**REPORT**



|  |  |
| --- | --- |
| **과목명** | 데이터베이스 설계 |
| **학과** | 정보통신공학과 |
| **학번** | 12151636, 121551, 12151629 |
| **이름** | 최진우, 라찬엽, 주현성 |
| **제출일자** | 2019-12-23 |

<응용에 대한 간단한 소개>

기존의 온라인 쇼핑몰은 개인 구매로 이루어지며 개인이 원하는 상품을 어느 곳에든 살 수 있는 시스템입니다. 온라인 쇼핑몰이어도 비싼 가격에 아쉬운 마음으로 포기하는 경우가 많았습니다.

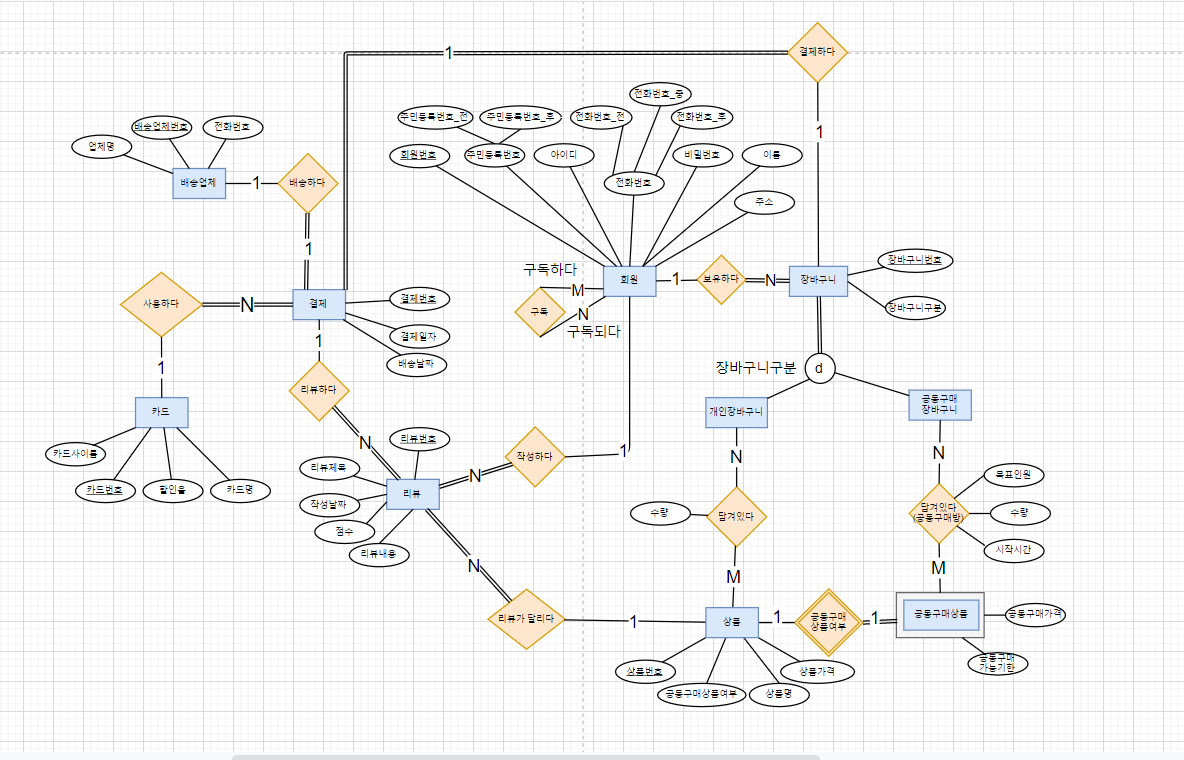
비싼 가격을 조금이나마 싸게 살 수 있는 방안은 공동구매라고 생각했습니다. 공동구매를 통해서 소비자는 값싼 가격에 구매하고 판매자는 대량의 물품을 팔 수 있기 때문에 서로 이득이라고 생각했습니다. 따라서 개인 구매 뿐만 아니라 공동구매라는 시스템을 도입하였습니다. 기존의 온라인 쇼핑몰의 장점들은 그대로 유지하면서 공동구매 시스템을 도입하여 보다 질 높은 프로그램을 구상하고 설계하였습니다.

<요구사항 명세서>

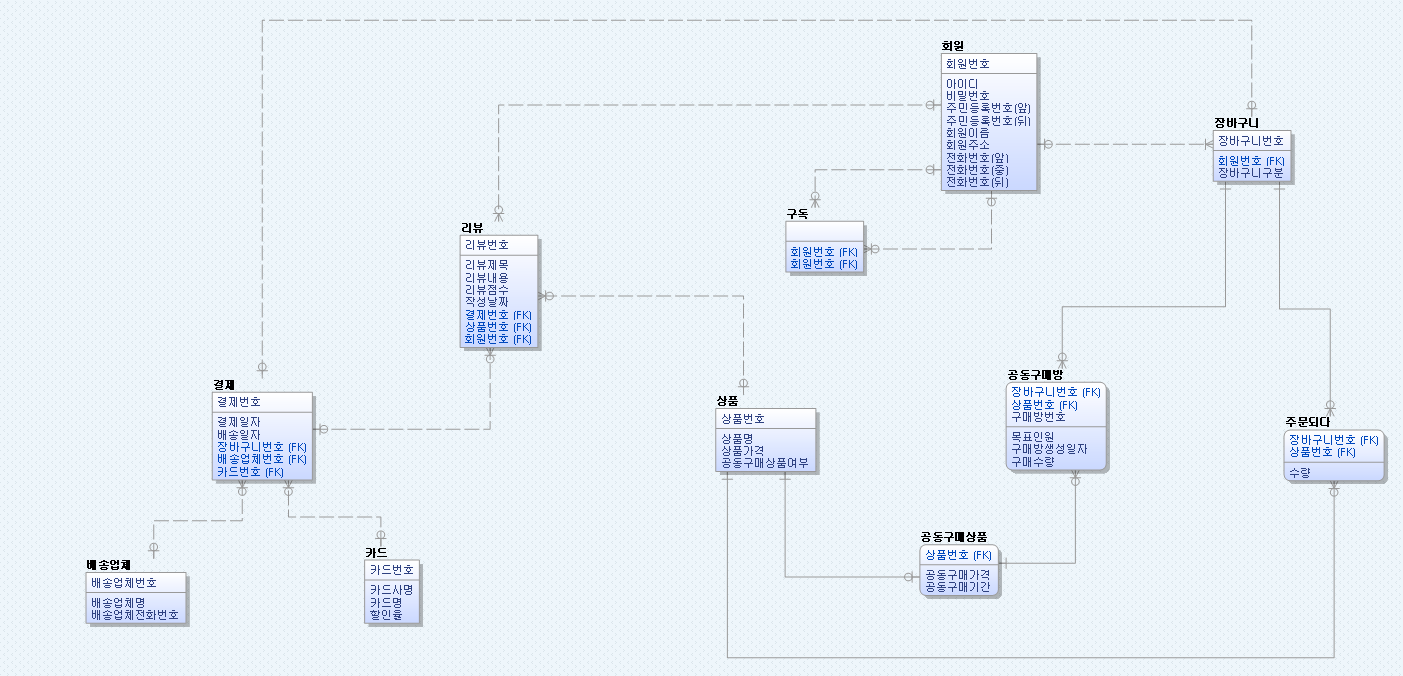
‘인하팡’에서는 물품거래와 공동 구매에 대한 시스템을 구축하려고 한다. 그를 위한 DB 설계를 위해 요구 사항을 다음과 같이 정리하였다.

* 회원은 기본적으로 아이디, 비밀번호, 이름, 회원번호, 주소, 전화번호, 주민등록번호로 이루어 진다.
* 원하는 상품을 담을 수 있는 장바구니가 있으며 장바구니는 여러 개의 상품을 담을 수 있고 한 상품은 여러 개의 장바구니에 담길 수 있다. 장바구니에는 장바구니 번호, 회원번호, 장바구니 구분 이 있다. 장바구니 구분은 공동구매 장바구니와 개인구매 장바구니를 구별하는 기준이 된다.
* 장바구니에 담긴 상품들을 담을 주문되다 와 공동구매방이 있다. 주문 되다는 개인 장바구니를 위한 것으로 개인 장바구니와 상품의 관계를 표시하고 개수를 가지고 있다. 공동구매방은 공동장바구니와 공동구매상품의 관계를 담고 있다. 장바구니 번호, 상품번호, 구매방 번호를 복합 키로 사용하고 같은 구매 그룹의 경우 같은 구매방 번호를 가진다. 또한 공동구매 목표인원, 구매방생성일자, 구매 수량으로 이루어 지며, 구매방은 생성 후 24시간이 지날 때 까지 목표인원에 도달하지 못하면 자동으로 취소된다. 24시간 이내에 목표인원에 도달하면 자동으로 결제가 완료된다.
* 상품은 상품번호, 상품명, 상품가격, 공동구매상품여부로 이루어지며 공동구매상품여부가 true 인 경우 공동구매상품을 가진다. 공동구매상품은 상품의 week entity로 공동구매가격, 공동구매기한 을 가진다.
* 결제는 결제번호, 결제일자, 배송일자, 장바구니번호, 배송업체번호, 카드번호 로 이루어 진다. 장바구니에 담긴 상품이 구매확정이 되면 결제가 생성된다.
* 리뷰는 리뷰번호, 리뷰제목, 리뷰내용, 리뷰점수, 작성날짜, 결제번호, 상품번호, 회원번호 로 이루어 진다. 결제번호를 갖는 이유는 구매한 상품에 대해서만 리뷰를 작성할 수 있도록 하기 위함이고 상품번호와 회원번호를 갖는 이유는 리뷰를 조회할 때의 효율성을 위해서 이다.
* 배송업체는 배송업체번호, 배송업체명, 배송업체전화번호를 갖는다.
* 카드는 카드번호, 카드사명, 카드명, 할인율 을 갖는다.
* 구독은 회원의 self relationship 이다. 선호하는 물건이 비슷하거나 성향이 비슷한 경우 구독하여 다양한 정보를 공유할 수 있다. 한 회원은 여러 회원을 구독할 수 있고 여러 회원으로부터 구독 될 수 있다.

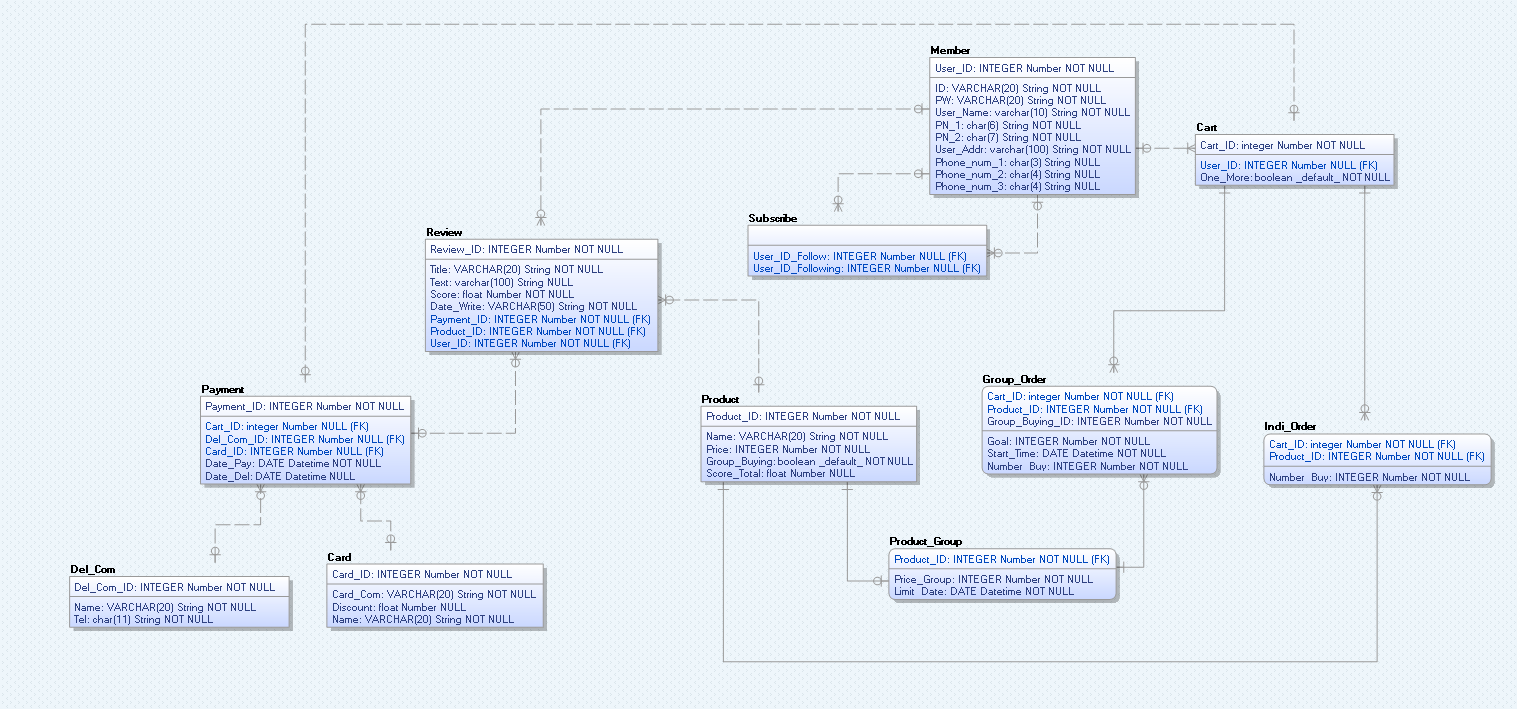
개념 다이어그램



개념 E-R 다이어그램

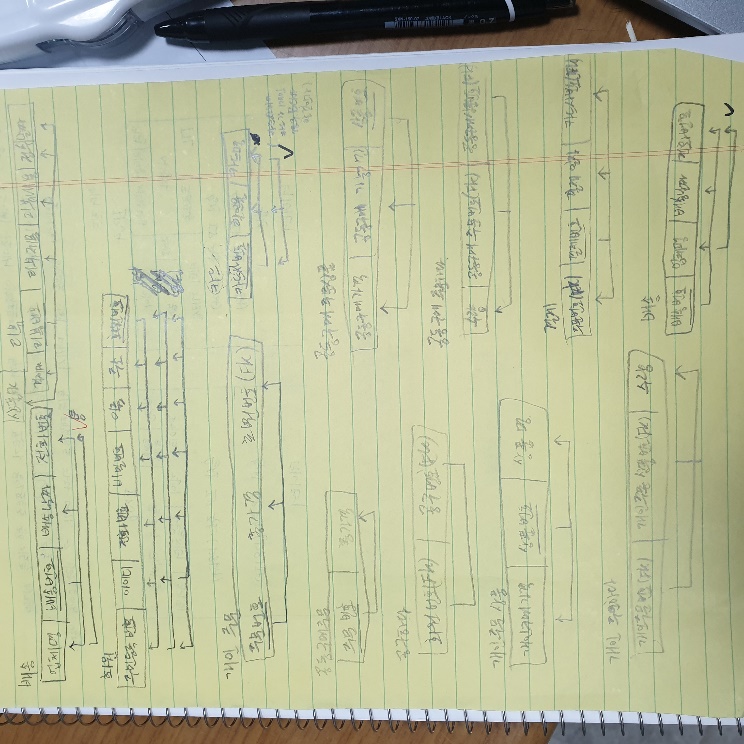


Physical 다이어그램

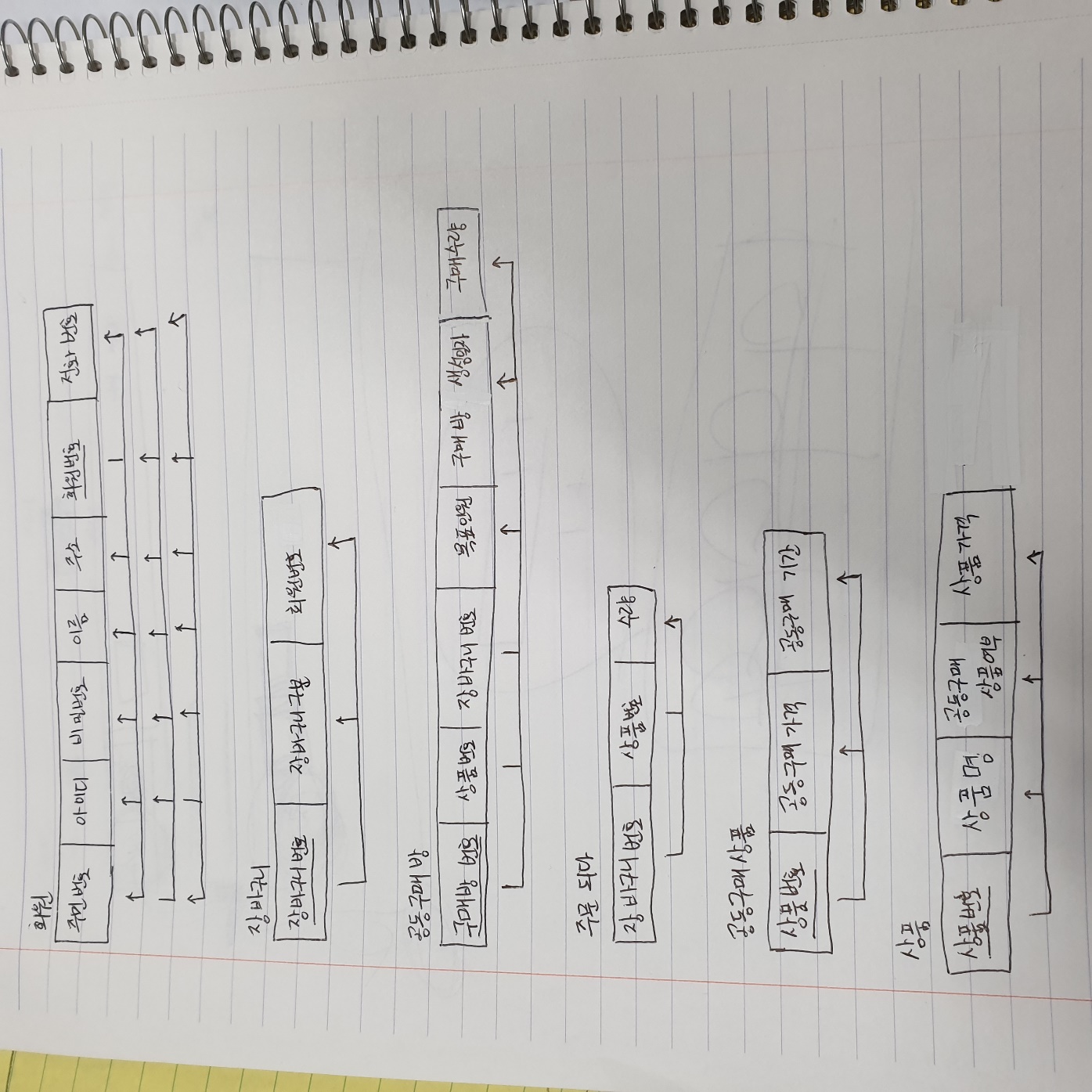


Normalize

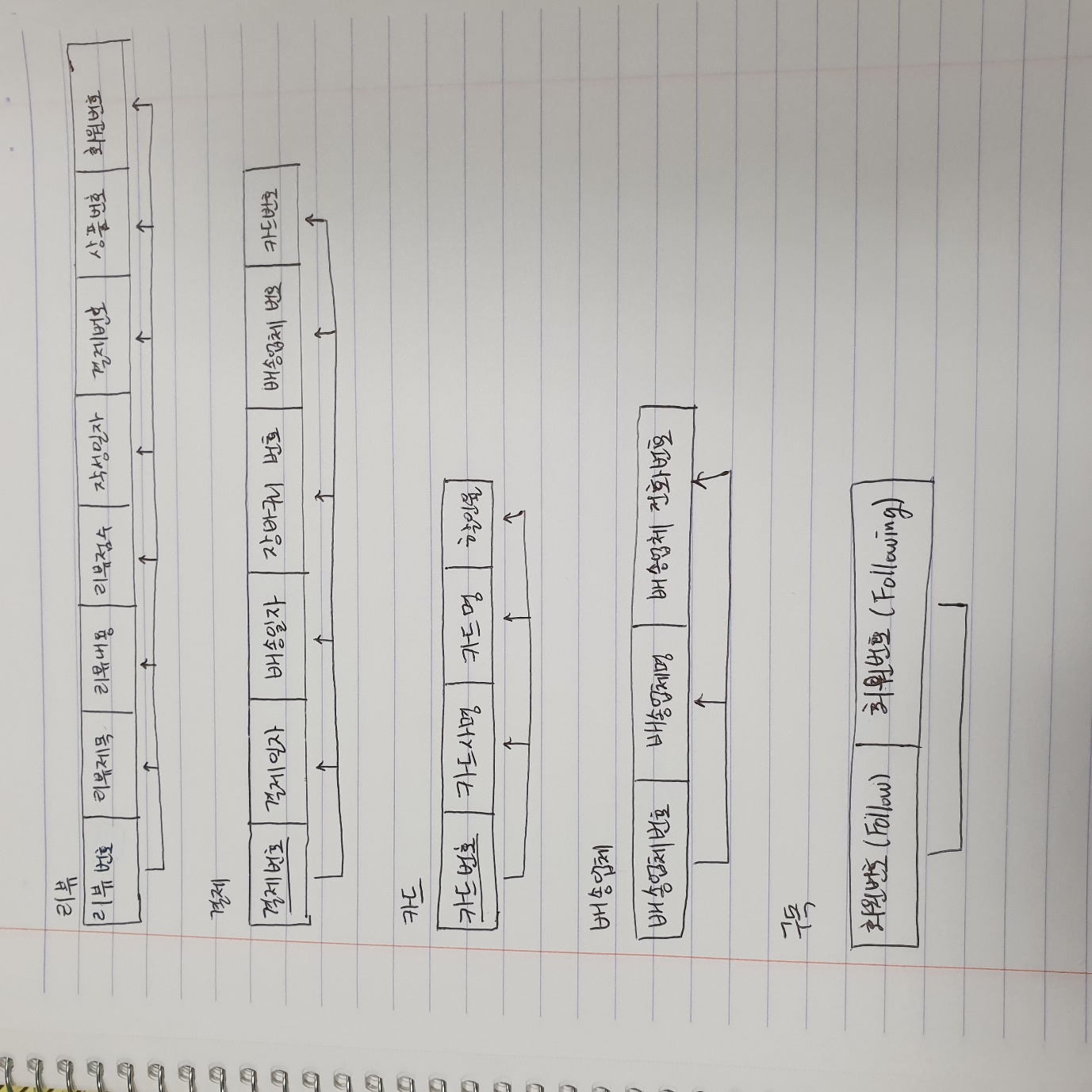
<첫 normalize>



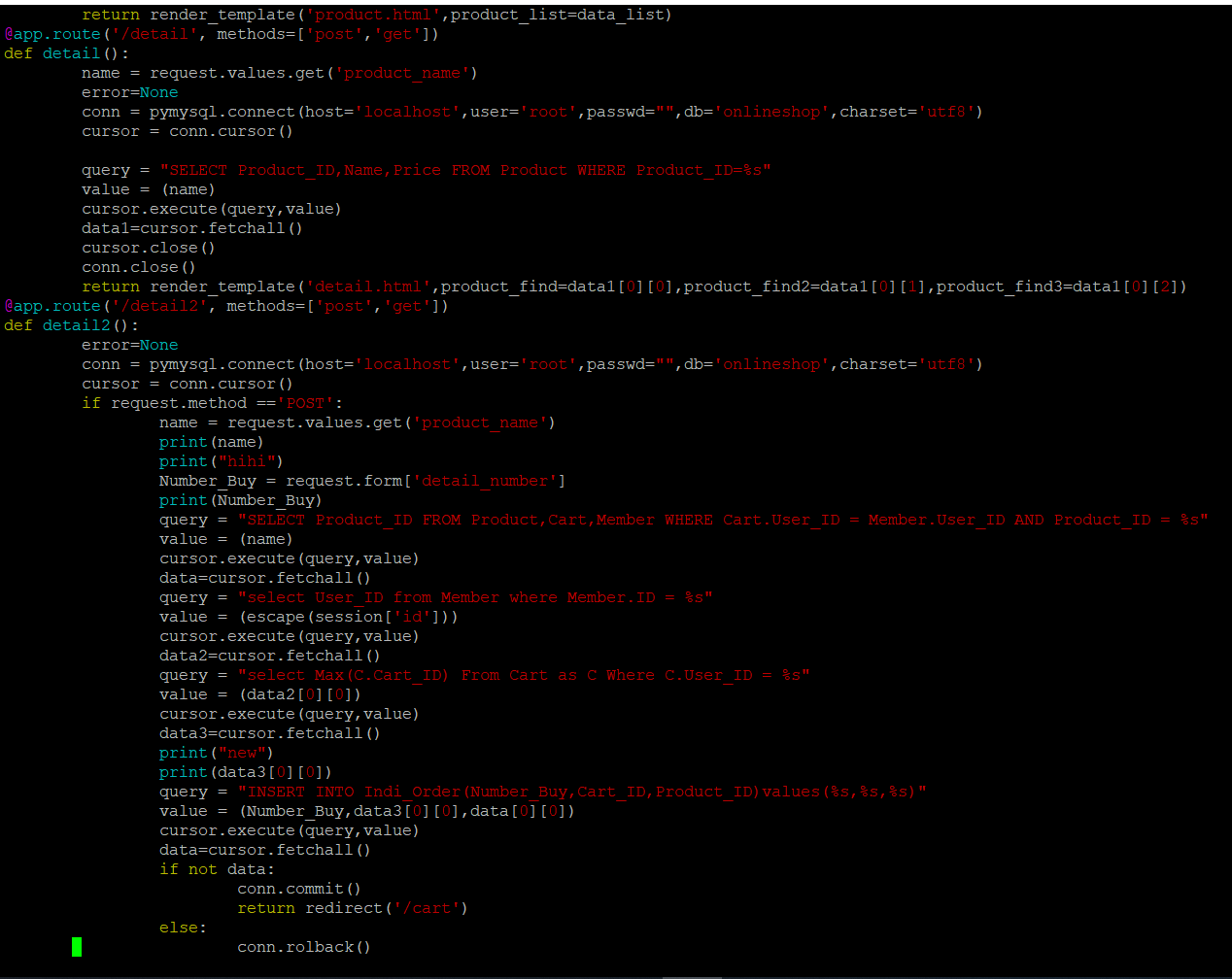
<최종 normalize>

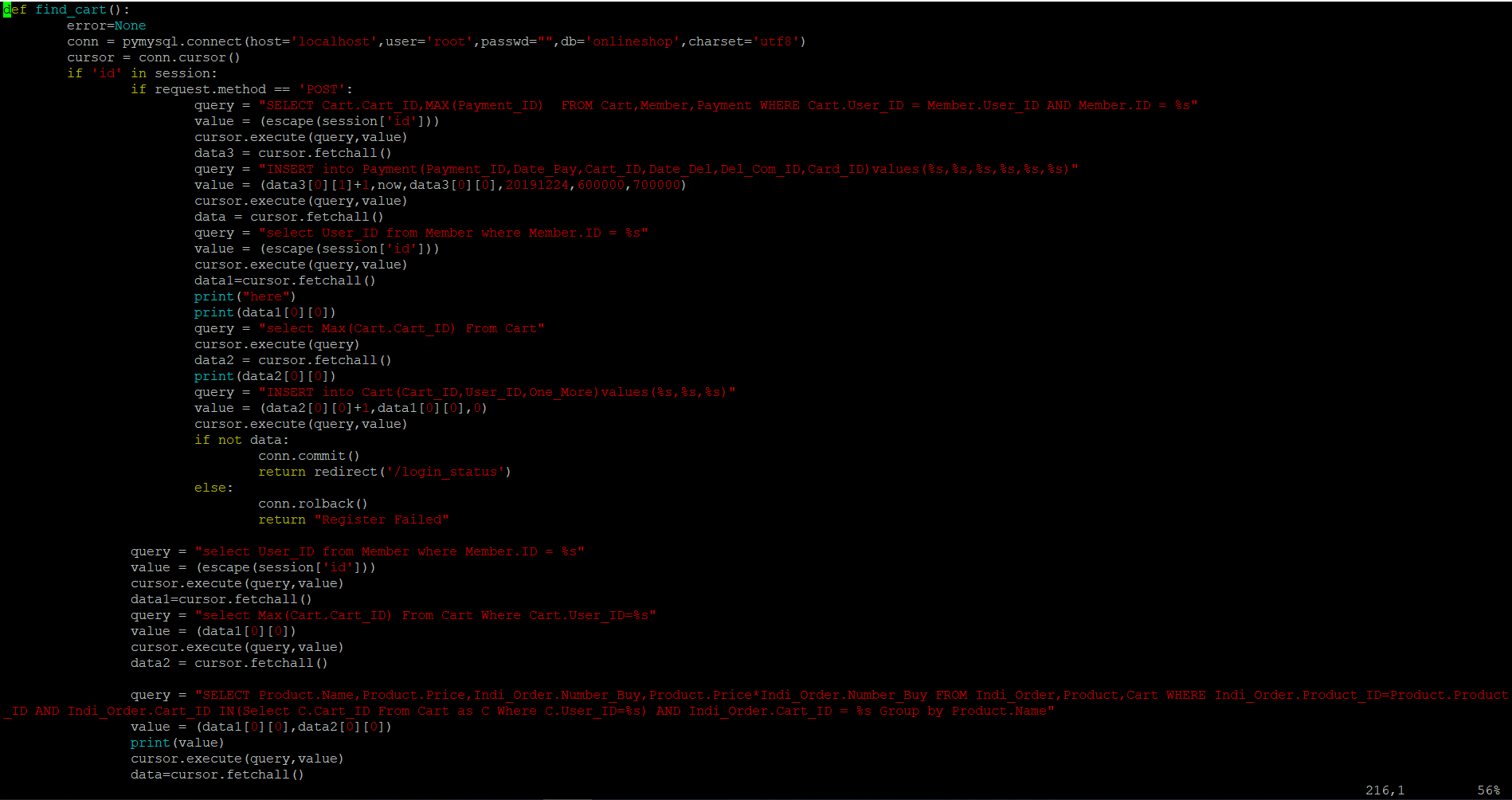


<최종 normalize>



Flask 구현





1. 자기가 구독하는 사람 중에 제일 구독자가 많은 회원의 쇼핑리스트

select \* from Cart,Subscribe,Indi\_Order Where Subscribe.User\_ID\_Follow=110025 and Cart.Cart\_ID = Indi\_Order.Cart\_ID and Cart.User\_ID = Subscribe.User\_ID\_Following and Subscribe.User\_ID\_Following in (select Subscribe.User\_ID\_Following

from Subscribe

where Subscribe.User\_ID\_Follow=110025 and Subscribe.User\_ID\_Following in (select User\_ID\_Following

from Subscribe

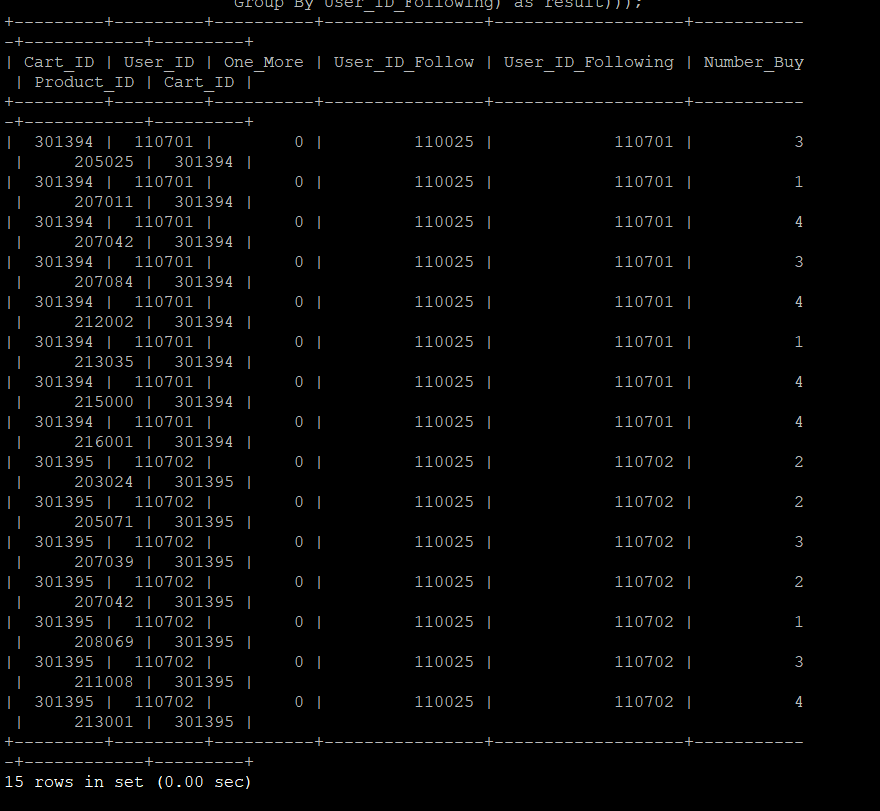
Group by User\_ID\_Following

Having count(\*) = (Select max(mycount)

from (select count(\*) as mycount

from Subscribe

Group By User\_ID\_Following) as result)));



1. 내가 구독한 사람이 개인주문을 하며 리뷰를 남겼고 결제를 했는가?

select \*

from Payment P

where EXISTS(select PP.Payment\_ID

from Payment PP,Review

where P.Payment\_ID= PP.Payment\_ID AND PP.Payment\_ID = Review.Payment\_ID AND Review.Payment\_ID in (select Review.Payment\_ID

from Review,Product

where Review.Product\_ID =Product.Product\_ID and Product.Product\_ID in ( select B.Product\_ID

from Product B

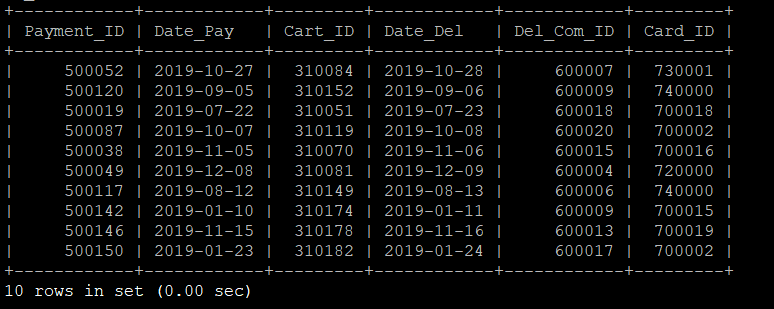
where B.Product\_ID in (select DISTINCT Product.Product\_ID

from Product, Cart, Member,Indi\_Order,Subscribe

where Product.Product\_ID = Indi\_Order.Product\_ID AND Cart.Cart\_ID=Indi\_Order.Cart\_ID AND Member.User\_ID=Cart.User\_ID AND Member.User\_ID = Subscribe.User\_ID\_Following and Subscribe.User\_ID\_Following in(select Subscribe.User\_ID\_Following

from Subscribe

where Subscribe.User\_ID\_Follow = '110025')))));

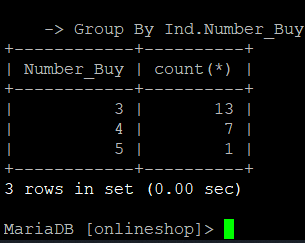


1. 서울사는 사람에 이씨 성을 가진 사람들 중에 구매 량이 3이상 5이하인 것을 정렬

select Ind.Number\_Buy,count(\*) from Indi\_Order Ind where Number\_Buy between 3 and 5 and Number\_Buy in (Select ln.Number\_Buy from Indi\_Order ln, Cart C where ln.Cart\_ID=Ind.Cart\_ID AND ln.Cart\_ID=C.Cart\_ID AND C.Cart\_ID in ( Select Cart.Cart\_ID from Member,Cart where Member.User\_ID = Cart.User\_ID AND Member.User\_ID in(Select User\_ID from Member

where User\_Addr Like "서울%" and User\_Name Like "이%")))

Group By Ind.Number\_Buy;

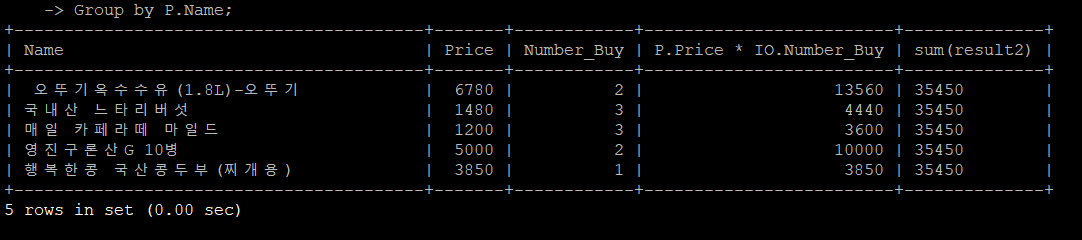


1. 특정ID의 장바구니와 총 가격 뽑기

Select P.Name, P.Price, IO.Number\_Buy, P.Price \* IO.Number\_Buy,sum(result2)

From Indi\_Order as IO, Product as P,(SELECT P2.Price\*IO2.Number\_Buy as result2 FROM Indi\_Order as IO2, Product as P2 WHERE IO2.Product\_ID = P2.Product\_ID AND IO2.Cart\_ID IN(Select C2.Cart\_ID From Cart as C2 Where C2.User\_ID = 120003) Group by P2.Name)result\_table Where P.Product\_ID = IO.Product\_ID and IO.Cart\_ID IN (Select C.Cart\_ID From Cart as C Where C.User\_ID =120003)

Group by P.Name;



1. 내가 구독한 사람 중 제일 구독자가 많은 사람들의 리뷰와 내 리뷰와 union

select DISTINCT Review\_ID,Title,Text,Score,Member.User\_ID from Review,Member,Subscribe where Review.User\_ID=Member.User\_ID AND Member.User\_ID=Subscribe.User\_ID\_Following and User\_ID\_Following in (select Subscribe.User\_ID\_Following

from Subscribe

where Subscribe.User\_ID\_Follow=110025 and Subscribe.User\_ID\_Following in (select User\_ID\_Following

from Subscribe

Group by User\_ID\_Following

Having count(\*) = (Select max(mycount)

from (select count(\*) as mycount

from Subscribe

Group By User\_ID\_Following) as result)))

UNION

select DISTINCT Review\_ID,Title,Text,Score,Member.User\_ID

from Review,Member

where Review.User\_ID = Member.User\_ID and Member.User\_ID=110025;

