

COMP3005 Project Report

Tom Lam – 101114541

1. Conceptual design

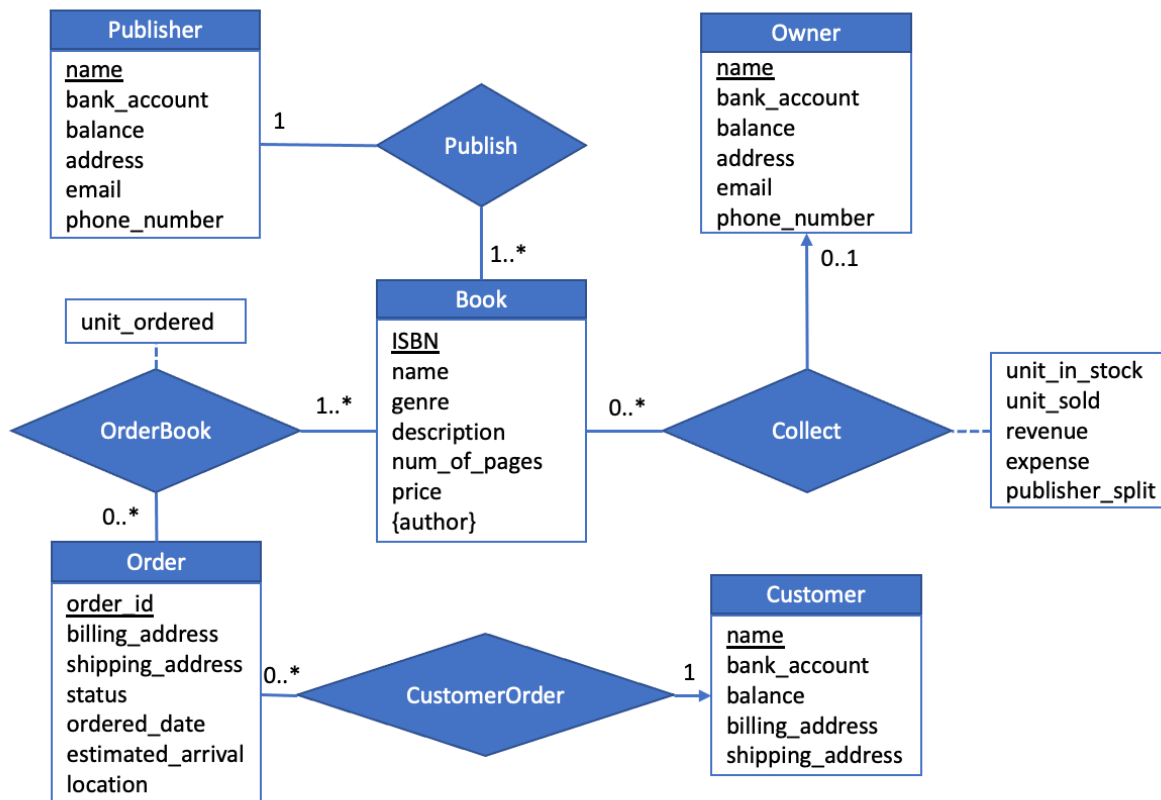
Cardinality:

- Participation type is indicated by the number of participants next to each entity in the ER diagram. Specifically, “1..*” or “1” means total participation while “0..*” or “0..1” means partial participation
- A book can be published by exactly 1 publisher, A publisher can publish 1 to many books
- A book can be collected by 0 to 1 owner, an owner can collect 0 to many book. There’s no book that belongs to 2 different owner
- A book can be ordered by 0 to many orders, an order can order 1 to many book
- An order can be placed by exactly 1 customer, a customer can place 0 to many order

Others:

- Different authors have different name (to avoid the use of author_id)
- Different publishes have different name (to avoid the use of publisher_id)
- Different customer has different name (to avoid the use of customer_id)

ER Diagram



2. Relational Schema

Book(ISBN, publisher_name, book_name, genre, description, num_of_pages, price)
Author(ISBN, name)
Publisher(name, email, bank_account, balance, address, phone_number)
Owner(name, bank_account, balance, address, email, phone_number)
Collect(ISBN, owner_name, unit_in_stock, unit_sold, revenue, expense, publisher_split)
Order(order_id, billing_address, shipping_address, status, ordered_date, estimated_arrivate, location)
Customer(name, bank_account, balance, billing_address, shipping_address)
CustomerOrder(order_id, customer_name)
OrderBook(ISBN, order_id, unit_ordered)

3. Normalization

Functional dependencies

Book(ISBN, publisher_name, book_name, genre, description, num_of_pages, price)
ISBN → name, publisher_name, book_name, genre, description, num_of_pages, price

Author(ISBN, name)
ISBN, name → ISBN, name

Publisher(name, email, bank_account, balance, address, phone_number)
name → email, bank_account, balance, address, email, phone_number

Owner(name, bank_account, balance, address, email, phone_number)
name → bank_account, balance, address, email, phone_number

Collect(ISBN, owner_name, unit_in_stock, unit_sold, revenue, expense, publisher_split)
ISBN → owner_name, unit_in_stock, unit_sold, revenue, expense, publisher_split

Order(order_id, billing_address, shipping_address, status, ordered_date, estimated_arrivate, location)
order_id → billing_address, shipping_address, status, ordered_date, estimated_arrivate, location

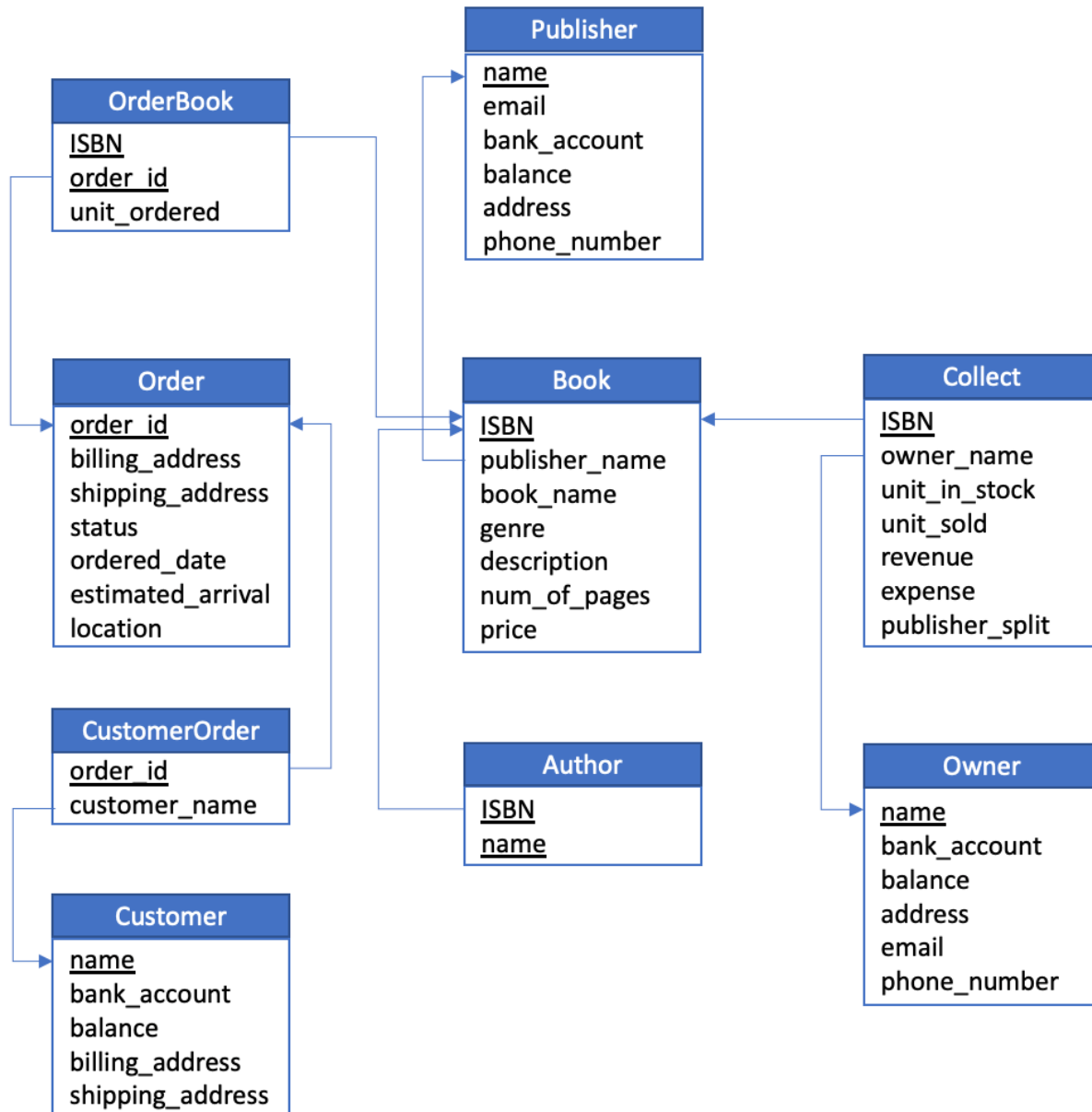
Customer(name, bank_account, balance, billing_address, shipping_address)
name → bank_account, balance, billing_address, shipping_address

CustomerOrder(order_id, customer_name)
order_id → customer_name

OrderBook(ISBN, order_id, unit_ordered)
ISBN, order_id → unit_ordered

All relations are already in BCNF because their functional dependency is also the superkey

4. Database Schema Diagram



5. Implementation:

5.1. Architectures:

The table creation “table.sql” and insertion “insertion.sql” files can be found in the “database_initialize” directory.

The program entry point is in Main.java. It initializes the Application and run it.

The functionalities are implemented by 3 main classes:

- View.java handles all interaction between the store and its users (Customer or Owner)
- JDBCController.java handles all interaction with the store and the database. This is where all database queries and updates are implemented
- Application.java implements all application requirements stated in the “**Problem Statement**” of Look-Inna-Book by using the View and JDBCController classes to interact with the store users with the store databases

The remainings are helper classes of which main functionalities are storage and display (toString)

- Customer.java stores informations about a customer
- Owner.java stores information about an owner
- Book.java stores temporary book information got from the database
- BookOrder.java stores a Customer’s chosen Book with the number of that book that’s going to be ordered when checkout.
- Basket.java stores Customer’s BookOrder(s) before checkout. Basket will be emptied after checkout or restart
- Order.java stores an order information when tracking an order

5.2. Program initialization:

1. Create the database and initialize it by running everything inside “table.sql” and “insertion.sql”.
2. In Main.java. Please change the following paramenters
PORT: database port
DATABASE_NAME: name of created database
USERNAME: user name to log into database
PASSWORD: password of the username
to the appropriate values so that JDBCController can connects to the database

5.3. Interface:

Main views:

When first started, you will be greeted with the Main view and required to choose either Customer or Owner view:

```
-----  
Welcome to LookInnaBook  
-----  
Menu: (type the number to choose)  
(1) Show Customer View  
(2) Show Owner View  
(0) Exit  
-----  
Enter your choice: █
```

Customer view (not logged in):

```
Enter your choice: 1  
  
-----  
Customer  
-----  
You are not logged in!  
  
Menu: (type the number to choose)  
(1) Browse book  
(2) Add book to basket  
(3) Remove book from basket  
(4) Show basket  
(5) Log in  
(0) Exit Customer View  
-----  
Enter your choice: █
```

Owner view (not logged in):

```
Enter your choice: 2  
  
-----  
Owner  
-----  
You are not logged in!  
  
Menu: (type the number to choose)  
(1) Browse unowned books  
(2) Log in  
(0) Exit Owner View  
-----  
Enter your choice: █
```

You are freely allow to switch between view by exiting back to the Main view.

Customer View:

When not logged in, the customer can browse books, add book to basket, remove book from basket, show basket, and log in.

```
Enter your choice: 1
-----
                        Customer
-----
You are not logged in!

Menu: (type the number to choose)
(1) Browse book
(2) Add book to basket
(3) Remove book from basket
(4) Show basket
(5) Log in
(0) Exit Customer View
-----

Enter your choice: █
```

When logged in, the customer gains the ability to checkout the basket, track order, and log out. And of course, loses the ability to log in.

```
Enter your choice: 5
Enter the name of the customer to log in:
Enter your choice: Customer1

Customer logged in successfully!

-----
                        Customer
-----
You are logged in
Customer information:
    name: Customer1
    billing address: BillingAddress1
    shipping address: ShippingAddress1
    balance: $98800.00

Menu: (type the number to choose)
(1) Browse book that are for sale
(2) Add book to basket
(3) Remove book from basket
(4) Show basket
(5) Track orders
(6) Checkout
(7) Log out
(0) Exit Customer View
-----

Enter your choice: █
```

The default insertion file “insertion.sql” gives 5 Customers to choose from: Customer1, Customer2, Customer3, Customer4, Customer5

Browsing books, adding and removing to/from basket works the same regardless of whether the user is logged in or not. More over, when logging in or logging out, the basket is not reseted.

Customer Browse Book:

When choose to browse book, you will be brought to the Browse Book view. Which gives to option to search using the filters, show, clear, and modifies the filters.

```
Enter your choice: 1
-----
                        Browse Book
-----
Browse book menu
(1) Search using the current filter
(2) Show all filters
(3) Clear all filters
(4) Add to Book Name filter
(5) Add to ISBN filter
(6) Add to Author filter
(7) Add to Genre filter
(8) Add to Publisher filter
(0) Exit from browse book view
-----

Enter your choice: █
```

Initially, all filters are empty:

```
Enter your choice: 2
Current Book Name Filter: []
Current ISBN Filter: []
Current Author Filter: []
Current Genre Filter: []
Current Publisher Filter: []
```

Add filters to search for books that are written by Author3 and published by either Publisher2 or Publisher1:

```
Enter your choice: 6
Available Authors:
(0) Author1
(1) Author10
(2) Author2
(3) Author3
(4) Author4
(5) Author5
(6) Author6
(7) Author7
(8) Author8
(9) Author9
Enter the index number (in the range 0 to 9):
Enter your choice: 3
```

```
Enter your choice: 8
Available Publishers:
(0) publisher0
(1) publisher1
(2) publisher2
Enter the index number (in the range 0 to 2):
Enter your choice: 2
```

```
Enter your choice: 8
Available Publishers:
(0) publisher0
(1) publisher1
(2) publisher2
Enter the index number (in the range 0 to 2):
Enter your choice: 0
```

Resulting filters:

```
Enter your choice: 2
Current Book Name Filter: []
Current ISBN Filter: []
Current Author Filter: [Author3]
Current Genre Filter: []
Current Publisher Filter: [publisher0, publisher2]
```

When choose to search for book, the application will show the books that are collected by an owner after applying all of the filters added above

```
Enter your choice: 1
select distinct ISBN from Book natural join Author where ISBN in (select ISBN from
Collect) and author.name in ('Author3') and publisher_name in ('publisher0','publ
isher2')order by ISBN;
Here are the books for sale:
ISBN: 3
Book name: Book3
Publisher: publisher0
Genre: Genre1
Description: Description3
Number of pages: 150
Price: $12.000000
Authors: Author3
```

Similarly, to browse for book by genre, publisher, book name, ISBN, adjust the corresponding filters

Add/remove book order to/from customer basket

Choose the ISBN from the list printed and the quantity. For example:

Add 10 books with ISBN 3:

```
Enter your choice: 2
List of Book ISBNs to add:
  ISBN: 3
  name: Book3
  price per book: 12.00

  ISBN: 4
  name: Book4
  price per book: 15.00

  ISBN: 5
  name: Book5
  price per book: 13.00

  ISBN: 7
  name: Book7
  price per book: 18.00

  ISBN: 8
  name: Book8
  price per book: 20.00

  ISBN: 9
  name: Book9
  price per book: 40.00
```

```
What is the ISBN of the book that you want to order?
Enter your choice: 3
How many do you want to order? (0 if you change your
Enter your choice: 10
```

Add 10 books with ISBN 5:

```
Enter your choice: 2
List of Book ISBNs to add:
  ISBN: 3
  name: Book3
  price per book: 12.00

  ISBN: 4
  name: Book4
  price per book: 15.00

  ISBN: 5
  name: Book5
  price per book: 13.00

  ISBN: 7
  name: Book7
  price per book: 18.00

  ISBN: 8
  name: Book8
  price per book: 20.00

  ISBN: 9
  name: Book9
  price per book: 40.00
```

```
What is the ISBN of the book that you want to order?
Enter your choice: 5
How many do you want to order? (0 if you change your
Enter your choice: 10
```

The basket information after adding:

```
Enter your choice: 4
Basket Information:
Unit order: 10
Book info:
  ISBN: 3
  Name: Book3
  Unit Price: $12.00
  Subtotal for this book: $120.00

Unit order: 10
Book info:
  ISBN: 5
  Name: Book5
  Unit Price: $13.00
  Subtotal for this book: $130.00

Your order total is: 250.00
```


Similarly, to remove, choose the ISBN and quantity to remove, if the quantity exceeds the current quantity in the basket. The application will completely remove the entry

```
Enter your choice: 3
Basket Information:
Unit order: 10
Book info:
    ISBN: 3
    Name: Book3
    Unit Price: $12.00
    Subtotal for this book: $120.00

Unit order: 10
Book info:
    ISBN: 5
    Name: Book5
    Unit Price: $13.00
    Subtotal for this book: $130.00

Your order total is: 250.00
What is the ISBN of the book that you want to remove? (0 to
Enter your choice: 3
How many do you want to remove? (0 if you change your mind)
Enter your choice: 10

Enter your choice: 3
Basket Information:
Unit order: 10
Book info:
    ISBN: 5
    Name: Book5
    Unit Price: $13.00
    Subtotal for this book: $130.00

Your order total is: 130.00
What is the ISBN of the book that you want to remove? (0 to
Enter your choice: 5
How many do you want to remove? (0 if you change your mind)
Enter your choice: 1

Enter your choice: 4
Basket Information:
Unit order: 9
Book info:
    ISBN: 5
    Name: Book5
    Unit Price: $13.00
    Subtotal for this book: $117.00

Your order total is: 117.00
```

Customer check out:

The application will show your basket information and prompt to enter your billing and shipping address. If you leave it blank, the store will use the default value stored in your profile. In this case, the default billing address is "BillingAddress1" for Customer1

```
Enter your choice: 6
Basket Information:
Unit order: 9
Book info:
    ISBN: 5
    Name: Book5
    Unit Price: $13.00
    Subtotal for this book: $117.00

Your order total is: 117.00
What is your billing address (leave empty to use default)
Enter your choice:

What is your shipping address (leave empty to use default)
Enter your choice: somewhere on earth

Order placed successful!
```

The basket will be emptied and your customer balance will be updated after placing the order.

Before, the balance was \$98800

After, it's \$98800 – \$117 = \$98683

```
-----
                        Customer
-----
You are logged in
Customer information:
    name: Customer1
    billing address: BillingAddress1
    shipping address: ShippingAddress1
    balance: $98683.00

Menu: (type the number to choose)
(1) Browse book that are for sale
(2) Add book to basket
(3) Remove book from basket
(4) Show basket
(5) Track orders
(6) Checkout
(7) Log out
(0) Exit Customer View
-----

Enter your choice: 4
Basket Information:
Your basket is empty
Your order total is: 0.00
```

After placing order, JDBCController will automatically order more books from Publisher if the number of books falls under the fixed MIN_BOOK_THRESHOLD (50).

Customer track order:

You will be showed the IDs of your placed order. When prompted, enter one of the ID and the application will show all information about that order

```
Enter your choice: 5
IDs of order placed:
    1
    2
What order do you want to track?
Enter your choice: 2
Information about order 2:
Order ID: 2
Billing address: BillingAddress1
Shipping address: somewhere on earth
Status: Order placed
Ordered data: 2021-12-19
Estimated arrival date: 2021-12-09
Current location: Warehouse
Basket Information:
Unit order: 9
Book info:
    ISBN: 5
    Name: Book5
    Unit Price: $13.00
    Subtotal for this book: $117.00

Your order total is: 117.00

Do you want to track another order (yes or no)?
Enter your choice: no
```

Owner View:

When not logged in, owner can only browse book and log in

```
-----
                        Owner
-----

You are not logged in!

Menu: (type the number to choose)
(1) Browse unowned books
(2) Log in
(0) Exit Owner View
-----

Enter your choice: █
```

The default insertion file "insertion.sql" gives 2 owner "Owner1" and "Owner2" to log in from. When logged in, owner gains additional ability

```
Enter your choice: 2
Enter the name of the owner to log in:
Enter your choice: Owner1

Owner logged in successfully!

-----
                        Owner
-----

You are logged in
Owner information:
    Name: Owner1
    Bank account: BankAccount1
    Email: Owner1@email.com
    Phone number: 613 456 7890
    Balance: $101317.00

Menu: (type the number to choose)
(1) Browse unowned books
(2) Add book to collection
(3) Remove book from collection
(4) Order book from publisher
(5) Show all books in collection
(6) Show sale records
(9) Log out
(0) Exit Owner View
-----

Enter your choice: █
```

Similar to Customer, browsing books works the same for regardless of whether the owner is logged in or not.

Owner browse book:

Browsing books for owner works very similar to Customer browse book. Indeed it has the same menu as Customer's and filter works the same as Customer browse book filters.

```
-----  
Browse Book  
-----  
Browse book menu  
(1) Search using the current filter  
(2) Show all filters  
(3) Clear all filters  
(4) Add to Book Name filter  
(5) Add to ISBN filter  
(6) Add to Author filter  
(7) Add to Genre filter  
(8) Add to Publisher filter  
(0) Exit from browse book view  
-----  
Enter your choice: █
```

The key difference is that Owner can only browse books that are not collected by any owner (including themselves).

```
Enter your choice: 1  
select distinct ISBN from Book natural join Author where ISBN not in (select ISBN  
from Collect)order by ISBN;  
Here are the books that can be added to collection:  
ISBN: 1  
Book name: Book1  
Publisher: publisher0  
Genre: Genre1  
Description: Description1  
Number of pages: 100  
Price: $0.000000  
Authors: Author1  
  
ISBN: 2  
Book name: Book2  
Publisher: publisher0  
Genre: Genre1  
Description: Description2  
Number of pages: 200  
Price: $0.000000  
Authors: Author2  
  
ISBN: 6  
Book name: Book6  
Publisher: publisher1  
Genre: Genre2  
Description: Description6  
Number of pages: 100  
Price: $0.000000  
Authors: Author6
```

Owner Add/Remove Book

To add book, enter the ISBN listed, choose the price to sell the book for as well as the ratio of revenue to share with the publisher (the publisher is very generous so you can literally not share any penny with them)

```
Enter your choice: 2
Here are the uncollected book ISBNs to choose from:
    1
    2
    6
What is the ISBN of the book that you want to add to collection?
Enter your choice: 6
What price do you want to set for the book?
Enter your choice: 60
What percentage do you want to split the revenue with the Publisher? (enter a ratio from 0 to 1)
Enter your choice: 0.3
Book added to collection successfully
```

After adding, the application will automatically order the fixed AMOUNT_TO_ORDER (300) numbers of books from the publisher, as well as send money from the Owner to the Publisher to pay for that.

```
-----
                        Owner
-----
You are logged in
Owner information:
    Name: Owner1
    Bank account: BankAccount1
    Email: Owner1@email.com
    Phone number: 613 456 7890
    Balance: $95917.00
```

Previous balance is \$101317

The new balance is $\$101317 - (0.3 \text{ split ratio} * 300 \text{ books} * \$60 \text{ per book}) = \$95917$

Similarly, removing books from collection requires entering the ISBN from the list. Although there won't be any refund when the owner removes the books from his collection.

```
Enter your choice: 3
select distinct ISBN from (((Owner join Collect on Owner.name=Collect.owner_name) join Book using
oin Author using (ISBN)) where Owner.name='Owner1' order by ISBN;
Here are the ISBN of books that belong to your collection:
    3
    4
    5
    6
What is the ISBN of the book that you want to remove?
Enter your choice: 6
Book removed from collection successfully
```

Owner show all book in collection

As the name suggest, this shows all books that are owned by this owner in collection

```
Enter your choice: 5
select distinct ISBN from (((Owner join Collect on Owner.name=Collect
.owner_name) join Book using (ISBN)) join Author using (ISBN)) where
Owner.name='Owner1' order by ISBN;
ISBN: 3
Book name: Book3
Publisher: publisher0
Genre: Genre1
Description: Description3
Number of pages: 150
Price: $12.000000
Authors: Author3

ISBN: 4
Book name: Book4
Publisher: publisher0
Genre: Genre1
Description: Description4
Number of pages: 175
Price: $15.000000
Authors: Author4

ISBN: 5
Book name: Book5
Publisher: publisher0
Genre: Genre2
Description: Description5
Number of pages: 250
Price: $13.000000
Authors: Author5
```

Owner show sale record

The owner sale record menu is similar to owner browse book menu and customer browse book menu. We can modify the filter to change how sale records is being aggregated

```
Enter your choice: 6

-----
                        Owner Record
-----

Sale record menu
(1) Show records using the current filters
(2) Show all filters
(3) Clear all filters
(4) Add to Genre filter
(5) Add to Author filter
(6) Add to Publisher filter
(0) Exit Owner Record View
-----

Enter your choice: █
```

Without any filter, “(1) Show records” shows all sale information of all books

```
Enter your choice: 1
select distinct ISBN from (((Owner join Collect on Owner.name=Collect
.owner_name) join Book using (ISBN)) join Author using (ISBN)) where
Owner.name='Owner1' order by ISBN;
ISBN: 3
Book name: Book3
Publisher: publisher0
Genre: Genre1
Description: Description3
Number of pages: 150
Price: $12.000000
Authors: Author3
Unit in stock: 200
Unit sold: 100
Revenue: $1200.00
Expense: $1440.00
Publisher split ratio: 40/60

ISBN: 4
Book name: Book4
Publisher: publisher0
Genre: Genre1
Description: Description4
Number of pages: 175
Price: $15.000000
Authors: Author4
Unit in stock: 300
Unit sold: 0
Revenue: $0.00
Expense: $2250.00
Publisher split ratio: 50/50

ISBN: 5
Book name: Book5
Publisher: publisher0
Genre: Genre2
Description: Description5
Number of pages: 250
Price: $13.000000
Authors: Author5
Unit in stock: 291
Unit sold: 9
Revenue: $117.00
Expense: $390.00
Publisher split ratio: 10/90

Total revenue: $1317.00
Total expense: $4080.00
```


To show sale information for a certain genre(s), adjust the genre filter
For example: Only show sales for Genre1

```
Enter your choice: 4
Available Genres:
(0) Genre1
(1) Genre2
(2) Genre3
(3) Genre4
Enter the index number (in the round bracket) to add to filter
Enter your choice: 0

Enter your choice: 2
Genre Filter: [Genre1]
Author Filter: []
Publisher Filter: []

Enter your choice: 1
select distinct ISBN from (((Owner join Collect on Owner.name=Collect
.owner_name) join Book using (ISBN)) join Author using (ISBN)) where
Owner.name='Owner1' and genre in ('Genre1')order by ISBN;
ISBN: 3
Book name: Book3
Publisher: publisher0
Genre: Genre1
Description: Description3
Number of pages: 150
Price: $12.000000
Authors: Author3
Unit in stock: 200
Unit sold: 100
Revenue: $1200.00
Expense: $1440.00
Publisher split ratio: 40/60

ISBN: 4
Book name: Book4
Publisher: publisher0
Genre: Genre1
Description: Description4
Number of pages: 175
Price: $15.000000
Authors: Author4
Unit in stock: 300
Unit sold: 0
Revenue: $0.00
Expense: $2250.00
Publisher split ratio: 50/50

Total revenue: $1200.00
Total expense: $3690.00
```

Similarly, to show sale per author and sale per publisher, just modify the corresponding filter

6. Github Repository:

<https://github.com/Tom1072/Look-Inna-Book>

7. Demo availability:

I am available any time from 9 am to 5 pm on December 20th. But here are my preferences:

11:00 AM to 11:20 AM

11:20 AM to 11:40 AM

11:40 AM to 12:00 PM