

# 데이터베이스

## 팀프로젝트 최종 발표

8조 : 정신이 88하조

# CONTENT

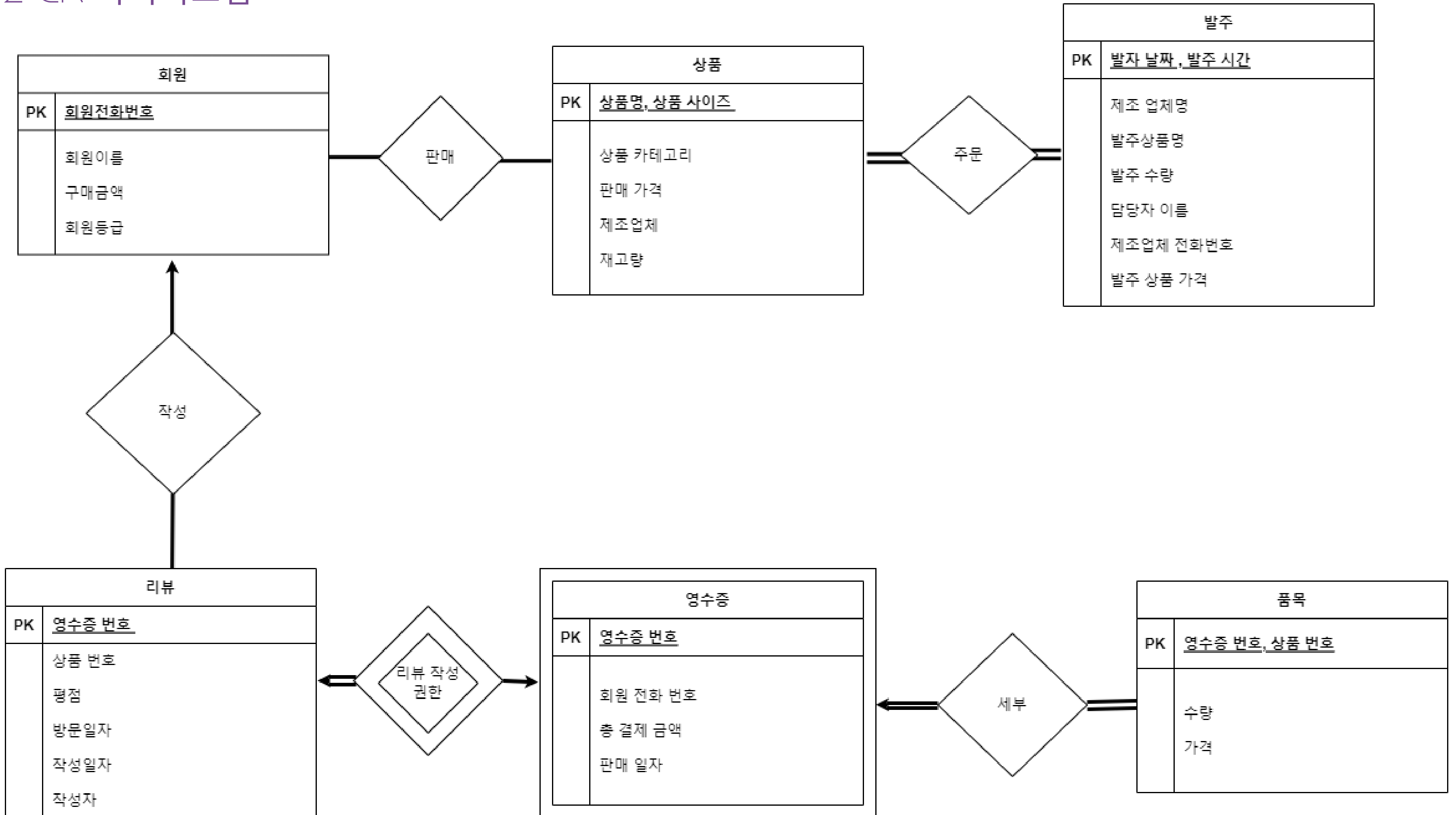
---

발표 순서

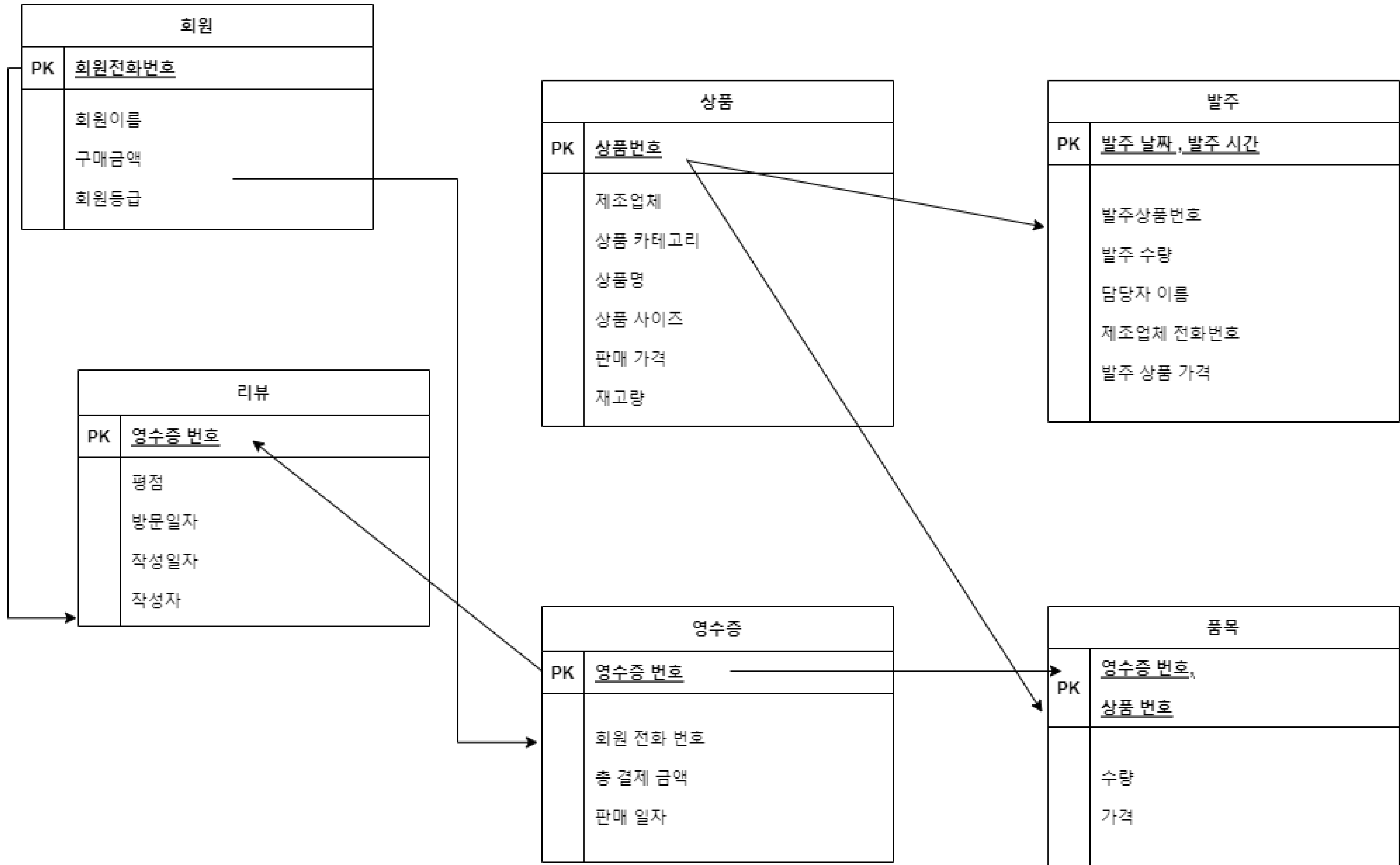
- 01 DB 필요성 설명
- 02 ER 다이어그램
- 03 스키마 다이어그램
- 04 코드 설명
- 05 자바 프로그램  
시연 영상

주류 매장 관리자 입장에서 고객과 주류매장 사이에 이루어지는  
**판매, 구매, 리뷰 정보를 관리**하고,  
제조업체로부터 발주를 넣어 **납품받은 주류의 재고를 관리**할 수 있는  
**주류 매장 관리 시스템**입니다.

## 02 ER 다이어그램



### 03 스키마 다이어그램



초기 데이터베이스를 생성하고 데이터를 삽입하는  
create.sql

## 01 데이터베이스 사용

```
use DB2022Team08
```

## 02 테이블 생성

#회원에 대한 정보를 저장하는 회원 table 생성

```
CREATE TABLE DB2022_회원 (
  전화번호 VARCHAR(30),
  회원이름 VARCHAR(10) NOT NULL,
  구매금액 INT DEFAULT 0,
  회원등급 VARCHAR(10) DEFAULT 'bronze',
  PRIMARY KEY(전화번호)
);
/*
구매금액    회원등급
0~99,999원  : bronze
100,000~499,999 : silver
500,000~999,999 : gold
1,000,000~   : vip
*/
```

#영수증에 대한 정보를 저장하는 영수증 테이블 생성

```
CREATE TABLE DB2022_영수증 (
  영수증번호 VARCHAR(10),
  전화번호 VARCHAR(30),
  총결제금액 INT DEFAULT 0,
  판매일자 DATE,
  PRIMARY KEY(영수증번호),
  FOREIGN KEY(전화번호) REFERENCES DB2022_회원(전화번호)
);
```

초기 데이터베이스를 생성하고 데이터를 삽입하는  
create.sql

#상품에 대한 정보를 저장하는 상품 테이블 생성

```
CREATE TABLE DB2022_상품 (  
    상품번호 VARCHAR(10),  
    상품카테고리 VARCHAR(30),  
    상품명 VARCHAR(30),  
    상품사이즈 VARCHAR(10),  
    판매가격 INT,  
    제조업체 VARCHAR(20),  
    재고량 INT,  
    PRIMARY KEY(상품번호)  
);
```

#발주에 대한 정보를 저장하는 발주 테이블 생성

```
CREATE TABLE DB2022_발주 (  
    발주상품번호 VARCHAR(10),  
    발주상품가격 INT,  
    발주날짜 DATE,  
    발주시간 TIME,  
    발주수량 INT,  
    담당자이름 VARCHAR(20),  
    제조업체전화번호 VARCHAR(20),  
    PRIMARY KEY(발주날짜, 발주시간),  
    FOREIGN KEY(발주상품번호) REFERENCES DB2022_상품(상품번호)  
);
```

초기 데이터베이스를 생성하고 데이터를 삽입하는  
create.sql

#영수증 번호에 따라 구매한  
한 품목의 정보를 저장하는 품목 테이블 생성

```
CREATE TABLE DB2022_품목 (  
    영수증번호 VARCHAR(10),  
    상품번호 VARCHAR(10),  
    수량 INT,  
    가격 INT,  
    PRIMARY key (영수증번호, 상품번호),  
    foreign key (영수증번호) references DB2022_영수증(영수증번호),  
    foreign key (상품번호) references DB2022_상품(상품번호)  
);
```

#하나의 영수증에 대해  
최대 하나의 리뷰 정보를 저장하는 리뷰 테이블 생성

```
CREATE TABLE DB2022_리뷰 (  
    영수증번호 VARCHAR(10),  
    방문일자 date,  
    작성자 VARCHAR(30),  
    평점 int,  
    작성일자 date,  
    PRIMARY key (영수증번호),  
    foreign key (작성자) references DB2022_회원(전화번호)  
);
```



## 뷰 생성 목적

---

01

고객 사용자에게 상품 테이블 중 상품번호, 재고량 등의 정보는 숨기고 상품명, 상품 사이즈, 판매가격만 공개

02

리뷰의 익명성 보장을 위해 평점과 작성일자만 공개

## 뷰 생성 SQL

---

```
CREATE VIEW DB2022_PRICE_VIEW AS  
SELECT 상품명, 상품사이즈, 판매가격 FROM DB2022_상품;
```

```
CREATE VIEW DB2022_REVIEW_VIEW AS  
SELECT 평점, 작성일자 FROM DB2022_리뷰;
```

## 뷰 사용 쿼리

---

### DB2022\_PRICE\_VIEW 사용

▶ DB2022\_Team08\_PriceViewDAO 클래스의 getProductList() 메소드 일부

```
String sql = "SELECT * FROM DB2022_price_view";
    PreparedStatement pstmt = conn.prepareStatement(sql);
    ResultSet r = pstmt.executeQuery();
```

### DB2022\_REVIEW\_VIEW 사용

▶ DB2022\_Team08\_ReviewViewDAO 클래스의 getReviewList() 메소드 일부

```
try {
    getConnection();
    String sql = "SELECT * FROM DB2022_REVIEW_VIEW"; //select 쿼리
    PreparedStatement pstmt = conn.prepareStatement(sql);
    ResultSet r = pstmt.executeQuery();
```

## 인덱스 지정

---

01

회원 삭제, 정보 수정, 검색 시 '전화번호'를 통해 검색하므로

02

리뷰 등록 시 영수증 번호를 통해 회원이 구매한 영수증이 맞는지 확인하므로

03

내가 쓴 리뷰 조회 시 전화번호를 통한 검색이 이루어지므로

04

리뷰 삭제, 정보 수정, 검색 시 '영수증 번호'를 통해 검색하므로

05

발주 삭제, 수정, 검색 시 '발주 날짜'를 통해 검색하므로

## 인덱스 생성 SQL

---

```
CREATE INDEX member_phone ON DB2022_회원 (전화번호);  
CREATE INDEX receipt_num ON DB2022_영수증 (영수증번호);  
CREATE INDEX review_phone ON DB2022_리뷰 (전화번호);  
CREATE INDEX receipt_num_review ON DB2022_리뷰 (영수증번호);  
CREATE INDEX orderdate ON DB2022_발주 (발주날짜);
```

## 인덱스 삭제 SQL (참고)

```
ALTER TABLE DB2022_회원 DROP INDEX member_phone;  
ALTER TABLE DB2022_영수증 DROP INDEX receipt_num;  
ALTER TABLE DB2022_리뷰 DROP INDEX review_phone;  
ALTER TABLE DB2022_리뷰 DROP INDEX receipt_num_review;  
ALTER TABLE DB2022_발주 DROP INDEX orderdate;
```

## 인덱스 사용 쿼리

---

DB2022\_회원 (전화번호) 인덱스인 member\_phone을 사용하는 쿼리

▶ DB2022\_Team08\_MemberDAO 클래스의 getMember(String) 메소드 일부

```
String sql = "SELECT * FROM DB2022_회원 WHERE 전화번호=?"; //select 쿼리
    PreparedStatement pstmt = conn.prepareStatement(sql);
    pstmt.setString(1, phone_num);
    ResultSet r = pstmt.executeQuery();
```

▶ DB2022\_Team08\_MemberDAO 클래스의 deleteMember(String) 메소드 일부

```
String sql = "DELETE FROM DB2022_회원 WHERE 전화번호 = ?"; //delete 쿼리
    PreparedStatement pstmt = conn.prepareStatement(sql);
    pstmt.setString(1, phone);
    int r = pstmt.executeUpdate();
```

## 인덱스 사용 쿼리

---

DB2022\_영수증 (영수증번호) 인덱스인 receipt\_num을 사용하는 쿼리

▶ DB2022\_Team08\_ReceiptDAO 클래스의 getReceipt(String) 메소드 일부

```
String sql = "SELECT 영수증번호, 전화번호, 총결제금액, 판매일자 FROM DB2022_영수증 WHERE 영수증번호 = ?";  
PreparedStatement pstmt = conn.prepareStatement(sql);  
pstmt.setString(1, num);  
ResultSet r = pstmt.executeQuery();
```

▶ DB2022\_Team08\_Receipt 클래스의 DAOdeleteReceipt(String) 메소드 일부

```
String sql = "DELETE FROM DB2022_영수증 WHERE 영수증번호 = ?";  
PreparedStatement pstmt = conn.prepareStatement(sql);  
pstmt.setString(1, num);  
int r = pstmt.executeUpdate();
```



## 인덱스 사용 쿼리

---

DB2022\_리뷰 (전화번호) 인덱스인 review\_phone을 사용하는 쿼리

▶ DB2022\_Team08\_ReviewDAO 클래스의 getMyReviewList(String) 메소드 일부

```
String sql = "SELECT * FROM DB2022_리뷰 WHERE 작성자 = ?"; //select 쿼리
    PreparedStatement pstmt = conn.prepareStatement(sql);
    pstmt.setString(1,phone);
    ResultSet r = pstmt.executeQuery();
```

## 인덱스 사용 쿼리

---

DB2022\_리뷰 (영수증번호) 인덱스인 receipt\_num\_review을 사용하는 쿼리

▶ DB2022\_Team08\_ReviewDAO 클래스의 getReview(String) 메소드 일부

```
String sql = "SELECT * FROM DB2022_리뷰 WHERE 영수증번호=?"; //select 쿼리
    PreparedStatement pstmt = conn.prepareStatement(sql);
    pstmt.setString(1, receipt_num);
    ResultSet r = pstmt.executeQuery();
```

▶ DB2022\_Team08\_ReviewDAO 클래스의 deleteReview(String) 메소드 일부

```
String sql = "DELETE FROM DB2022_리뷰 WHERE 영수증번호 = ?"; //delete 쿼리
    PreparedStatement pstmt = conn.prepareStatement(sql);
    pstmt.setString(1, receipt);
    int r = pstmt.executeUpdate();
```

## 인덱스 사용 쿼리

---

DB2022\_발주 (발주날짜) 인덱스인 orderdate을 사용하는 쿼리

▶ DB2022\_Team08\_OrderDAO 클래스의 getOrder(Date, Time) 메소드 일부

```
String sql = "SELECT * FROM DB2022_발주 WHERE 발주날짜 = ? and 발주시간=?";
    PreparedStatement pstmt = conn.prepareStatement(sql);
    pstmt.setDate(1, date);
    pstmt.setTime(2, time);
    ResultSet r = pstmt.executeQuery();
```

▶ DB2022\_Team08\_OrderDAO 클래스의 deleteOrder(Date, Time) 메소드 일부

```
String sql = "DELETE FROM DB2022_발주 WHERE 발주날짜 = ? and 발주시간=?";
    PreparedStatement pstmt = conn.prepareStatement(sql);
    pstmt.setDate(1, date);
    pstmt.setTime(2, time);
    int r = pstmt.executeUpdate();
```

초기 데이터베이스를 생성하고 데이터를 삽입하는  
create.sql

### 03 데이터 삽입

#회원 table에 tuples 삽입

```
INSERT INTO DB2022_회원 VALUES
  ('010-4024-2434', '춘식이', 64000, 'bronze'),
  ('010-1234-1254', '라이언', 38500, 'bronze'),
  ('010-1245-2356', '어피치', 17500, 'bronze'),
  ('010-1233-4566', '프로도', 295000, 'silver'),
  ('010-4566-4566', '무지', 295000, 'silver'),
  ('010-7899-4566', '콘', 295000, 'silver'),
  ('010-7410-4566', '네오', 695000, 'gold'),
  ('010-1245-4578', '튜브', 1009000, 'vip');
```

#영수증 table에 tuples 삽입

#회원 table의 '전화번호'를 참조하므로 회원 table보다 나중에 삽입

```
INSERT INTO DB2022_영수증 VALUES
  ('001', '010-4024-2434', 37700, '2021-05-25'),
  ('002', '010-1245-2356', 7500, '2022-05-18'),
  ('003', '010-4566-4566', 2700, '2022-03-02'),
  ('004', '010-7899-4566', 15000, '2022-05-15'),
  ('005', '010-7410-4566', 16500, '2022-05-08'),
  ('006', '010-1245-4578', 509000, '2022-04-17');
```

초기 데이터베이스를 생성하고 데이터를 삽입하는  
create.sql

#상품 table에 tuples 삽입

INSERT INTO DB2022\_상품 VALUES

```
("001", "맥주", "테라캔", "500ml" , 2350, "하이트진로", 300),  
("002", "맥주", "테라병", "1600ml", 6100, "하이트진로", 50),  
("003", "맥주", "칭다오", "500ml" , 2500, "TSINGDAO BREWERY", 200),  
("004", "소주", "처음처럼병", "360ml", 1800, "롯데칠성", 170),  
("005", "소주", "처음처럼페트", "640ml", 3000, "롯데칠성", 140),  
("006", "소주", "참이슬", "360ml", 1650, "하이트진로", 70),  
("007", "막걸리", "국순당생막걸리", "750ml", 1800, "국순당", 5),  
("008", "양주", "조니워커블루라벨", "750ml", 250000, "조니워커", 15);
```

초기 데이터베이스를 생성하고 데이터를 삽입하는  
create.sql

```
#발주 table에 tuples 삽입
#회원 table의 상품명 attribute를 참조하고 있으니 회원 table보다 나중에 INSERT
INSERT INTO DB2022_발주 VALUES
  ("001", 1900, "2022-05-03", "15:13:33", 90, "김책임", "02-4321-9876"),
  ("002", 5100, "2022-05-03", "15:13:35", 80, "김책임", "02-4321-9876"),
  ("003", 2000, "2022-05-09", "19:18:49", 70, "이담당", "031-456-789"),
  ("004", 1200, "2022-05-13", "09:05:01", 150, "박사원", "02-4141-4141"),
  ("005", 2400, "2022-05-13", "09:06:01", 50, "박사원", "02-4141-4141"),
  ("006", 1000, "2022-05-16", "13:45:25", 120, "최대리", "032-1478-5236"),
  ("007", 1100, "2022-05-17", "18:00:00", 30, "송주임", "051-7410-7410"),
  ("008", 150000, "2022-05-19", "11:50:40", 10, "Heather", "02-0519-0519");
```

초기 데이터베이스를 생성하고 데이터를 삽입하는  
create.sql

```
#품목 table에 tuples 삽입
#상품번호, 영수증번호 attribute를 참조하고 있으니까
| 상품, 영수증 table보다 나중에 INSERT
INSERT INTO DB2022_품목 VALUES
    ('001', '002', 5, 30500),
    ('001', '004', 4, 7200),
    ('002', '003', 3, 7500),
    ('003', '001', 2, 2700),
    ('004', '005', 5, 15000),
    ('005', '006', 10, 16500),
    ('006', '007', 5, 9000),
    ('006', '008', 2, 50000);
```

```
#리뷰 table에 tuples 삽입
#영수증번호 attribute를 참조하고 있으니까
| 영수증 table보다 나중에 INSERT
INSERT INTO DB2022_리뷰 VALUES
    ('001', '2021-05-25', '010-4024-2434', 5, '2022-01-02'),
    ('002', '2022-05-18', '010-4024-2434', 5, '2022-02-02'),
    ('003', '2022-03-02', '010-1234-1254', 3, '2022-03-02'),
    ('004', '2022-05-15', '010-7899-4566', 5, '2022-04-02'),
    ('005', '2022-04-17', '010-7410-4566', 5, '2022-05-02');
```

모든 테이블과 데이터를 데이터베이스로부터 삭제하는  
dropdb.sql

```
use DB2022Team08

drop view DB2022_REVIEW_VIEW;
drop table DB2022_리뷰;
drop view DB2022_PRICE_VIEW;
drop table DB2022_품목;
drop table DB2022_팔주;
drop table DB2022_상품;
drop table DB2022_영수증;
drop table DB2022_회원;
```



# THANK YOU

이어서 자바 프로그램 시연이 진행될 예정입니다!

8조 : 정신이 88하조