1.book

```
CREATE TABLE Book (
  Book_ID NUMBER(10) PRIMARY KEY,
  Title VARCHAR2(100) NOT NULL,
  Author VARCHAR2(100) NOT NULL,
  Publisher VARCHAR2(100) NOT NULL,
  Publication_Date DATE NOT NULL,
  Price NUMBER(10,2) NOT NULL,
  Genre VARCHAR2(50) NOT NULL
);


INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
VALUES (1, 'The Great Gatsby', 'F. Scott Fitzgerald', 'Scribner', TO_DATE('1925-04-
10', 'YYYY-MM-DD'), 9.99, 'Classic Fiction');

INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
VALUES (2, 'To Kill a Mockingbird', 'Harper Lee', 'J.B. Lippincott & Co.',
TO_DATE('1960-07-11', 'YYYY-MM-DD'), 10.99, 'Classic Fiction');

INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
VALUES (3, '1984', 'George Orwell', 'Secker and Warburg', TO_DATE('1949-06-08',
'YYYY-MM-DD'), 8.99, 'Science Fiction');

INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
VALUES (4, 'The Catcher in the Rye', 'J.D. Salinger', 'Little, Brown and Company',
TO_DATE('1951-07-16', 'YYYY-MM-DD'), 7.99, 'Classic Fiction');

INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
VALUES (5, 'Pride and Prejudice', 'Jane Austen', 'T. Egerton, Whitehall',
TO_DATE('1813-01-28', 'YYYY-MM-DD'), 6.99, 'Romance');

INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
VALUES (6, 'The Hobbit', 'J.R.R. Tolkien', 'George Allen & Unwin', TO_DATE('1937-
09-21', 'YYYY-MM-DD'), 11.99, 'Fantasy');

INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
VALUES (7, 'The Hitchhiker''s Guide to the Galaxy', 'Douglas Adams', 'Pan Books',
TO_DATE('1979-10-12', 'YYYY-MM-DD'), 9.99, 'Science Fiction');

INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
VALUES (8, 'The Da Vinci Code', 'Dan Brown', 'Doubleday', TO_DATE('2003-03-18',
'YYYY-MM-DD'), 12.99, 'Mystery');

INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
```

```
VALUES (9, 'The Lord of the Rings', 'J.R.R. Tolkien', 'George Allen & Unwin',
TO_DATE('1954-07-29', 'YYYY-MM-DD'), 19.99, 'Fantasy');

INSERT INTO Book (Book_ID, Title, Author, Publisher, Publication_Date, Price,
Genre)
VALUES (10, 'The Chronicles of Narnia', 'C.S. Lewis', 'Geoffrey Bles',
TO_DATE('1950-10-16', 'YYYY-MM-DD'), 14.99, 'Fantasy');
```

table:

```
SELECT * FROM book
ORDER BY book_id ASC;
```

结果 解释 说明 保存的 SQL 历史记录

BOOK_ID	TITLE	AUTHOR	PUBLISHER	PUBLICATION_DATE	PRICE	GENRE
1	The Great Gatsby	F. Scott Fitzgerald	Scribner	1925-04-10	9.99	Classic Fiction
2	To Kill a Mockingbird	Harper Lee	J.B. Lippincott & Co.	1960-07-11	10.99	Classic Fiction
3	1984	George Orwell	Secker and Warburg	1949-06-08	8.99	Science Fiction
4	The Catcher in the Rye	J.D. Salinger	Little, Brown and Company	1951-07-16	7.99	Classic Fiction
5	Pride and Prejudice	Jane Austen	T. Egerton, Whitehall	1813-01-28	6.99	Romance
6	The Hobbit	J.R.R. Tolkien	George Allen & Unwin	1937-09-21	11.99	Fantasy
7	The Hitchhiker's Guide to the Galaxy	Douglas Adams	Pan Books	1979-10-12	9.99	Science Fiction
8	The Da Vinci Code	Dan Brown	Doubleday	2003-03-18	12.99	Mystery
9	The Lord of the Rings	J.R.R. Tolkien	George Allen & Unwin	1954-07-29	19.99	Fantasy
10	The Chronicles of Narnia	C.S. Lewis	Geoffrey Bles	1950-10-16	14.99	Fantasy

---

2.customer

```
CustomerCREATE TABLE Customer (
  Customer_ID NUMBER(10) PRIMARY KEY,
  First_Name VARCHAR2(50) NOT NULL,
  Last_Name VARCHAR2(50) NOT NULL,
  Email VARCHAR2(100) NOT NULL,
  Address VARCHAR2(200) NOT NULL,
  Phone_Number VARCHAR2(20) NOT NULL
);

INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (1, 'John', 'Doe', 'johndoe@example.com', '123 Main St, Anytown, USA', '555-
1234');
INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (2, 'Jane', 'Smith', 'janesmith@example.com', '456 Elm St, Anytown, USA',
'555-5678');
INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (3, 'Bob', 'Johnson', 'bobjohnson@example.com', '789 Oak St, Anytown, USA',
'555-9012');
INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (4, 'Alice', 'Williams', 'alicewilliams@example.com', '321 Pine St, Anytown,
USA', '555-3456');
INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (5, 'Mike', 'Brown', 'mikebrown@example.com', '654 Cedar St, Anytown, USA',
```

```
'555-7890');
INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (6, 'Samantha', 'Jones', 'samanthajones@example.com', '987 Maple St,
Anytown, USA', '555-2345');
INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (7, 'David', 'Lee', 'davidlee@example.com', '369 Willow St, Anytown, USA',
'555-6789');
INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (8, 'Karen', 'Davis', 'karendavis@example.com', '258 Birch St, Anytown,
USA', '555-0123');
INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (9, 'Tom', 'Wilson', 'tomwilson@example.com', '147 Oak St, Anytown, USA',
'555-4567');
INSERT INTO Customer (customer_id, first_name, last_name, Email, Address,
phone_number)
VALUES (10, 'Emily', 'Taylor', 'emilytaylor@example.com', '369 Maple St, Anytown,
USA', '555-8901');
```

```
SELECT * FROM customer
ORDER BY customer_id ASC;
```

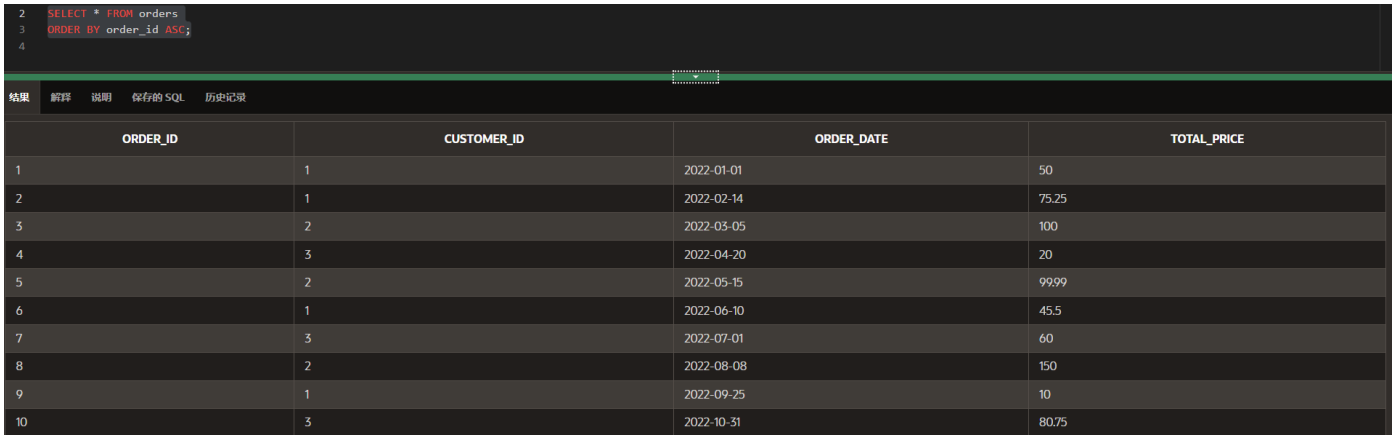
结果 解释 说明 保存的 SQL 历史记录

CUSTOMER_ID	FIRST_NAME	LAST_NAME	EMAIL	ADDRESS	PHONE_NUMBER
1	John	Doe	johndoe@example.com	123 Main St, Anytown, USA	555-1234
2	Jane	Smith	janesmith@example.com	456 Elm St, Anytown, USA	555-5678
3	Bob	Johnson	bobjohnson@example.com	789 Oak St, Anytown, USA	555-9012
4	Alice	Williams	alicewilliams@example.com	321 Pine St, Anytown, USA	555-3456
5	Mike	Brown	mikebrown@example.com	654 Cedar St, Anytown, USA	555-7890
6	Samantha	Jones	samanthajones@example.com	987 Maple St, Anytown, USA	555-2345
7	David	Lee	davidlee@example.com	369 Willow St, Anytown, USA	555-6789
8	Karen	Davis	karendavis@example.com	258 Birch St, Anytown, USA	555-0123
9	Tom	Wilson	tomwilson@example.com	147 Oak St, Anytown, USA	555-4567
10	Emily	Taylor	emilytaylor@example.com	369 Maple St, Anytown, USA	555-8901

3.order

```
CREATE TABLE Orders (
  Order_ID NUMBER(10) PRIMARY KEY,
  Customer_ID NUMBER(10) NOT NULL,
  Order_Date DATE NOT NULL,
  Total_Price NUMBER(10,2) NOT NULL,
  CONSTRAINT fk_order_customer FOREIGN KEY (Customer_ID) REFERENCES
Customer(Customer_ID)
);
INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (1, 1, TO_DATE('2022-01-01', 'YYYY-MM-DD'), 50);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (2, 1, TO_DATE('2022-02-14', 'YYYY-MM-DD'), 75.25);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (3, 2, TO_DATE('2022-03-05', 'YYYY-MM-DD'), 100.00);
```

```sql
INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (4, 3, TO_DATE('2022-04-20', 'YYYY-MM-DD'), 20.00);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (5, 2, TO_DATE('2022-05-15', 'YYYY-MM-DD'), 99.99);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (6, 1, TO_DATE('2022-06-10', 'YYYY-MM-DD'), 45.50);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (7, 3, TO_DATE('2022-07-01', 'YYYY-MM-DD'), 60.00);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (8, 2, TO_DATE('2022-08-08', 'YYYY-MM-DD'), 150.00);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (9, 1, TO_DATE('2022-09-25', 'YYYY-MM-DD'), 10.00);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (10, 3, TO_DATE('2022-10-31', 'YYYY-MM-DD'), 80.75);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (11, 2, TO_DATE('2022-11-11', 'YYYY-MM-DD'), 89.99);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (12, 1, TO_DATE('2022-12-25', 'YYYY-MM-DD'), 149.99);

INSERT INTO Orders (order_id, customer_id, order_date, total_price)
VALUES (13, 3, TO_DATE('2023-01-01', 'YYYY-MM-DD'), 19.99);
```



```sql
SELECT * FROM orders
ORDER BY order_id ASC;
```

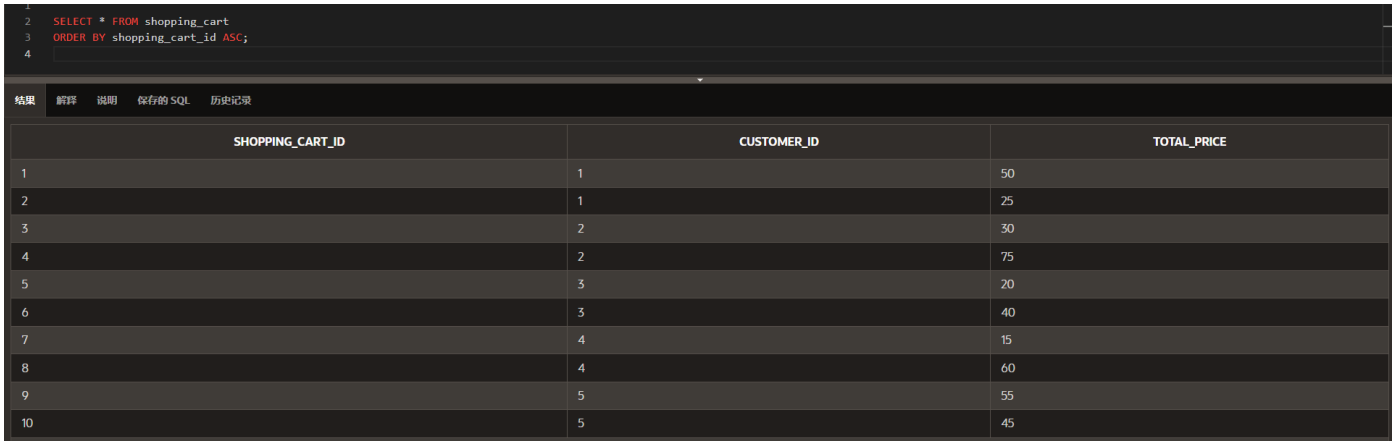
结果 解释 说明 保存的 SQL 历史记录

ORDER_ID	CUSTOMER_ID	ORDER_DATE	TOTAL_PRICE
1	1	2022-01-01	50
2	1	2022-02-14	75.25
3	2	2022-03-05	100
4	3	2022-04-20	20
5	2	2022-05-15	99.99
6	1	2022-06-10	45.5
7	3	2022-07-01	60
8	2	2022-08-08	150
9	1	2022-09-25	10
10	3	2022-10-31	80.75

4.shopping cart

```sql
CREATE TABLE Shopping_Cart (
  Shopping_Cart_ID NUMBER(10) PRIMARY KEY,
  Customer_ID NUMBER(10) NOT NULL,
  Total_Price NUMBER(10,2) NOT NULL,
  CONSTRAINT fk_shoppingcart_customer FOREIGN KEY (Customer_ID) REFERENCES
```

```
Customer(Customer_ID)
);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (1, 1, 50.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (2, 1, 25.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (3, 2, 30.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (4, 2, 75.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (5, 3, 20.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (6, 3, 40.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (7, 4, 15.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (8, 4, 60.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (9, 5, 55.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (10, 5, 45.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (11, 6, 10.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (12, 6, 30.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (13, 7, 35.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (14, 7, 20.00);

INSERT INTO shopping_cart (shopping_cart_id, customer_id, total_price)
VALUES (15, 8, 50.00);
```



```
SELECT * FROM shopping_cart
ORDER BY shopping_cart_id ASC;
```

结果 解释 说明 保存的 SQL 历史记录

SHOPPING_CART_ID	CUSTOMER_ID	TOTAL_PRICE
1	1	50
2	1	25
3	2	30
4	2	75
5	3	20
6	3	40
7	4	15
8	4	60
9	5	55
10	5	45

5.payment

```
CREATE TABLE Payment (
  Payment_ID NUMBER(10) PRIMARY KEY,
  Order_ID NUMBER(10) NOT NULL,
  Payment_Method VARCHAR2(50) NOT NULL,
  Amount NUMBER(10,2) NOT NULL,
  CONSTRAINT fk_payment_order FOREIGN KEY (Order_ID) REFERENCES Orders(Order_ID)
);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (1, 1,
'Credit Card', 50.00);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (2, 2,
'Debit Card', 75.25);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (3, 3,
'PayPal', 100.00);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (4, 4,
'Cash', 20.00);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (5, 5,
'Credit Card', 99.99);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (6, 6,
'Debit Card', 45.50);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (7, 7,
'PayPal', 60.00);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (8, 8,
'Credit Card', 150.00);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (9, 9,
'Cash', 10.00);

INSERT INTO Payment (Payment_ID, Order_ID, Payment_Method, Amount) VALUES (10, 10,
'Debit Card', 80.75);
```

```
SELECT * FROM payment
ORDER BY payment_id ASC;
```

结果 解释 说明 保存的 SQL 历史记录

PAYMENT_ID	ORDER_ID	PAYMENT_METHOD	AMOUNT
1	1	Credit Card	50
2	2	Debit Card	75.25
3	3	PayPal	100
4	4	Cash	20
5	5	Credit Card	99.99
6	6	Debit Card	45.5
7	7	PayPal	60
8	8	Credit Card	150
9	9	Cash	10
10	10	Debit Card	80.75

6.shipping

```
CREATE TABLE Shipping (
  ShippingID NUMBER(10) PRIMARY KEY,
  Order_ID NUMBER(10) NOT NULL,
  Shipping_Date DATE NOT NULL,
  Shipping_Address VARCHAR2(200) NOT NULL,
  CONSTRAINT fk_shipping_order FOREIGN KEY (Order_ID) REFERENCES Orders(Order_ID)
);

INSERT INTO Shipping (shipping_id, order_id, shipping_date, shipping_address)
VALUES (1, 1, TO_DATE('2022-02-01', 'YYYY-MM-DD'), '123 Main St, Anytown, USA');

INSERT INTO Shipping (shipping_id, order_id, shipping_date, shipping_address)
VALUES (2, 2, TO_DATE('2022-03-01', 'YYYY-MM-DD'), '456 Elm St, Anytown, USA');

INSERT INTO Shipping (shipping_id, order_id, shipping_date, shipping_address)
VALUES (3, 3, TO_DATE('2022-03-02', 'YYYY-MM-DD'), '789 Oak St, Anytown, USA');

INSERT INTO Shipping (shipping_id, order_id, shipping_date, shipping_address)
VALUES (4, 4, TO_DATE('2022-03-03', 'YYYY-MM-DD'), '321 Pine St, Anytown, USA');

INSERT INTO Shipping (shipping_id, order_id, shipping_date, shipping_address)
VALUES (5, 5, TO_DATE('2022-03-04', 'YYYY-MM-DD'), '654 Cedar St, Pine St, Anytown,
USA');
```

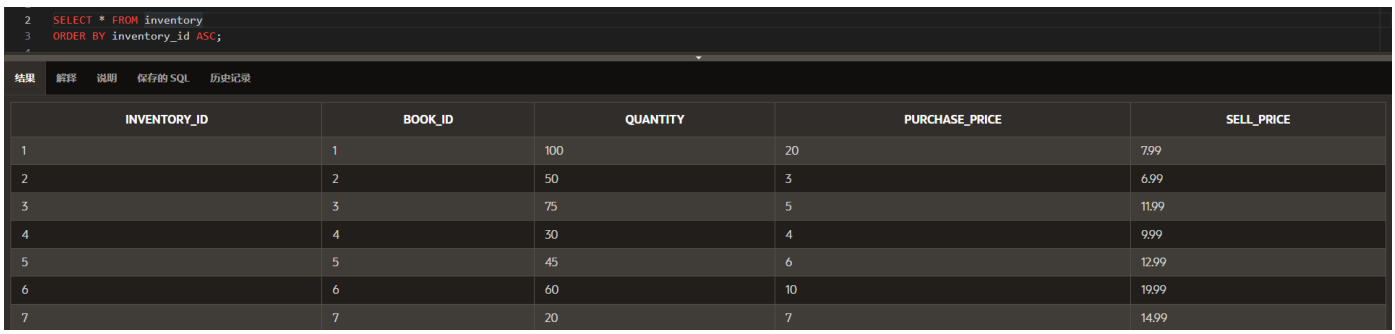
```
SELECT * FROM shipping
ORDER BY shipping_id ASC;
```

结果 解释 说明 保存的 SQL 历史记录

SHIPPING_ID	ORDER_ID	SHIPPING_DATE	SHIPPING_ADDRESS
1	1	2022-02-01	123 Main St, Anytown, USA
2	2	2022-03-01	456 Elm St, Anytown, USA
3	3	2022-03-02	789 Oak St, Anytown, USA
4	4	2022-03-03	321 Pine St, Anytown, USA
5	5	2022-03-04	654 Cedar St, Pine St, Anytown, USA

7.inventory

```sql
CREATE TABLE Inventory (
  Inventory_ID NUMBER(10) PRIMARY KEY,
  Book_ID NUMBER(10) NOT NULL,
  Quantity NUMBER(10) NOT NULL,
  Purchase_Price NUMBER(10,2) NOT NULL,
  Sell_Price NUMBER(10,2) NOT NULL,
  CONSTRAINT fk_inventory_book FOREIGN KEY (Book_ID) REFERENCES book(Book_ID)
);

INSERT INTO Inventory (inventory_id, book_id, quantity, purchase_price, sell_price)
VALUES (1, 1, 100, 20.00, 7.99);

INSERT INTO Inventory (inventory_id, book_id, quantity, purchase_price, sell_price)
VALUES (2, 2, 50, 3.00, 6.99);

INSERT INTO Inventory (inventory_id, book_id, quantity, purchase_price, sell_price)
VALUES (3, 3, 75, 5.00, 11.99);

INSERT INTO Inventory (inventory_id, book_id, quantity, purchase_price, sell_price)
VALUES (4, 4, 30, 4.00, 9.99);

INSERT INTO Inventory (inventory_id, book_id, quantity, purchase_price, sell_price)
VALUES (5, 5, 45, 6.00, 12.99);

INSERT INTO Inventory (inventory_id, book_id, quantity, purchase_price, sell_price)
VALUES (6, 6, 60, 10.00, 19.99);

INSERT INTO Inventory (inventory_id, book_id, quantity, purchase_price, sell_price)
VALUES (7, 7, 20, 7.00, 14.99);
```



```sql
SELECT * FROM inventory
ORDER BY inventory_id ASC;
```

结果 解释 说明 保存的 SQL 历史记录

INVENTORY_ID	BOOK_ID	QUANTITY	PURCHASE_PRICE	SELL_PRICE
1	1	100	20	7.99
2	2	50	3	6.99
3	3	75	5	11.99
4	4	30	4	9.99
5	5	45	6	12.99
6	6	60	10	19.99
7	7	20	7	14.99