**题目：**

Design and implement a relational database for an online shopping website. The

basic requirements for the shopping website database are: it should record

customers and products information, as well as purchase/order information and

payment details.

**一、实体（Entities）**：

1. **Customer（顾客）**
   * **解释**：存储所有顾客的详细信息，包括顾客ID（主键，使用INT类型以节省存储空间和提高搜索效率）、用户名、密码、名字、姓氏、电话号码、电子邮件和地址。
   * **介绍**：这个实体允许网站跟踪和管理顾客信息，以支持用户注册、登录、修改个人资料和查看订单历史等功能。
2. **Seller（商家）**
   * **解释**：存储所有卖家的详细信息，包括卖家ID（主键，使用INT类型）和卖家名称。
   * **介绍**：这个实体用于网站跟踪和管理商家的信息，以支持商家注册、创建用户账户、描述其产品类别和用户评价等功能。
3. **Product（产品）**
   * **解释**：（在文档中未直接给出表结构，但根据上下文理解）应该存储所有产品的详细信息，可能包括产品ID、产品名称、描述、价格、库存量等。
   * **介绍**：这个实体是网站的核心，用于展示和管理销售的所有产品，确保顾客能够看到并购买所需的产品。
4. **ShoppingCart（购物车）**
   * **解释**：（在文档中未直接给出表结构，但根据上下文理解）用于记录顾客添加到购物车中的产品信息，包括购物车ID（主键）和产品ID。
   * **介绍**：这个实体允许顾客浏览网站时选择产品并添加到购物车中，准备进行购买。
5. **Order（订单）**
   * **解释**：记录整个订单相关的信息，包括订单ID（可能为主键）、顾客ID（外键）、订单状态等。
   * **介绍**：这个实体允许网站跟踪和管理顾客的订单，确保订单能够准确、及时地被处理。
6. **OrderDetail（订单明细）**
   * **解释**：记录订单中每个单独商品的具体信息，包括订单明细ID（可能为主键）、订单ID（外键）、产品ID（外键）、购买数量等。
   * **介绍**：这个实体为订单提供了更详细的信息，包括顾客购买了哪些产品、购买的数量等，有助于商家和顾客准确了解订单内容。
7. **Payment（付款信息）**
   * **解释**：记录付款信息，包括付款ID（可能为主键）、订单ID（外键）、付款金额、付款方式等。
   * **介绍**：这个实体允许网站跟踪和管理订单的付款信息，确保支付过程的安全和准确。
8. **Coupon（优惠券信息）**
   * **解释**：记录优惠券信息，包括优惠券ID（主键）、优惠券的条件和有效期等。
   * **介绍**：这个实体允许网站提供优惠券功能，吸引顾客购买产品，同时验证优惠券的有效性和适用条件。
9. **Customer（存所有的顾客）**
   * customerId (PK) INT
   * userName VARCHAR(20)
   * password VARCHAR(20)
   * fName VARCHAR(15)
   * sName VARCHAR(15)
   * phoneNumber INT
   * email VARCHAR(45)
   * address TEXT(100)
10. **Seller（存所有的卖家）**
    * sellerId (PK) INT
    * sellerName VARCHAR(45)
    * description TEXT(255)
11. **Product(存所有的商品)**
    * productId (PK) INT
    * productName VARCHAR(45)
    * description TEXT(255)
    * price INT
    * stockQuantity INT
    * category VARCHAR(45)
    * salesVolume INT
    * sellerId (FK) INT
12. **Shopping****Cart（存所有的顾客的购物车）**
    * cartId (PK) INT
    * customerId (FK) INT
13. **Order（存所有的订单信息）**
    * orderId (PK) INT
    * orderDate DATETIME(3)
    * totalPrice INT
    * paymentStatus VARCHAR(45)
    * shippingAddress TEXT(100)
    * createdAt TIMESTAMP(3)
    * customerId(FK) INT
14. **OrderDetail（存每个订单的具体信息，从Order里访问）**
    * orderDetailId (PK) INT
    * quantity INT
    * priceAtPurchase (记录购买时的价格，以防之后产品价格变动) INT
    * orderId (FK，引用Order表)  INT
    * productId (FK，引用Product表) INT
15. **Payment（存所有的支付信息）**
    * paymentId (PK) INT
    * paymentMethod (例如：信用卡、微信支付、数字人民币等) VARCHAR(45)
    * transactionId (支付网关或银行提供的事务ID) VARCHAR(45)
    * paymentDate DATETIME(3)
    * amount (支付金额) INT
    * updatedAt TIMESTAMP(3)
    * orderId (FK，引用Order表) INT
16. **Coupon（优惠券）**
    * couponId (PK) INT
    * productId INT
    * discount FLOAT
    * startDate DATETIME(3)
    * endDate DATETIME(3)
    * usageCondition VARCHAR(45)

**Associated relations**

**1. Owned\_Coupon(记录顾客持有哪些优惠券):**

* + Owned\_Coupon\_customerId(PK) INT
  + Owned\_Coupon\_couponId(PK) INT

**2. Product\_in\_Shoppingcart(记录顾客购物车里有哪些产品):**

* + Product\_in\_ShoppingCart\_cartId (PK) INT
  + Product\_in\_ShoppingCart\_productId (PK) INT

**二、关系（Relationships）:**

1. 一名**Customer** 可以下多个 **Order**，一个**Order**只能对应一个**Customer** ，即**1:m**。这可以通过 **Order** 表中的 **CustomerID** 外键实现。
2. 每个 **Order** 可以包含多个 **OrderDetail**，一个**OrderDetail**只能对应一个**Order** ，即**1:m**。这可以通过 **OrderDetail** 表中的 **OrderID** 外键实现。
3. 每个 **OrderDetail** 对应某种 **Product**中的一个，而一种**Product**可以对应多个**OrderDetail** （一个**Customer** 可能买了多个相同的，不同**Customer** 买了相同的），即**m:1**。这可以通过 **OrderDetail** 表中的 **ProductID** 外键实现。
4. 一个 **Order** 可以对应多个 **Payment**(如退款等)，而一个**Payment**也只能对应一个**Order** ，即**1:m**。这通过 **Payment** 表中的 **OrderID** 外键实现。
5. 一个**Customer** 只能有一个**ShoppingCart**，一个ShoppingCart也只能对于一个**Customer** ，即**1:1**。
6. 一个**ShoppingCart**里可以有多种**Product**，一种**Product**也能在不同的**ShoppingCart**里，即**m:m。**因为是**m:m**所以需要添加一个**associate relation, Product\_in\_Shoppingcart**用于专门保存具体的购物车内的商品。
7. 一个**Seller**可以有多种**Product**，一种**Product**只能对应着一个**Seller**，即**1:m。**
8. 一个**Customer** 可以有多个**Coupon**，一种**Coupon**可以对应多个**Customer** ，即**m:m**。因为是**m:m**所以需要添加一个**associate relation, Owned\_Coupon**用于专门保存具体的顾客的优惠券。
9. 一种**Coupon**只能对应一种**Product**，一种**Product**可以有多个**Coupon**，即**m:1**

**Requirement：**

1. We need to design an online shopping database. it should record customers and products information, as well as purchase/order information and payment details. 2. The core entity sets should contain Customer, Seller, Product, ShoppingCart, Order, OrderDetail, Payment, Coupon.

3. Customer entity has 8 attributes: customerId, userName, password, fName, sName, phoneNumber, Email, Address. Seller entity has 3 attributes: sellerId, sellerName, description. Product entity has 7 attributes: productId, sellerId, productName, description, price, stockQuantity, category. ShoppingCart has 3 attributes: cartId, customerId, productId. Order eneity has 6 attributes: orderId, customerId, orderDate, totalPrice, paymentStatus, shippingAddress. OrderDetail eneity has 5 attributes: OrderDetailId, orderId, productId, quantity, priceAtPurchase. Payment entity has 6 attributes: paymentId, orderId, paymentMethod, transactionId, paymentDate, amount. Coupon entity has 7 attributes: couponId, customerId, productId, discount, startDate, endDate, usageCondition.

4. Each customer could has many orders, and each order only belong to one customer. Each Order can contain many OrderDetails, and one OrderDetail can only correspond to one Order. Each OrderDetail contains one Product, and a Product can exist in many OrderDetails. Each Order can contain many Payments, one Payment corresponding to one Order. Each Customer can only have one ShoppingCart, and a ShoppingCart can only be used for one Customer. One ShoppingCart can have many Products, and one Product can also be in different Shoppingcarts. A Seller can sell many products, and a Product can only be sold by one Seller. A Customer can have many Coupons, and a Coupon can be owned by many Customers. A Coupon can only correspond to one Product, and a Product can have many Coupons.

需求：

1. Customer Management

* The system should allow customers to register and create user accounts.
* Customers should be able to update and modify their personal information such as name, address, phonenumber and email.
* Customers should have the ability to view and manage their order history.
* Users should be able to view their favorites.

2. Seller Management

* The system should allow sellers to register and create user accounts.
* Sellers should have the ability to describe themselves, such as the product category and user reviews
* Sellers should have the ability to manage their products, including adding new products, updating and modifying product information, and tracking product inventory.

3. Product Management

* The system should provide a platform for sellers to add, edit, and remove products.
* Each product should have attributes such as name, description, price, category, and sales volume.
* Customers should be able to view product details, including description, price and seller.
* Customers should be able to search for the product they want
* Customers should have the ability to record their desired product collection for easy search
* Customers should have the ability to see coupons for products they own

4. Order Management

* The system should store the details of the order, including the orderID, the type of products and the quantity purchased, and the price of the products.
* Customers should be able to add products to their shopping orders and proceed to checkout.
* Sellers should receive notifications of new orders and be able to update the order status.
* Customers should be able to track the status of their orders, including shipping and delivery information.
* The customer should have the ability to check the order history.

5. Payment Management

* The system should support secure payment processing for customer orders.
* The system should have the ability to record payment information, including payment time, payment amount, payment method, and transaction ID provided by the bank.
* Customers should be able to choose payment methods.

6. ShoppingCart Management

* The system should record the basic information of the product that the customer puts in the cart, so that the customer can find it quickly
* Customers should have the ability to view, add and remove products from the shopping cart.

7. Coupon Management

* The system should have the ability to record the basic information of the coupon, including the start and end time, the scope of application, the conditions of use, and the discount
* Customers should have the ability to view coupon information
* When the coupon is about to expire, users should be reminded to use it as soon as possible.

**Normalization**

**3NF:**

1. **Customer****：**

*customerId → userName, password, fName, sName, phoneNumber, Email, Address*

1. **Seller：**

*sellerId → sellerName, description*

1. **Product：**

*productId → sellerId, productName, description, price, stockQuantity, category,* *salesVolume*

1. **ShoppingCart：**

*cartId→ customerId, productId*

1. **Order：**

*orderId→ customerId, orderDate, totalPrice, paymentStatus, shippingAddress*

1. **OrderDetail：**

*orderItemId→ orderId, productId, quantity, priceAtPurchase*

1. **Payment：**

*paymentId→ orderId, paymentMethod, transactionId, paymentDate, amount*

1. **Coupon：**

*couponId→ customerId, productId, discount, startDate, endDate, usageCondition*

**Relational Schema**

**Customer** ( customerId, userName, password, fName, sName, phoneNumber, email, address)

**Seller** ( sellerId, sellerName, description)

**Product**( productId, sellerId, productName, description, price, stockQuantity, category, salesVolume)

**ShoppingCart** ( cartId, customerId, productId)

**Order** ( orderId, customerId, orderDate, totalPrice, paymentStatus, shippingAddress, createdAt, updatedAt)

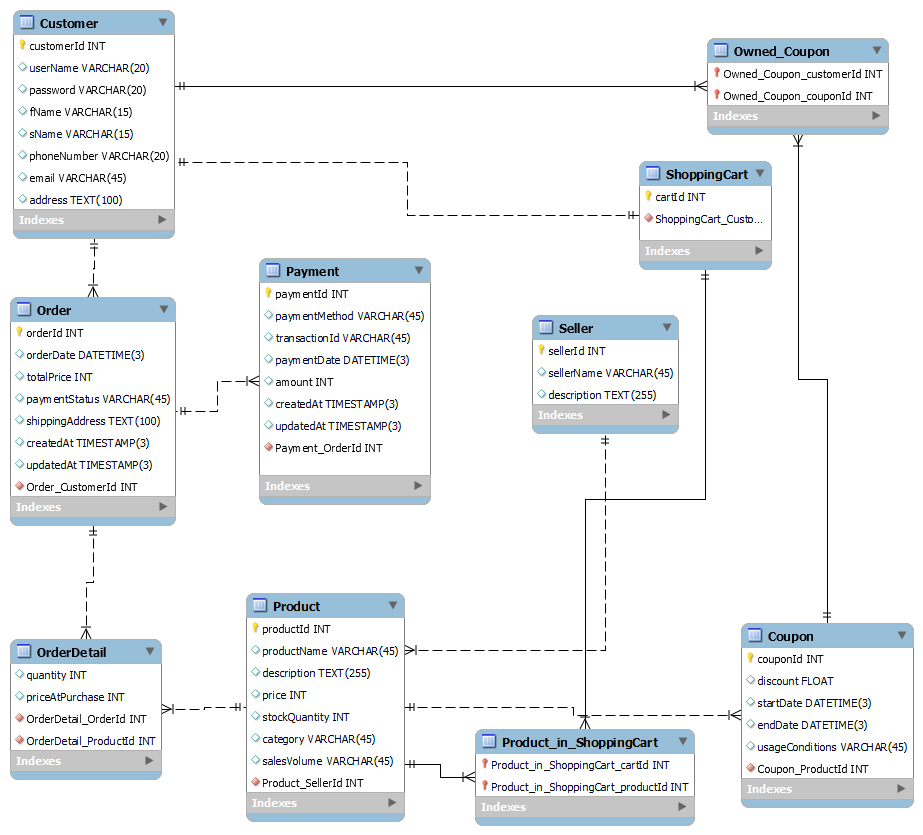
**OrderDetail** ( orderItemId, orderId, productId, quantity, priceAtPurchase)

**Payment** ( paymentId, orderId, paymentMethod, transactionId, paymentDate, amount)

**Coupon** ( couponId, customerId, productId, discount, startDate, endDate, usageCondition)

**Assumptions（这个我不太确定，看着改改）**：

1. **用户角色明确**：
   * 用户必须明确注册为顾客（customer）或商家（seller），且每种角色具有不同的权限和功能访问。
2. **产品信息完整**：
   * 假设所有在平台上销售的产品（product）都有完整且准确的信息，包括名称、描述、价格、库存量等。
3. **购物车功能**：
   * 顾客可以将产品添加到购物车（shoppingcart）中，并且购物车将记录产品添加的时间。
   * 购物车中的产品将保留其被添加时的价格和数量，即使在之后产品价格或库存发生变化。
4. **订单处理**：
   * 顾客可以从购物车中生成订单（order），并且每个订单将包含详细的订单明细（orderdetail），包括产品、数量和价格。
   * 一旦订单生成，商家将收到通知，并能够更新订单状态，如已发货、已完成等。
5. **支付安全**：
   * 顾客可以选择多种支付方式（payment）来完成订单支付，所有支付信息都将被安全地处理和存储。
6. **优惠券使用**：
   * 优惠券（coupon）只能应用于符合特定条件的产品，且每个优惠券只能使用一次。
   * 顾客在使用优惠券时，系统将验证优惠券的有效性和适用条件。
7. **订单追踪与物流**：
   * 顾客能够追踪其订单的状态，包括发货状态、物流信息等。
   * 商家有责任更新订单的物流信息，以确保顾客能够准确追踪。
8. **数据安全与完整性**：
   * 假设系统采取了适当的安全措施来保护用户数据和交易信息的安全性和完整性。
   * 数据库设计遵循数据完整性的最佳实践，如使用外键约束来维护数据的一致性。



 -- The customers

 INSERT INTO `customer` VALUES('49','DeMorgans','demorgans','Chao','Liu','+8617888888888','jp2022215115@qmul.cn.uk','GuangZhou');

 INSERT INTO `customer` VALUES(7,'The Boy Who Lived','Ginny Weasley','Harry','Potter',null,null,' Scottish Highlands, Hogwarts');

 INSERT INTO `customer` VALUES('1001', 'DanK1ng', 'password123', 'Dan', 'King', '1234567890', 'danking@example.com', '123 Dan Street');

 INSERT INTO `customer` VALUES('1002', 'L1LyQueen', 'pass456789', 'Lily', 'Queen', '0987654321', 'lilyqueen@example.com', '456 Lily Road');

 INSERT INTO `customer` VALUES('1003', 'SunnyDay', 'sunny234', 'Sunny', 'Day', '1231231234', 'sunnyday@example.com', '789 Sunny Blvd');

 INSERT INTO `customer` VALUES('1004', 'StarNight', 'night321', 'Star', 'Night', '3213214321', 'starnight@example.com', '1010 Star Way');

 INSERT INTO `customer` VALUES('1005', 'RiverFlow', 'flow987', 'River', 'Flow', '9876543210', 'riverflow@example.com', '2022 River Rd');

 INSERT INTO `customer` VALUES('1006', 'GreenLeaf', 'leaf123', 'Green', 'Leaf', '6543219870', 'greenleaf@example.com', '3030 Leaf Ln');

 INSERT INTO `customer` VALUES('1007', 'MoonLight', 'moon456', 'Moon', 'Light', '4567891230', 'moonlight@example.com', '4040 Moon St');

 INSERT INTO `customer` VALUES('1008', 'SunRise', 'sunrise123', 'Sun', 'Rise', '1237894560', 'sunrise@example.com', '5050 Sunrise Ave');

 INSERT INTO `customer` VALUES('1009', 'WindWhisper', 'wind987', 'Wind', 'Whisper', '9873216540', 'windwhisper@example.com', '6060 Wind Way');

 INSERT INTO `customer` VALUES('1010', 'CloudSky', 'cloud123', 'Cloud', 'Sky', '3216549870', 'cloudsky@example.com', '7070 Cloud Rd');

 select \* from `customer`;

 -- The sellers

 INSERT INTO `seller` VALUES(40000,'Dark Source Flagship Store','A toy store specializing in warhammer 40K');

 INSERT INTO `seller` VALUES(2406,'Ollivanders wand shop','A wand shop with 2,000 years of history.');

 INSERT INTO `seller` VALUES('8001', 'Top technology', 'Focus on high-tech electronics');

 INSERT INTO `seller` VALUES('8002', 'Acoustic music', 'Provide quality audio equipment');

 INSERT INTO `seller` VALUES('8003', 'Home appliance monopoly', 'Supplier of high quality home appliances');

 INSERT INTO `seller` VALUES('8004', 'Outdoors Sports', 'Professional outdoor sports equipment supply');

 INSERT INTO `seller` VALUES('8005', 'Underwater world', 'Various types of underwater sports equipment are provided');

 INSERT INTO `seller` VALUES('8006', 'Tea Garden', 'Professional supply of organic tea');

 INSERT INTO `seller` VALUES('8007', 'Moonlight jewelry', 'Specializing in the supply of all kinds of jewelry');

 INSERT INTO `seller` VALUES('8008', 'Sunrise food', 'Provide healthy and nutritious breakfast foods');

 INSERT INTO `seller` VALUES('8009', 'Wind talk toy', 'Focus on outdoor entertainment toys');

 INSERT INTO `seller` VALUES('8010', 'Cloud home', 'Comfortable household items are provided');

 select \* from `seller`;

**-- The products**

 INSERT INTO `product` VALUES(401,'The Imperial Forbidden Army','Forbidden Guard',209,40,'figure toy',40,40000);

 INSERT INTO `product` VALUES(3,'Custom Wand','The wand chooses you.',7,328,'Daily Necessities',null,2406);

 INSERT INTO `product` VALUES('5001', 'Smartphone', 'The latest powerful smart phone', '150.00', '50', 'electronics', 12,'8001');

 INSERT INTO `product` VALUES('6001', 'Top-tier smart wristband', 'A high-tech smart bracelet made by top technology', '2888', '148', 'electronics', 11,'8001');

 INSERT INTO `product` VALUES('5002', 'Bluetooth Headset', 'High quality sound, comfortable to wear', '50.00', '100', 'electronics', 161,'8002');

 INSERT INTO `product` VALUES('6002', 'Audio premium Headphones', 'An unparalleled music experience', 1280, '110', 'electronics', 141,'8002');

 INSERT INTO `product` VALUES('5003', 'HDTV', 'Ultra HD smart TV, big screen experience', '4888', '30', 'home appliances',1140, '8003');

 INSERT INTO `product` VALUES('6003', 'Energy-saving refrigerator', 'Home appliance monopoly recommended energy-saving refrigerator, large capacity, low noise', 3888, 43, 'home appliances',1242, '8003');

 INSERT INTO `product` VALUES('5004', 'sports bike', 'A high-performance bike for long rides', '250.00', '40', 'outdoor exercise',204, '8004');

 INSERT INTO `product` VALUES('6004', 'hiking shoes', 'Designed for outdoor sports, non-slip, wear resistant', 980.00, '140', 'outdoor exercise',315, '8004');

 INSERT INTO `product` VALUES('5005', 'dive watch', 'Waterproof diving watch for diving enthusiasts', 90.00, 60, 'outdoor equipment', 50,'8005');

 INSERT INTO `product` VALUES('6005', 'underwater movie camera', 'Underwater World provides underwater cameras with high definition image quality and excellent water resistance', 298.00, 115, 'outdoor equipment', 144,'8005');

 INSERT INTO `product` VALUES('5006', 'organic tea', 'Pure organic tea from the mountains', 420.00, '80', 'food',2504, '8006');

 INSERT INTO `product` VALUES('6006', 'Green Tea Garden special Longjing', 'The aroma is fresh and the taste is mellow', 680.00, 32, 'food',22, '8006');

 INSERT INTO `product` VALUES('5007', 'Moonstone necklace', 'A moonstone necklace with an air of mystery', 120.00, '70', 'jewelry',9, '8007');

 INSERT INTO `product` VALUES('6007', 'Moonlight custom necklace', 'Selected gems, exquisite workmanship', 88, 70, 'jewelry',11, '8007');

 INSERT INTO `product` VALUES('5008', 'breakfast cereal', 'Nutritious and healthy breakfast options', '55.00', '100', 'food', 4894,'8008');

 INSERT INTO `product` VALUES('6008', 'Sunrise cereal', 'Rich in vitamins and minerals', 120, 120, 'food', 3551,'8008');

 INSERT INTO `product` VALUES('5009', 'kite', 'High quality kites for all ages', 130.00, '90', 'outdoor toy',112, '8009');

 INSERT INTO `product` VALUES('6009', 'Wind talk outdoor kite', 'Unique design and stable flight', 380.00, 54, 'outdoor toy',38, '8009');

 INSERT INTO `product` VALUES('5010', 'Cloud cushion', 'Comfortable and soft, suitable for family use', 150.00, 85, 'household supplies',58, '8010');

 INSERT INTO `product` VALUES('6010', 'Cloud memory cotton mattress', 'Ergonomically designed to provide the ultimate sleep experience', 588.00, 58, 'household supplies',85, '8010');

**-- The order, orderdetail, payment**

 INSERT INTO `order` VALUES('449','2028-10-1 16:19:25','209','paid','GuangZhou','2028-10-1 16:19:25','2028-10-1 16:19:25',49);

 INSERT INTO `orderdetail` VALUES('1','209',449,'401');

 INSERT INTO `payment` VALUES('449','WeChat Pay',"TXN2024214394",'2028-10-1 16:20:03',1,'2028-10-1 16:20:03','2028-10-1 16:20:03','449');

 INSERT INTO `order` VALUES(7,'1991-07-03 13:07:13','7','paid','Hogwarts, Gryffindor Common Room','1991-07-03 13:07:13','1991-07-03 13:07:13',7);

 INSERT INTO `orderdetail` VALUES('1',7,'7','3');

 INSERT INTO `payment` VALUES(3,'By galleons',null,'1991-07-03 13:07:33',1,'1991-07-03 13:07:33','1991-07-03 13:07:33',7);

 INSERT INTO `order` VALUES('901', '2024-05-15 00:00:00', '300', 'paid', '123 Ding Street', '2024-05-15 00:00:00', '2024-05-15 00:00:00', '1001');

 INSERT INTO `orderdetail` VALUES('2', '150.00', '901', '5001');

 INSERT INTO `payment` VALUES('601', 'Alipay', 'TXN12345', '2024-05-15 15:17:43', '300.00', '2024-05-15 15:17:43', '2024-05-15 15:17:43', '901');

 INSERT INTO `order` VALUES('902', '2024-05-16 14:16:13', '450.00', 'non-payment', '456 Lily Road', '2024-05-16 14:16:13', '2024-05-16 14:16:13', '1002');

 INSERT INTO `orderdetail` VALUES('3', '150.00', '902', '5002');

 INSERT INTO `payment` VALUES('602', 'WeChat Pay', 'TXN67890', '2024-05-16 14:16:52', '450.00', '2024-05-16 14:16:52', '2024-05-16 14:16:52', '902');

 INSERT INTO `order` VALUES('903', '2024-05-17 09:35:25', '550.00', 'paid', '789 Sunny Blvd', '2024-05-17 09:35:25', '2024-05-17 09:35:25', '1003');

 INSERT INTO `orderdetail` VALUES( '1', '550.00', '903', '5003');

 INSERT INTO `payment` VALUES('603', 'credit card', 'TXN123456', '2024-05-17 09:36:12', '550.00', '2024-05-17 09:36:12', '2024-05-17 09:36:12', '903');

 INSERT INTO `order` VALUES('904', '2024-05-18 18:22:16', '750.00', 'non-payment', '1010 Star Way', '2024-05-18 18:22:16', '2024-05-18 18:22:16', '1004');

 INSERT INTO `orderdetail` VALUES( '3', '250.00', '904', '5004');

 INSERT INTO `payment` VALUES('604', 'PayPal', 'TXN234567', '2024-05-18 18:22:54', '750.00', '2024-05-18 18:22:54', '2024-05-18 18:22:54', '904');

 INSERT INTO `order` VALUES('905', '2024-05-19 14:16:15', '180.00', 'paid', '2022 River Rd', '2024-05-19 14:16:15', '2024-05-19 14:16:15', '1005');

 INSERT INTO `orderdetail` VALUES( '2', '90.00', '905', '5005');

 INSERT INTO `payment` VALUES('605', 'WeChat Pay', 'TXN345678', '2024-05-19 14:18:11' , '180.00', '2024-05-19 14:18:11', '2024-05-19 14:18:11', '905');

 INSERT INTO `order` VALUES('906', '2024-05-20 17:20:21', '420.00', 'non-payment', '3030 Leaf Ln', '2024-05-20 17:20:21', '2024-05-20 17:20:21', '1006');

 INSERT INTO `orderdetail` VALUES('1', '420.00', '906', '5006');

 INSERT INTO `payment` VALUES('606', 'Alipay', 'TXN456789', '2024-05-20 17:20:34', '420.00', '2024-05-20 17:20:34', '2024-05-20 17:20:34', '906');

 INSERT INTO `order` VALUES('907', '2024-05-21  21:44:59', '120.00', 'paid', '4040 Moon St', '2024-05-21  21:44:59', '2024-05-21  21:44:59', '1007');

 INSERT INTO `orderdetail` VALUES( '1', '120.00', '907', '5007');

 INSERT INTO `payment` VALUES('607', '信用卡', 'TXN567890', '2024-05-21 21:45:14', '120.00', '2024-05-21 21:45:14', '2024-05-21 21:45:14', '907');

 INSERT INTO `order` VALUES('908', '2024-05-22 16:01:43', '55.00', 'non-payment', '5050 Sunrise Ave', '2024-05-22 16:01:43', '2024-05-22 16:01:43', '1008');

 INSERT INTO `orderdetail` VALUES( '1', '55.00', '908', '5008');

 INSERT INTO `payment` VALUES('608', 'WeChat Pay', 'TXN678901', '2024-05-22 16:02:31', '55.00', '2024-05-22 16:02:31', '2024-05-22 16:02:31', '908');

 INSERT INTO `order` VALUES('909', '2024-05-23 10:26:23', '260.00', 'paid', '6060 Wind Way', '2024-05-23 10:26:23', '2024-05-23 10:26:23', '1009');

 INSERT INTO `orderdetail` VALUES('2', '130.00', '909', '5009');

 INSERT INTO `payment` VALUES('609', 'credit card', 'TXN789012', '2024-05-23 10:26:59', '260.00', '2024-05-23 10:26:59', '2024-05-23 10:26:59', '909');

 INSERT INTO `order` VALUES('910', '2024-05-24 11:45:22', '450.00', 'non-payment', '7070 Cloud Rd', '2024-05-24 11:45:22', '2024-05-24 11:45:22', '1010' );

 INSERT INTO `orderdetail` VALUES( '3', '150.00', '910', '5010');

 INSERT INTO `payment` VALUES('610', 'Alipay', 'TXN890123', '2024-05-24 11:46:22', '450.00', '2024-05-24 11:46:22', '2024-05-24 11:46:22', '910');

**-- The coupon**

 INSERT INTO `coupon` VALUES('449','0.8','2028-10-1 00:00:00','2028-10-7 23:59:59','null','401');

 INSERT INTO `coupon` VALUES(7,'0.7','1991-07-03 13:07:03','1991-07-13 13:07:03','You name is Harry Potter','3');

 INSERT INTO `coupon` VALUES('101', '0.85', '2024-05-01 00:00:00 ', '2024-05-31 23:59:59', 'It can be used after 100 yuan', '5001');

 INSERT INTO `coupon` VALUES('102', '0.90', '2024-05-28 00:00:00', '2024-06-10 23:59:59', 'It can be used after 200 yuan', '5002');

 INSERT INTO `coupon` VALUES('103', '0.80', '2024-04-15 00:00:00', '2024-06-15 23:59:59', 'It can be used after 500 yuan', '5003');

 INSERT INTO `coupon` VALUES('104', '0.75', '2024-05-01 00:00:00', '2024-07-01 23:59:59', 'It can be used after 300 yuan', '5004');

 INSERT INTO `coupon` VALUES('105', '0.70', '2024-04-20 00:00:00', '2024-05-20 23:59:59', 'It can be used after 200 yuan', '5005');

 INSERT INTO `coupon` VALUES('206', '0.70', '2024-04-20 00:00:00', '2024-05-20 23:59:59', 'It can be used after 300 yuan', '5006');

 INSERT INTO `coupon` VALUES('106', '0.85', '2024-04-20 00:00:00', '2024-05-20 23:59:59','It can be used after 400 yuan', '5006');

 INSERT INTO `coupon` VALUES('107', '0.80', '2024-05-01 00:00:00', '2024-06-01 23:59:59', 'It can be used after 100 yuan', '5007');

 INSERT INTO `coupon` VALUES('108', '0.9', '2024-06-01 00:00:00', '2024-07-15 23:59:59', 'It can be used after 50 yuan', '5008');

 INSERT INTO `coupon` VALUES('208', '0.9', '2024-06-01 00:00:00', '2024-07-15 23:59:59', 'It can be used after 100 yuan', '5008');

 INSERT INTO `coupon` VALUES('109', '0.85', '2024-04-01 00:00:00', '2024-05-31 23:59:59', 'It can be used after 250 yuan', '5009');

 INSERT INTO `coupon` VALUES('110','0.75', '2024-05-15 00:00:00', '2024-06-30 23:59:59', 'It can be used after 400 yuan', '5010');

 INSERT INTO `coupon` VALUES('210','0.85', '2024-05-15 00:00:00', '2024-06-30 23:59:59', 'It can be used after 200 yuan', '5010');

**-- The shoppingcart**

 INSERT INTO `shoppingcart` VALUES('49','49');

 INSERT INTO `shoppingcart` VALUES(7,7);

 INSERT INTO `shoppingcart` VALUES('201', '1001');

 INSERT INTO `shoppingcart` VALUES('202', '1002');

 INSERT INTO `shoppingcart` VALUES('203', '1003');

 INSERT INTO `shoppingcart` VALUES('204', '1004');

 INSERT INTO `shoppingcart` VALUES('205', '1005');

 INSERT INTO `shoppingcart` VALUES('206', '1006');

 INSERT INTO `shoppingcart` VALUES('207', '1007');

 INSERT INTO `shoppingcart` VALUES('208', '1008');

 INSERT INTO `shoppingcart` VALUES('209', '1009');

 INSERT INTO `shoppingcart` VALUES('210', '1010');

**-- The products in shopping cart**

 INSERT INTO `product\_in\_shoppingcart` VALUES(49,401);

 INSERT INTO `product\_in\_shoppingcart` VALUES(7,3);

 INSERT INTO `product\_in\_shoppingcart` VALUES('201',  '5001');

 INSERT INTO `product\_in\_shoppingcart` VALUES('202',  '5002');

 INSERT INTO `product\_in\_shoppingcart` VALUES('202',  '6002');

 INSERT INTO `product\_in\_shoppingcart` VALUES('202',  '5001');

 INSERT INTO `product\_in\_shoppingcart` VALUES('203',  '5003');

 INSERT INTO `product\_in\_shoppingcart` VALUES('203',  '6003');

 INSERT INTO `product\_in\_shoppingcart` VALUES('203',  '5009');

 INSERT INTO `product\_in\_shoppingcart` VALUES('204',  '5004');

 INSERT INTO `product\_in\_shoppingcart` VALUES('204',  '6004');

 INSERT INTO `product\_in\_shoppingcart` VALUES('205',  '5005');

 INSERT INTO `product\_in\_shoppingcart` VALUES('205',  '6005');

 INSERT INTO `product\_in\_shoppingcart` VALUES('206',  '5006');

 INSERT INTO `product\_in\_shoppingcart` VALUES('206',  '6006');

 INSERT INTO `product\_in\_shoppingcart` VALUES('206',  '6010');

 INSERT INTO `product\_in\_shoppingcart` VALUES('206',  '6008');

 INSERT INTO `product\_in\_shoppingcart` VALUES('207',  '5007');

 INSERT INTO `product\_in\_shoppingcart` VALUES('207',  '6007');

 INSERT INTO `product\_in\_shoppingcart` VALUES('207',  '6006');

 INSERT INTO `product\_in\_shoppingcart`VALUES('208',  '6008');

 INSERT INTO `product\_in\_shoppingcart`VALUES('208',  '5008');

 INSERT INTO `product\_in\_shoppingcart` VALUES('209',  '5009');

 INSERT INTO `product\_in\_shoppingcart` VALUES('210',  '5010');

 INSERT INTO `product\_in\_shoppingcart` VALUES('210',  '5003');

 INSERT INTO `product\_in\_shoppingcart` VALUES('210',  '6003');

**-- The coupons that owned by customers**

 INSERT INTO `owned\_coupon` VALUES(49,449);

 INSERT INTO `owned\_coupon` VALUES(7,7);

 INSERT INTO `owned\_coupon` VALUES(1001,101);

 INSERT INTO `owned\_coupon` VALUES(1002,102);

 INSERT INTO `owned\_coupon` VALUES(1003,103);

 INSERT INTO `owned\_coupon` VALUES(1004,104);

 INSERT INTO `owned\_coupon` VALUES(1005,105);

 INSERT INTO `owned\_coupon` VALUES(1005,206);

 INSERT INTO `owned\_coupon` VALUES(1006,106);

 INSERT INTO `owned\_coupon` VALUES(1007,107);

 INSERT INTO `owned\_coupon` VALUES(1008,108);

 INSERT INTO `owned\_coupon` VALUES(1008,208);

 INSERT INTO `owned\_coupon` VALUES(1009,109);

 INSERT INTO `owned\_coupon` VALUES(1010,110);

 INSERT INTO `owned\_coupon` VALUES(1010,210);