



QUEEN MARY UNIVERSITY OF LONDON

EBU5503 DATABASE SYSTEMS

---

# Coursework Report

---

Shuang Liang	(BUPT ID: 2022213449, QM ID: 221169221)
Zecheng Zuo	(BUPT ID: 2022213434, QM ID: 221169081)
Jiapeng Wang	(BUPT ID: 2022213436, QM ID: 221169106)
Yan Li	(BUPT ID: 2022213446, QM ID: 221169209)
Wenhao Ding	(BUPT ID: 2022213431, QM ID: 221169058)
Shangbai Li	(BUPT ID: 2022213481, QM ID: 221169553)
Yuxuan Zhong	(BUPT ID: 2022213435, QM ID: 221169092)

Date: May 20, 2024

---

# Contents

<b>1</b>	<b>Description</b>	<b>2</b>
1.1	System Overview . . . . .	2
1.2	Requirements . . . . .	2
1.2.1	User Management . . . . .	2
1.2.2	Product and Inventory Management . . . . .	2
1.2.3	Order and Shopping Cart Management . . . . .	2
1.2.4	Payments and Security . . . . .	2
1.2.5	Coupons and Promotions . . . . .	2
<b>2</b>	<b>Conceptual Schema</b>	<b>3</b>
2.1	EER Diagram . . . . .	3
2.2	Explanation . . . . .	3
2.2.1	Entity and Attribute . . . . .	3
2.2.2	Relationship . . . . .	5
<b>3</b>	<b>Relational Database Schema</b>	<b>6</b>
3.1	Relational Schema . . . . .	6
3.2	Validation using normalisation technique . . . . .	6
<b>4</b>	<b>Sample Data</b>	<b>7</b>
<b>A</b>	<b>Appendix: Queries and Results</b>	<b>11</b>
A.1	Queries . . . . .	11
A.2	Results . . . . .	12

---

# 1 Description

## 1.1 System Overview

This section outlines the fundamental requirements and functionalities of the proposed system, which includes managing user roles, product information, orders, payments, and security.

## 1.2 Requirements

### 1.2.1 User Management

- Users must register as either a customer or a merchant, each with specific permissions.
- Customers can register, update personal details, view, and manage their order history and favorites.
- Sellers can manage their accounts, add, update, and describe products, and receive order notifications.

### 1.2.2 Product and Inventory Management

- Platform for sellers to manage product listings including adding, editing, and removal.
- Products feature attributes like name, description, price, category, and inventory details.
- Customers can view detailed product information and search for products based on various criteria.

### 1.2.3 Order and Shopping Cart Management

- Customers can add products to their shopping carts, which capture product details and timestamp of addition.
- Support for generating orders from the cart, with each order capturing detailed product quantities and prices.

### 1.2.4 Payments and Security

- Support for multiple payment methods ensuring secure transaction processing.

### 1.2.5 Coupons and Promotions

- Coupons are applicable under specific conditions and are verified for validity at the time of use.

---

## 2 Conceptual Schema

### 2.1 EER Diagram

The EER Diagram is shown below as Figure 1.

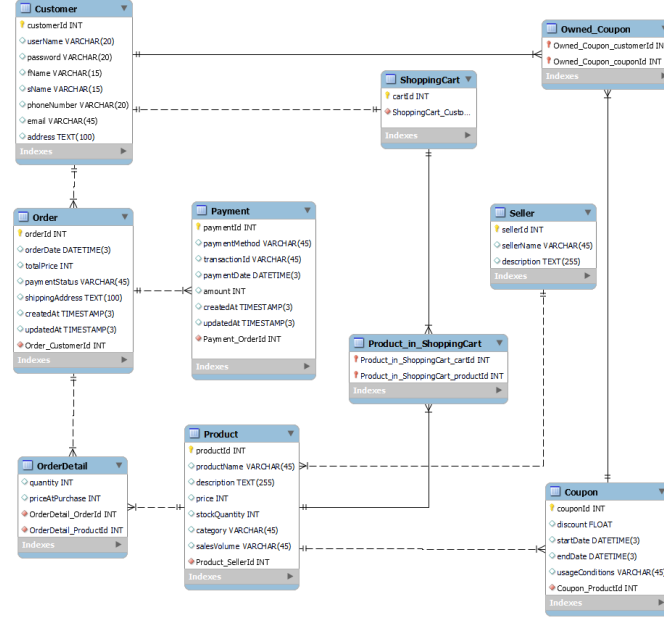


Figure 1: EER diagram

### 2.2 Explanation

#### 2.2.1 Entity and Attribute

The database design includes the following entities and their attributes:

##### 1. Customer

- **Explanation:** Stores all detailed information of customers, including Customer ID (primary key, using INT type to save storage space and enhance search efficiency), username, password, first name, last name, phone number, email, and address.
- **Introduction:** This entity allows the website to track and manage customer information, supporting functionalities such as user registration, login, personal profile modification, and viewing order history.

##### 2. Seller

- **Explanation:** Stores all detailed information of sellers, including Seller ID (primary key, using INT type) and seller name.

- 
- **Introduction:** This entity is used for tracking and managing information of merchants on the website, supporting functionalities such as merchant registration, account creation, product category description, and customer reviews.

### 3. Product

- **Explanation:** Stores all detailed information of products, likely including Product ID, product name, description, price, and stock quantity.
- **Introduction:** This entity is the core of the website, used for displaying and managing all products for sale, ensuring customers can see and purchase the products they need.

### 4. Shopping Cart

- **Explanation:** Records information about products added to the customer's shopping cart, including Cart ID (primary key) and Product ID.
- **Introduction:** This entity allows customers to select products and add them to their shopping cart while browsing the website, preparing for purchase.

### 5. Order

- **Explanation:** Records information related to the entire order, including Order ID (possibly a primary key), Customer ID (foreign key), and order status.
- **Introduction:** This entity allows the website to track and manage customer orders, ensuring that orders are processed accurately and timely.

### 6. Order Detail

- **Explanation:** Records detailed information about each item in an order, including Order Detail ID (possibly a primary key), Order ID (foreign key), Product ID (foreign key), and quantity purchased.
- **Introduction:** This entity provides more detailed information about orders, including which products were purchased, in what quantity, helping merchants and customers understand the order contents accurately.

### 7. Payment

- **Explanation:** Records payment information, including Payment ID (possibly a primary key), Order ID (foreign key), payment amount, and payment method.
- **Introduction:** This entity allows the website to track and manage payment information of orders, ensuring the security and accuracy of the payment process.

### 8. Coupon

- **Explanation:** Records coupon information, including Coupon ID (primary key), conditions, and validity period.
- **Introduction:** This entity enables the website to offer coupon functionalities, attracting customers to purchase products, while also validating the effectiveness and applicability of coupons.

---

### 2.2.2 Relationship

The database design includes the following relationships between entities:

#### 1. Customer and Order Relationship

- A single Customer can place multiple Orders, establishing a one-to-many (1:m) relationship. This relationship is implemented through the CustomerID foreign key in the Order table.

#### 2. Order and OrderDetail Relationship

- Each Order can include multiple OrderDetails, a one-to-many (1:m) relationship. This is facilitated by the OrderID foreign key in the OrderDetail table.

#### 3. OrderDetail and Product Relationship

- Each OrderDetail corresponds to a specific Product, while a single Product can be associated with multiple OrderDetails, forming a many-to-one (m:1) relationship. This is managed through the ProductID foreign key in the OrderDetail table.

#### 4. Order and Payment Relationship

- An Order can be associated with multiple Payments (such as refunds), a one-to-many (1:m) relationship. This relationship is managed through the OrderID foreign key in the Payment table.

#### 5. Customer and ShoppingCart Relationship

- Each Customer has exactly one ShoppingCart, and each ShoppingCart corresponds to exactly one Customer, establishing a one-to-one (1:1) relationship.

#### 6. ShoppingCart and Product Relationship

- A ShoppingCart can contain multiple Products, and a Product can be in different ShoppingCarts, forming a many-to-many (m:m) relationship. This necessitates an associate relationship, Product\_in\_Shoppingcart, to specifically store the products in each ShoppingCart.

#### 7. Seller and Product Relationship

- A Seller can offer multiple Products, while each Product is associated with only one Seller, forming a one-to-many (1:m) relationship.

#### 8. Customer and Coupon Relationship

- A Customer can have multiple Coupons, and a Coupon can apply to multiple Customers, establishing a many-to-many (m:m) relationship. This requires an associate relationship, Owned\_Coupon, to specifically store the coupons owned by each customer.

---

## 9. Coupon and Product Relationship

- Each Coupon corresponds to one specific Product, while a Product can have multiple Coupons, forming a many-to-one (m:1) relationship.

# 3 Relational Database Schema

## 3.1 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)

## 3.2 Validation using normalisation technique

**Customer:**

- $\text{customerId} \rightarrow \text{userName, password, fName, sName, phoneNumber, Email, Address}$

**Seller:**

- $\text{sellerId} \rightarrow \text{sellerName, description}$

**Product:**

- $\text{productId} \rightarrow \text{sellerId, productName, description, price, stockQuantity, category, salesVolume}$

**ShoppingCart:**

- $\text{cartId} \rightarrow \text{customerId, productId}$

**Order:**

- $\text{orderId} \rightarrow \text{customerId, orderDate, totalPrice, paymentStatus, shippingAddress}$

**OrderDetail:**

- $\text{orderItemId} \rightarrow \text{orderId, productId, quantity, priceAtPurchase}$

**Payment:**

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

Coupon:

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

This database is in 3NF (Third Normal Form).

## 4 Sample Data

The following SQL Insert statements are used to insert sample data:

```

1  -- The customers
2  INSERT INTO 'customer' VALUES('49','DeMorgans',
   'demorgans','Chao','Liu','+861788888888','
   jp2022215115@qmul.cn.uk','GuangZhou');
3  INSERT INTO 'customer' VALUES(7,'The Boy Who
   Lived','Ginny Weasley','Harry','Potter',null
   ,null,'Scottish Highlands, Hogwarts');
4  INSERT INTO 'customer' VALUES('1001','DanKing'
   , 'password123','Dan','King','1234567890'
   , 'danking@example.com','123 Dan Street');
5  INSERT INTO 'customer' VALUES('1002','
   LilyQueen','pass456789','Lily','Queen','
   0987654321','lilyqueen@example.com','456
   Lily Road');
6  INSERT INTO 'customer' VALUES('1003','SunnyDay
   ', 'sunny234','Sunny','Day','1231231234',
   'sunnyday@example.com','789 Sunny Blvd');
7  INSERT INTO 'customer' VALUES('1004','
   StarNight','night321','Star','Night','
   3213214321','starnight@example.com','1010
   Star Way');
8  INSERT INTO 'customer' VALUES('1005','
   RiverFlow','flow987','River','Flow','
   9876543210','riverflow@example.com','2022
   River Rd');
9  INSERT INTO 'customer' VALUES('1006','
   GreenLeaf','leaf123','Green','Leaf','
   6543219870','greenleaf@example.com','3030
   Leaf Ln');
10 INSERT INTO 'customer' VALUES('1007','
   MoonLight','moon456','Moon','Light','
   4567891230','moonlight@example.com','4040
   Moon St');
11 INSERT INTO 'customer' VALUES('1008','SunRise'
   , 'sunrise123','Sun','Rise','1237894560',
   'sunrise@example.com','5050 Sunrise Ave');
12 INSERT INTO 'customer' VALUES('1009','
   WindWhisper','wind987','Wind','Whisper',
   '9873216540','windwhisper@example.com',
   '6060 Wind Way');
13 INSERT INTO 'customer' VALUES('1010','CloudSky
   ', 'cloud123','Cloud','Sky','3216549870',
   'cloudsky@example.com','7070 Cloud Rd');
14
15 select * from 'customer';
16
17
18
19 -- The sellers
20 INSERT INTO 'seller' VALUES(40000,'Dark Source
   Flagship Store','A toy store specializing in
   warhammer 40K');
21 INSERT INTO 'seller' VALUES(2406,'Ollivanders
   wand shop','A wand shop with 2,000 years of
   history. ');
22 INSERT INTO 'seller' VALUES('8001','Top
   technology','Focus on high-tech electronics
   ');
23 INSERT INTO 'seller' VALUES('8002','Acoustic
   music','Provide quality audio equipment');
24 INSERT INTO 'seller' VALUES('8003','Home
   appliance monopoly','Supplier of high
   quality home appliances');
25 INSERT INTO 'seller' VALUES('8004','Outdoors
   Sports','Professional outdoor sports
   equipment supply');
26 INSERT INTO 'seller' VALUES('8005','Underwater
   world','Various types of underwater sports
   equipment are provided');
27 INSERT INTO 'seller' VALUES('8006','Tea Garden
   ', 'Professional supply of organic tea');
28 INSERT INTO 'seller' VALUES('8007','Moonlight
   jewelry','Specializing in the supply of all
   kinds of jewelry');
29 INSERT INTO 'seller' VALUES('8008','Sunrise
   food','Provide healthy and nutritious
   breakfast foods');
30 INSERT INTO 'seller' VALUES('8009','Wind talk
   toy','Focus on outdoor entertainment toys')
   ;
31 INSERT INTO 'seller' VALUES('8010','Cloud home
   ', 'Comfortable household items are provided
   ');
32
33 select * from 'seller';
34
35
36
37 -- The products
38 INSERT INTO 'product' VALUES(401,'The Imperial
   Forbidden Army','Forbidden Guard',209,40,'
   figure toy',40,40000);
39 INSERT INTO 'product' VALUES(3,'Custom Wand',
   'The wand chooses you.',7,328,'Daily
   Necessities',null,2406);
40 INSERT INTO 'product' VALUES('5001','
   Smartphone','The latest powerful smart
   phone','150.00','50','electronics',12,'

```



```

8001');
41 INSERT INTO 'product' VALUES('6001', 'Top-tier smart wristband', 'A high-tech smart
bracelet made by top technology', '2888', '148', 'electronics', 11, '8001');
42 INSERT INTO 'product' VALUES('5002', 'Bluetooth Headset', 'High quality sound, comfortable
to wear', '50.00', '100', 'electronics', 161, '8002');
43 INSERT INTO 'product' VALUES('6002', 'Audio premium Headphones', 'An unparalleled music
experience', 1280, '110', 'electronics', 141, '8002');
44 INSERT INTO 'product' VALUES('5003', 'HDTV', 'Ultra HD smart TV, big screen experience', '4888', '30', 'home appliances', 1140, '8003');
45 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');
46 INSERT INTO 'product' VALUES('5004', 'sports bike', 'A high-performance bike for long rides', '250.00', '40', 'outdoor exercise', 204, '8004');
47 INSERT INTO 'product' VALUES('6004', 'hiking shoes', 'Designed for outdoor sports, non-slip, wear resistant', 980.00, '140', 'outdoor exercise', 315, '8004');
48 INSERT INTO 'product' VALUES('5005', 'dive watch', 'Waterproof diving watch for diving enthusiasts', 90.00, 60, 'outdoor equipment', 50, '8005');
49 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');
50 INSERT INTO 'product' VALUES('5006', 'organic tea', 'Pure organic tea from the mountains', 420.00, '80', 'food', 2504, '8006');
51 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');
52 INSERT INTO 'product' VALUES('5007', 'Moonstone necklace', 'A moonstone necklace with an air of mystery', 120.00, '70', 'jewelry', 9, '8007');
53 INSERT INTO 'product' VALUES('6007', 'Moonlight custom necklace', 'Selected gems, exquisite workmanship', 88, 70, 'jewelry', 11, '8007');
54 INSERT INTO 'product' VALUES('5008', 'breakfast cereal', 'Nutritious and healthy breakfast options', '55.00', '100', 'food', 4894, '8008');
55 INSERT INTO 'product' VALUES('6008', 'Sunrise cereal', 'Rich in vitamins and minerals', 120, 120, 'food', 3551, '8008');
56 INSERT INTO 'product' VALUES('5009', 'kite', 'High quality kites for all ages', 130.00, '90', 'outdoor toy', 112, '8009');
57 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');

```

```

84 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');
85 INSERT INTO `orderdetail` VALUES('3', '250.00
, '904', '5004');
86 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');
87
88 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');
89 INSERT INTO `orderdetail` VALUES('2', '90.00'
, '905', '5005');
90 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');
91
92 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');
93 INSERT INTO `orderdetail` VALUES('1', '420.00'
, '906', '5006');
94 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');
95
96 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');
97 INSERT INTO `orderdetail` VALUES('1', '120.00'
, '907', '5007');
98 INSERT INTO `payment` VALUES('607', 'Credit
Card', 'TXN567890', '2024-05-21 21:45:14', '
120.00', '2024-05-21 21:45:14', '2024-05-21
21:45:14', '907');
99
100 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');
101 INSERT INTO `orderdetail` VALUES('1', '55.00'
, '908', '5008');
102 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');
103
104 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');
105 INSERT INTO `orderdetail` VALUES('2', '130.00'
, '909', '5009');
106 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');
107
108 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'
);

```

```

138 INSERT INTO 'shoppingcart' VALUES('203', '1003')
139 );
139 INSERT INTO 'shoppingcart' VALUES('204', '1004')
140 );
140 INSERT INTO 'shoppingcart' VALUES('205', '1005')
141 );
141 INSERT INTO 'shoppingcart' VALUES('206', '1006')
142 );
142 INSERT INTO 'shoppingcart' VALUES('207', '1007')
143 );
143 INSERT INTO 'shoppingcart' VALUES('208', '1008')
144 );
144 INSERT INTO 'shoppingcart' VALUES('209', '1009')
145 );
145 INSERT INTO 'shoppingcart' VALUES('210', '1010')
146 );
146
147 -- The products in shopping cart
148 INSERT INTO 'product_in_shoppingcart' VALUES
149 (49,401);
149 INSERT INTO 'product_in_shoppingcart' VALUES
150 (7,3);
150 INSERT INTO 'product_in_shoppingcart' VALUES('
151 201', '5001');
151 INSERT INTO 'product_in_shoppingcart' VALUES('
152 202', '5002');
152 INSERT INTO 'product_in_shoppingcart' VALUES('
153 202', '6002');
153 INSERT INTO 'product_in_shoppingcart' VALUES('
154 202', '5001');
154 INSERT INTO 'product_in_shoppingcart' VALUES('
155 203', '5003');
155 INSERT INTO 'product_in_shoppingcart' VALUES('
156 203', '6003');
156 INSERT INTO 'product_in_shoppingcart' VALUES('
157 203', '5009');
157 INSERT INTO 'product_in_shoppingcart' VALUES('
158 204', '5004');
158 INSERT INTO 'product_in_shoppingcart' VALUES('
159 204', '6004');
159 INSERT INTO 'product_in_shoppingcart' VALUES('
160 205', '5005');
160 INSERT INTO 'product_in_shoppingcart' VALUES('
161 205', '6005');
161 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);

```

Listing 1: MySQL Example Code

---

## A Appendix: Queries and Results

Below are the SQL queries and corresponding results related to our database:

### A.1 Queries

#### Query 1: Customers with the Most Purchases

```
1 -- Query the customers who purchased the most
  and the total number of purchases
2 SELECT c.customerId, c.fName, c.sName, SUM(od.
  quantity) AS totalQuantity
3 FROM customer c
4 JOIN 'order' o ON c.customerId = o.
  Order_CustomerId
5 JOIN orderdetail od ON o.orderId = od.
  OrderDetail_OrderId
6 GROUP BY c.customerId, c.fName, c.sName
7 ORDER BY totalQuantity DESC
8 LIMIT 1;
```

#### Query 2: Products Purchased and Still In Stock

```
1 -- Query all products that customers have
  purchased and still have in stock
2 SELECT DISTINCT p.productId, p.productName
3 FROM product p
4 JOIN orderdetail od ON p.productId = od.
  OrderDetail_ProductId
5 WHERE p.stockQuantity > 0;
```

#### Query 3: Total Discount for All Customers

```
1 -- Query the total discount for all customers
  and their coupons
2 SELECT c.customerId, c.fName, c.sName, SUM(cou.
  discount) AS totalDiscount
3 FROM customer c
4 LEFT JOIN owned_coupon oc ON c.customerId = oc.
  Owned_Coupon_customerId
5 LEFT JOIN coupon cou ON oc.Owned_Coupon_couponId
  = cou.couponId
6 GROUP BY c.customerId, c.fName, c.sName;
```

#### Query 4: Date of Last Purchase for Each Customer

```
1 -- Query the date of each customer's last
  purchase
```

```
2 SELECT c.customerId, c.fName, c.sName, MAX(o.
  orderDate) AS lastPurchaseDate
3 FROM customer c
4 JOIN 'Order' o ON c.customerId = o.
  Order_CustomerId
5 GROUP BY c.customerId, c.fName, c.sName;
```

#### Query 5: Customers Who Placed Orders Within a Certain Date Range

```
1 -- Query customers who placed orders within a
  certain date range (within 2024) and list
  the names and quantities of the products
  they purchased
2 SELECT MIN(o.orderDate) AS earliestOrderDate, c.
  customerId, c.fName, c.sName, p.productName,
  SUM(od.quantity) AS quantityPurchased
3 FROM customer c
4 JOIN 'order' o ON c.customerId = o.
  Order_CustomerId
5 JOIN orderdetail od ON o.orderId = od.
  OrderDetail_OrderId
6 JOIN product p ON od.OrderDetail_ProductId = p.
  productId
7 WHERE o.orderDate BETWEEN '2024-01-01 00:00:00'
  AND '2024-12-31 00:00:00'
8 GROUP BY c.customerId, c.fName, c.sName, p.
  productName;
```

#### Query 6: Average Price Per Customer

```
1 -- Query the average price per customer
2 SELECT c.customerId, c.fName, c.sName, AVG(od.
  priceAtPurchase) AS averagePrice
3 FROM customer c
4 JOIN 'order' o ON c.customerId = o.
  Order_CustomerId
5 JOIN orderdetail od ON o.orderId = od.
  OrderDetail_OrderId
6 GROUP BY c.customerId, c.fName, c.sName;
```

#### Query 7: Total Price of Items in Each Customer's Cart

```

1 -- Query the total price of the items in each
  customer's cart
2 SELECT sc.cartId, c.customerId, c.fName, c.sName
   , SUM(p.price) AS totalCartPrice
3 FROM shoppingcart sc
4 JOIN customer c ON sc.ShoppingCart_CustomerId =
  c.customerId
5 JOIN product_in_shoppingcart pisc ON sc.cartId =
  pisc.Product_in_ShoppingCart_cartId
6 JOIN product p ON pisc.
  Product_in_ShoppingCart_productId = p.
  productId
7 GROUP BY sc.cartId, c.customerId, c.fName, c.
  sName;

```

## Query 8: Check for Expiring Coupons in Customer Carts

```

1 -- Check each customer's shopping cart to see if
  there is an item corresponding to an
  expiring coupon
2 SELECT DISTINCT c.customerId, c.fName, c.sName
3 FROM customer c
4 JOIN shoppingcart sc ON c.customerId = sc.
  ShoppingCart_CustomerId
5 JOIN product_in_shoppingcart pisc ON sc.cartId =
  pisc.Product_in_ShoppingCart_cartId
6 JOIN product p ON pisc.
  Product_in_ShoppingCart_productId = p.
  productId
7 JOIN owned_coupon oc ON c.customerId = oc.
  Owned_Coupon_customerId
8 JOIN coupon cou ON oc.Owned_Coupon_couponId =
  cou.couponId
9 WHERE cou.endDate BETWEEN CURTIME() AND DATE_ADD
  (CURDATE(), INTERVAL 7 DAY)
10 AND p.productId = cou.Coupon_ProductId;

```

### A.2 Results

	customerId	fName	sName	totalQuantity
►	1002	Lily	Queen	3

Result of Query 1

	productId	productName
►	3	Custom Wand
	401	The Imperial Forbidden Army
	5001	Smartphone
	5002	Bluetooth Headset
	5003	HDTV
	5004	sports bike
	5005	dive watch
	5006	organic tea
	5007	Moonstone necklace
	5008	breakfast cereal
	5009	kite
	5010	Cloud cushion

Result of Query 2



	customerId	fName	sName	totalDiscount
▶	7	Harry	Potter	0.699999988079071
	49	Chao	Liu	0.800000011920929
	1001	Dan	King	0.8500000238418579
	1002	Lily	Queen	0.8999999761581421
	1003	Sunny	Day	0.800000011920929
	1004	Star	Night	0.75
	1005	River	Flow	1.399999976158142
	1006	Green	Leaf	0.8500000238418579
	1007	Moon	Light	0.800000011920929
	1008	Sun	Rise	1.7999999523162842
	1009	Wind	Whisper	0.8500000238418579
	1010	Cloud	Sky	1.600000023841858

Result of Query 3

	earliestOrderDate	customerId	fName	sName	productName	quantityPurchased
▶	2024-05-15 00:00:00.000	1001	Dan	King	Smartphone	2
	2024-05-16 14:16:13.000	1002	Lily	Queen	Bluetooth Headset	3
	2024-05-17 09:35:25.000	1003	Sunny	Day	HDTV	1
	2024-05-18 18:22:16.000	1004	Star	Night	sports bike	3
	2024-05-19 14:16:15.000	1005	River	Flow	dive watch	2
	2024-05-20 17:20:21.000	1006	Green	Leaf	organic tea	1
	2024-05-21 21:44:59.000	1007	Moon	Light	Moonstone necklace	1
	2024-05-22 16:01:43.000	1008	Sun	Rise	breakfast cereal	1
	2024-05-23 10:26:23.000	1009	Wind	Whisper	kite	2
	2024-05-24 11:45:22.000	1010	Cloud	Sky	Cloud cushion	3

Result of Query 4

	customerId	fName	sName	lastPurchaseDate
▶	7	Harry	Potter	1991-07-03 13:07:13.000
	49	Chao	Liu	2028-10-01 16:19:25.000
	1001	Dan	King	2024-05-15 00:00:00.000
	1002	Lily	Queen	2024-05-16 14:16:13.000
	1003	Sunny	Day	2024-05-17 09:35:25.000
	1004	Star	Night	2024-05-18 18:22:16.000
	1005	River	Flow	2024-05-19 14:16:15.000
	1006	Green	Leaf	2024-05-20 17:20:21.000
	1007	Moon	Light	2024-05-21 21:44:59.000
	1008	Sun	Rise	2024-05-22 16:01:43.000
	1009	Wind	Whisper	2024-05-23 10:26:23.000
	1010	Cloud	Sky	2024-05-24 11:45:22.000

Result of Query 5

	customerId	fName	sName	averagePrice
▶	7	Harry	Potter	7.0000
	49	Chao	Liu	209.0000
	1001	Dan	King	150.0000
	1002	Lily	Queen	150.0000
	1003	Sunny	Day	550.0000
	1004	Star	Night	250.0000
	1005	River	Flow	90.0000
	1006	Green	Leaf	420.0000
	1007	Moon	Light	120.0000
	1008	Sun	Rise	55.0000
	1009	Wind	Whisper	130.0000
	1010	Cloud	Sky	150.0000

Result of Query 6

	cartId	customerId	fName	sName	totalCartPrice
▶	7	7	Harry	Potter	7
	49	49	Chao	Liu	209
	201	1001	Dan	King	150
	202	1002	Lily	Queen	1480
	203	1003	Sunny	Day	8906
	204	1004	Star	Night	1230
	205	1005	River	Flow	388
	206	1006	Green	Leaf	1808
	207	1007	Moon	Light	888
	208	1008	Sun	Rise	175
	209	1009	Wind	Whisper	130
	210	1010	Cloud	Sky	8926

Result of Query 7

	customerId	fName	sName
▶	1005	River	Flow
	1006	Green	Leaf

Result of Query 8