiple payment methods, such as online payments via credit/debit cards, digital wallets, or cash on delivery.

Customers also have the option to leave special instructions for the restaurant, such as "No onions" or "Extra spicy," and for the delivery agent, such as "Call upon arrival."

Orders are timestamped, capturing the date and time when they were placed, and customers are given a unique Order ID for reference. The platform automatically assigns each order to a delivery agent based on factors like proximity to the restaurant and availability. Customers can track the real-time status of their order, such as "Order Accepted," "Being Prepared," "Out for Delivery," or "Delivered." Once the order is

completed, customers are encouraged to leave feedback, which helps improve the service and maintain quality.

Delivery agents are an integral part of the platform. To join, they must provide their name, phone number, vehicle details (e.g., type, model, registration number), and upload proof of identity. Each delivery agent is assigned a unique Agent ID. They are notified about new delivery assignments through the platform's app, which includes details such as pickup location, drop-off address, and the estimated time to complete the delivery. Delivery agents are rated after each delivery, and their overall performance metrics are tracked, including the number of deliveries completed, average delivery time, and customer feedback.

To enhance customer engagement, the platform also includes a rewards program, where customers earn points for every order. These points can be redeemed for discounts or free items. Customers are notified about promotions or special offers via email or in-app notifications. The platform also supports group orders, allowing multiple customers to contribute to a shared order, useful for office lunches or family gatherings.

The system ensures all transactions and activities are securely logged, enabling administrators to perform analytics on customer preferences, restaurant performance, and delivery efficiency. This data helps the startup optimize its operations and offer personalized recommendations to customers.