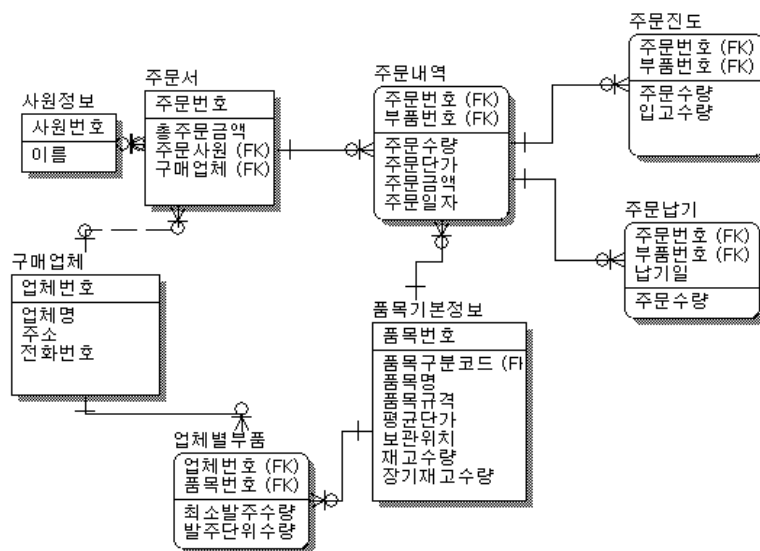


울산대학교
IT융합학부

Term Project (TP #1)

과목명 : 데이터베이스관리 및 실험



과제 내용		오라클 인터넷 서점 웹사이트 구축 프로젝트
제출자	전공	IT 융합
	학년	3 학년
	학번	20184258
	이름	박선재
제출일자		2020. 05. 25.

담당교수 : 안형근

(sign)

과제점수		A	B	C	D	F
------	--	---	---	---	---	---

Good Luck !

<목 차>

1. 요구사항 명세	3
1.1 요구사항	4
1.2 요구사항 명세서	4
2. 개념적 설계 : 엔티티/속성 정의서	4
2.1 엔티티 분석서	4
2.2 엔티티 속성/정의서	5
3. 개념적 설계 : 엔티티별 정의 모델	5
3.1 회원 엔티티	6
3.2 신용카드 엔티티	7
3.3 배송지 엔티티	7
3.4 장바구니 엔티티	8
3.5 주문목록 엔티티	8
3.6 도서 엔티티	8
4. 개념적 간의 관계 분석	7
5. 엔티티 간의 관계 ER다이어그램	8
6. 개념 설계 : 최종 E-R다이어그램	9
7. 논리적 설계 : 관계 모델 변환	9
7.1 엔티티를 릴레이션 모델로 변환	9
7.2 관계 엔티티를 릴레이션 모델로 변환	10
8. 최종 릴레이션 스키마	13
9. 논리 설계 : 테이블 명세서	14
7.1 회원 테이블	14
7.2 신용카드 테이블	14
7.3 배송지 테이블	14
7.4 장바구니 테이블	14

7.5 주문목록 테이블	15
7.6 도서 테이블	15
10. 논리적 모델 구축 : IE 표기법	15
11. 물리적 모델 구축 : IE 표기법	16
12. 정규화 작업	18
13. 데이터베이스 구축 sql문	20
14. 한 학기를 마치며	23

1. 요구사항 명세서

1.1) 요구사항

우리 오라클 서점이 인터넷 서점의 웹사이트를 구축하게 되면 웹사이트에서 회원으로 가입해야만 인터넷에서 도서를 구입할 수 있다. 인터넷 회원에 가입할 때 반드시 입력해야 하는 정보에는 회원번호, 회원명, 연락처, 비밀번호, 아이디 등이 있다. 더불어 신용카드 정보는 여러 개 저장할 수 있는데, 번호, 유효기간, 카드 종류를 저장할 수 있다. 배송지정보도 마찬가지로 여러 개 저장할 수 있으며 배송지 우편번호와 기본주소, 상세주소를 저장할 수 있다. 인터넷을 통해 등록한 회원은 구매할 도서목록을 선택하여 주문하기 전에 장바구니에 담을 수 있다. 이 때 장바구니 번호와 생성일자를 저장한다. 인터넷 통해 등록한 회원은 구매할 도서목록을 선택하여 장바구니에 담지 않고 인터넷 주문서에 주문 내용을 입력하여 바로 주문할 수도 있다.

주문할 때는 주문목록에 대한 상세정보와 주문번호, 주문일자, 주문총액, 신용카드 종류, 신용카드 번호, 신용카드 유효기간, 배송지 우편번호, 배송지 기본주소, 배송지 상세주소를 지정해야 한다. 주문목록에 대한 상세정보에는 수량이 있다.

도서정보에는 도서번호, 도서명, 재고량, 판매가를 저장한다.

1.2) 요구 사항 명세서

- ① 오라클 서점의 웹사이트에 회원으로 가입하려면 회원번호, 회원명, 연락처, 비밀번호, 아이디를 입력해야 한다.
- ② 신용카드 정보는 여러 개 저장할 수 있고, 번호, 유효기간, 카드 종류를 저장할 수 있다.
- ③ 배송지 정보는 여러 개 저장할 수 있고, 배송지 우편번호, 기본주소, 상세주소를 저장할 수 있다.
- ④ 회원은 구매할 도서목록을 선택하여 주문하기 전에 장바구니에 담을 수 있다.
- ⑤ 회원은 주문 전 장바구니에 담으면 장바구니 번호와 생성일자를 저장한다.
- ⑥ 회원은 구매할 도서목록을 선택하여 장바구니에 담지 않고 인터넷 주문서에 주문 내용을 바로 주문할 수 있다.
- ⑦ 주문할 때는 주문 목록에 대한 상세정보, 주문일자, 주문총액, 신용카드 종류, 신용카드 번호, 신용카드 유효기간, 배송지 우편번호, 배송지 기본주소, 배송지 상세주소를 지정해야 한다.
- ⑧ 주문목록에 대한 상세정보에는 수량이 있다.
- ⑨ 도서 정보에는 도서번호, 도서명, 재고량, 판매가를 저장한다.

2. 개념적 설계 : 엔티티/속성 정의서

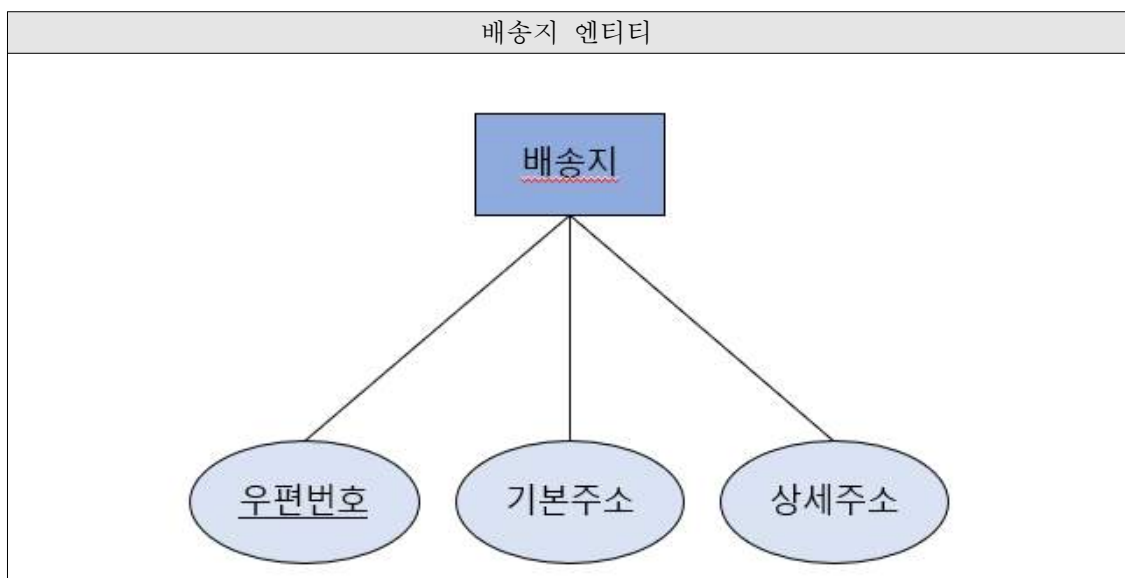
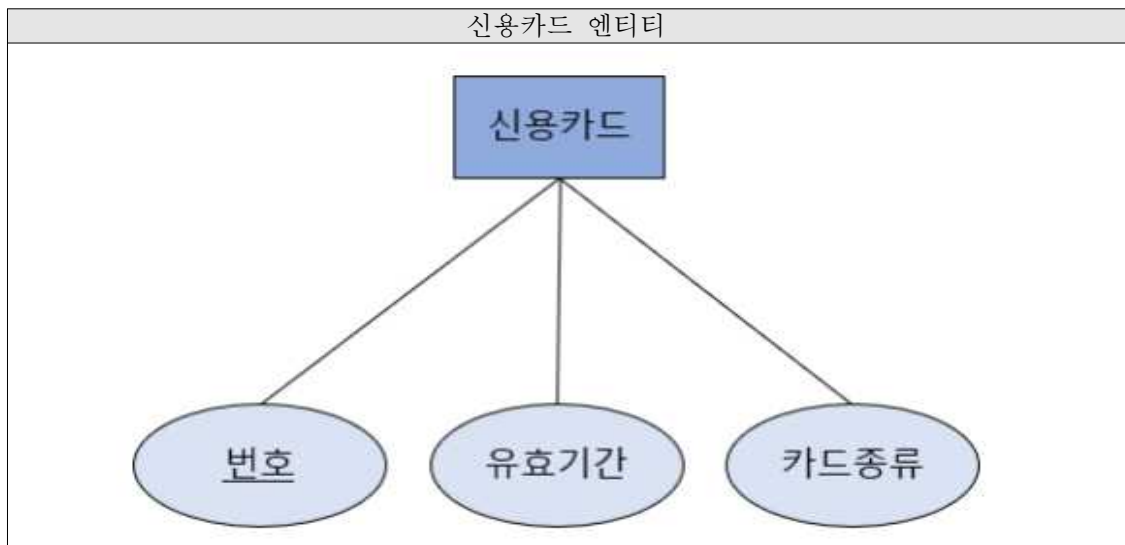
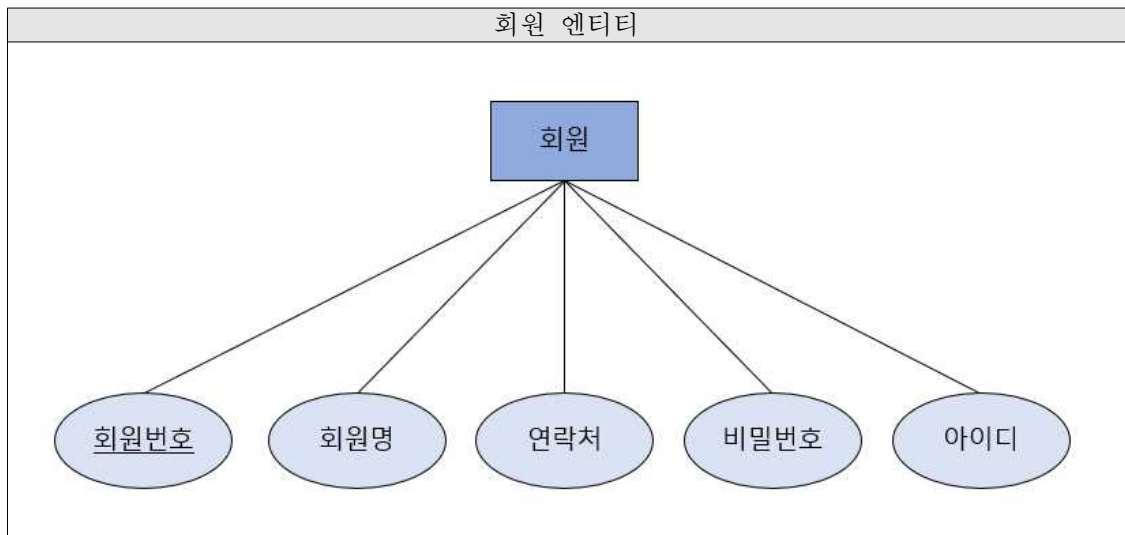
2.1) 엔티티 분석서

- ① 오라클 서점의 웹사이트에 **회원**으로 가입하려면 회원번호, 회원명, 연락처, 비밀번호, 아이디를 입력해야 한다.
- ② **신용카드** 정보는 여러 개 저장할 수 있고, 번호, 유효기간, 카드 종류를 저장할 수 있다.
- ③ **배송지 정보**는 여러 개 저장할 수 있고, 배송지 우편번호, 기본주소, 상세주소를 저장할 수 있다.
- ④ **회원**은 구매할 **도서목록**을 선택하여 주문하기 전에 장바구니에 담을 수 있다, (**회원 - 도서**) : 주문 관계
- ⑤ **회원**은 주문 전 **장바구니**에 담으면 장바구니 번호와 생성일자를 저장한다.
- ⑥ **회원**은 구매할 **도서목록**을 선택하여 장바구니에 담지 않고 인터넷 주문서에 주문 내용을 바로 주문할 수 있다.
- ⑦ 주문할 때는 **주문 목록**에 대한 주문번호, 상세정보, 주문일자, 주문총액, 신용카드 종류, 신용카드 번호, 신용카드 유효기간, 배송지 우편번호, 배송지 기본주소, 배송지 상세주소를 지정해야 한다.
- ⑧ 주문목록에 대한 상세정보에는 수량이 있다.
- ⑨ **도서** 정보에는 도서번호, 도서명, 재고량, 판매가를 저장한다.

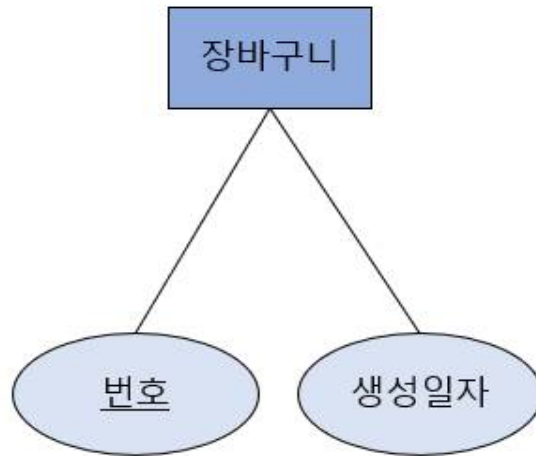
2.2) 엔티티/속성 정의서

엔티티	엔티티 설명	속성	비고
회원	오라클 서점의 웹사이트에 가입한 회원의 정보.	회원번호 회원명 연락처 비밀번호 아이디	기본키 : 회원번호 or 아이디
신용카드	회원이 도서목록을 주문하여 결제 할 때 사용할 신용카드의 정보.	번호 유효기간 카드 종류	기본키 : 번호
배송지	회원이 주문한 도서목록을 받을 배송지에 대한 정보.	우편번호 기본주소 상세주소	기본키 : 우편번호
장바구니	회원은 주문 전 장바구니에 담을시 장바구니에 대한 정보.	번호 생성일자	기본키 : 번호
주문목록	회원이 주문할 도서에 대한 정보.	상세정보(수량) 주문번호 주문일자 주문총액 신용카드 종류 신용카드 번호 신용카드 유효기간 배송지 우편번호 배송지 기본주소 배송지 상세주소	기본키 : 주문번호
도서	도서에 대한 정보	도서번호 도서명 재고량 판매가	기본키 : 도서번호

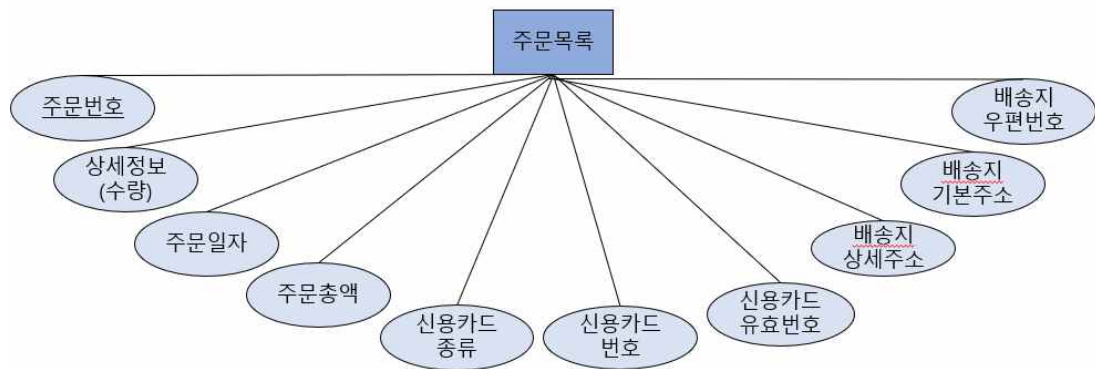
3. 개념적 설계 : 엔티티별 정의 모델



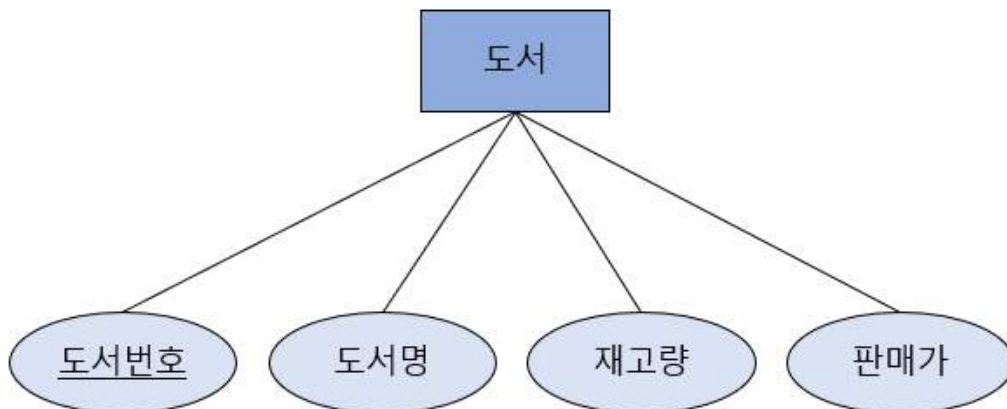
장바구니 엔티티



주문목록 엔티티



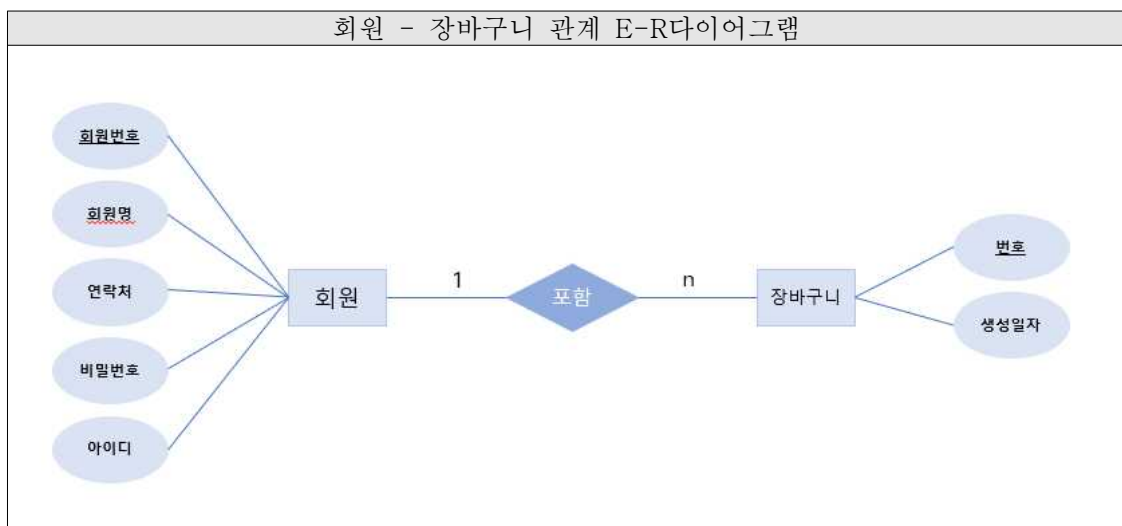
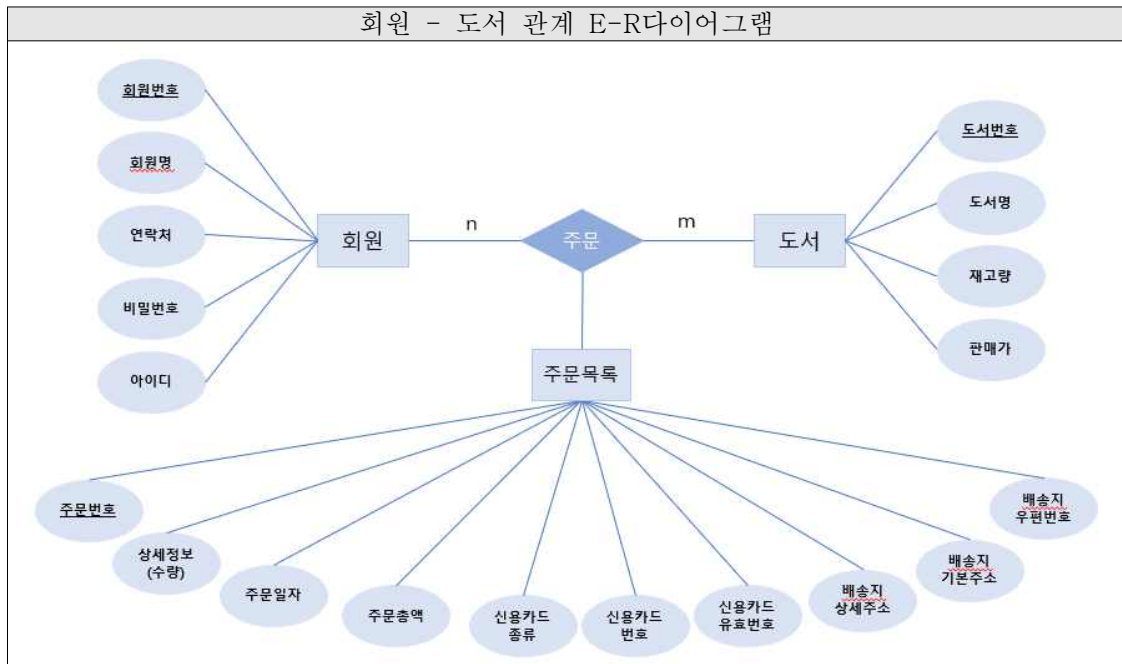
도서 엔티티



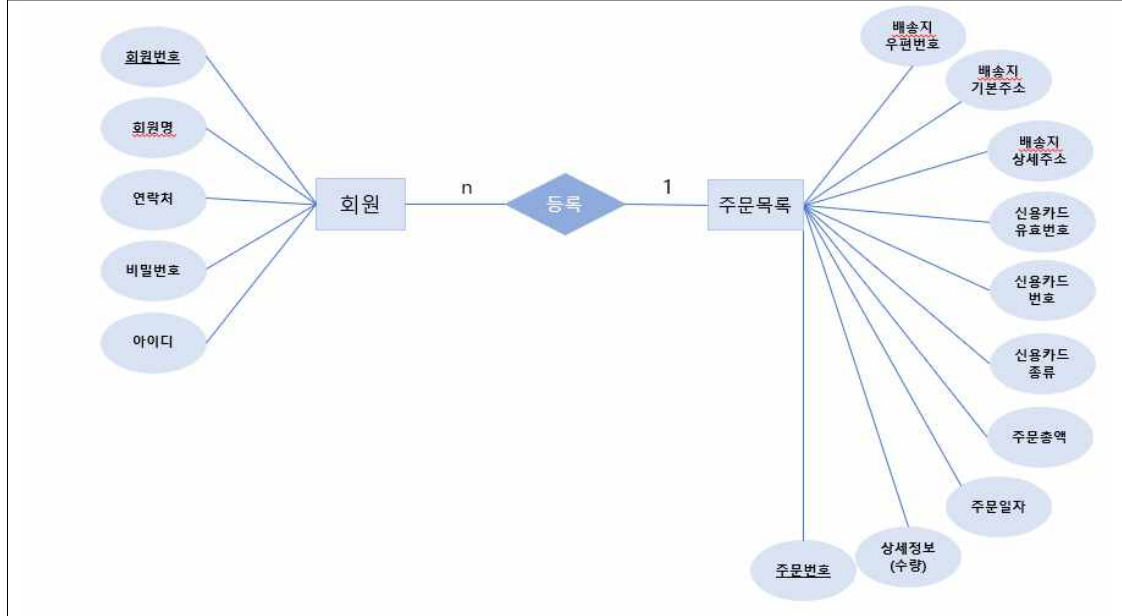
4. 엔티티 간의 관계 분석

엔티티	관계 설명	관계 형태	관계 엔티티
회원	회원은 여러 도서를 주문할 수 있다. 하나의 도서는 여러 회원이 주문할 수 있다.	선택(0..m) 선택(0..m)	도서
		m : n	
장바구니	회원은 여러 도서를 장바구니에 담을 수 있다.	n : 1	회원
도서	도서는 주문한 회원에게 공급된다.	m : n	회원
주문목록	회원이 주문 할 때 필요한 정보들을 주문 목록에 등록한다.	n : 1	회원
장바구니	장바구니에 도서를 담을 수 있다.	1 : n	도서

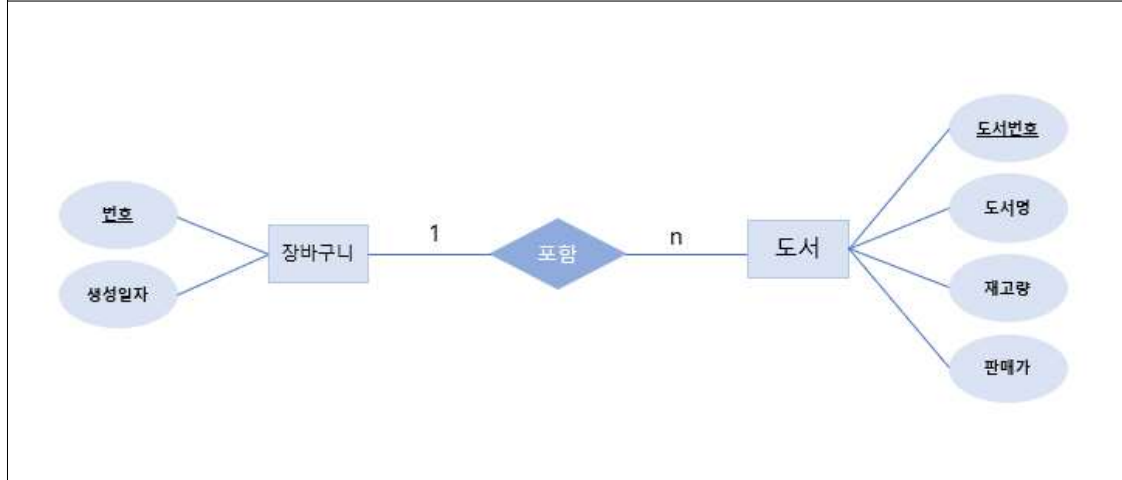
5. 엔티티 간의 관계 ER다이어그램



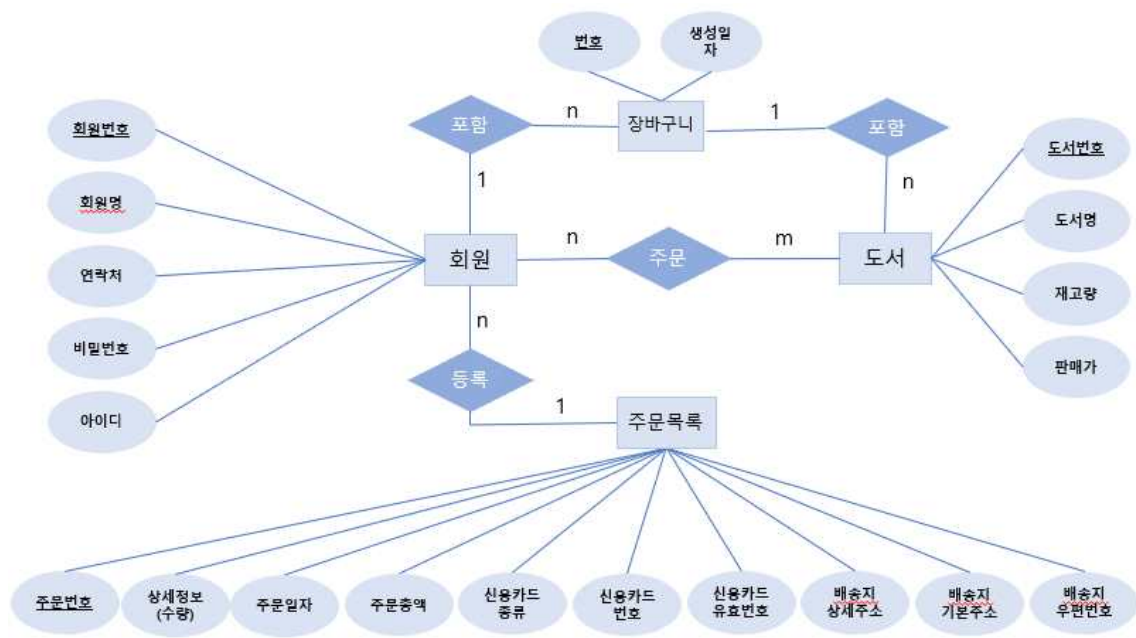
회원 - 주문목록 관계 E-R 다이어그램



장바구니 - 도서 관계 E-R 다이어그램



6. 개념 설계 : 최종 E-R 다이어그램



7. 논리적 설계 : 관계 모델 변환

7.1 엔티티를 릴레이션 모델로 변환

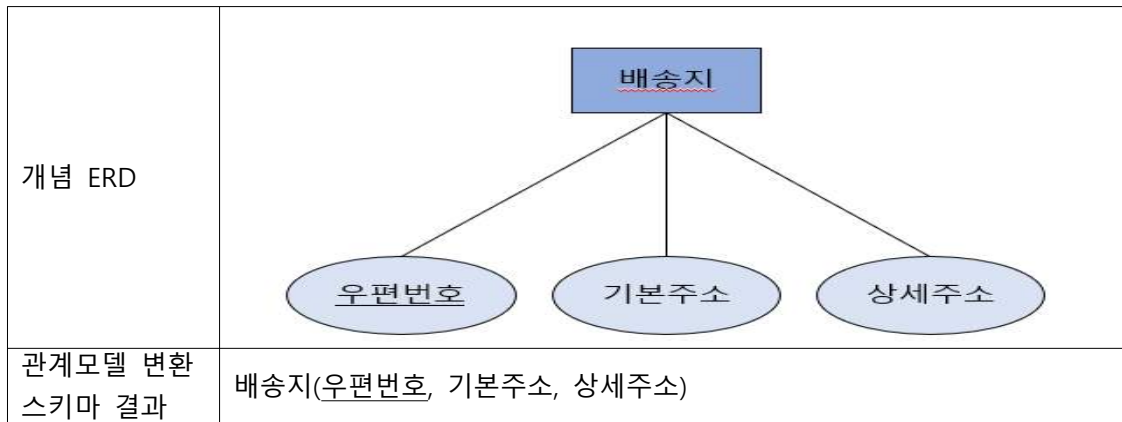
1) 회원 엔티티 → 회원 릴레이션 스키마

개념 ERD	
관계모델 변환 스키마 결과	회원(회원번호, 회원명, 연락처, 비밀번호, 아이디)

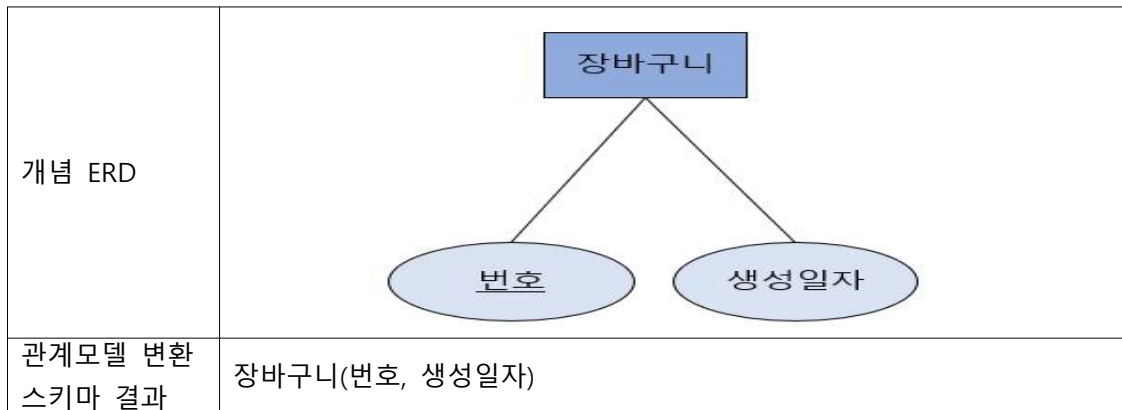
2) 상품 엔티티 → 상품 릴레이션 스키마

개념 ERD	
관계모델 변환 스키마 결과	신용카드(번호, 유효기간, 카드종류)

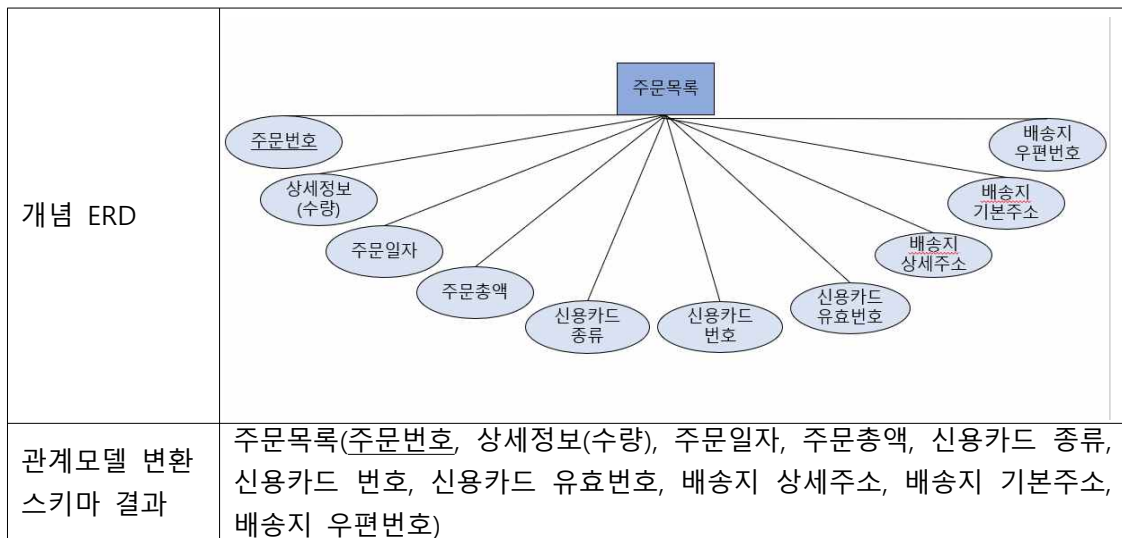
3) 배송지 엔티티 → 배송지 릴레이션 스키마



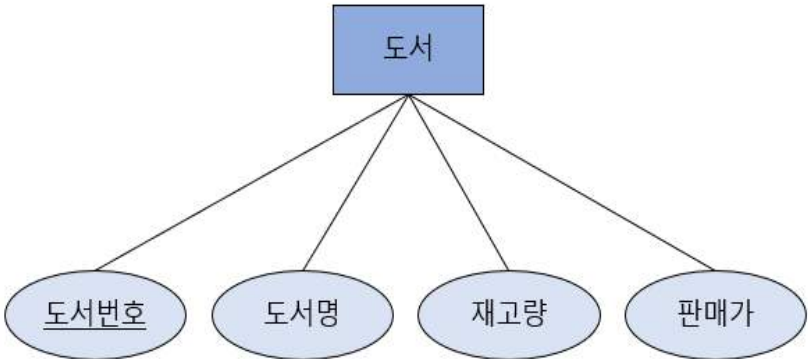
4) 장바구니 엔티티 → 장바구니 릴레이션 스키마



5) 주문목록 엔티티 → 주문목록 릴레이션 스키마

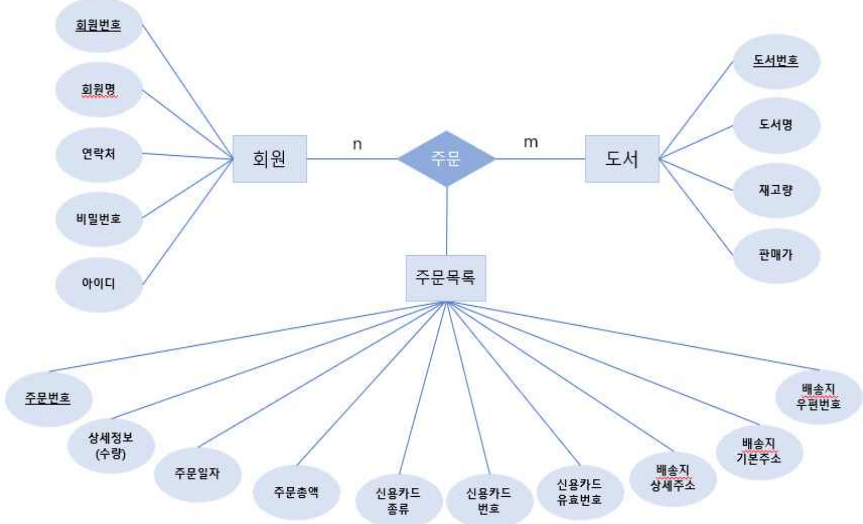


6) 도서 엔티티 → 도서 릴레이션 스키마

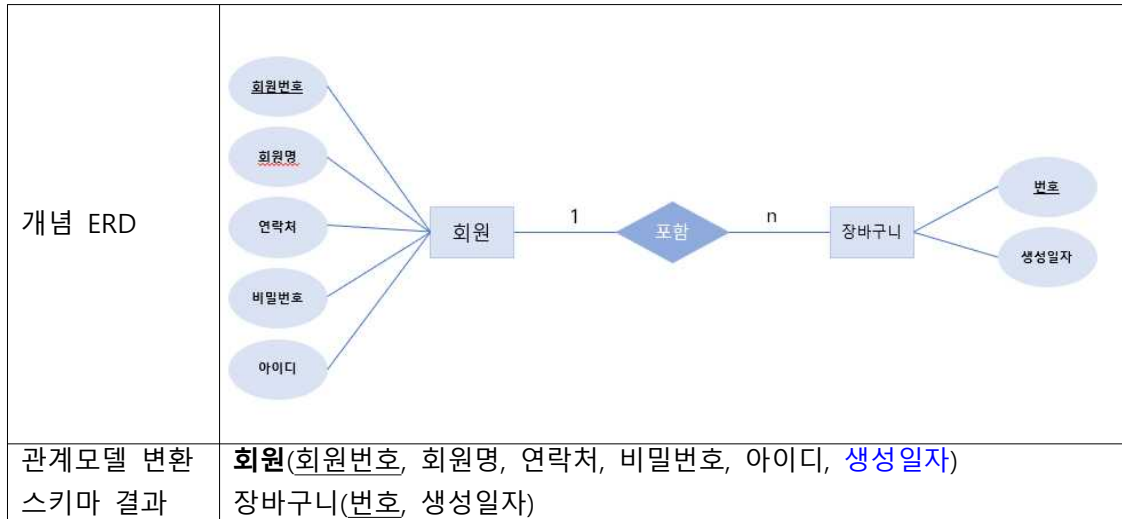
개념 ERD	
관계모델 변환 스키마 결과	<p>도서 (도서번호, 도서명, 재고량, 판매가)</p>

7) 관계 엔티티를 릴레이션 모델로 변환

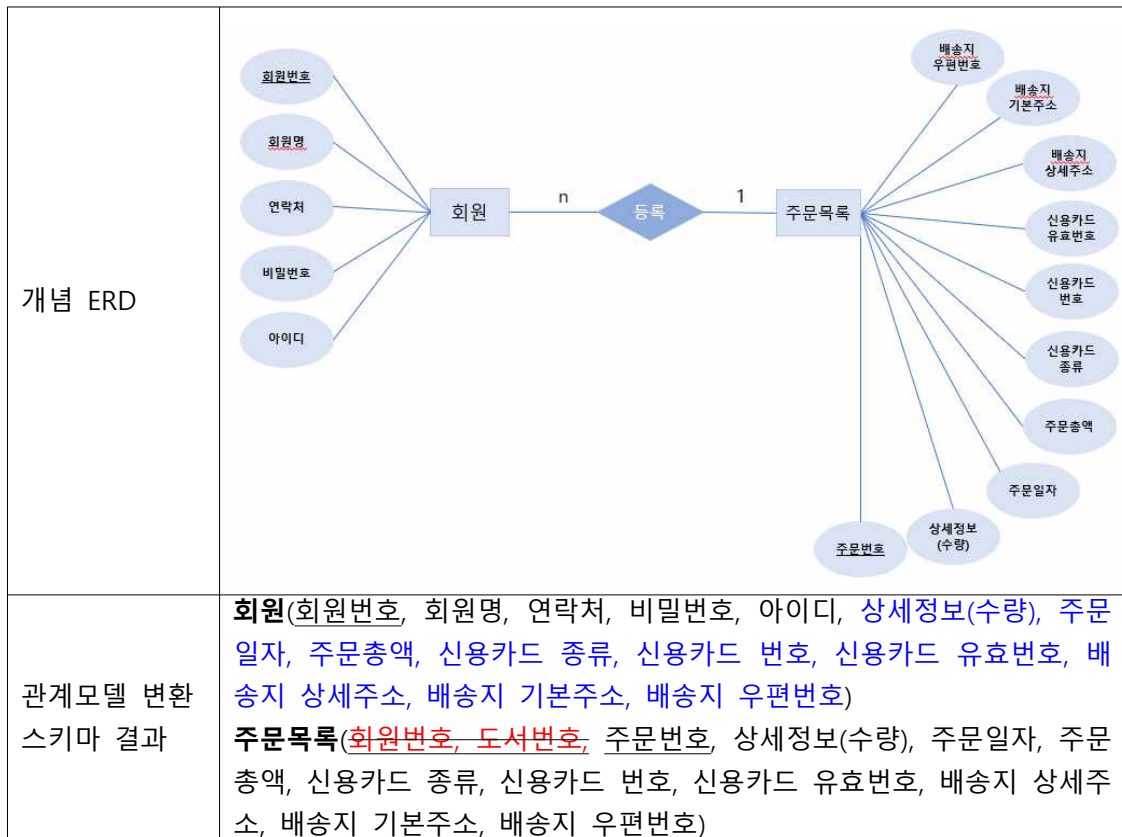
1) 회원 엔티티와 도서 엔티티의 관계 모델 변환 / 릴레이션 스키마

개념 ERD	
관계모델 변환 스키마 결과	<p>회원(회원번호, 회원명, 연락처, 비밀번호, 아이디)</p> <p>도서(도서번호, 도서명, 재고량, 판매가)</p> <p>주문목록(회원번호, 도서번호, 주문번호, 상세정보(수량), 주문일자, 주문총액, 신용카드 종류, 신용카드 번호, 신용카드 유효번호, 배송지 상세주소, 배송지 기본주소, 배송지 우편번호)</p>

2) 회원 엔티티와 장바구니 엔티티 간의 관계 모델 변화/ 스키마



3) 회원 엔티티와 주문목록 엔티티 간의 관계 모델 변환 / 스키마



4) 장바구니 엔티티와 도서 엔티티 간의 관계 모델 변환 / 스키마

개념 ERD	
관계모델 변환 스키마 결과	장바구니(번호, 생성일자, 도서명, 재고량, 판매가) 도서(도서번호, 도서명, 재고량, 판매가)

8) 최종 릴레이션 스키마

회원 릴레이션	<table><tr><td><u>회원 번호</u></td><td>회원명</td><td>연락처</td><td>비밀번호</td><td>아이디</td></tr><tr><td><u>카드번호</u></td><td>유효기간</td><td>카드종류</td><td></td><td></td></tr></table>	<u>회원 번호</u>	회원명	연락처	비밀번호	아이디	<u>카드번호</u>	유효기간	카드종류		
<u>회원 번호</u>	회원명	연락처	비밀번호	아이디							
<u>카드번호</u>	유효기간	카드종류									
	<table><tr><td>외래키</td></tr></table>	외래키									
외래키											
도서 릴레이션	<table><tr><td><u>도서 번호</u></td><td>도서명</td><td>재고량</td><td>판매가</td></tr></table>	<u>도서 번호</u>	도서명	재고량	판매가						
<u>도서 번호</u>	도서명	재고량	판매가								
신용카드 릴레이션	<table><tr><td><u>카드번호</u></td><td>유효기간</td><td>카드종류</td></tr></table>	<u>카드번호</u>	유효기간	카드종류							
<u>카드번호</u>	유효기간	카드종류									
배송지 릴레이션	<table><tr><td><u>우편 번호</u></td><td>상세 주소</td><td>기본 주소</td></tr></table>	<u>우편 번호</u>	상세 주소	기본 주소							
<u>우편 번호</u>	상세 주소	기본 주소									
	<table><tr><td>외래키</td></tr></table>	외래키									
외래키											
장바구니 릴레이션	<table><tr><td><u>번호</u></td><td>생성일자</td><td><u>도서번호</u></td><td>재고량</td><td>판매가</td></tr></table>	<u>번호</u>	생성일자	<u>도서번호</u>	재고량	판매가					
<u>번호</u>	생성일자	<u>도서번호</u>	재고량	판매가							
주문목록 릴레이션	<table><tr><td><u>주문 번호</u></td><td>상세정보(수량)</td><td>주문일자</td><td>주문총액</td></tr></table>	<u>주문 번호</u>	상세정보(수량)	주문일자	주문총액						
<u>주문 번호</u>	상세정보(수량)	주문일자	주문총액								
	<table><tr><td>신용카드 종류</td><td>신용카드 번호</td><td>신용카드 유효번호</td><td>배송지 상세주소</td><td>배송지 기본주소</td><td>배송지 우편번호</td></tr></table>	신용카드 종류	신용카드 번호	신용카드 유효번호	배송지 상세주소	배송지 기본주소	배송지 우편번호				
신용카드 종류	신용카드 번호	신용카드 유효번호	배송지 상세주소	배송지 기본주소	배송지 우편번호						

9. 논리 설계 :테이블 명세서

(1) 회원 테이블

테이블 명	member	테이블기술서	작성일		2020/06/15		Page1/		
시스템 명	인터넷 서점의 웹사이트 DB 구축		작성자		울산대 박선재				
테이블설명	회원 테이블								
No	Attribute	Data type	NN	PK	FK	FK.tab	FK.col	Def.	Description
1	mem_num	number	O	O	O	신용카드 배송지	카드번호 우편번호		회원번호
2	mem_name	varchar2(10)	O						회원명
3	mem_pnum	varchar2(20)							연락처
4	mem_pw	varchar2(20)	O						비밀번호
5	mem_id	varchar2(20)	O						아이디

(2) 신용카드 테이블

테이블 명	CreditCard	테이블기술서	작성일		2020/06/15		Page1/		
시스템 명	인터넷 서점의 웹사이트 DB 구축		작성자		울산대 박선재				
테이블설명	신용카드 테이블								
No	Attribute	Data type	NN	PK	FK	FK.tab	FK.col	Def.	Description
1	Card_num	number	O	O	O				카드번호
2	Card_VValidity	varchar2(20)	O						유효기간
3	Card_kind	carchar2(20)	O						카드종류

(3) 배송지 테이블

테이블 명	Shipping	테이블기술서	작성일		2020/06/15		Page1/		
시스템 명	인터넷 서점의 웹사이트 DB 구축		작성자		울산대 박선재				
테이블설명	배송지 테이블								
No	Attribute	Data type	NN	PK	FK	FK.tab	FK.col	Def.	Description
1	Ship_num	number	O	O	O				우편번호
2	Ship_address	varchar(30)	O						기본주소
3	Ship_detail	varchar(30)	O						상세주소

(4) 장바구니 테이블

테이블 명	basket	테이블기술서	작성일		2020/06/15		Page1/		
시스템 명	인터넷 서점의 웹사이트 DB 구축		작성자		울산대 박선재				
테이블설명	장바구니 테이블								
No	Attribute	Data type	NN	PK	FK	FK.tab	FK.col	Def.	Description
1	Basket_num	number	O	O	O	회원 도서	회원번호 도서번호		번호
2	Basket_date	varchar(30)	O						생성일자

테이블 명	Order list	테이블기술서	작성일		2020/06/15		Page1/		
시스템 명	인터넷 서점의 웹사이트 DB 구축		작성자		울산대 박선재				
테이블설명	주문목록 테이블								
No	Attribute	Data type	NN	PK	FK	FK.tab	FK.col	Def.	Description
1	Order_num	number	O	O	O	회원 도서	회원번호 도서번호		주문번호
2	Order_detail	varchar(30)	O						상세정보(수량)
3	Order_date	varchar(30)	O						주문일자
4	Order_tatal	number	O						주문총액
5	Order_Ckind	varchar(30)	O						신용카드종류
6	Order_Cnum	number	O		O	신용카드	카드번호		신용카드번호
7	Order_Cvalidity	varchar(30)	O						카드유효번호
8	Order_Sdetail	varchar(30)	O						배송지상세주소
9	Order_Saddress	varchar(30)	O						배송지기본주소
10	Order_Snum	number	O		O	배송지	우편번호		배송지우편번호

[illegible]

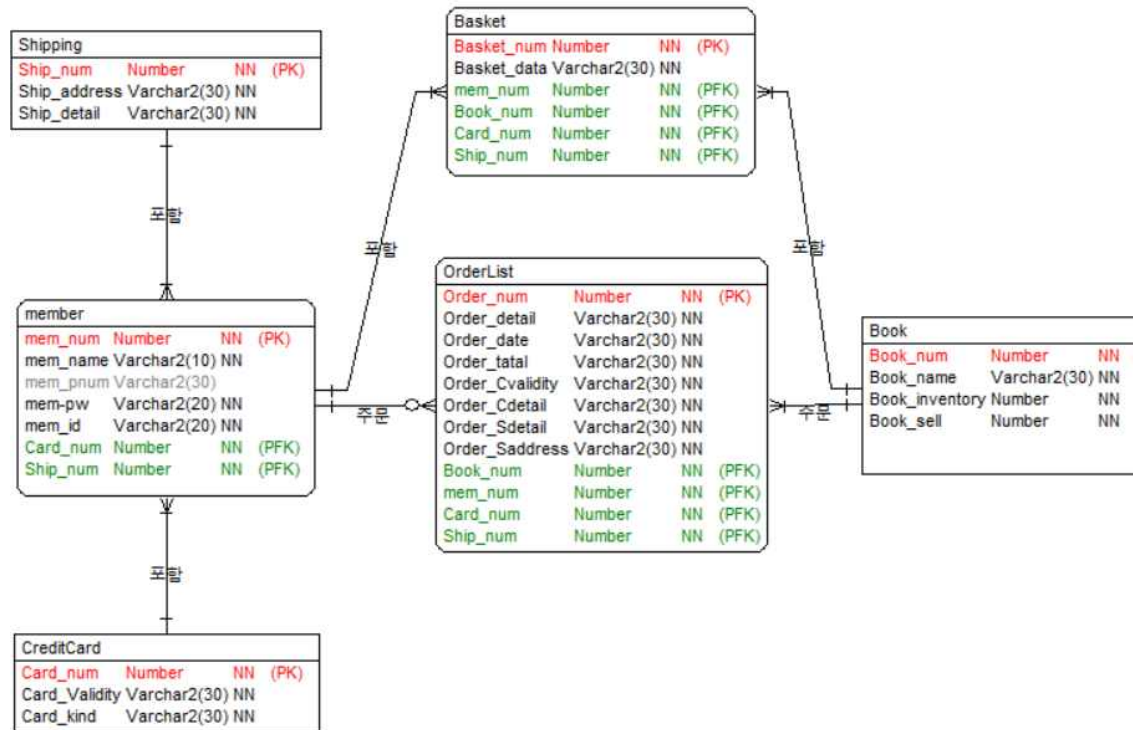
The diagram illustrates the following entities and their attributes:

- 배송지 (Delivery Address):** Attributes include 우편번호 (PK), 기본주소, and 상세주소.
- 장바구니 (Shopping Cart):** Attributes include 장바구니 번호 (PK), 생성일자, 회원번호 (PFK), and 도서번호 (PFK).
- 회원 (Member):** Attributes include 회원번호 (PK), 회원명, 연락처, 비밀번호, 아이디, 카드번호 (PFK), and 우편번호 (PFK).
- 주문목록 (Order List):** Attributes include 주문번호 (PK), 상세정보 (수량), 주문일자, 주문총액, 신용카드종류, 신용카드유효번호, 배송지 상세주소, 배송지 기본주소, 도서번호 (PFK), 회원번호 (PFK), 카드번호 (PFK), and 우편번호 (PFK).
- 도서 (Book):** Attributes include 도서번호 (PK), 도서명, 재고량, and 판매가.

The relationships are as follows:

- 배송지** and **회원** are connected by a **포함 (Containment)** relationship.
- 회원** and **주문목록** are connected by a **주문 (Order)** relationship.
- 장바구니** and **주문목록** are connected by a **주문 (Order)** relationship.
- 장바구니** and **도서** are connected by a **주문 (Order)** relationship.
- 회원** and **도서** are connected by a **주문 (Order)** relationship.

11. 물리적 모델 구축 : IE 표기법



12. 정규화 작업

13. 데이터베이스 구축 sql문

--테이블 삭제--

```
DROP TABLE member CASCADE CONSTRAINTS;  
DROP TABLE CreditCard CASCADE CONSTRAINTS;  
DROP TABLE Shipping CASCADE CONSTRAINTS;  
DROP TABLE basket CASCADE CONSTRAINTS;  
DROP TABLE basket Book CASCADE CONSTRAINTS;  
DROP TABLE OrderList CASCADE CONSTRAINTS;  
DROP TABLE OrderList2 CASCADE CONSTRAINTS;  
DROP TABLE OrderList3 CASCADE CONSTRAINTS;  
DROP TABLE OrderList4 CASCADE CONSTRAINTS;
```

--멤버 테이블--

```
create table member(  
  mnum number(4),  
  mname VARCHAR2(10),  
  mpnum VARCHAR2(20),  
  mpwd VARCHAR2(20),  
  constraint member_mnum_pk primary key(mnum)  
);
```

--insert 문

```
insert into member values('1','박선재','010-6318-9098','1234');  
insert into member values('2','홍길동','010-1111-1111','5678');  
insert into member values('3','이순신','010-2222-3333','1111');  
insert into member values('4','김구','010-2222-3333','1111');  
insert into member values('5','김유신','010-2222-3333','1111');
```

--update 문

```
update member set mname='홍길동' where mnum='2';  
update member set mname='이순신' where mnum='3';
```

--DELETE 문

```
DELETE FROM member where mname = '박선재';  
Delete from member where mnum between '1' and '3';
```

--Select 문--

```
select * from member;
select * from member where mpwd='1111';
select * from member where mnum between '1' and '3';
select mname, mnum from member where mname = '이순신' or mnum
= '010-2222-3333';
select mname,mnum from member where mname LIKE '%신%';
```

--신용카드 테이블--

```
create table CreditCard(
Cnum number(4),
Cvalidity VARCHAR2(20),
Ckind VARCHAR2(20),
constraint CreditCard_Cnum_pk primary key(cnum),
constraint CreditCard_Cnum_fk foreign key(cnum)REFERENCES member(mnum)
);
```

```
insert into CreditCard values('1','01/31','국민');
insert into CreditCard values('2','02/31','국민');
insert into CreditCard values('3','03/31','국민');
insert into CreditCard values('4','04/31','신한');
insert into CreditCard values('5','05/31','신한');
```

--배송지 테이블--

```
create table Shipping(
Ship_num NUMBER(4),
Ship_address VARCHAR2(30),
Ship_detail VARCHAR2(30),
constraint Shipping_ship_num_pk primary key(ship_num),
constraint Shipping_ship_num_fk foreign key(ship_num)REFERENCES member(mnum)
);
insert into shipping values('1', '울산광역시','중구 달동');
insert into shipping values('2', '울산광역시','남구 삼산동');
insert into shipping values('3', '울산광역시','중구 성남동');
insert into shipping values('4', '울산광역시','남구 무거동');
insert into shipping values('5', '울산광역시','중구 태화동');
```

--장바구니 테이블--

```
create table basket(  
Basket_num NUMBER(4),  
Basket_date varchar(30),  
constraint basket_Basket_num_PK primary key(Basket_num),  
constraint basket_Basket_num_fk foreign key(Basket_num)REFERENCES  
member(mnum)  
);  
insert into basket values('1', '2020/01/01');  
insert into basket values('2', '2020/02/01');  
insert into basket values('3', '2020/02/01');  
insert into basket values('4', '2020/04/01');  
insert into basket values('5', '2020/04/01');
```

--장바구니 테이블--

```
create table basketBook(  
Basket_num NUMBER(4),  
Basket_date varchar(30),  
constraint basketBook_Basket_num_PK primary key(Basket_num),  
constraint basketBook_Basket_num_fk foreign key(Basket_num)REFERENCES  
Book(Book_num)  
);  
insert into basketBook values('1', '2020/01/01');  
insert into basketBook values('2', '2020/02/01');  
insert into basketBook values('3', '2020/02/01');  
insert into basketBook values('4', '2020/04/01');  
insert into basketBook values('5', '2020/04/01');
```

--도서 테이블--

```
create table Book(  
Book_num number(4),  
Book_name varchar(30),  
Book_inventory varchar(30),  
Book_sell varchar(30),  
constraint Book_Book_num_pk primary key(Book_num)  
);  
insert into Book values('1','데이터베이스','5개','20000원');
```

```

insert into Book values('2','운영체제','7개','300000원');
insert into Book values('3','네트워크','2개','30000원');
insert into Book values('4','JSP','3개','23000원');
insert into Book values('5','파이썬','4개','25000원');

```

--주문 테이블--

```

create table OrderList(
Order_num number(4),
Order_detail varchar(30),
Order_date varchar(30),
Order_total varchar(30),
Order_Ckind varchar(30),
Order_Cnum number(4),
Order_Cvalidity varchar(30),
Order_Sdetail varchar(30),
Order_Saddress varchar(30),
Order_Snum number(4),
constraint OrderList_Order_num_pk primary key(Order_num),
constraint OrderList_Order_num_fk foreign key(Order_num)REFERENCES
member(mnum)
);

```

```

insert into OrderList values('1','데이터베이스','2020/01/01','20000원','국민카드','389','
국민은행','울산','중구 옥교동','1');
insert into OrderList values('2','네트워크','2020/04/31','30000원','신한카드','239','신한
은행','울산','중구 옥교동','2');
insert into OrderList values('3','운영체제','2020/05/01','30000원','농협카드','239','농협
은행','울산','중구 옥교동','3');
insert into OrderList values('4','자료구조','2020/09/21','30000원','우리카드','239','우리
은행','울산','중구 옥교동','4');
insert into OrderList values('5','파이썬','2020/09/01','30000원','토스카드','239','토스은
행','울산','중구 옥교동','5');

```

--주문 테이블--

```

create table OrderList2(
Order_num number(4),
Order_detail varchar(30),

```

```

Order_date varchar(30),
Order_total varchar(30),
Order_Ckind varchar(30),
Order_Cnum number(4),
Order_Cvalidity varchar(30),
Order_Sdetail varchar(30),
Order_Saddress varchar(30),
Order_Snum number(4),
constraint OrderList2_Order_num_pk primary key(Order_num),
constraint OrderList2_Order_num_fk foreign key(Order_num)REFERENCES
Book(Book_num)
);

```

--주문 테이블--

```

create table OrderList3(
Order_num number(4),
Order_detail varchar(30),
Order_date varchar(30),
Order_total varchar(30),
Order_Ckind varchar(30),
Order_Cnum number(4),
Order_Cvalidity varchar(30),
Order_Sdetail varchar(30),
Order_Saddress varchar(30),
Order_Snum number(4),
constraint OrderList3_Order_num_pk primary key(Order_num),
constraint OrderList3_Order_num_fk foreign key(Order_Cnum)REFERENCES
CreditCard(Cnum)
);

```

--주문 테이블--

```

create table OrderList4(
Order_num number(4),
Order_detail varchar(30),
Order_date varchar(30),
Order_total varchar(30),
Order_Ckind varchar(30),

```

```

Order_Cnum number(4),
Order_Cvalidity varchar(30),
Order_Sdetail varchar(30),
Order_Saddress varchar(30),
Order_Snum number(4),
constraint OrderList4_Order_num_pk primary key(Order_num),
constraint OrderList4_Order_num_fk foreign key(Order_Snum)REFERENCES
Shipping(Ship_num)
);

```

14. 한 학기를 마치며

14.1) 온라인 강의 후기

코로나로 인해 대면강의를 하지 못해 온라인 강의 수업으로 대체 된 점이 생각보다 나쁘지 않았다고 생각합니다. 기초 지식 없이 처음 접하게 되어서 어렵겠다는 편견과 달리 교수님의 강의 스타일은 신선하고도 내용은 학생들이 이해하기 쉽게 설명되어 좋았습니다. 무엇보다도 실습에 있어서는 누구나 따라 갈 수 있게끔 영상으로 제작되었던 점이 좋았고 중간 중간에 교수님께서 학생들에게 포기하지 않도록 북돋아 주신 메시지와 과제의 강도 등을 보았을 때 교수님의 수업에 대한 열정이나 배려하시는 모습 등이 보여 너무 좋았습니다. 온라인 강의의 한가지 단점을 뽑자면 시간이 지남에 따라 점점 수업을 들을 때 와 과제를 함에 있어서 집중도가 떨어지는 것이 제일 컸습니다. 마지막으로, 이번 데이터베이스 수업을 하면서 진로에 대해 한번 생각해 볼 수 있는 계기가 되었습니다. 개인적으로 데이터베이스 수업이 이번 학기 가장 재미있고 제대로 수업을 들은 기분입니다. 감사합니다.

14.2) 프로젝트기반 강의

프로젝트기반 강의를 통해 학습자가 상당한 시간을 투자하여 과제를 해결하고 문제를 발견하고 문제해결의 틀을 스스로 구성하게 되는 구조화된 학습형태로 실질적으로 더 많은 도움이 되었던 것 같습니다.

그리고 탐구해야 할 하나의 의미있는 질문이나 해결 과제를 가지고 학습자들 스스로가 해결 방안을 설계, 실천 과정에서 교육과정에서 비롯된 인지적, 정의적, 심동적인 목표들을 달성해 나갔다고 생각합니다. 한 가지 아쉬운 점은 대면 수업이었다면 교수님 및 동료들과 함께 의견을 나누고 유용한 지식을 공유 할 수 있지 않았을까 하는 아쉬움이 남습니다. 감사합니다.