

# 논리 데이터베이스 설계

# 관계의 종류 및 문제해결



## 학습내용

- 관계의 종류 및 식별
- 다대다 관계 식별 및 문제해결

## 학습목표

- 관계의 종류 및 식별 방법에 대해 설명할 수 있다.
- 다대다 관계를 식별하고 문제를 해결할 수 있다.

# 관계의 종류 및 식별

## 1 관계의 종류

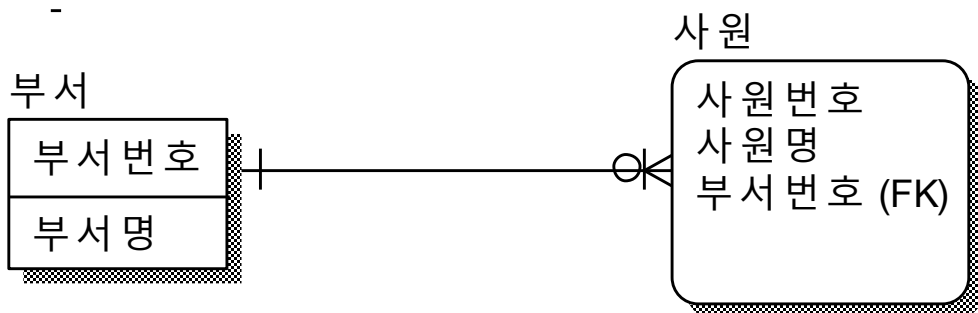
- 1 정상 관계(Normal Relationship)
- 2 중복 관계(Redundant Relationship)
- 3 직렬 관계(Serial Relationship)
- 4 병렬 관계(Parallel Relationship)
- 5 자기참조 관계(Recursive Relationship)
- 6 BOM(Bill of Materials) 관계
- 7 Arc(Mutually Exclusive) 관계
- 8 수퍼 타입 서브 타입 관계
- 9 주 식별자 관계/비 식별자 관계

# 관계의 종류 및 식별

## 1 관계의 종류

### 1 정상 관계(Normal Relationship)

- 엔티티와 엔티티가 독립적으로 분리되어 있으면서 **한 개의 관계만 상호 간 존재**하는 형태의 관계



### 2 중복 관계(Redundant Relationship)

- 엔티티와 엔티티 간에 **두 번 이상 발생**하는 종속 관계
- 꼭 필요한 경우에만 사용(모델링이 복잡하면 최적화가 어려움)

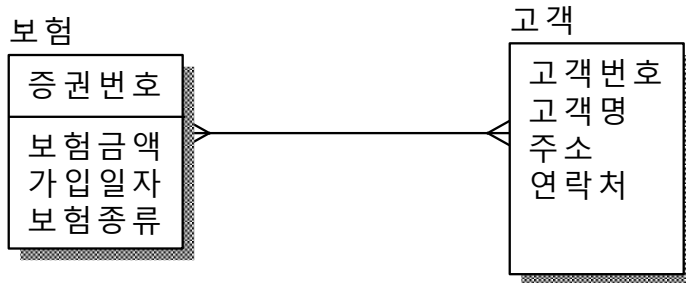


# 관계의 종류 및 식별

## 1 관계의 종류

### 3 직렬 관계(Serial Relationship)

- 두 엔티티 사이에 존재하는 몇 개의 관계를 모아 상위 개념으로 통합하여 **하나의 관계로 관리**하는 방법
- M : N 관계를 해소시키기 위해 새로운 엔티티 추가



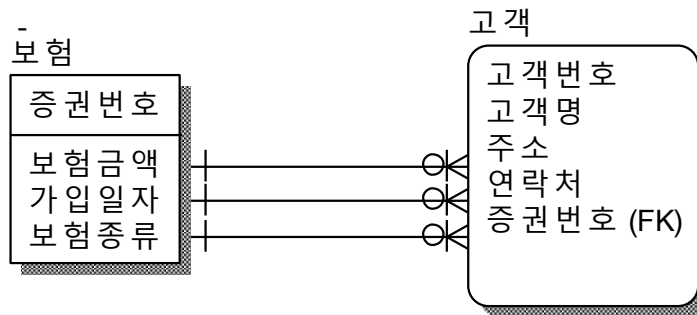
계약자, 피보험자, 수익자 등을  
통합 관리하기 위해 M : N의 관계로 구성

# 관계의 종류 및 식별

## 1 관계의 종류

### 4 병렬 관계(Parallel Relationship)

- 두 엔티티 사이에 존재하는 관계들을 별도의 관계로 간주함으로써 **여러 개의 관계 선분이 나란히 병렬로** 관계 형성



계약자, 피보험자, 수익자 등을 **별도**  
로 관리하기 위해 1 : M의 관계로 구성

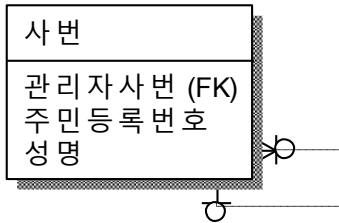
# 관계의 종류 및 식별

## 1 관계의 종류

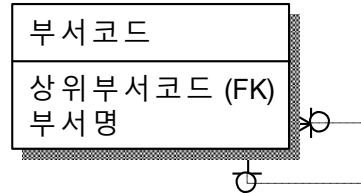
### 5 자기참조 관계(Recursive Relationship)

- 하나의 엔티티가 다른 엔티티가 아닌 **자기 자신과 관계를 맺는 관계**

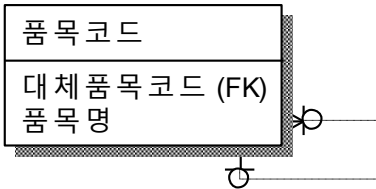
사원



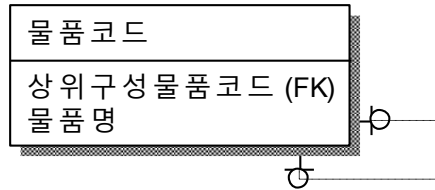
부서



품목



물품구성

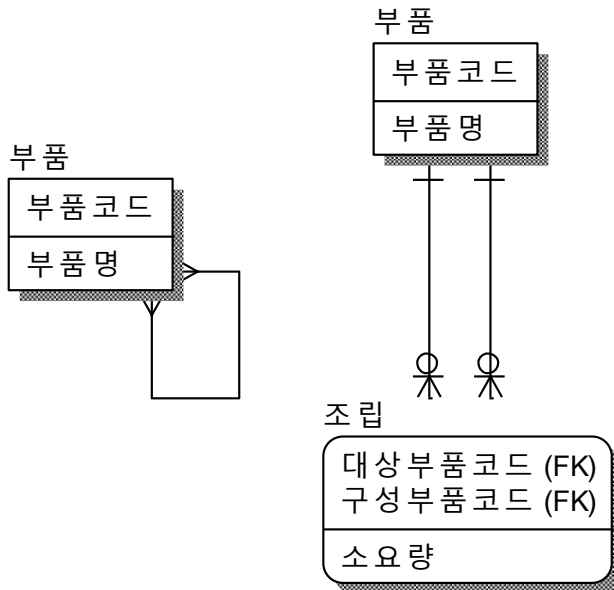


# 관계의 종류 및 식별

## 1 관계의 종류

### 6 BOM(Bill of Materials) 관계

- 제조업에서 부품의 소요량 파악 등의 모델에 유용하게 이용
- **M : M 순환 관계**라고 함



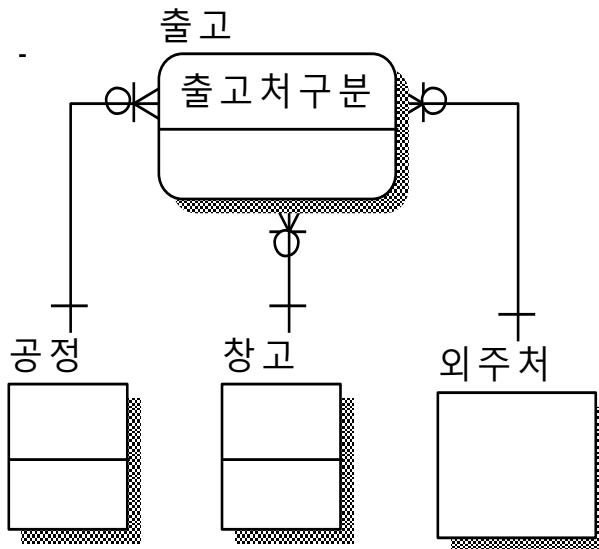


# 관계의 종류 및 식별

## 1 관계의 종류

### 7 Arc(Mutually Exclusive) 관계

- 어떤 엔티티가 **두 개 이상의 다른 엔티티의 합집합과 관계를 가지는 것**
- 배타적(Exclusive) 관계 혹은 아크(Arc) 관계**

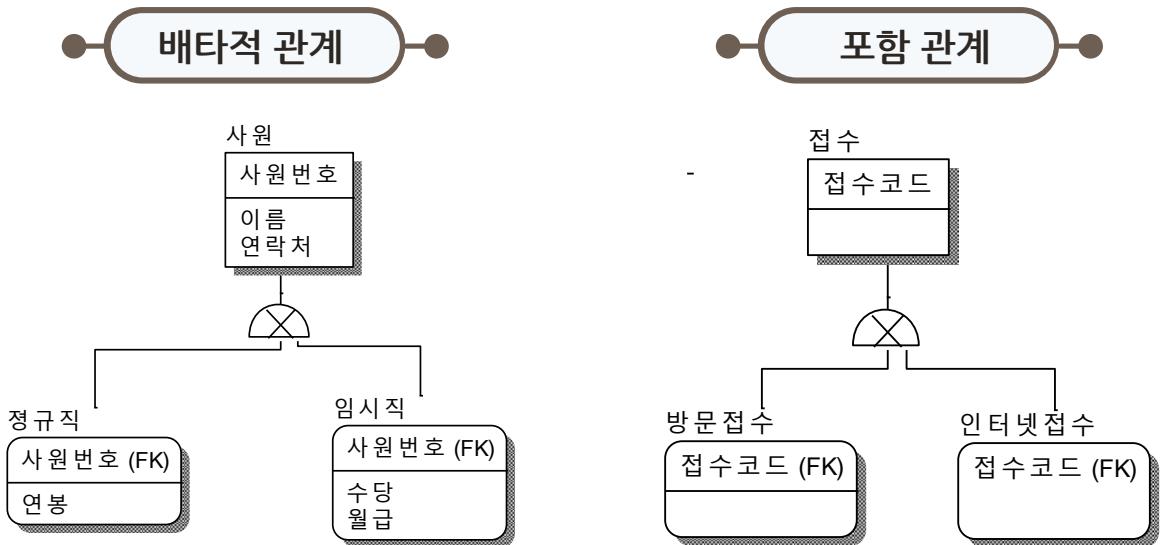


# 관계의 종류 및 식별

## 1 관계의 종류

### 8 수퍼 타입 서브 타입 관계

- 공통 속성을 가지는 슈퍼 타입과 공통 부분을 제외하고, 두 개 이상의 엔티티 간의 속성에 차이가 있을 때 **별도의 서브 타입**이 존재



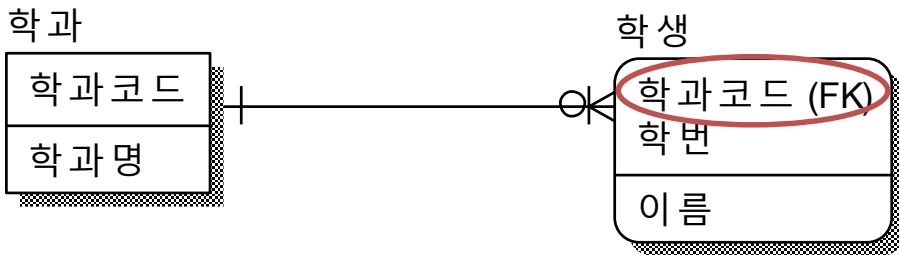
# 관계의 종류 및 식별

## 1 관계의 종류

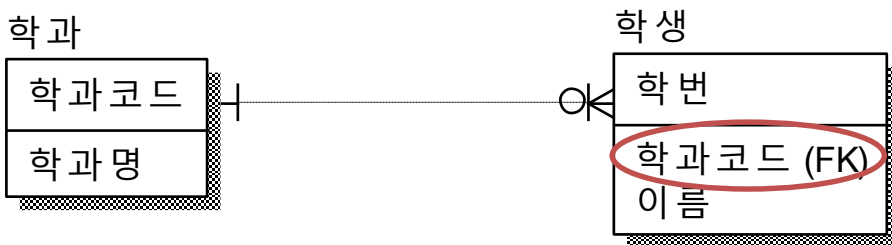
### 9 주 식별자 관계/비 식별자 관계

- 공통 속성을 가지는 슈퍼 타입과 공통 부분을 제외하고, 두 개 이상의 엔티티 간의 속성에 차이가 있을 때 **별도의 서브 타입**이 존재

#### 주 식별자 관계



#### 비 식별자 관계

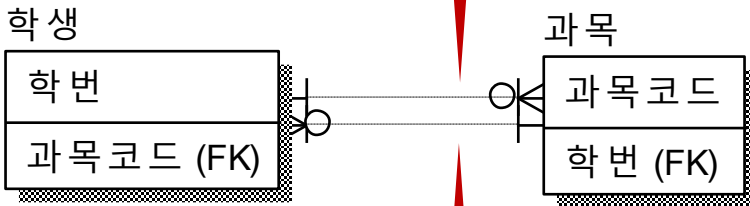


# 다대다 관계 식별 및 문제해결

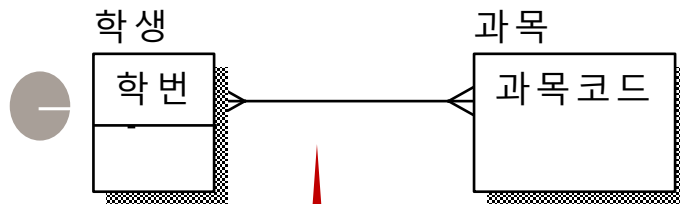
## 1 다대다 관계 식별

- 관계를 도출하는 과정에서 상호 간의 관계가 1 : M, M : 1로 되는 경우를 다대다(M : N) 관계로 표현
- 다대다 관계는 반드시 1:M의 관계로 변환시켜줘야 함

(1 : M) 한 명의 학생은  
여러 개의 강의를 수강할 수 있음



(M:1) 하나의 강의는  
여러 명의 수강생이 있을 수 있음



(M : N) 한 명의 학생은 여러 개의 강  
의를 수강할 수 있고, 하나의 강의는 여  
러 명의 수강생이 있을 수 있음

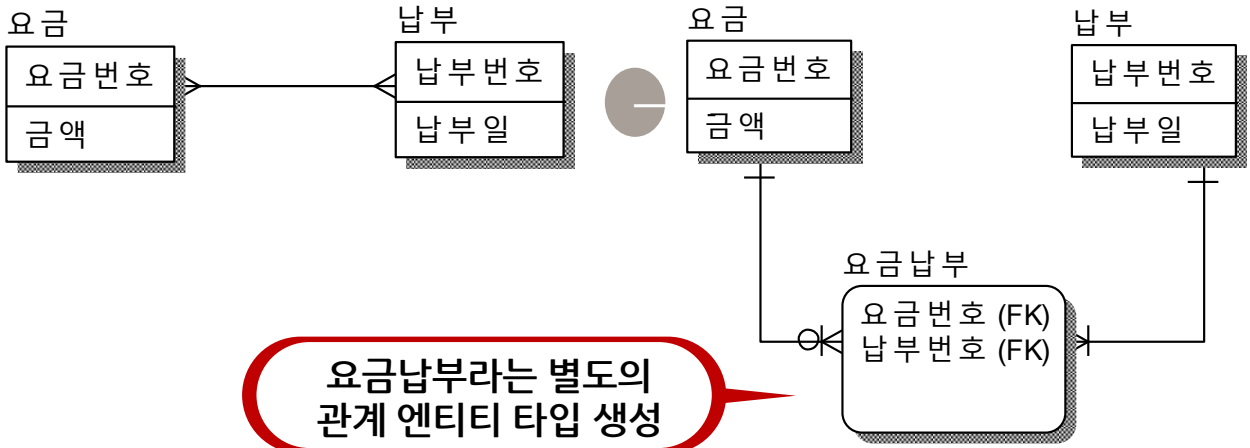
# 다대다 관계 식별 및 문제해결

## 2 다대다 관계 문제해결

### 1 관계 엔티티 타입 생성을 통한 문제해결

예시

한 번 납부할 때 여러 개의 요금 고지서를  
한꺼번에 납부할 수 있고, 또 다른 요금 고지서는  
여러 번에 걸쳐 납부 가능



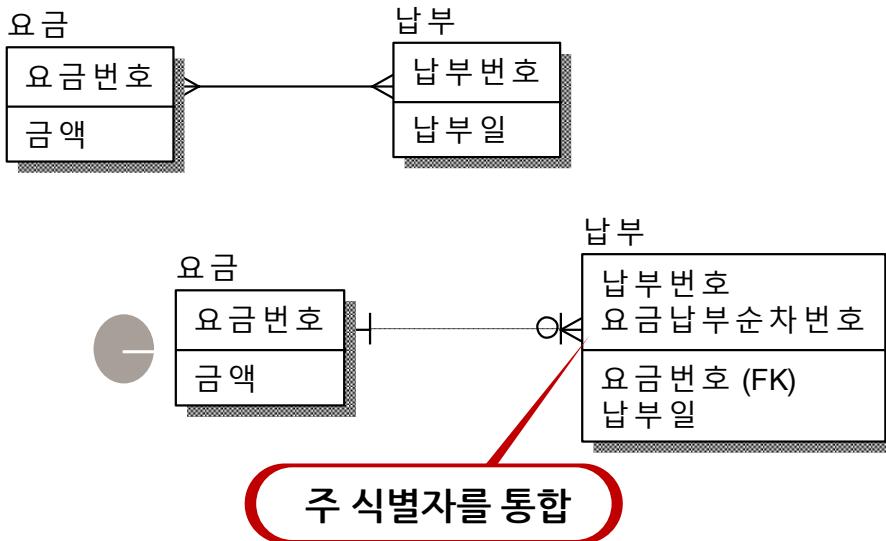
# 다대다 관계 식별 및 문제해결

## 2 다대다 관계 문제해결

### 2 식별자에 의한 통합으로 문제해결

예시

한 번 납부할 때 여러 개의 요금 고지서를  
한꺼번에 납부할 수 있고, 또 다른 요금 고지서는  
여러 번에 걸쳐 납부 가능



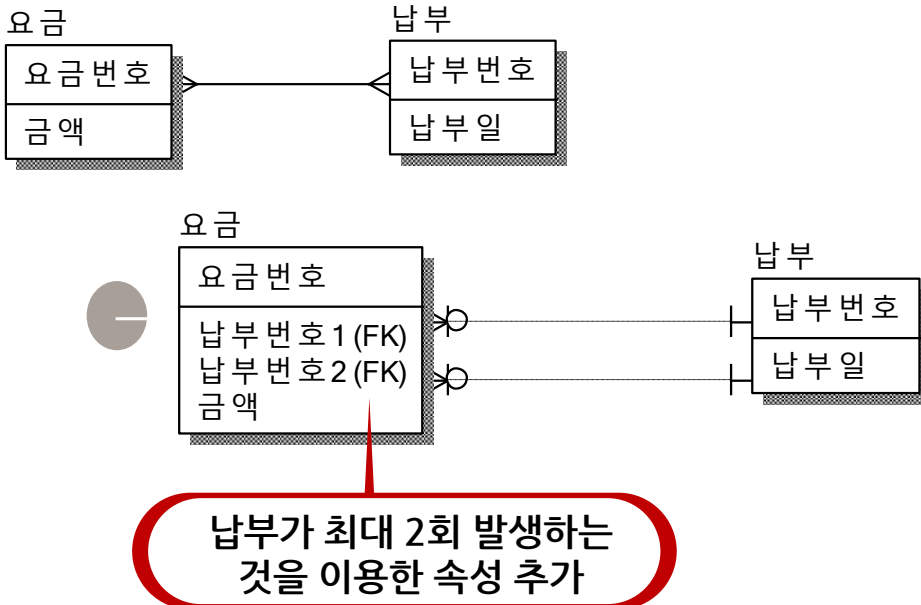
# 다대다 관계 식별 및 문제해결

## 2 다대다 관계 문제해결

### 3 속성에 의한 통합으로 문제해결

예시

한 번 납부할 때 여러 개의 요금 고지서를  
한꺼번에 납부할 수 있고, 또 다른 요금 고지서는  
최대 2번에 걸쳐 납부 가능



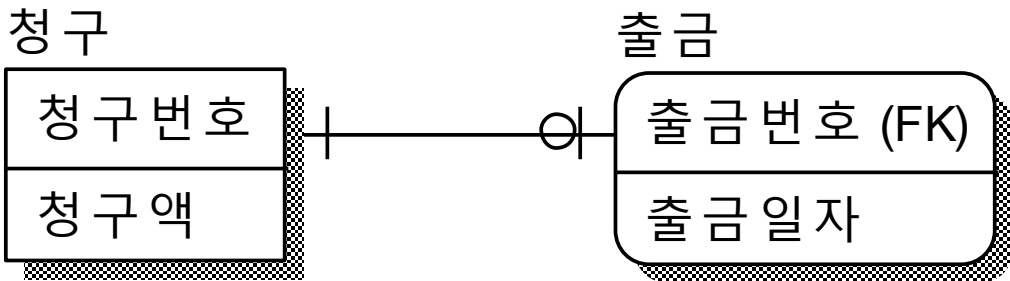
# 다대다 관계 식별 및 문제해결

## 3 일대일 관계 문제해결

### 1 별개의 엔티티 타입으로 따로 표현하는 방법

예시

한 번 청구에 대해 한 번의 출금이 발생되고,  
한 번의 출금은 반드시 하나의 청구서에  
의해서만 발생





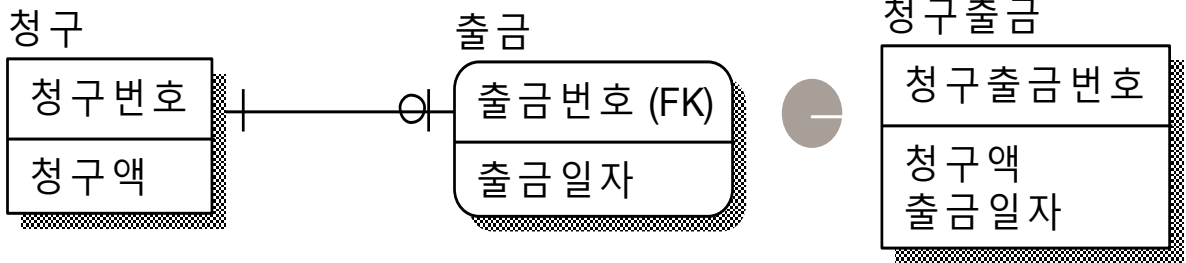
# 다대다 관계 식별 및 문제해결

## 3 일대일 관계 문제해결

### 2 하나의 엔티티 타입 통합(완전 통합)

예시

한 번 청구에 대해 한 번의 출금이 발생되고,  
한 번의 출금은 반드시 하나의 청구서에  
의해서만 발생



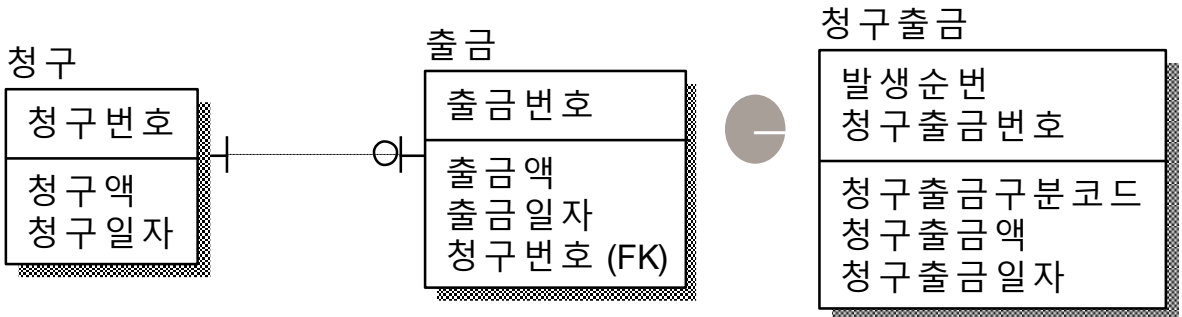
# 다대다 관계 식별 및 문제해결

## 3 일대일 관계 문제해결

### 3 하나의 엔티티 타입 통합(부분 통합)

예시

- 청구 번호와 출금 번호 각각이 고객의 업무에서 유용하게 사용되는 속성
- 청구 번호에 따라 일자, 금액, 장소 등이 있음
- 출금 번호에 따라 일자, 장소, 금액 등이 서로 다른 의미를 가질 경우
- 청구와 출금을 별도의 엔티티 타입으로 유지하거나 하나의 엔티티 타입으로 통합할 때에는 기본 키(PK)를 변경하여 통합



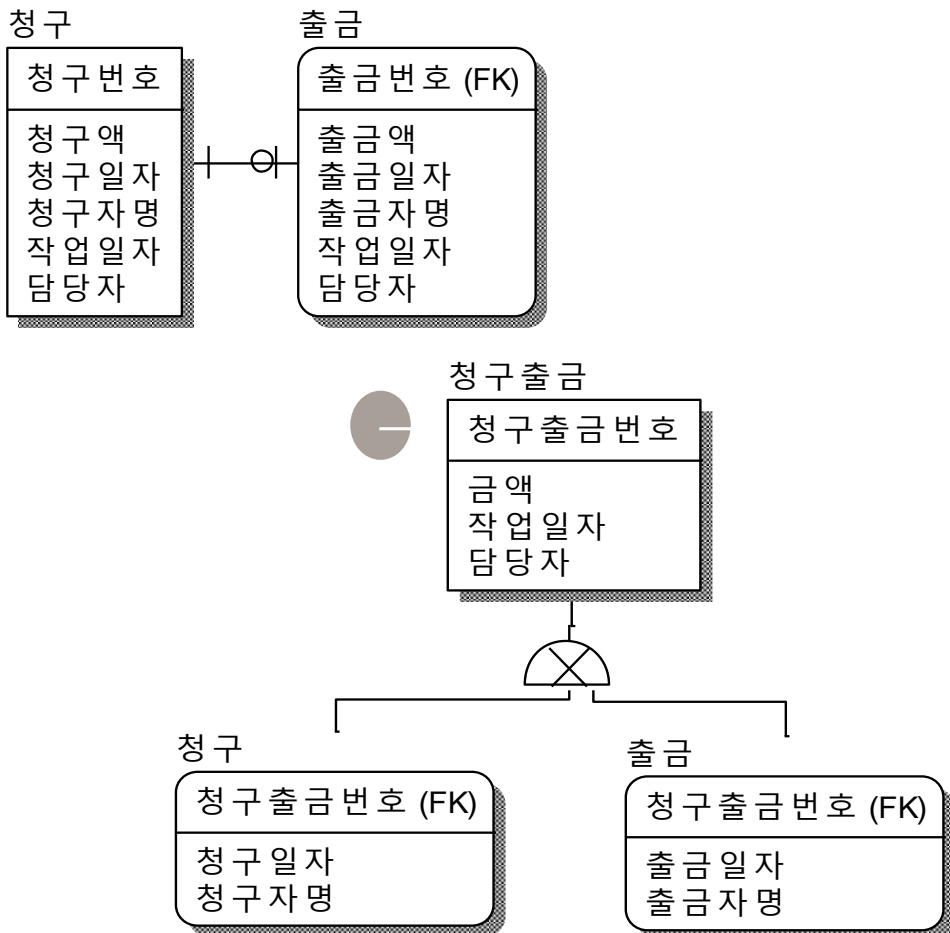
# 다대다 관계 식별 및 문제해결

## 3 일대일 관계 문제해결

### 4 슈퍼 엔티티 타입 생성

예시

- 청구와 출금의 기본 키(PK)와 그 의미가 동일하고 속성의 일부만 다를 경우
- 엔티티 타입을 슈퍼 타입으로 통합



## 학습정리

## 1. 관계의 종류 및 식별

- 정상 관계(Normal Relationship)
  - 엔티티와 엔티티가 독립적으로 분리되어 있으면서 한 개의 관계만 상호 간 존재하는 형태의 관계
- 중복 관계(Redundant Relationship)
  - 엔티티와 엔티티 간에 두 번 이상 발생하는 종속 관계
- 직렬 관계(Serial Relationship)
  - 두 엔티티 사이에 존재하는 몇 개의 관계를 모아 상위 개념으로 통합하여 하나의 관계로 관리하는 방법
- 병렬 관계(Parallel Relationship)
  - 두 엔티티 사이에 존재하는 관계들을 별도의 관계로 간주함으로써 여러 개의 관계 선분이 나란히 병렬로 관계 형성
- 자기참조 관계(Recursive Relationship)
  - 하나의 엔티티가 다른 엔티티가 아닌 자기 자신과 관계를 맺는 관계
- BOM(Bill of Materials) 관계
  - 제조업에서 부품의 소요량 파악 등의 모델에 유용하게 이용되며 M : M 순환 관계라고 함
- Arc(Mutually Exclusive) 관계
  - 어떤 엔티티가 두 개 이상의 다른 엔티티의 합집합과 관계를 가지는 것(배타적(Exclusive) 관계 혹은 아크(Arc) 관계)

## 학습정리

### 1. 관계의 종류 및 식별

- 수퍼 타입 서브 타입 관계
  - 공통 속성을 가지는 슈퍼 타입과 공통 부분을 제외하고, 두 개 이상의 엔티티 간의 속성에 차이가 있을 때 별도의 서브 타입이 존재
- 주 식별자/비 식별자 관계
  - 주 식별자 관계 : 부모의 주 식별자가 자식의 주 식별자로 상속되는 경우
  - 비 식별자 관계 : 부모의 주 식별자가 자식의 일반 속성으로 상속되는 경우

## 학습정리

## 2. 다대다 관계 식별 및 문제해결

- 다대다 관계 문제해결
  - 관계 엔티티 타입 생성, 식별자에 의한 통합, 속성에 의한 통합
- 일대일 관계 문제해결
  - 별개의 엔티티 타입으로 따로 표현, 하나의 엔티티 타입 통합(완전 통합), 하나의 엔티티 타입 통합(부분 통합), 슈퍼 엔티티 타입 생성