Project Report

Deployment

Run the following lines in MySQL:

```
CREATE USER 'user'@'localhost' IDENTIFIED BY 'user';
GRANT ALL PRIVILEGES ON *.* TO 'user'@'localhost';
CREATE DATABASE book_ordering_system;
```

Open cmd on the directory of the project and run the program.

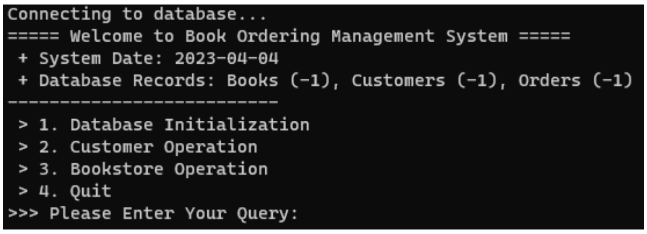
Functions

Database Initialization

1. Initialize the database
   The program creates all the necessary tables by executing the following SQL statement:

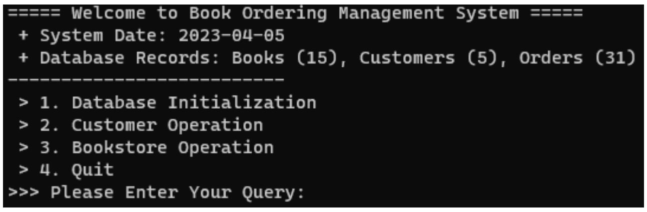
```
CREATE TABLE Customer
        (uid CHAR(10) not NULL,
         name CHAR(50) not NULL,
         address CHAR(200) not NULL,
         PRIMARY KEY ( uid ));
CREATE TABLE Book
        (isbn CHAR(13) not NULL,
         title CHAR(100) not NULL,
         price INTEGER,
         inventory_quantity INTEGER,
         PRIMARY KEY ( isbn ));
CREATE TABLE Orders
        (oid CHAR(8) not NULL,
         uid CHAR(10) not NULL,
         isbn CHAR(13) not NULL,
         order_date DATE,
         order_quantity INTEGER,
         shipping_status CHAR(8),
         FOREIGN KEY ( uid ) REFERENCES Customer( uid ),
         FOREIGN KEY ( isbn ) REFERENCES Book( isbn ),
         PRIMARY KEY ( oid, uid, isbn ));
CREATE TABLE Author
        (aid CHAR(10) not NULL,
         aname CHAR(50) not NULL,
         PRIMARY KEY ( aid ));
CREATE TABLE Writes
        (isbn CHAR(13) not NULL,
         aid CHAR(10) not NULL,
         FOREIGN KEY ( isbn ) REFERENCES Book( isbn ),
         FOREIGN KEY ( aid ) REFERENCES Author( aid ),
         PRIMARY KEY ( isbn, aid ))
```

Before initialization:



```
Connecting to database...
===== Welcome to Book Ordering Management System =====
 + System Date: 2023-04-04
 + Database Records: Books (-1), Customers (-1), Orders (-1)
--------------------------
 > 1. Database Initialization
 > 2. Customer Operation
 > 3. Bookstore Operation
 > 4. Quit
>>> Please Enter Your Query:
```

After initialization:

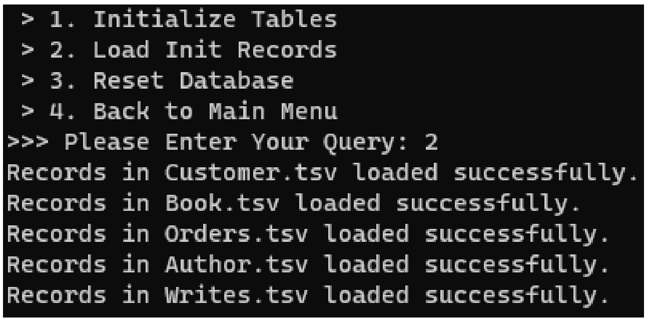


```
===== Welcome to Book Ordering Management System =====
 + System Date: 2023-04-05
 + Database Records: Books (15), Customers (5), Orders (31)
--------------------------
 > 1. Database Initialization
 > 2. Customer Operation
 > 3. Bookstore Operation
 > 4. Quit
>>> Please Enter Your Query:
```

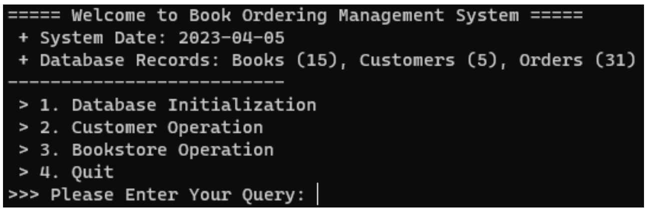
2. Load Init Records

   The program loads the records in the tsv files to the corresponding tables.

```
INSERT INTO Customer (uid,name,address) VALUES (?, ?, ?);
INSERT INTO Book (isbn, title, price, inventory_quantity) VALUES (?, ?, ?, ?);
INSERT INTO Orders (oid, uid, isbn, order_date, order_quantity, shipping_status) VALUES (?, ?, ?, ?, ?, ?);
INSERT INTO Author (aid, aname) VALUES (?, ?);
INSERT INTO Writes (isbn, aid) VALUES (?, ?);
```



```
 > 1. Initialize Tables
 > 2. Load Init Records
 > 3. Reset Database
 > 4. Back to Main Menu
>>> Please Enter Your Query: 2
Records in Customer.tsv loaded successfully.
Records in Book.tsv loaded successfully.
Records in Orders.tsv loaded successfully.
Records in Author.tsv loaded successfully.
Records in Writes.tsv loaded successfully.
```



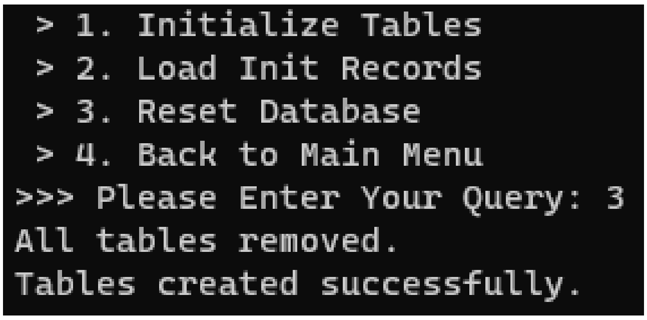
```
===== Welcome to Book Ordering Management System =====
 + System Date: 2023-04-05
 + Database Records: Books (15), Customers (5), Orders (31)
--------------------------
 > 1. Database Initialization
 > 2. Customer Operation
 > 3. Bookstore Operation
 > 4. Quit
>>> Please Enter Your Query: |
```

   The records are successfully inserted as shown. Note: the files are placed at ./tsv/, make sure the terminal is at the correct directory.

3. Reset Database

   The program resets the database by dropping all the tables then creating the tables again.

```sql
DROP TABLE Orders;
DROP TABLE Writes;
DROP TABLE Author;
DROP TABLE Book;
DROP TABLE Customer;
```

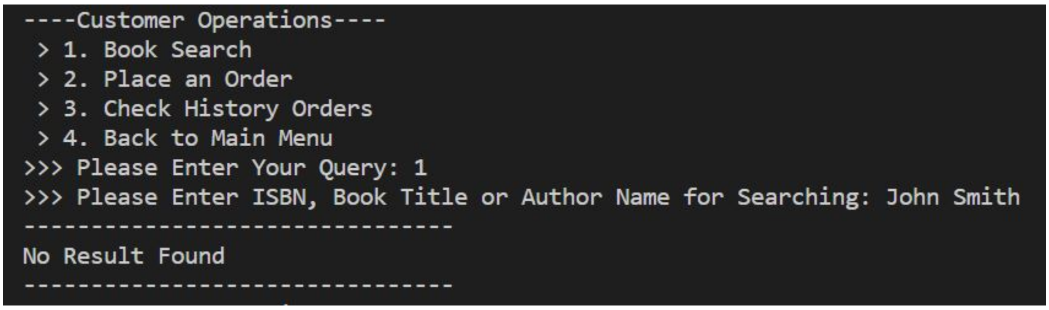


```
 > 1. Initialize Tables
 > 2. Load Init Records
 > 3. Reset Database
 > 4. Back to Main Menu
>>> Please Enter Your Query: 3
All tables removed.
Tables created successfully.
```

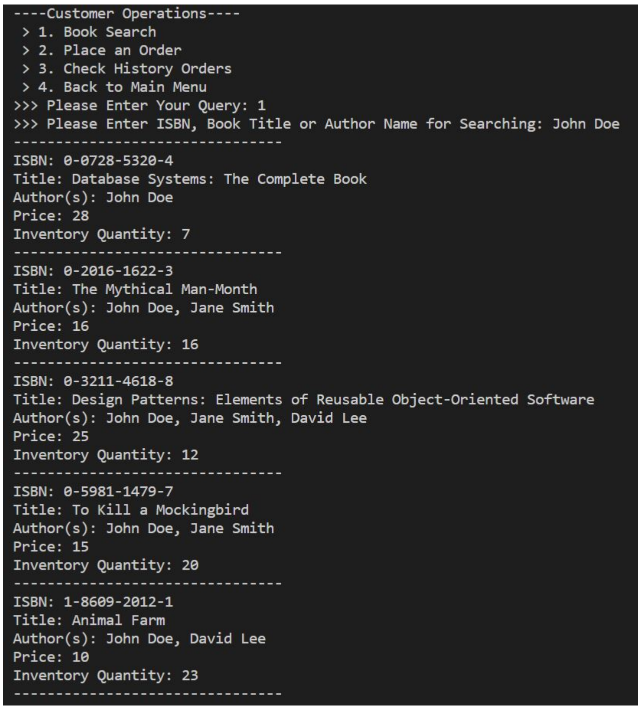
Customer Operation

1. Book Search

   The program allows users to search books by ISBN, book title or author name.



```
----Customer Operations----
 > 1. Book Search
 > 2. Place an Order
 > 3. Check History Orders
 > 4. Back to Main Menu
>>> Please Enter Your Query: 1
>>> Please Enter ISBN, Book Title or Author Name for Searching: John Smith
-------------------------------
No Result Found
-------------------------------
```

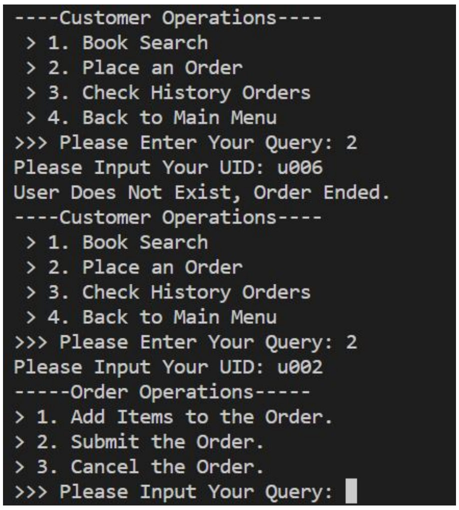


```
----Customer Operations----
 > 1. Book Search
 > 2. Place an Order
 > 3. Check History Orders
 > 4. Back to Main Menu
>>> Please Enter Your Query: 1
>>> Please Enter ISBN, Book Title or Author Name for Searching: John Doe
-------------------------------
ISBN: 0-0728-5320-4
Title: Database Systems: The Complete Book
Author(s): John Doe
Price: 28
Inventory Quantity: 7
-------------------------------
ISBN: 0-2016-1622-3
Title: The Mythical Man-Month
Author(s): John Doe, Jane Smith
Price: 16
Inventory Quantity: 16
-------------------------------
ISBN: 0-3211-4618-8
Title: Design Patterns: Elements of Reusable Object-Oriented Software
Author(s): John Doe, Jane Smith, David Lee
Price: 25
Inventory Quantity: 12
-------------------------------
ISBN: 0-5981-1479-7
Title: To Kill a Mockingbird
Author(s): John Doe, Jane Smith
Price: 15
Inventory Quantity: 20
-------------------------------
ISBN: 1-8609-2012-1
Title: Animal Farm
Author(s): John Doe, David Lee
Price: 10
Inventory Quantity: 23
-------------------------------
```

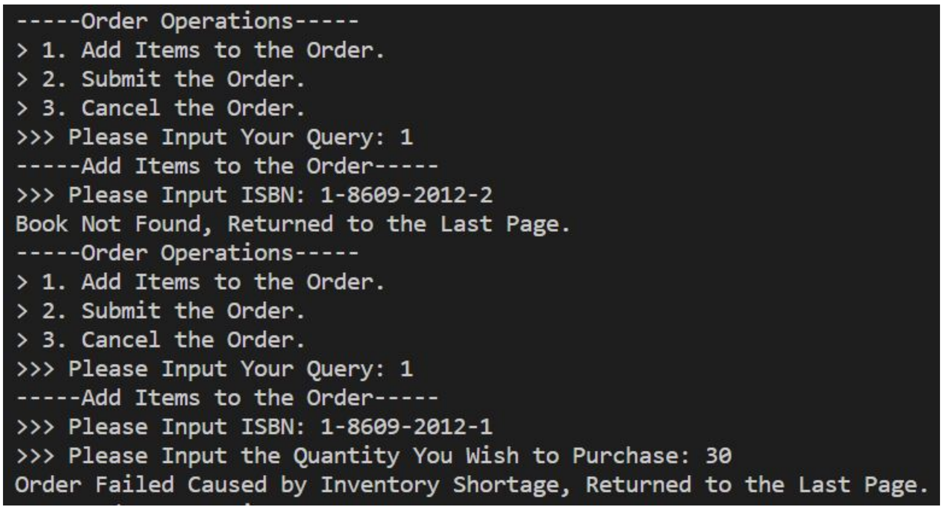
```sql
SELECT DISTINCT B.isbn, B.title, B.price, B.inventory_quantity
FROM Author A, Writes W, Book B
WHERE W.aid = A.aid AND W.isbn = B.isbn AND A.aname = keyword
```

```
UNION
SELECT DISTINCT B.isbn, B.title, B.price, B.inventory_quantity
FROM Author A, Writes W, Book B
WHERE W.aid = A.aid AND W.isbn = B.isbn AND B.title = keyword
UNION
SELECT DISTINCT B.isbn, B.title, B.price, B.inventory_quantity
FROM Author A, Writes W, Book B
WHERE W.aid = A.aid AND W.isbn = B.isbn AND B.isbn = keyword;
SELECT DISTINCT A.aname FROM Author A, Writes W WHERE A.aid = W.aid AND
W.isbn = isbn;
```

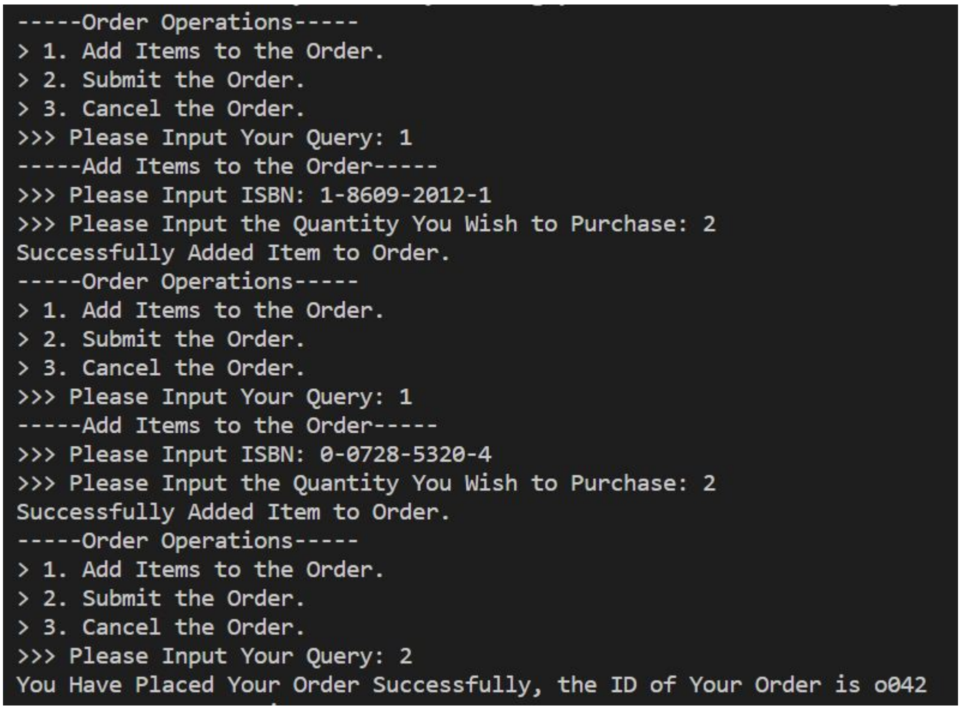
2. Place an Order
   User can add different books with different quantity to the order, submit the order or cancel the order.



```
----Customer Operations----
 > 1. Book Search
 > 2. Place an Order
 > 3. Check History Orders
 > 4. Back to Main Menu
>>> Please Enter Your Query: 2
Please Input Your UID: u006
User Does Not Exist, Order Ended.
----Customer Operations----
 > 1. Book Search
 > 2. Place an Order
 > 3. Check History Orders
 > 4. Back to Main Menu
>>> Please Enter Your Query: 2
Please Input Your UID: u002
-----Order Operations-----
> 1. Add Items to the Order.
> 2. Submit the Order.
> 3. Cancel the Order.
>>> Please Input Your Query:
```



```
-----Order Operations-----
> 1. Add Items to the Order.
> 2. Submit the Order.
> 3. Cancel the Order.
>>> Please Input Your Query: 1
-----Add Items to the Order-----
>>> Please Input ISBN: 1-8609-2012-2
Book Not Found, Returned to the Last Page.
-----Order Operations-----
> 1. Add Items to the Order.
> 2. Submit the Order.
> 3. Cancel the Order.
>>> Please Input Your Query: 1
-----Add Items to the Order-----
>>> Please Input ISBN: 1-8609-2012-1
>>> Please Input the Quantity You Wish to Purchase: 30
Order Failed Caused by Inventory Shortage, Returned to the Last Page.
```

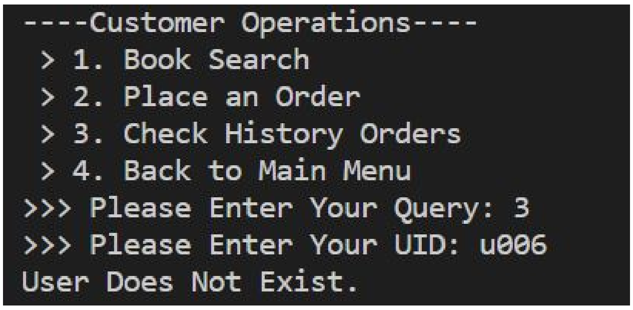


```
-----Order Operations-----
> 1. Add Items to the Order.
> 2. Submit the Order.
> 3. Cancel the Order.
>>> Please Input Your Query: 1
-----Add Items to the Order-----
>>> Please Input ISBN: 1-8609-2012-1
>>> Please Input the Quantity You Wish to Purchase: 2
Successfully Added Item to Order.
-----Order Operations-----
> 1. Add Items to the Order.
> 2. Submit the Order.
> 3. Cancel the Order.
>>> Please Input Your Query: 1
-----Add Items to the Order-----
>>> Please Input ISBN: 0-0728-5320-4
>>> Please Input the Quantity You Wish to Purchase: 2
Successfully Added Item to Order.
-----Order Operations-----
> 1. Add Items to the Order.
> 2. Submit the Order.
> 3. Cancel the Order.
>>> Please Input Your Query: 2
You Have Placed Your Order Successfully, the ID of Your Order is o042
```

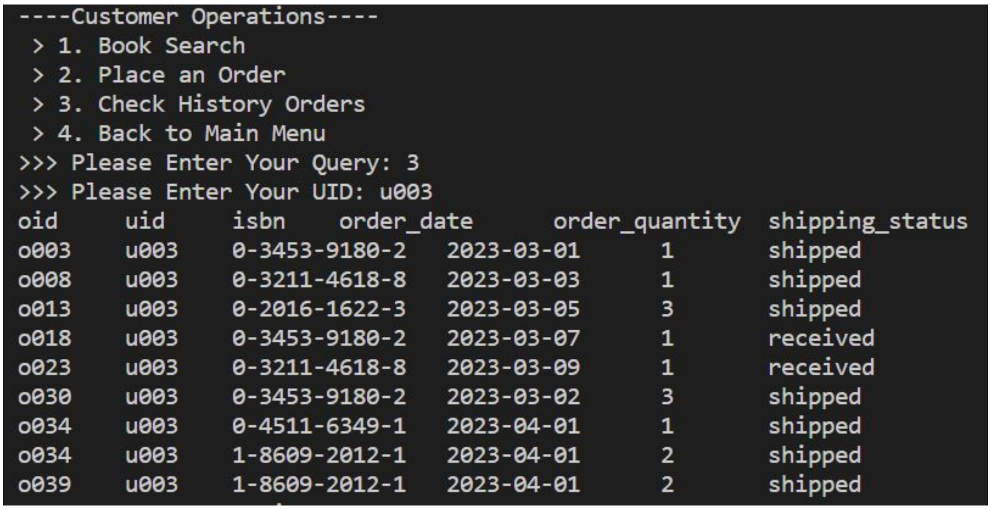
```sql
SELECT * FROM Customer C WHERE C.uid = uid;
SELECT * FROM Book B WHERE B.isbn = isbn;
SELECT B.inventory_quantity FROM Book B WHERE B.isbn = isbn;
SELECT DISTINCT O.oid FROM Orders O;
INSERT INTO Orders (oid, uid, isbn, order_date, order_quantity,
shipping_status)
VALUES (?, ?, ?, ?, ?, ?);
```

3. Check History Orders

   The program allows users to check their history orders by entering their uid.



```
----Customer Operations----
 > 1. Book Search
 > 2. Place an Order
 > 3. Check History Orders
 > 4. Back to Main Menu
>>> Please Enter Your Query: 3
>>> Please Enter Your UID: u006
User Does Not Exist.
```



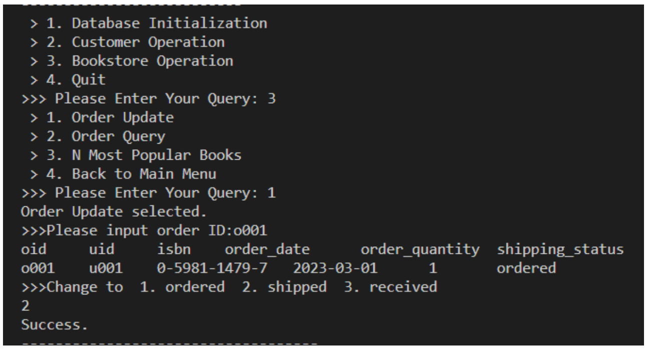
```
----Customer Operations----
 > 1. Book Search
 > 2. Place an Order
 > 3. Check History Orders
 > 4. Back to Main Menu
>>> Please Enter Your Query: 3
>>> Please Enter Your UID: u003
oid     uid     isbn     order_date      order_quantity  shipping_status
o003    u003    0-3453-9180-2   2023-03-01      1       shipped
o008    u003    0-3211-4618-8   2023-03-03      1       shipped
o013    u003    0-2016-1622-3   2023-03-05      3       shipped
o018    u003    0-3453-9180-2   2023-03-07      1       received
o023    u003    0-3211-4618-8   2023-03-09      1       received
o030    u003    0-3453-9180-2   2023-03-02      3       shipped
o034    u003    0-4511-6349-1   2023-04-01      1       shipped
o034    u003    1-8609-2012-1   2023-04-01      2       shipped
o039    u003    1-8609-2012-1   2023-04-01      2       shipped
```

```
SELECT * FROM Orders O WHERE O.uid = uid;
```

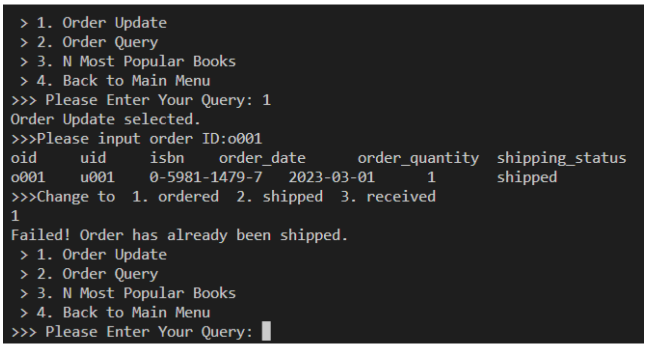
Bookstore Operation

1. Order Update

   The program allows users to update the shipping status of an order.



```
 > 1. Database Initialization
 > 2. Customer Operation
 > 3. Bookstore Operation
 > 4. Quit
>>> Please Enter Your Query: 3
 > 1. Order Update
 > 2. Order Query
 > 3. N Most Popular Books
 > 4. Back to Main Menu
>>> Please Enter Your Query: 1
Order Update selected.
>>>Please input order ID:o001
oid     uid     isbn     order_date      order_quantity  shipping_status
o001    u001    0-5981-1479-7   2023-03-01      1       ordered
>>>Change to  1. ordered  2. shipped  3. received
2
Success.
-----------------------------------
```



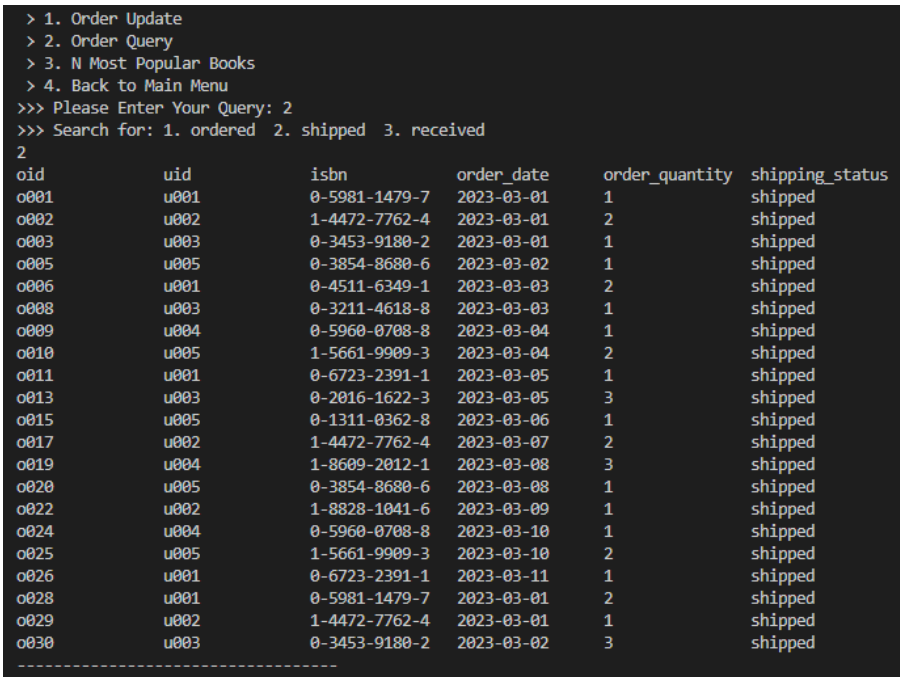
```
 > 1. Order Update
 > 2. Order Query
 > 3. N Most Popular Books
 > 4. Back to Main Menu
>>> Please Enter Your Query: 1
Order Update selected.
>>>Please input order ID:o001
oid     uid     isbn     order_date      order_quantity  shipping_status
o001    u001    0-5981-1479-7   2023-03-01      1       shipped
>>>Change to  1. ordered  2. shipped  3. received
1
Failed! Order has already been shipped.
 > 1. Order Update
 > 2. Order Query
 > 3. N Most Popular Books
 > 4. Back to Main Menu
>>> Please Enter Your Query:
```

```
UPDATE Orders
SET shipping_status = status
```

```
WHERE O.oid = oid;
```

2. Order Query

   The program allows users to query all the order grouped by shipping status.

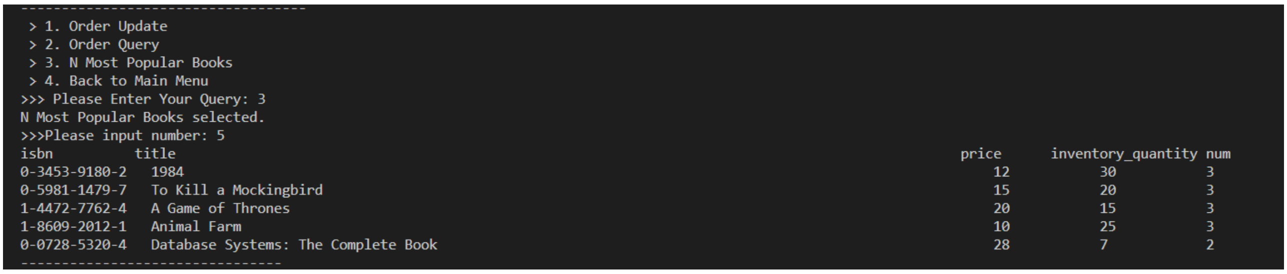


```
 > 1. Order Update
 > 2. Order Query
 > 3. N Most Popular Books
 > 4. Back to Main Menu
>>> Please Enter Your Query: 2
>>> Search for: 1. ordered  2. shipped  3. received
2
oid            uid            isbn           order_date     order_quantity shipping_status
o001           u001           0-5981-1479-7  2023-03-01     1              shipped
o002           u002           1-4472-7762-4  2023-03-01     2              shipped
o003           u003           0-3453-9180-2  2023-03-01     1              shipped
o005           u005           0-3854-8680-6  2023-03-02     1              shipped
o006           u001           0-4511-6349-1  2023-03-03     2              shipped
o008           u003           0-3211-4618-8  2023-03-03     1              shipped
o009           u004           0-5960-0708-8  2023-03-04     1              shipped
o010           u005           1-5661-9909-3  2023-03-04     2              shipped
o011           u001           0-6723-2391-1  2023-03-05     1              shipped
o013           u003           0-2016-1622-3  2023-03-05     3              shipped
o015           u005           0-1311-0362-8  2023-03-06     1              shipped
o017           u002           1-4472-7762-4  2023-03-07     2              shipped
o019           u004           1-8609-2012-1  2023-03-08     3              shipped
o020           u005           0-3854-8680-6  2023-03-08     1              shipped
o022           u002           1-8828-1041-6  2023-03-09     1              shipped
o024           u004           0-5960-0708-8  2023-03-10     1              shipped
o025           u005           1-5661-9909-3  2023-03-10     2              shipped
o026           u001           0-6723-2391-1  2023-03-11     1              shipped
o028           u001           0-5981-1479-7  2023-03-01     2              shipped
o029           u002           1-4472-7762-4  2023-03-01     1              shipped
o030           u003           0-3453-9180-2  2023-03-02     3              shipped
-----------------------------------
```

```
SELECT *
FROM Orders O
WHERE shipping_status = status;
```

3. N Most Popular Books

   The program allows users to check the most popular book by entering the number that you want to show.



```
-----------------------------------
 > 1. Order Update
 > 2. Order Query
 > 3. N Most Popular Books
 > 4. Back to Main Menu
>>> Please Enter Your Query: 3
N Most Popular Books selected.
>>>Please input number: 5
isbn          title                                                                                     price     inventory_quantity num
0-3453-9180-2   1984                                                                                        12           30               3
0-5981-1479-7   To Kill a Mockingbird                                                                       15           20               3
1-4472-7762-4   A Game of Thrones                                                                           20           15               3
1-8609-2012-1   Animal Farm                                                                                 10           25               3
0-0728-5320-4   Database Systems: The Complete Book                                                         28           7                2
--------------------------------
```

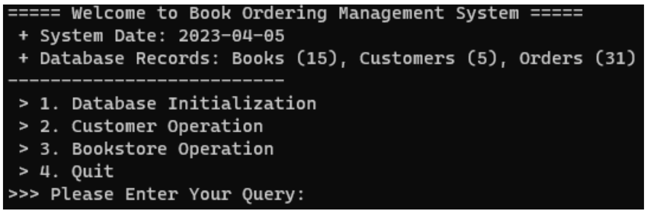
```
SELECT B.isbn, B.title, B.price, COUNT(O.oid) AS num
FROM Book B, Orders O
WHERE B.isbn = O.isbn
GROUP BY B.isbn
ORDER BY num DESC
LIMIT N;
```

Other Utilities

1. Print Current DateTime
   The system datetime is shown on the main menu.

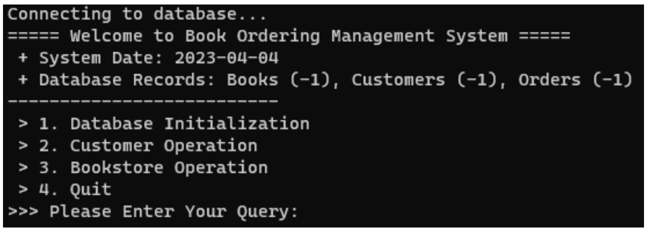
2. Print Database Overview The system displays the number of records in the Book, Customer, Orders table respectively.

   ```
   SELECT COUNT(*) FROM table_name;
   ```

   ```
   ===== Welcome to Book Ordering Management System =====
    + System Date: 2023-04-05
    + Database Records: Books (15), Customers (5), Orders (31)
   --------------------------
    > 1. Database Initialization
    > 2. Customer Operation
    > 3. Bookstore Operation
    > 4. Quit
   >>> Please Enter Your Query:
   ```

   -1 is displayed if such table does not exist.

   ```
   Connecting to database...
   ===== Welcome to Book Ordering Management System =====
    + System Date: 2023-04-04
    + Database Records: Books (-1), Customers (-1), Orders (-1)
   --------------------------
    > 1. Database Initialization
    > 2. Customer Operation
    > 3. Bookstore Operation
    > 4. Quit
   >>> Please Enter Your Query:
   ```

3. Control and Navigation
   The user should input '4' to navigate to the previous page or quit the program.

4. Change shipping status
   The system changes the shipping status of the orders from "ordered" to "shipped" every 30 seconds.