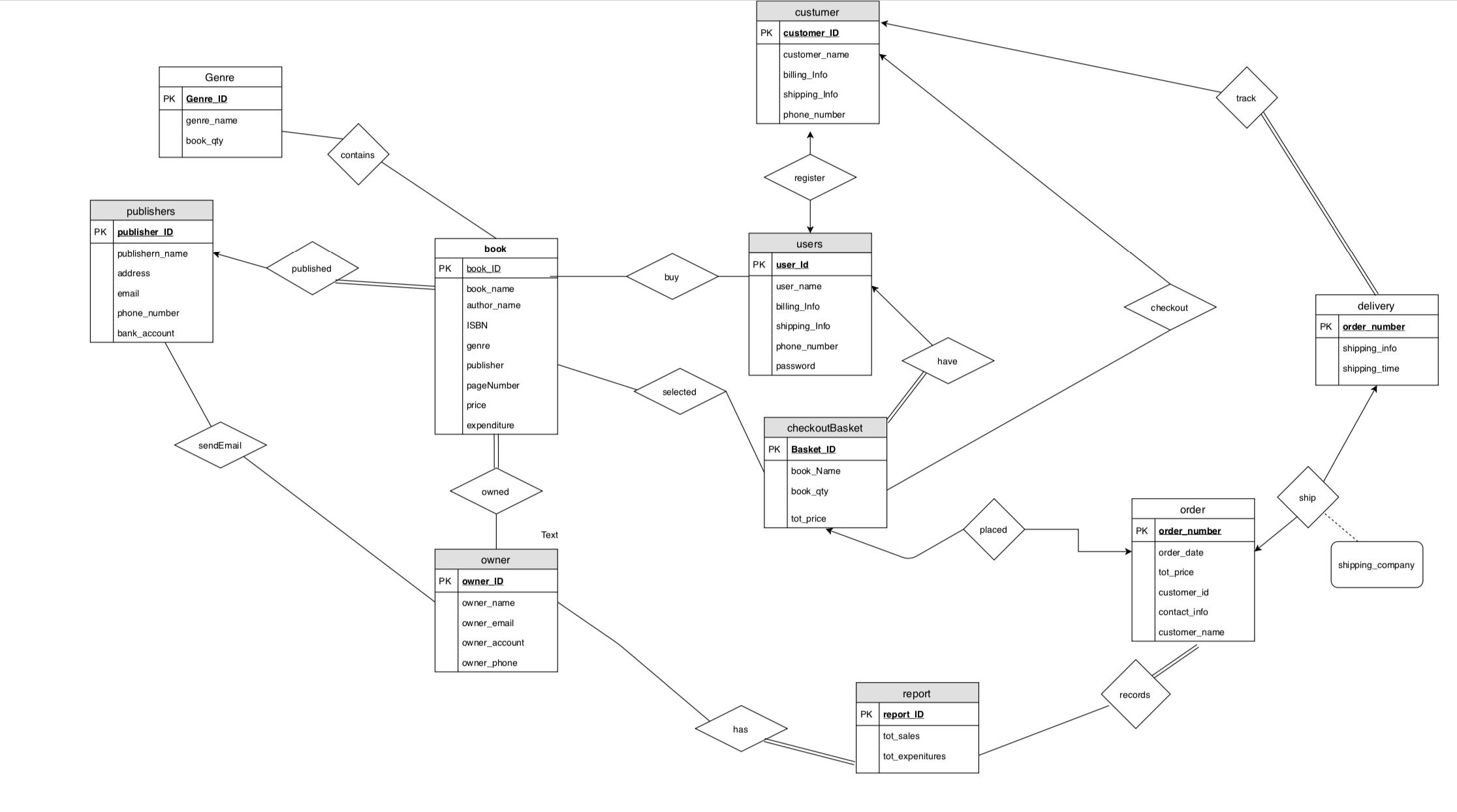
COMP 3005 Porject

Yuqing Guo

101106497

ER diagram：

**2.1. Conceptual Design**

* Explanation of the assumption:

The users have the checkoutBasket, they can buy book to the checkoutBasket, in order to buy the book, they need to register to become customers, customers can checkout the books from the basket. Then the order can be placed and ship from the shipping company to delivery, the customers can track the delivery by the order number. The book contains many genres. The books are published by the publisher and theses book are owned by owner, when the remaining quality is not enough, the owner can send email to publisher. Also, the owner can report the order to show the tot\_sale and expenditure.

* Each entity:

1. The primary key of the book entity is book\_ID.
2. The primary key of the users entity is user\_Id.
3. The primary key of the customer entity is \_ID.
4. The primary key of the checkoutBasket entity is Basket\_ID.
5. The primary key of the report entity is report\_ID.
6. The primary key of the order entity is user\_number.
7. The primary key of the owner entity is owner\_Id.
8. The primary key of the publishers entity is publisher\_Id.
9. The primary key of the delivery entity is order\_number.
10. The primary key of the Genre entity is Genre\_ID.

* cardinalities and participation types
* The relationship from user to book is buy, it is a many to many relationship and participations of book and user in buy are partial.
* The relationship from book to checkoutBasket is select, it is a many to many relationship and participation of book and checkoutBasket in select are partial.
* The relationship from user to checkoutBasket is have, it is a one to many relationship, one basket is owned by one user. The participation of checkoutBasket in have relation is total, every checkoutBasket must have a user. The participation of user in the relation is partial.
* The relationship from user to customer is register, it is a one to one relation, one user can register once and one customer can be registed by one user. The participations of user and customer in register relation are partial.
* The relationship form customer to checkoutBasket is checkout, it is a one to many relation, one basket can only be checked out by one customer. The participations of customer and checkoutBasket in checkout relation are partial.
* The relationship from checkoutBasket to order is placed, it is a one to one relation, one order can only be placed by one basket, and one basket can make only one order. The participations of checkoutBasket and order in placed relation are partial.
* The relationship from order to delivery is ship, and the attribute of ship relation is shipping\_company. It is a one to one relation, one order must only have one delivery option, one delivery must ship one order. The participations of order and delivery in ship relation is partial.
* The relationship from customer to delivery is track, it is a one to many relation, a customer can track many delivery, but one delivery can only be tracked by one customer. The participation of customer in the relation is partial. The participation of delivery in track relation is total which means every order can be tracked by customer.
* The relationship from book to genre is contains, it is a one to one relation. The participations of book and genre in contains relation are partial.
* The relationship from book to publisher is published, it is a many to one relation, one publisher can publish many books, but one book only can by published by one publisher. The participation of book in the relation is total, every book must have publisher, but some publisher might did not publish book, so the participation of publisher in the relation is partial.
* The relationship from book to owner is owned, it is a many to many relation. The participation of book in the relation is total, every book must have an owner. The participation of owner in owned relation is partial.
* The relationship from owner to publisher is sendEmail, it is a many to many relation. The participations of owner and publishers in the relation are partial.
* The relationship from owner to report is has, it is a many to many relation. The participation of owner in relation is partial and the participation of report in relation is total, every report must be report by owner.
* The relationship from report to order is record, it is a many to many relation. The participation of report in record relation is partial. The participation of order in relation is total, every order must be record to the report.

**2.2 Reduction to Relation Schema**

* book (book\_ID, book\_name, author\_name, ISBN, genre, publisher, pageNumber, price, expenditure)
* user (user\_Id, user\_name, billing\_Info, shipping\_Info, phone\_number, password)
* customer (customer\_ID, customer\_name, billing\_Info, shipping\_Info, phone\_number)
* checkourBasket (basket\_ID, book\_Name, book\_qty, tot\_price)
* owner (owner\_ID, owner\_name, owner\_email, owner\_account, owner\_phone)
* order (order\_number, order\_date, tot\_price, customer\_id, contact\_Info, customer\_name)
* delivery (order\_number, shipping\_info, shipping\_time)
* Genre (Genre\_ID, genre\_name, book\_qty)
* Publishers (publisher\_ID, publisher\_name, address, email, phone\_email, bank\_account)
* report (report\_ID, tot\_sales, tot\_expenditure)

**2.3 Normalization of Relation Schemas**

* In the book table, book\_ID is the primary key and superkey.

book\_ID->book\_name, author\_name, ISBN, genre, publisher, pageNumber, price, expenditure.

It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form and it doesn’t need to decompose.

* In the users table, user\_Id is the superkey.

user\_Id->user\_name, billing\_Info, shipping\_Info, phone\_number, password

It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form.

* In the checkoutBasket table, basket\_ID is the superkey.

Basket\_Id-> book\_name, book\_qty, tot\_price

It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form.

* In the customer table, customer\_ID is the superkey.

Customer->customer\_name, billing\_Info, shipping\_Info, phone\_number

It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form.

* In the order table, order\_number is the superkey.

Order\_number->order\_name, tot\_price, customer\_id, contact\_Info, customer\_name

It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form.

* In the delivery table, the order\_number is the superkey

Order\_number->shipping\_Info, shipping\_time

It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form.

* In the genre table, the genre\_ID is the superkey.

Genre\_ID->genre\_name, book\_qty

It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form.

* In the publisher table, the publisher\_ID is the superkey

Publisher\_ID->publisher\_name, address, email, phone\_number, bank\_account

It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form.

* In the owner table, the owner\_ID is the superkey

Owner\_ID-> owner\_name, owner\_email, owner,account, owner\_phone

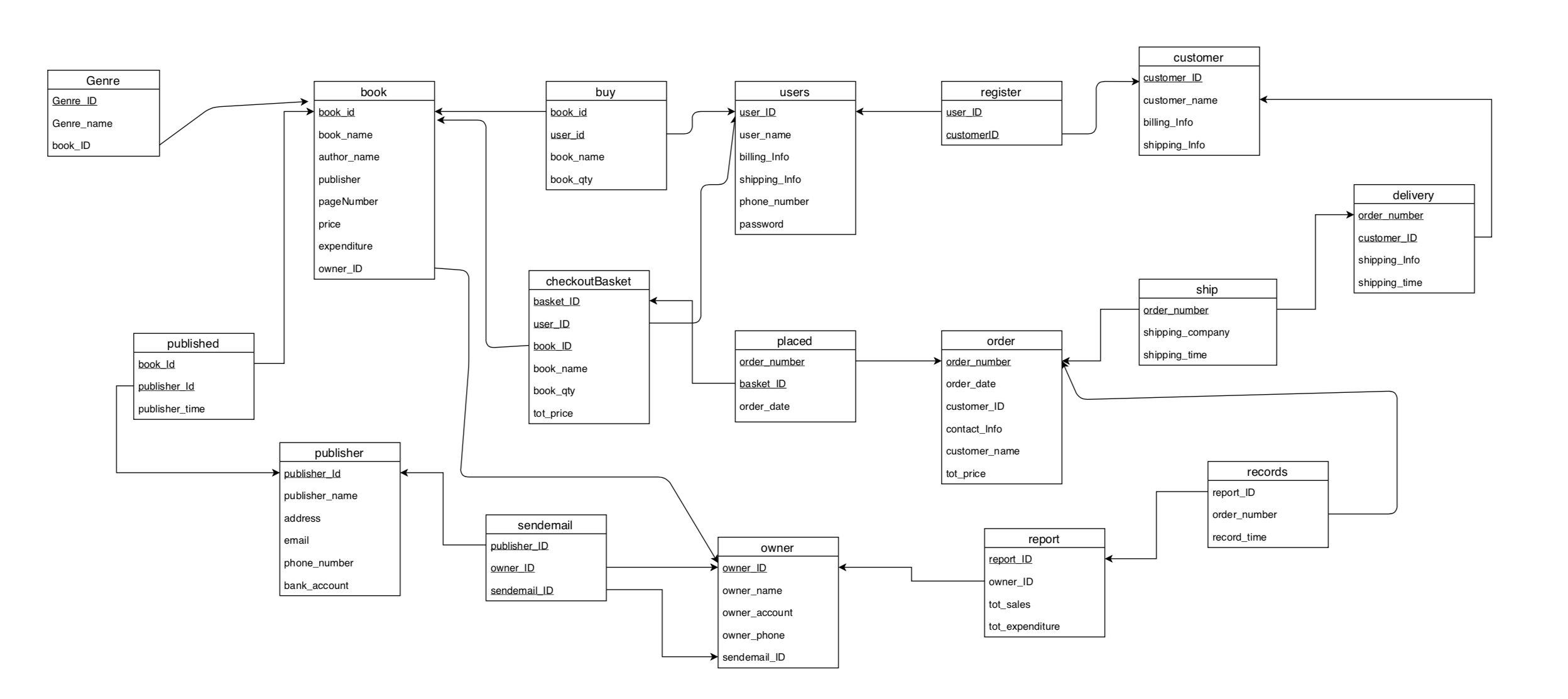
It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form.

* In the report table, the report\_ID is the superkey

Report\_ID-> tot\_sales, tot\_expeniture

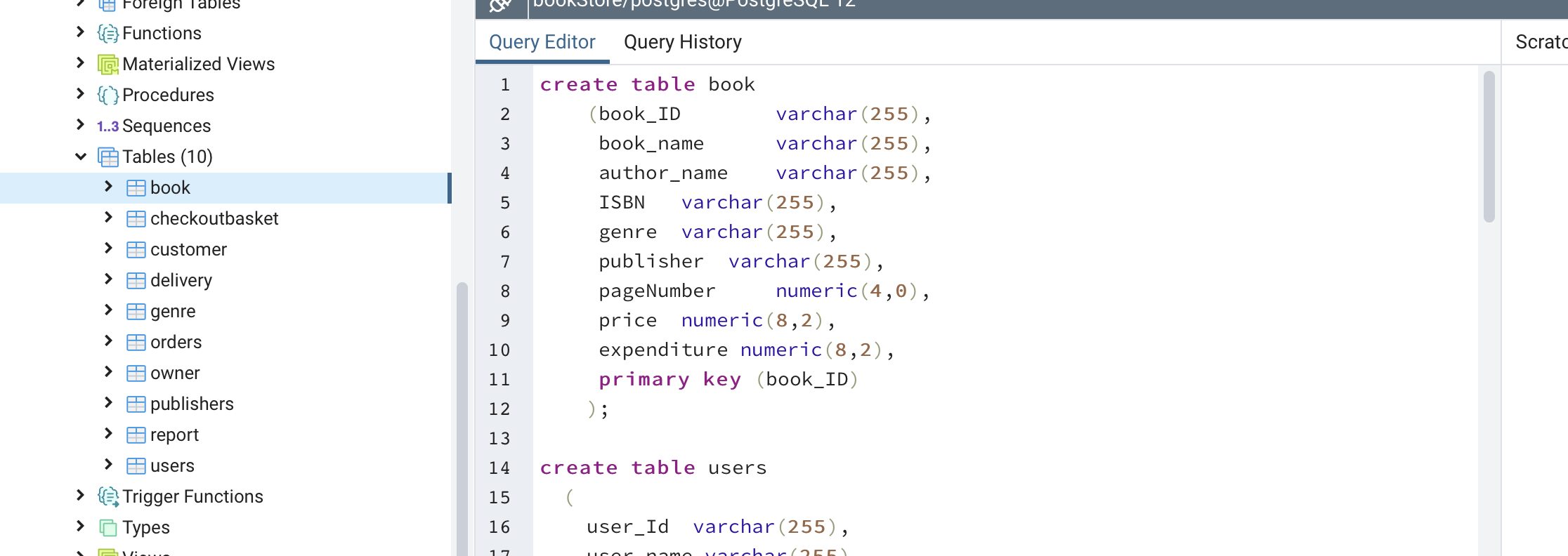
It satisfies that for all functional dependencies in α→β in F+, α is a superkey for R, so the relation schema R is in BCNF, it is a good form.

**2.4 Database Schema Diagram**

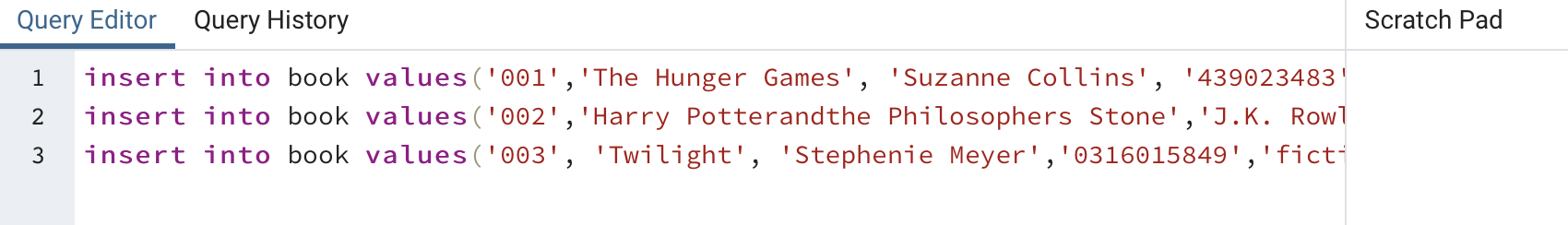


**2.5 Implementation**

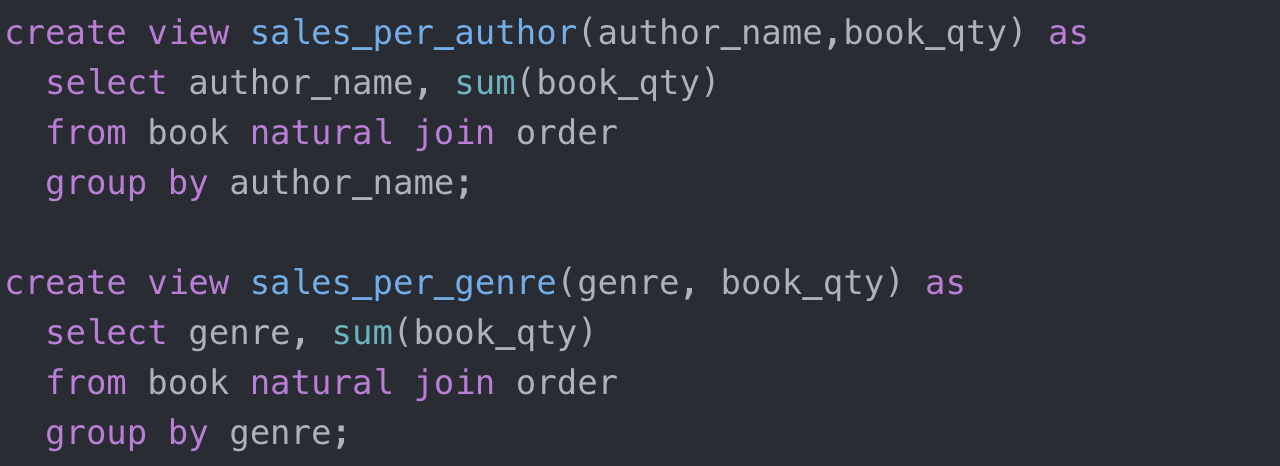
create the schemas:



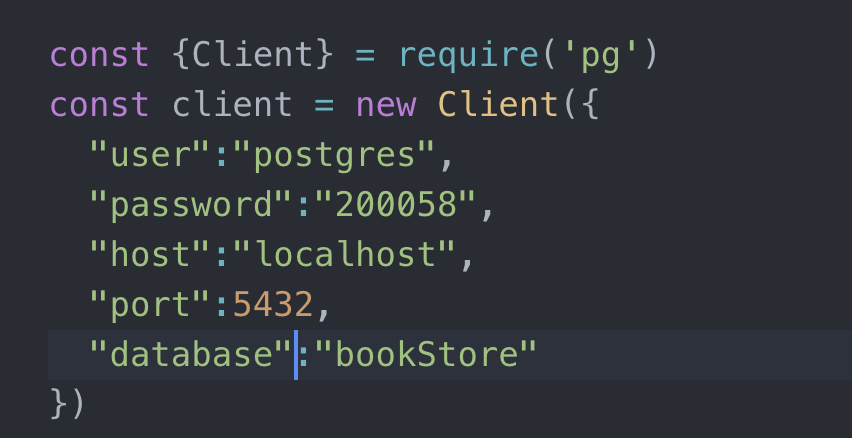
insert the data:



get the sales per genre, sales per author:



I tried to used JavaScript to connect the pgAdmin database.



I used the data from the JSON file. Users can add book to their checkout Basket.

