**ACCOUNT/BUYING AND SELLING BOOKS**

(\*)Designing an ERD conceptual schema

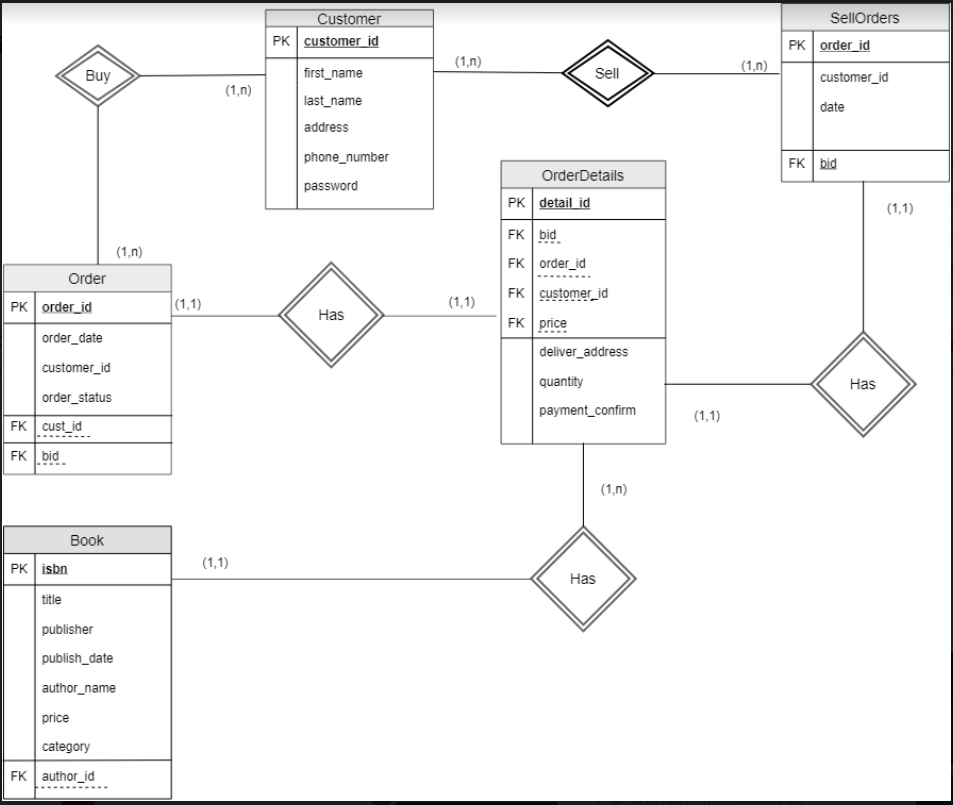
As we can see from this diagram,there are 5 objects

Firstly, customer and book are the strong entities which hold the datas.

Secondly is buying and selling. The customer will choose which to order or which to buy. So,the relation between *Customer* and Order/*SellOrders* are Many to Many as orders can be taken by many customers, and many customers can take many orders.

Lastly,there will be a log file, which is name as OrderDetails. It does the work of recording the details of the transaction (price,date…). And it is linked to the initial Orders of Customer of which the relation is straight forward - 1 to 1.Which means, one order can only have one detail information.

Tool : draw.io



**Fig.ERD Diagram**

(\*)ERD to relational database schema conversion

- In this part, I follow the step and define the strong and weak entities, the primary key and foreign key -> the relation of attributes.

**Customer** = (customer\_id,password,first\_name,last\_name,email,address,phone\_number)

From Customer.password *to* Register.password

**Book** = (isbn, title, author\_name,publisher, publish\_date,price,category,author\_id)

**Order** = (order\_id, customer\_id, isbn,quantity)

From Order.customer\_id *to* Customer.customer\_id

From Order.isbn to Book.isbn

**Buy** = (order\_id,customer\_id)

**SellOrders** = (sell\_id, customer\_id, bid,quantity)

From Sell.customer\_id *to* Customer.customer\_id

From Sell.bid to Book,bid

**Sell** = (order\_id,customer\_id)

**OrderDetails** = (detail\_id, bid, price,quantity,order\_status,order\_id)

From OrderDetails.order\_id to Order.order\_id

From OrderDetails.order\_id to Sell.sell\_id

From OrderDetails.price to Book.price

From OrderDetails.bid to Book.isbn