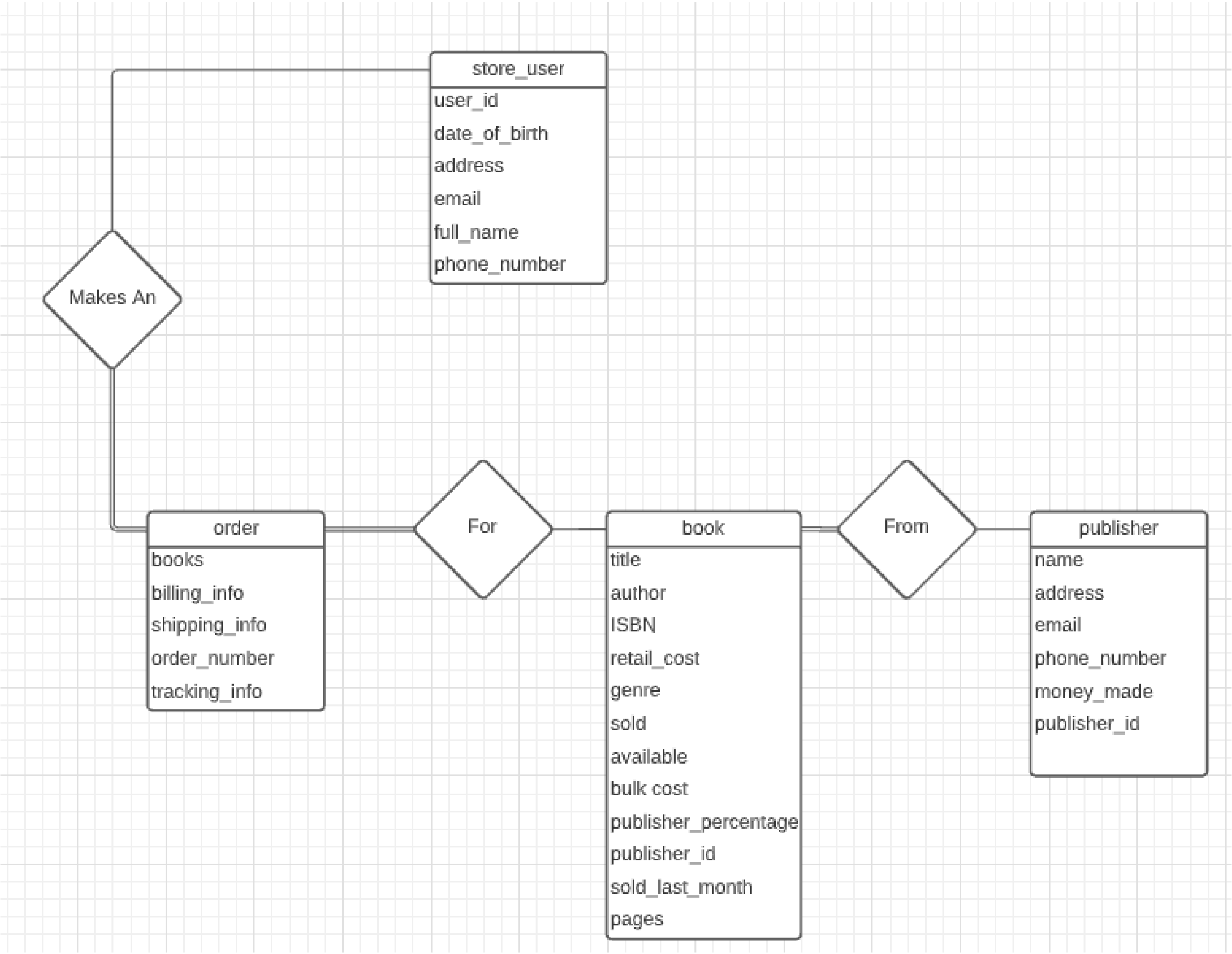
Braedon Dillon
101068213
COMP3005 Final Project

Solo Project:

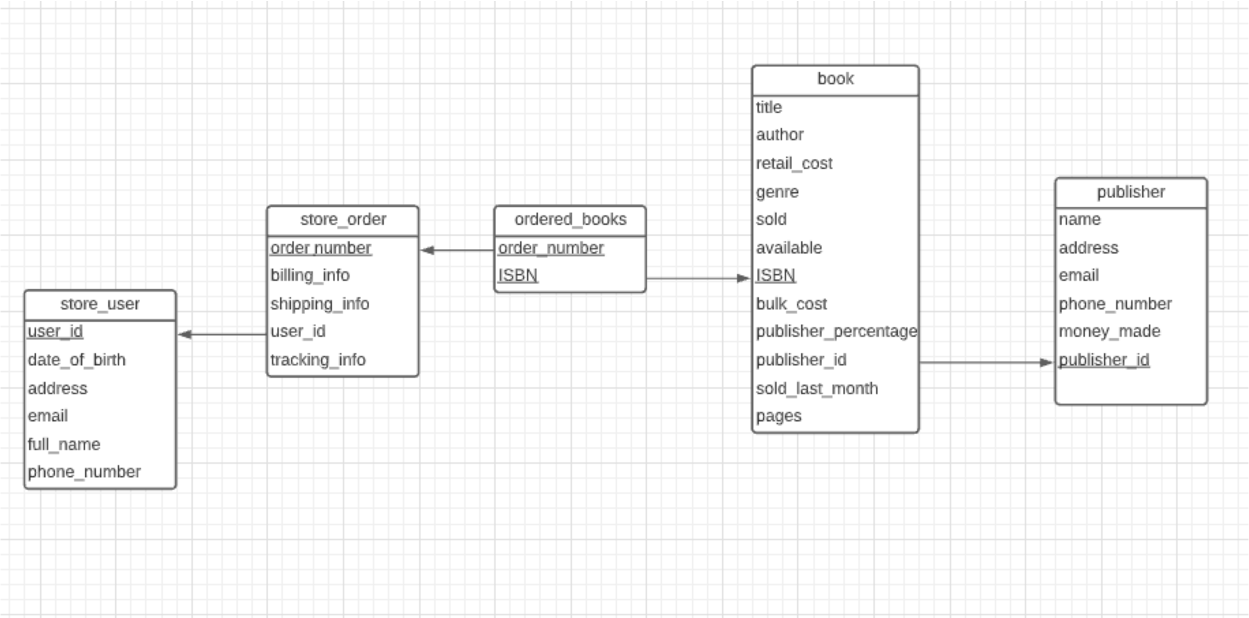
Conceptual Design:



Assumptions I make:

1. All orders must be made by a user
2. All orders must contain at least one book
3. All books must have a publisher

Reduction to Relation Schemas



Or
(primary key underlined, foreign keys shown above with arrows)

Store_user(user_id, date_of_birth, address, email, name, phone_number)
Store_order(order_number, billing_info, shipping_info, user_id, tracking_info)
Ordered_books(order_number, ISBN)
Book(title, author, retail_cost, genre, sold, available, ISBN, bulk_cost, publisher_percentage, publisher_id, sold_last_month, pages)
Publisher(name, address, email, phone_number, money_made, publisher_id)

Functional Dependencies:

User:

1. User_id->(date_of_birth, address, email, name, phone_number)

No other dependencies can be enforced as there are edge cases for each
For example,
Someone with the same D.O.B could have varying everything else
Someone with the same address could have varying everything else
Someone with the same email could have varying everything else (shared email used in multiple accounts for a couple living separately)
Someone with the same name could have varying everything else
Someone with the same phone_number could have varying everything else (child account with their own email who lives with one parent uses the other parent's phone number)

Order:

1. OrderNumber->(billing_info, shipping_info, user_id, tracking_info)

No other dependencies can be enforced
An order with the same billing_info can differ in everything else
An order with the same shipping_info can differ in everything else
An order with the same user_id can differ in everything else
An order with the same tracking_info can differ in everything else

Ordered_books:

1. (order_number, ISBN)->(order_number) (trivial)
2. (order_number, ISBN)->(ISBN) (trivial)

Book:

1. ISBN -> (title, author, genre, sold, available, bulk_cost, publisher_percentage, publisher_id, retail_cost, sold_last_month, pages)

No other dependencies can be enforced

None of the attributes other than ISBN can be used to define any others

For example,

(title, author) cannot be used to define a book as it may be a different edition or format with a different ISBN

Publisher:

1. publisher_id->(name, address, email, phone_number, money_made)
2. email -> (name, address, email, phone_number, money_made)

No other dependencies can be enforced

None of the attributes other than publisher_id and email can be used to uniquely identify the others

For example,

phone_number is not necessarily unique as it may be the number of a call centre shared by multiple publishers

address is not necessarily unique as there may be multiple publishers working out of the same space

Testing Good Form

User: BCNF

Functional Dependencies:

1. User_id->(date_of_birth, address, email, name, phone_number)

Explanation:

User is in BCNF because in all possible F^+ of User, the functional dependency is either trivial, or has superkey user_id as the determinant

Order: BCNF

Functional Dependencies:

1. Order_number->(billing_info, shipping_info, user_id, tracking_info)

Explanation:

Order is in BCNF because in all possible F^+ of order, the functional dependency is either trivial, or has superkey order_number as the determinant

Ordered_books: BCNF

Functional Dependencies:

1. (order_number, ISBN)->(order_number)
2. (order_number, ISBN)->(ISBN)

Explanation:

Ordered_books is in BCNF because in all possible F^+ of order, only trivial function dependencies exist

Book: BCNF

Functional Dependencies:

1. ISBN -> (title, author, genre, sold, available, bulk_cost, publisher_percentage, publisher_id, retail_cost, sold_last_month, pages)

Explanation:

book is in BCNF because in all possible F^+ of order, the functional dependency is either trivial, or has superkey ISBN as the determinant

Publisher: BCNF

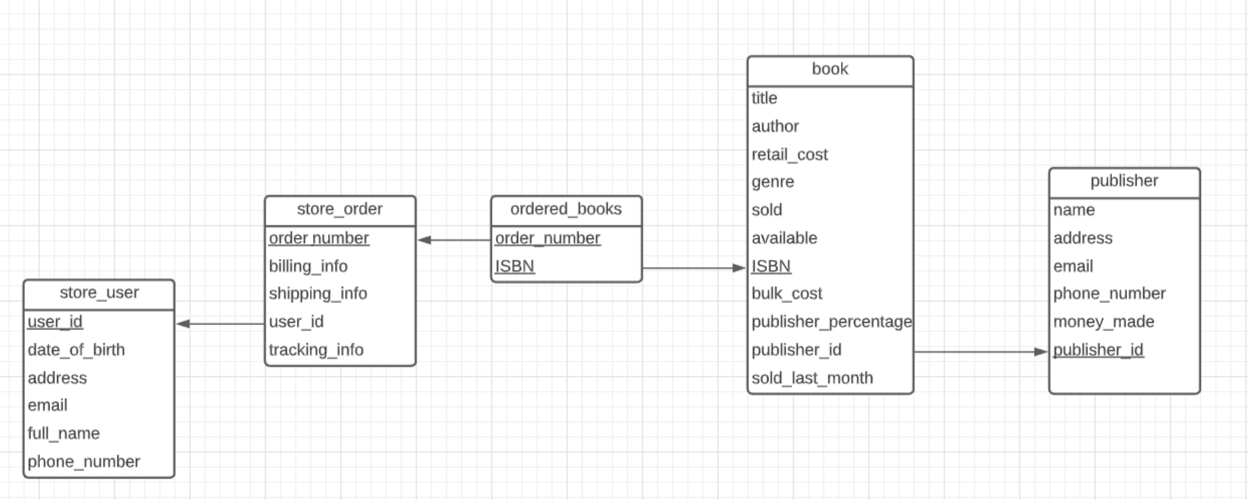
Functional Dependencies:

1. publisher_id->(name, address, email, phone_number, money_made)
2. email -> (name, address, email, phone_number, money_made)

Explanation:

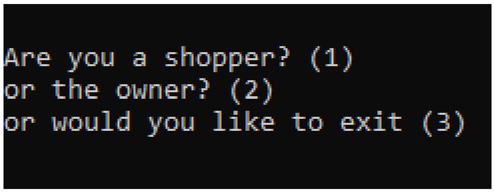
publisher is in BCNF because in all possible F^+ of order, the functional dependency is either trivial, or has either superkey publisher_id or superkey email as the determinant.

Database Schema Diagram:



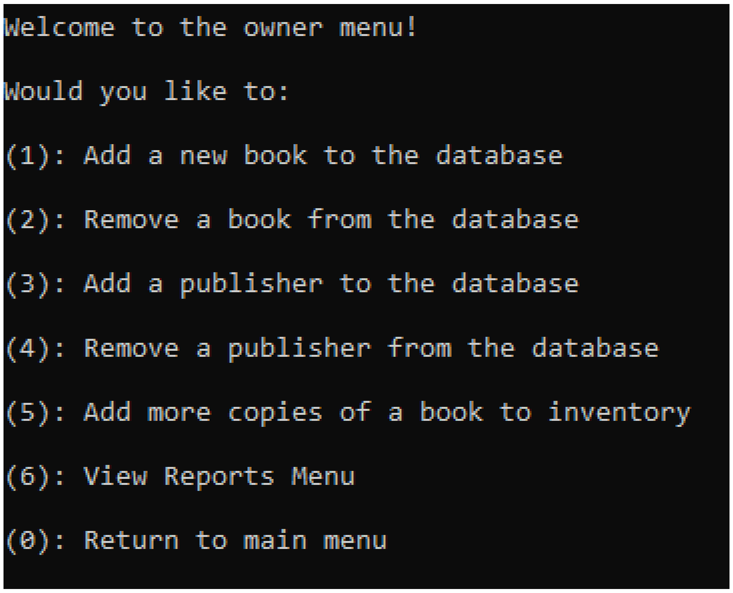
Implementation:

The workflow of my application is very simple. Upon entering the application it will show a menu asking if you'd like the shopper options or the owner options. As seen here



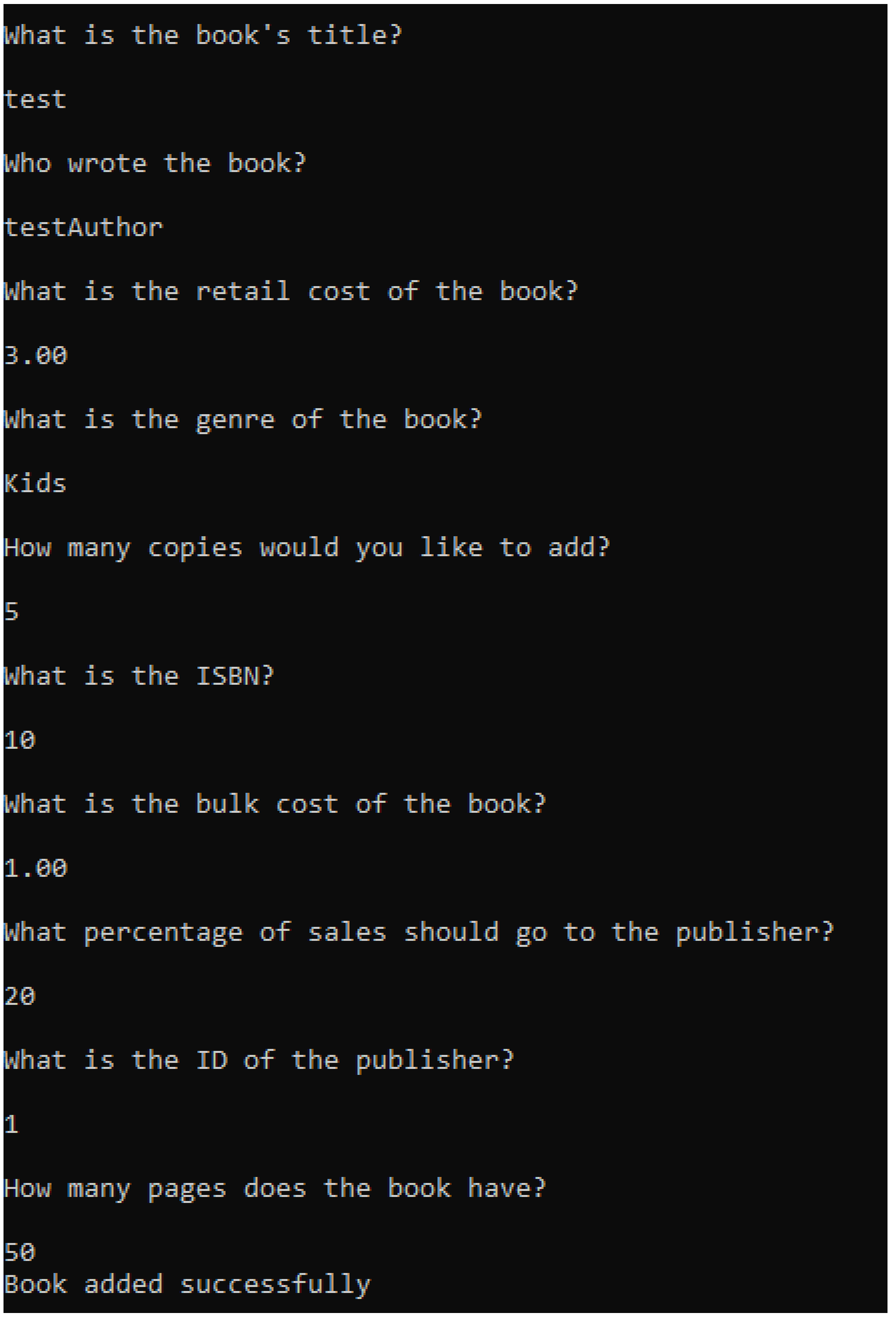
```
Are you a shopper? (1)
or the owner? (2)
or would you like to exit (3)
```

Choosing the owner menu will take you to the following menu



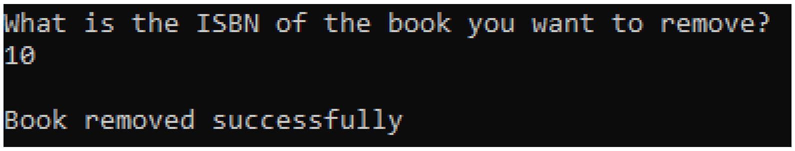
```
Welcome to the owner menu!

Would you like to:

(1): Add a new book to the database

(2): Remove a book from the database

(3): Add a publisher to the database

(4): Remove a publisher from the database

(5): Add more copies of a book to inventory

(6): View Reports Menu

(0): Return to main menu
```

You then can choose to do any of the options on the list. For example, to add a book would look like this



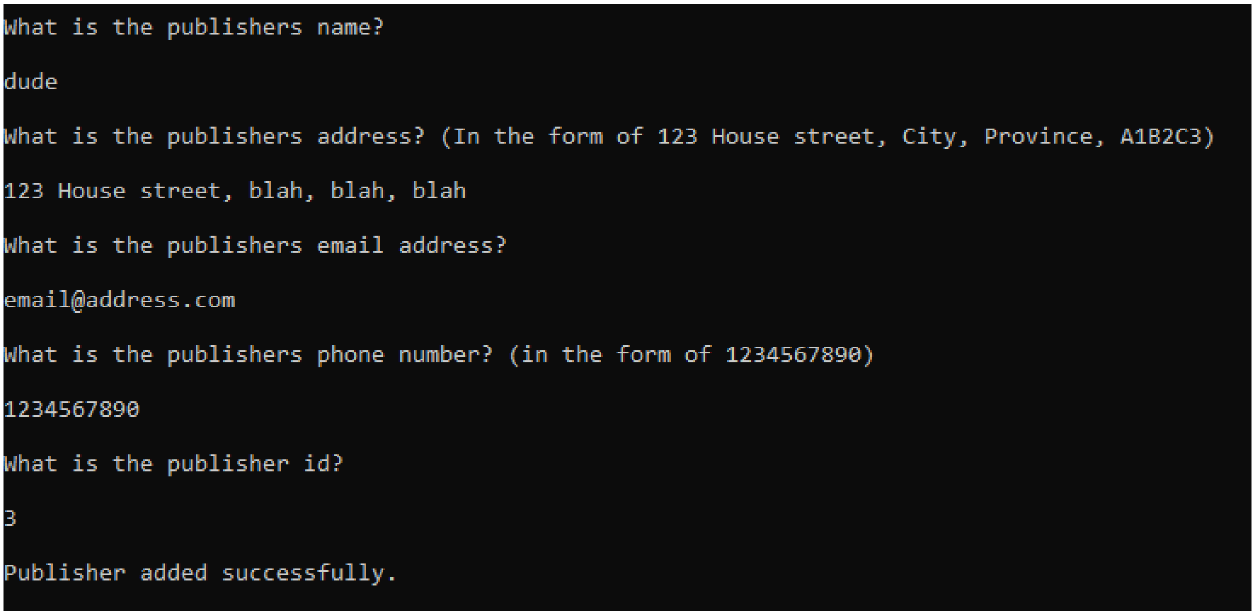
```
What is the book's title?

test

Who wrote the book?

testAuthor

What is the retail cost of the book?

3.00

What is the genre of the book?

Kids

How many copies would you like to add?

5

What is the ISBN?

10

What is the bulk cost of the book?

1.00

What percentage of sales should go to the publisher?

20

What is the ID of the publisher?

1

How many pages does the book have?

50
Book added successfully
```

Removing a book would look like this:



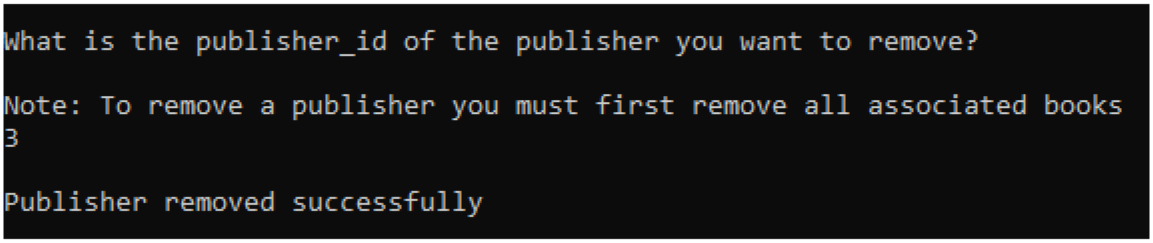
```
What is the ISBN of the book you want to remove?
10

Book removed successfully
```

Adding a publisher would look like this



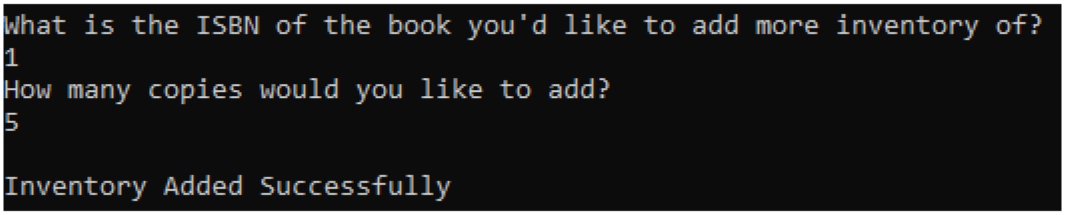
```
What is the publishers name?

dude

What is the publishers address? (In the form of 123 House street, City, Province, A1B2C3)

123 House street, blah, blah, blah

What is the publishers email address?

email@address.com

What is the publishers phone number? (in the form of 1234567890)

1234567890

What is the publisher id?

3

Publisher added successfully.
```

Removing a publisher would look like this:



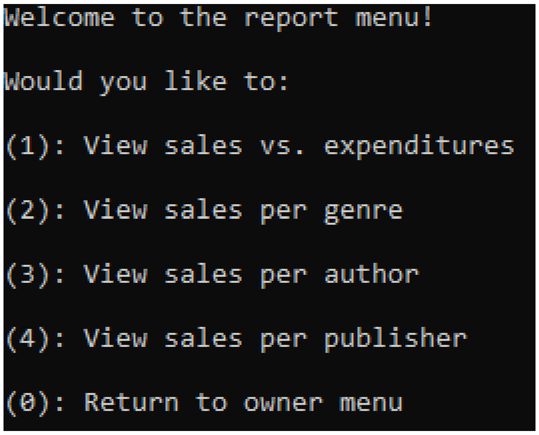
```
What is the publisher_id of the publisher you want to remove?

Note: To remove a publisher you must first remove all associated books
3

Publisher removed successfully
```

Adding more copies of a book would look like this:



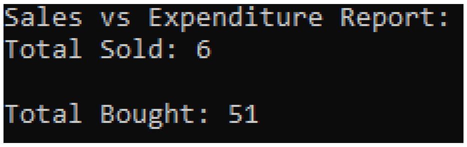
```
What is the ISBN of the book you'd like to add more inventory of?
1
How many copies would you like to add?
5

Inventory Added Successfully
```

Lastly, entering the report menu can be done by entering 6. After doing that you will see the following



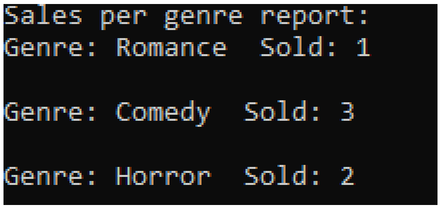
```
Welcome to the report menu!

Would you like to:

(1): View sales vs. expenditures

(2): View sales per genre

(3): View sales per author

(4): View sales per publisher

(0): Return to owner menu
```

The first report will something look like this (sales from my testing)



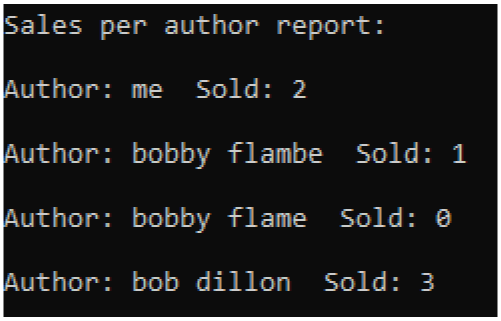
```
Sales vs Expenditure Report:
Total Sold: 6

Total Bought: 51
```

The second report will look something like this



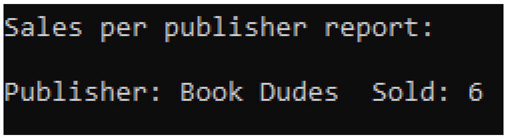
```
Sales per genre report:
Genre: Romance  Sold: 1

Genre: Comedy  Sold: 3

Genre: Horror  Sold: 2
```

The third report will look something like this



```
Sales per author report:

Author: me  Sold: 2

Author: bobby flambe  Sold: 1

Author: bobby flame  Sold: 0

Author: bob dillon  Sold: 3
```

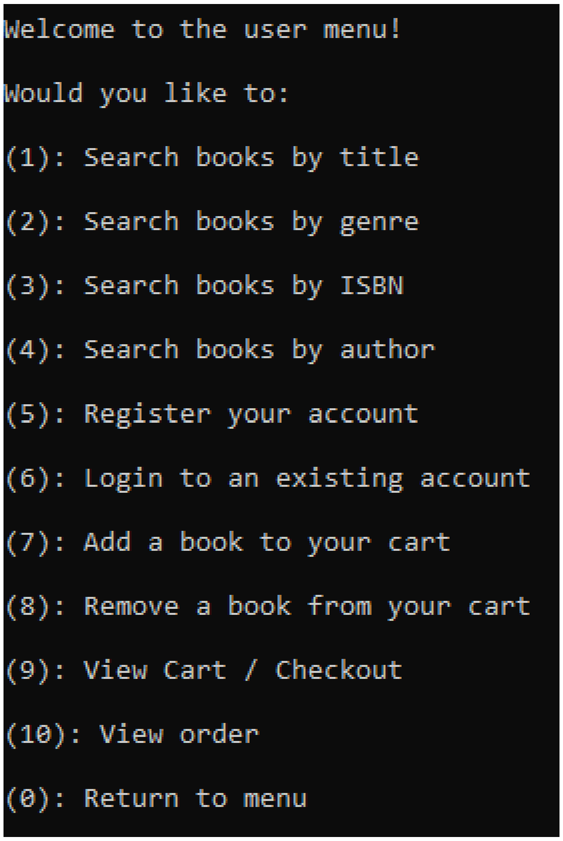
The final report will look like this



```
Sales per publisher report:

Publisher: Book Dudes  Sold: 6
```

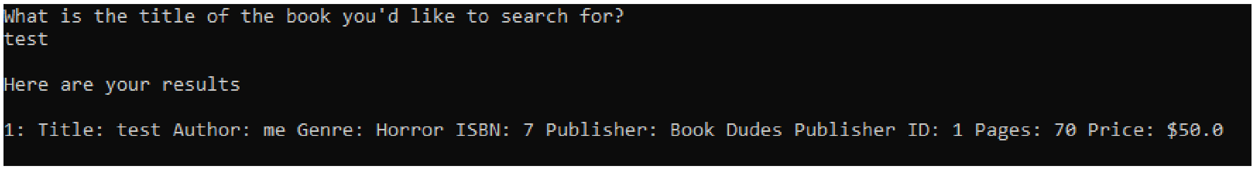
The menus can be traveled back through to the main menu where we can now quit or chose user/owner again. I will show the the user side now

The user menu looks like this:



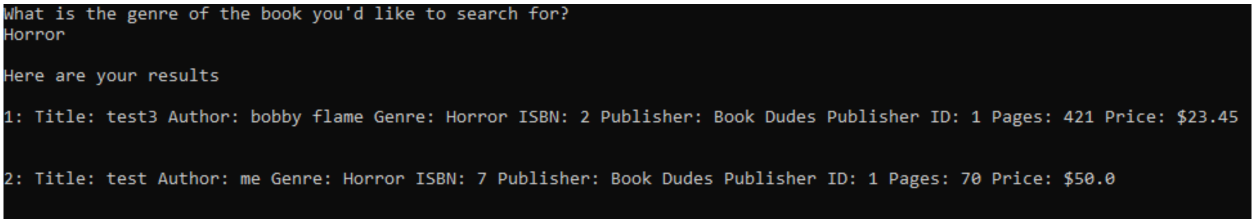
```
Welcome to the user menu!

Would you like to:

(1): Search books by title

(2): Search books by genre

(3): Search books by ISBN

(4): Search books by author

(5): Register your account

(6): Login to an existing account

(7): Add a book to your cart

(8): Remove a book from your cart

(9): View Cart / Checkout

(10): View order

(0): Return to menu
```

If you choose search by title, It looks like this



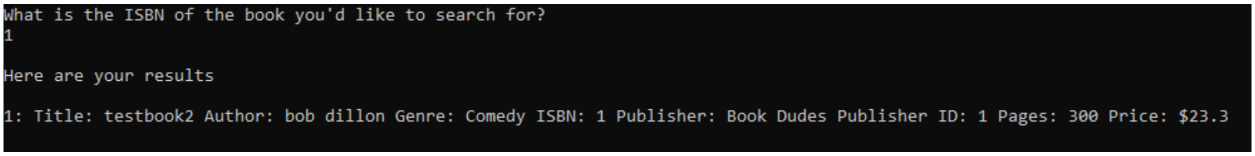
```
What is the title of the book you'd like to search for?
test

Here are your results

1: Title: test Author: me Genre: Horror ISBN: 7 Publisher: Book Dudes Publisher ID: 1 Pages: 70 Price: $50.0
```

If you choose search by genre, it looks like this



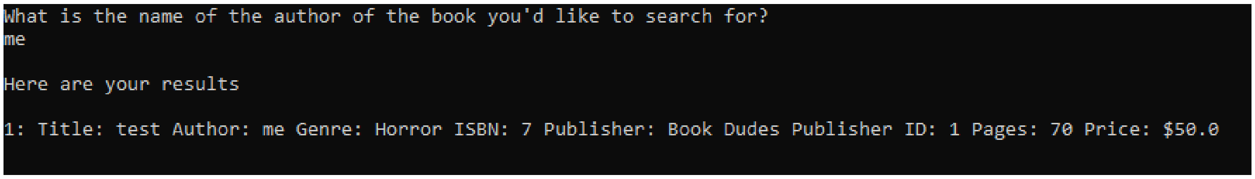
```
What is the genre of the book you'd like to search for?
Horror

Here are your results

1: Title: test3 Author: bobby flame Genre: Horror ISBN: 2 Publisher: Book Dudes Publisher ID: 1 Pages: 421 Price: $23.45


2: Title: test Author: me Genre: Horror ISBN: 7 Publisher: Book Dudes Publisher ID: 1 Pages: 70 Price: $50.0
```

If you choose search by ISBN, it looks like this



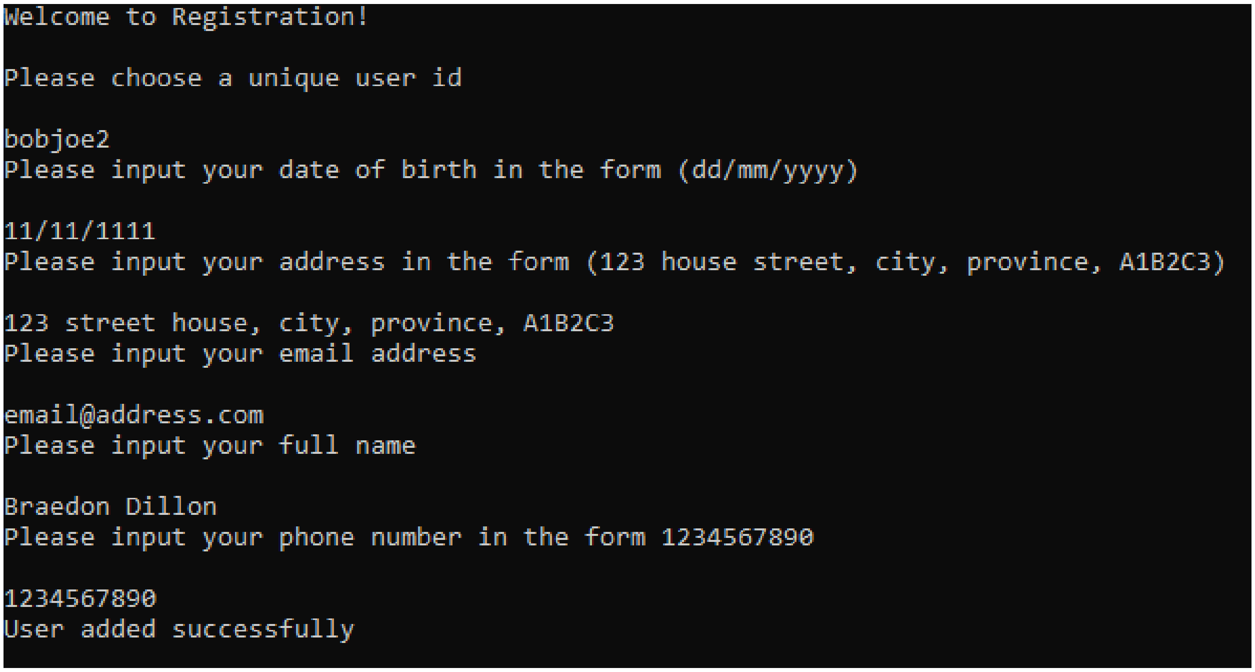
```
What is the ISBN of the book you'd like to search for?
1

Here are your results

1: Title: testbook2 Author: bob dillon Genre: Comedy ISBN: 1 Publisher: Book Dudes Publisher ID: 1 Pages: 300 Price: $23.3
```

If you choose search by author, it looks like this



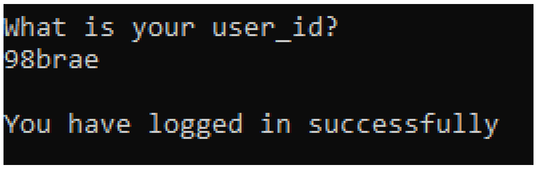
```
What is the name of the author of the book you'd like to search for?
me

Here are your results

1: Title: test Author: me Genre: Horror ISBN: 7 Publisher: Book Dudes Publisher ID: 1 Pages: 70 Price: $50.0
```

If you choose to register an account, it looks like this



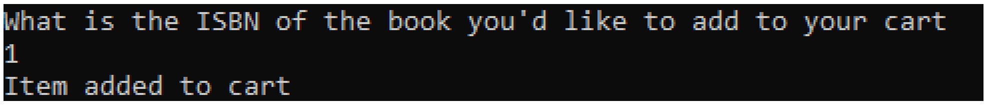
```
Welcome to Registration!

Please choose a unique user id

bobjoe2
Please input your date of birth in the form (dd/mm/yyyy)

11/11/1111
Please input your address in the form (123 house street, city, province, A1B2C3)

123 street house, city, province, A1B2C3
Please input your email address

email@address.com
Please input your full name

Braedon Dillon
Please input your phone number in the form 1234567890

1234567890
User added successfully
```

If you choose login, it looks like this



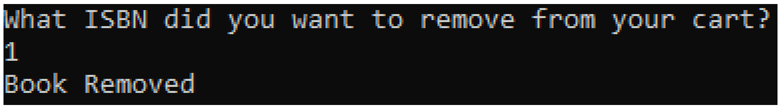
```
What is your user_id?
98brae

You have logged in successfully
```

If you choose add a book to cart, it looks like this



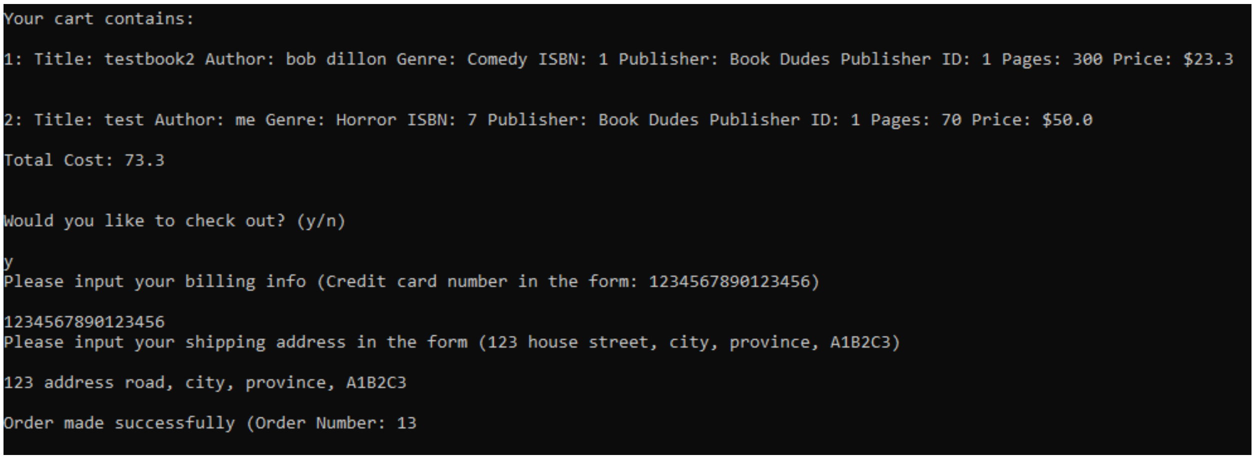
```
What is the ISBN of the book you'd like to add to your cart
1
Item added to cart
```

If you chose remove a book from your cart it looks like this



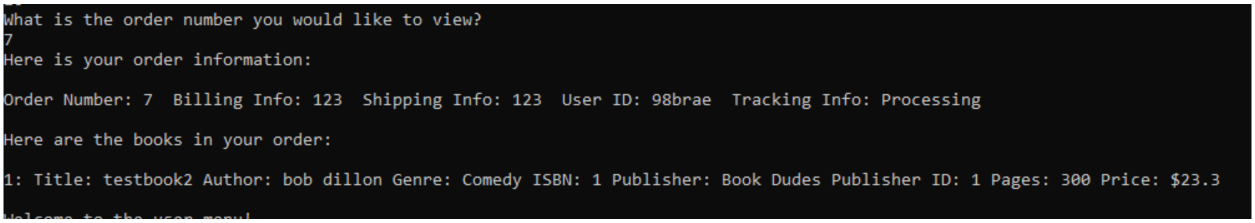
```
What ISBN did you want to remove from your cart?
1
Book Removed
```

If you choose to look at your cart / checkout it looks like this (I added some books to the cart)



```
Your cart contains:

1: Title: testbook2 Author: bob dillon Genre: Comedy ISBN: 1 Publisher: Book Dudes Publisher ID: 1 Pages: 300 Price: $23.3


2: Title: test Author: me Genre: Horror ISBN: 7 Publisher: Book Dudes Publisher ID: 1 Pages: 70 Price: $50.0

Total Cost: 73.3


Would you like to check out? (y/n)

y
Please input your billing info (Credit card number in the form: 1234567890123456)

1234567890123456
Please input your shipping address in the form (123 house street, city, province, A1B2C3)

123 address road, city, province, A1B2C3

Order made successfully (Order Number: 13
```

If you choose view order it looks like this



```
What is the order number you would like to view?
7
Here is your order information:

Order Number: 7  Billing Info: 123  Shipping Info: 123  User ID: 98brae  Tracking Info: Processing

Here are the books in your order:

1: Title: testbook2 Author: bob dillon Genre: Comedy ISBN: 1 Publisher: Book Dudes Publisher ID: 1 Pages: 300 Price: $23.3

Welcome to the user menu!
```

GitHub Repository:

https://github.com/99brae/DatabasesFinalProject
Repository will be public at roughly 11:30 pm December 10th

Appendix

I am available any time from noon to 5 pm on December 11th.

I found this class very interesting! Thanks for a good term!