

## 7. 데이터 종속성과 정규화

# ❖ 데이터의 논리적 표현

## ◆ 관계 스키마(**relational scheme**)의 설계

- 관계 모델을 이용하여 어떻게 실세계를 정확히 표현할 것인가?
  - i. 애트리뷰트, 엔티티, 관계성을 파악
  - ii. 관련된 애트리뷰트들을 릴레이션으로 묶음
    - 데이터 종속성 : 애트리뷰트들간의 관계성
    - 효율적인 데이터 조작
    - 데이터의 중복성
  - iii. 변칙적 성질의 예방
    - 이상(anomaly)

## ▶ 이상 (anomaly)

◆ example : 수강 릴레이션

수강

학번	과목번호	성적	학년
100	C413	A	4
100	E412	A	4
200	C123	B	3
300	C312	A	1
300	C324	C	1
300	C413	A	1
400	C312	A	4
400	C324	A	4
400	C413	B	4
400	E412	C	4
500	C312	B	2

기본키 : 학번, 과목 번호

## ▶ 이상(2)

### ◆ 삭제이상(deletion anomaly)

- 200번 학생이 'C123'의 등록을 취소  
⇒ 3학년이라는 정보도 함께 삭제됨
- 연쇄 삭제(triggered deletion)에 의한 정보의 손실(loss of information)

### ◆ 삽입이상(insertion anomaly)

- 600번 학생이 2학년이라는 사실을 삽입  
⇒ 어떤 과목을 등록하지 않는 한 삽입이 불가능  
(∵ 과목 번호가 기본 키)
- 원하지 않는 정보의 강제 삽입

### ◆ 갱신이상(update anomaly)

- 400번 학생의 학년을 4에서 3으로 변경  
⇒ 학번이 400인 4개의 튜플 모두를 갱신시켜야 함
- 중복데이터의 일부 갱신으로 정보의 모순성(inconsistency) 발생

## ▶ 이상의 원인과 해결책

### ◆ 이상의 원인

- 애틀리뷰트들 간에 존재하는 여러 종속관계를 하나의 릴레이션에 표현

### ◆ 이상의 해결

- 애틀리뷰트들 간의 종속관계를 분석하여 여러개의 릴레이션으로 분해(decomposition)  
⇒ 정규화(normalization)

# ▶ 스키마 설계와 변환

## ◆ 스키마 설계 : 데이터베이스의 논리적 설계

- ① 애트리뷰트들과 이들의 제약 조건 (종속성)들을 수집
  - ② 수집된 결과를 명시된 제약 조건에 따라 여러 개의 릴레이션으로 분할
- ⇒ 스키마 변환 (schema transformation)

## ◆ 스키마 변환의 원리

- ① 정보의 무손실
- ② 데이터의 중복성 감소
- ③ 분리의 원칙

## ❖ 함수 종속(FD)

### ◆ 정의

- 어떤 릴레이션  $R$ 에서, 애트리뷰트  $X$ 의 값 각각에 대해 애트리뷰트  $Y$ 의 값이 하나만 연관

$\Leftrightarrow$  애트리뷰트  $Y$ 는 애트리뷰트  $X$ 에 함수 종속  $X \rightarrow Y$

- 애트리뷰트  $X$ 는  $Y$ 를 (함수적으로) 결정
  - ◆  $X$ 를 결정자(determinant)
  - ◆  $Y$ 를 종속자(dependent)
- $X, Y$ 는 복합 애트리뷰트일 수 있음

## ❖ 함수 종속(2)

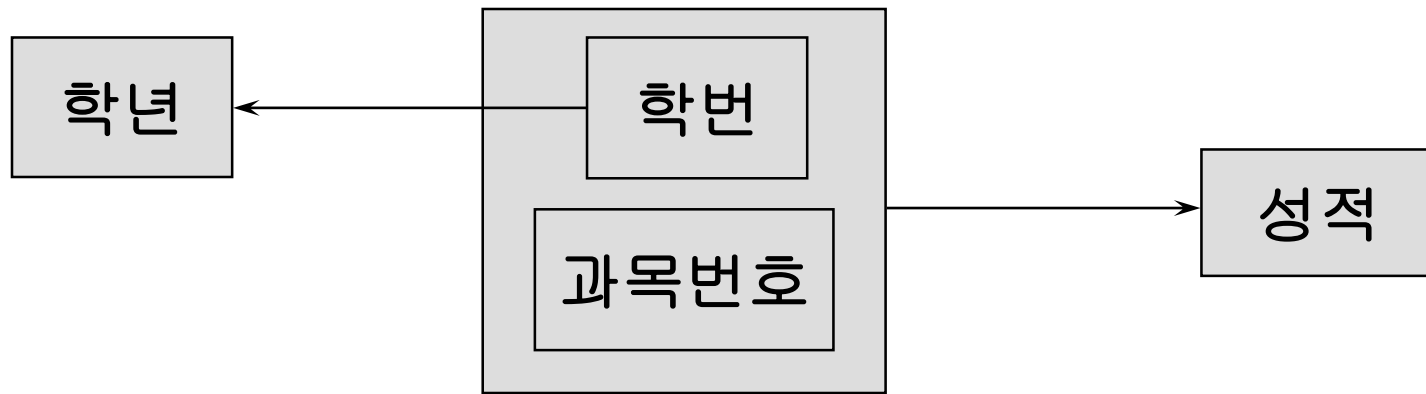
### 👉 Note

- 릴레이션  $R$ 에서 애트리뷰트  $X$ 가 키이면,  
 $R$ 의 모든 애트리뷰트  $Y$ 에 대해  $X \rightarrow Y$  성립
- 함수종속  $X \rightarrow Y$ 의 경우  
애트리뷰트  $X$ 가 반드시 키(유일 값)라는 것을 요건으로  
하지 않음  
즉, 하나 이상의 튜플 값으로 존재가능



## ※ 함수 종속 다이어그램

- ◆ 수강 릴레이션 ( 기본키: 학번, 과목번호 )



{학번, 과목번호} → 성적  
학번 → 학년

## ▶ 완전 함수 종속과 부분 함수 종속

- ◆ 복합 애트리뷰트  $X$ 에 대하여  $X \rightarrow Y$ 가 성립할 때
- ◆ 완전 함수 종속 (**full functional dependency**)
  - $X' \subset X$  이고  $X' \rightarrow Y$  를 만족하는 애트리뷰트  $X'$ 이 존재하지 않음
- ◆ 부분 함수 종속 (**partial functional dependency**)
  - $X' \subset X$  이고  $X' \rightarrow Y$  를 만족하는 애트리뷰트  $X'$ 이 존재함

## ▶ 완전 함수 종속과 부분 함수 종속(2)

### ◆ 예: 수강 릴레이션의 함수 종속

학번  $\rightarrow$  학년

{학번, 과목번호}  $\rightarrow$  성적

(학년)은 (학번)에 완전 함수 종속,

그러나 {학번, 과목번호}에는 부분 함수 종속,

(성적)은 {학번, 과목번호}에 완전 함수 종속

## ▶ 완전 함수 종속과 부분 함수 종속(3)

### ◆ 함수 종속에 대한 추론 규칙

R1: (반사, reflexive)  $A \supseteq B$ 이면  $A \rightarrow B$ 이다. 또한  $A \rightarrow A$ 이다

R2: (첨가, augmentation)  $A \rightarrow B$ 이면  $AC \rightarrow BC$ 이고  $AC \rightarrow B$ 이다.

R3: (이행, transitive)  $A \rightarrow B$ 이고  $B \rightarrow C$ 이면  $A \rightarrow C$ 이다.

R4: (분해, decomposition)  $A \rightarrow BC$ 이면  $A \rightarrow B$ 이다.

R5: (결합, union)  $A \rightarrow B$ 이고  $A \rightarrow C$ 이면  $A \rightarrow BC$ 이다.

### 👉 Note

- 함수 종속은 데이터의 의미(data semantics)를 표현
  - ◆ 예: “학번  $\rightarrow$  학년”의 의미는 “학생은 하나의 학년에만 속한다”
  - ◆ 의미적 제약 조건

# ❖ 기본 정규형

## ◆ 정규형(Normal Form)

- 어떤 일련의 제약 조건을 만족하는 릴레이션

## ◆ 정규화(Normalization)의 원칙

정규화 = 스키마 변환 ( $S \rightarrow S'$ )

### ① 무손실 표현

- ◆ 같은 의미의 정보 유지
- ◆ 그러나 더 바람직한 구조

### ② 데이터의 중복성 감소

### ③ 분리의 원칙

- ◆ 독립적인 관계는 별개의 릴레이션으로 표현
- ◆ 릴레이션 각각에 대해 독립적 처리가 가능

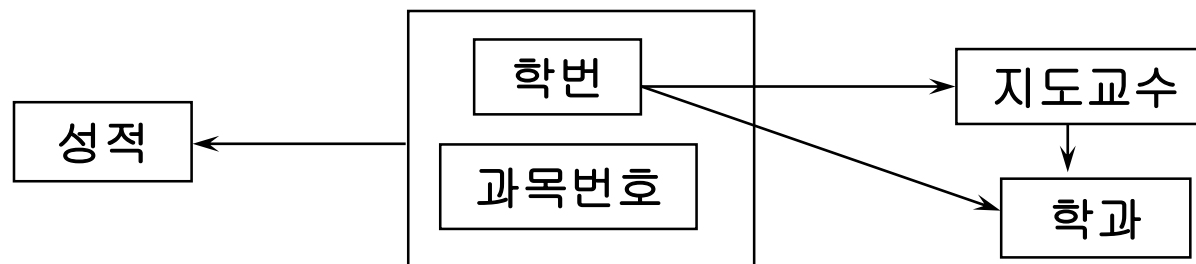
# ▶ 제1정규형 (1NF)

## ◆ 정의

- 모든 도메인이 원자값(atomic value)만으로 된 릴레이션

## ◆ 예 : 수강지도 릴레이션

- 수강지도 (학번, 지도교수, 학과, 과목번호, 성적)
- 기본 키 : {학번, 과목번호}
- 함수 종속 : {학번, 과목번호} → 성적  
학번 → 지도교수  
학번 → 학과  
지도교수 → 학과



## ▶ 제1정규형(2)

수강 지도

학번	지도교수	학과	과목번호	성적
100	P1	컴퓨터	C413	A
100	P1	컴퓨터	E412	A
200	P2	전기	C123	B
300	P3	컴퓨터	C312	A
300	P3	컴퓨터	C324	C
300	P3	컴퓨터	C413	A
400	P1	컴퓨터	C312	A
400	P1	컴퓨터	C324	A
400	P1	컴퓨터	C413	B
400	P1	컴퓨터	E412	C

## ▶ 제1정규형(3)

### ◆ 1NF에서의 이상

#### ① 삽입이상

- ◆ 500번 학생의 지도교수가 P4라는 사실의 삽입은 어떤 교과목을 등록하지 않는 한 삽입 불가능

#### ② 삭제이상

- ◆ 200번 학생이 C123의 등록을 취소하여 이 튜플을 삭제할 경우 지도교수가 P2라는 정보까지 손실됨

#### ③ 갱신이상

- ◆ 400번 학생의 지도교수를 P1에서 P3로 변경할 경우 학번이 400인 4개 튜플의 지도교수 값을 모두 P3로 변경해야 함



## ▶ 제1정규형(4)

### ◆ 1NF 이상의 원인

- 기본키에 부분 함수 종속된 애트리뷰트가 존재
  - 기본키로 식별되는 개체와 무관한 애트리뷰트가 존재
  - 두가지 상이한 정보가 포함

### ◆ 1NF 이상의 해결

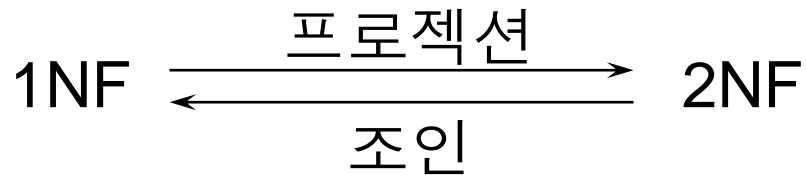
- 프로젝션으로 릴레이션을 분해 (부분 함수 종속을 제거)
  - ⇒ 2NF

## ▶ 제2정규형 (2NF)

### ◆ 정의

- 1NF이고, 키에 속하지 않는 애트리뷰트들은 모두 기본키에 완전 함수 종속

### ◆ 무손실 분해(nonloss decomposition)



- 프로젝션하여 분해된 릴레이션들은 자연 조인을 통해 원래의 릴레이션으로 복귀 가능
- 원래의 릴레이션에서 얻을 수 있는 정보는 분해된 릴레이션들로부터도 얻을 수 있음 그러나, 그 역은 성립하지 않음  
(500번 학생의 지도교수가 P4라는 정보는 원래의 릴레이션에서 표현할 수 없음)

## ▶ 제2정규형(2)

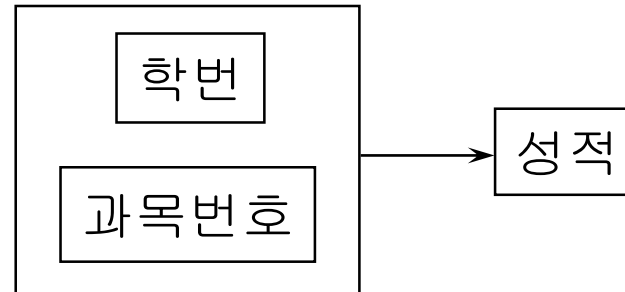
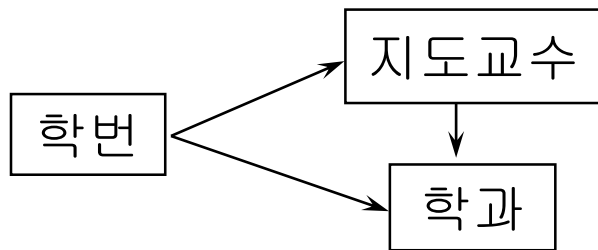
☞ **Note: Heath의 무손실 분해**

- $R(A,B,C)$ 에서 함수 종속  $A \rightarrow B$ 가 성립하면  
 $\Rightarrow R1(A,B), R2(A,C)$  로 무손실 분해 가능

## ▶ 제2정규형(3)

### ◆ 예 : 수강지도 ⇒ 지도, 수강 릴레이션

- 지도 (학번, 지도교수, 학과)  
기본 키 : {학번}
- 수강 (학번, 과목번호, 성적)  
기본 키 : {학번, 과목번호}  
외래 키 : {학번} 참조 : 지도



## ▶ 제2정규형(4)

지도

학번	지도교수	학과
100	P1	컴퓨터
200	P2	전기
300	P3	컴퓨터
400	P1	컴퓨터

수강

학번	과목번호	성적
100	C413	A
100	E412	A
200	C123	B
300	C312	A
300	C324	C
300	C413	A
400	C312	A
400	C324	A
400	C413	B
400	E412	C

## ▶ 제2정규형(5)

### ◆ 2NF(지도 릴레이션)에서의 이상

#### ① 삽입이상

- ◆ 어떤 지도교수가 특정 학과에 속한다는 사실의 삽입 불가능

#### ② 삭제이상

- ◆ 300번 학생의 튜플을 삭제하면 지도교수 P3가 컴퓨터공학과에 속한다는 정보 손실

#### ③ 갱신이상

- ◆ 지도교수 P1의 소속이 컴퓨터공학과에서 전자과로 변경된다면 학번이 100과 400번인 두개의 튜플을 모두 변경하여야 함

### ◆ 2NF 이상의 원인

- 이행적 함수 종속이 존재

## ▶ 제2정규형(6)

### ☞ Note :

- 이행적 함수 종속 (TD, Transitive Dependency)

$$A \rightarrow B \text{이고 } B \rightarrow C$$

$$\Rightarrow A \rightarrow C$$

(즉, 애트리뷰트 C는 애트리뷰트 A에 이행적 함수 종속)

### ◆ 2NF 이상의 해결

- 프로젝션으로 릴레이션 분해 (이행적 함수 종속을 제거)

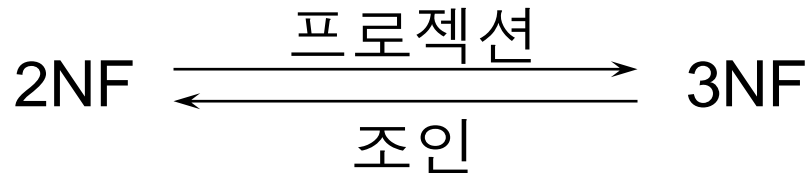
$$\Rightarrow 3NF$$

## ▶ 제3정규형 (3NF)

### ◆ 정의(3NF)

- 2NF이고, 키가 아닌 모든 애트리뷰트들은 기본키에 이행적 함수 종속되지 않음

### ◆ 무손실 분해



- 원래의 릴레이션에서 얻을 수 있는 정보는 분해된 릴레이션들로부터도 얻을 수 있으나 그 역은 성립하지 않음

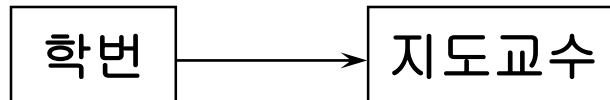
(지도교수 P4가 수학과에 속한다는 정보의 표현)



## ▶ 제3정규형(2)

◆ 예 : 지도 ⇒ 학생지도, 지도교수학과 릴레이션

- 학생지도 (학번, 지도교수)  
기본 키 : {학번}  
외래 키 : {지도교수} 참조 : 지도교수학과
- 지도교수학과 (지도교수, 학과)  
기본 키 : {지도교수}



학생지도

학번	지도교수
100	P1
200	P2
300	P3
400	P1

지도교수학과

지도교수	학과
P1	컴퓨터
P2	전기
P3	컴퓨터

## ▶ 제3정규형(3)

### 👉 Note

- 키가 아닌 애트리뷰트 값의 갱신시 불필요한 부작용(이상) 발생 없음
- 모든 이진 릴레이션은 3NF에 속함

## ▶ 제3정규형(4)

### ◆ 3NF의 약점

- i . 복수의 후보키를 가지고 있고
- ii . 후보키들이 복합 애트리뷰트들로 구성되고
- iii . 후보키들이 서로 중첩되는 경우

⇒ 적용 불가능

⇒ 보다 일반적인 Boyce/Codd Normal Form(BCNF)을 제안

## ▶ 보이스/코드 정규형(BCNF)

### ◆ 정의

- 릴레이션  $R$ 의 모든 결정자가 후보키이면 릴레이션  $R$ 은 BCNF에 속한다.

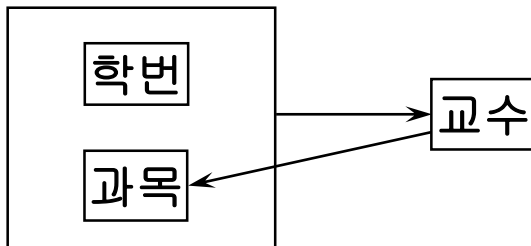
### ◆ 릴레이션 $R$ 이 BCNF에 속하면 $R$ 은 제1, 제2, 제3 정규형에 속함

### ◆ 강한 제3정규형(strong 3NF)이라고도 함

## ▶ 보이스/코드 정규형(2)

### ◆ 예(3NF) : 수강과목 릴레이션

- 제약 조건
  - ◆ 각 과목에 대한 한 학생은 오직 한 교수의 강의만 수강
  - ◆ 각 교수는 한 과목만 담당
  - ◆ 한 과목은 여러 교수가 담당할 수 있음
- 수강과목 (학번,과목,교수)
- 후보키 : {학번,과목}, {학번,교수}
- 기본키 : {학번,과목}
- 함수종속 : {학번,과목} → 교수  
교수 → 과목



수강과목

학번	과목	교수
100	프로그래밍	P1
100	자료구조	P2
200	프로그래밍	P1
200	자료구조	P3
300	자료구조	P3
300	프로그래밍	P4

## ▶ 보이스/코드 정규형(3)

### ◆ 3NF(수강과목 릴레이션)에서의 이상

#### ① 삽입이상

- ◆ 교수 P5가 자료구조를 담당한다는 사실의 삽입은 학번(수강 학생)이 있어야 가능

#### ② 삭제이상

- ◆ 100번 학생이 자료구조를 취소하여 투플을 삭제하면 P2가 담당교수라는 정보도 삭제됨

#### ③ 갱신이상

- ◆ P1이 프로그래밍 과목 대신 자료구조를 담당하게 되면 P1이 나타난 모든 투플을 변경하여야 함

⇒ 원인 : 교수가 결정자이지만 후보키가 아님

## ▶ 보이스/코드 정규형(4)

### ◆ 예(BCNF) : 수강과목 ⇒ 수강교수, 과목교수

수강교수(학번, 교수)

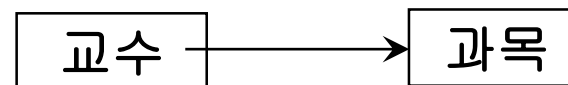
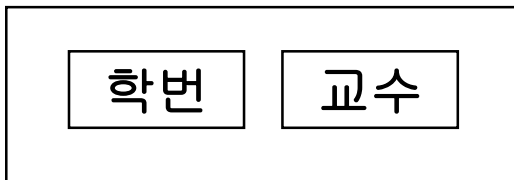
기본 키 : {학번, 교수}

외래키 : {교수} 참조 : 과목교수

과목교수(교수, 과목)

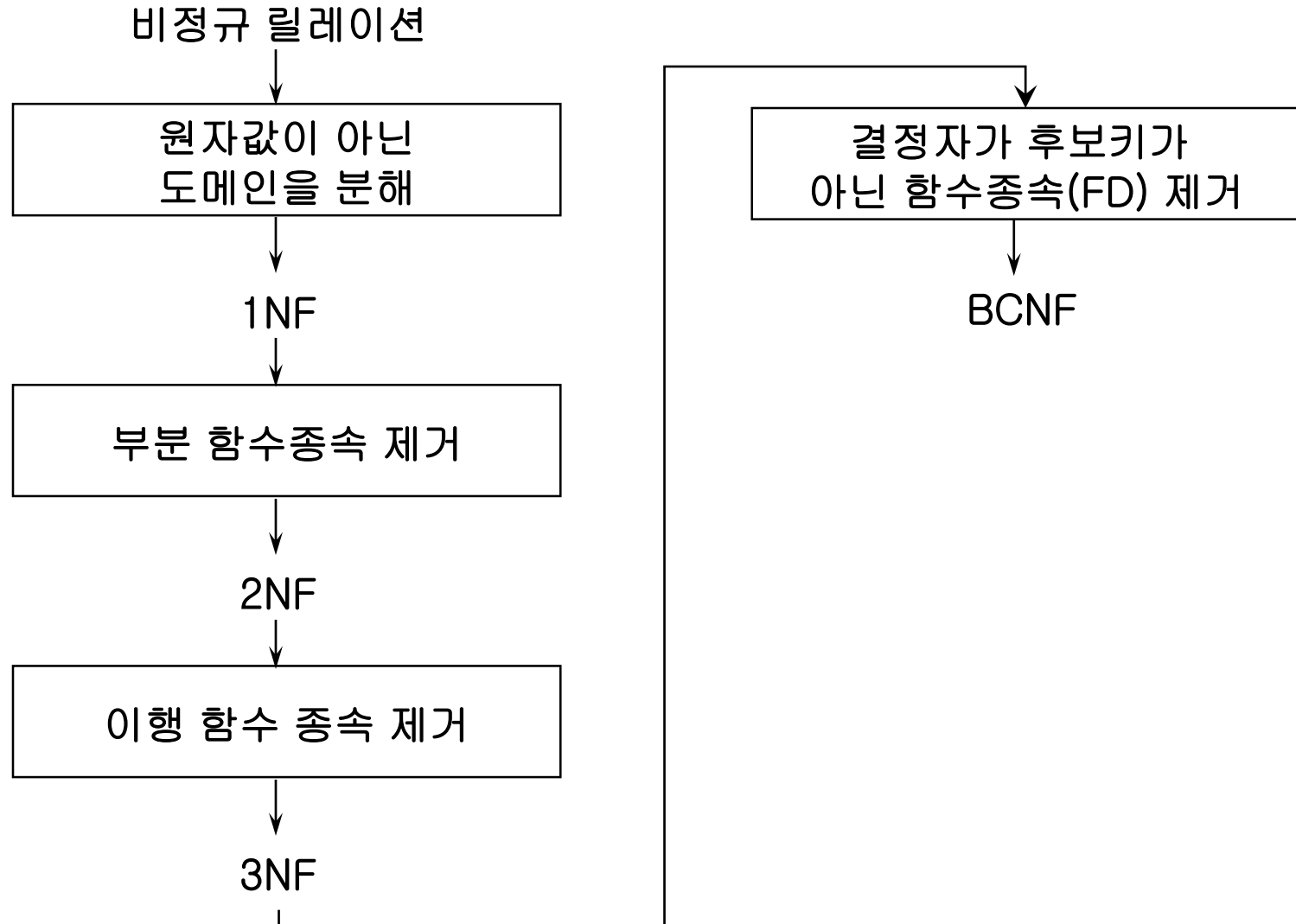
기본 키 : {교수}

수강교수	학번	교수	과목교수	교수	과목
	100	P1		P1	프로그래밍
	100	P2		P2	자료구조
	200	P1		P3	자료구조
	200	P3		P4	프로그래밍
	300	P3			
	300	P4			



# ❖ 정규형들 간의 관계

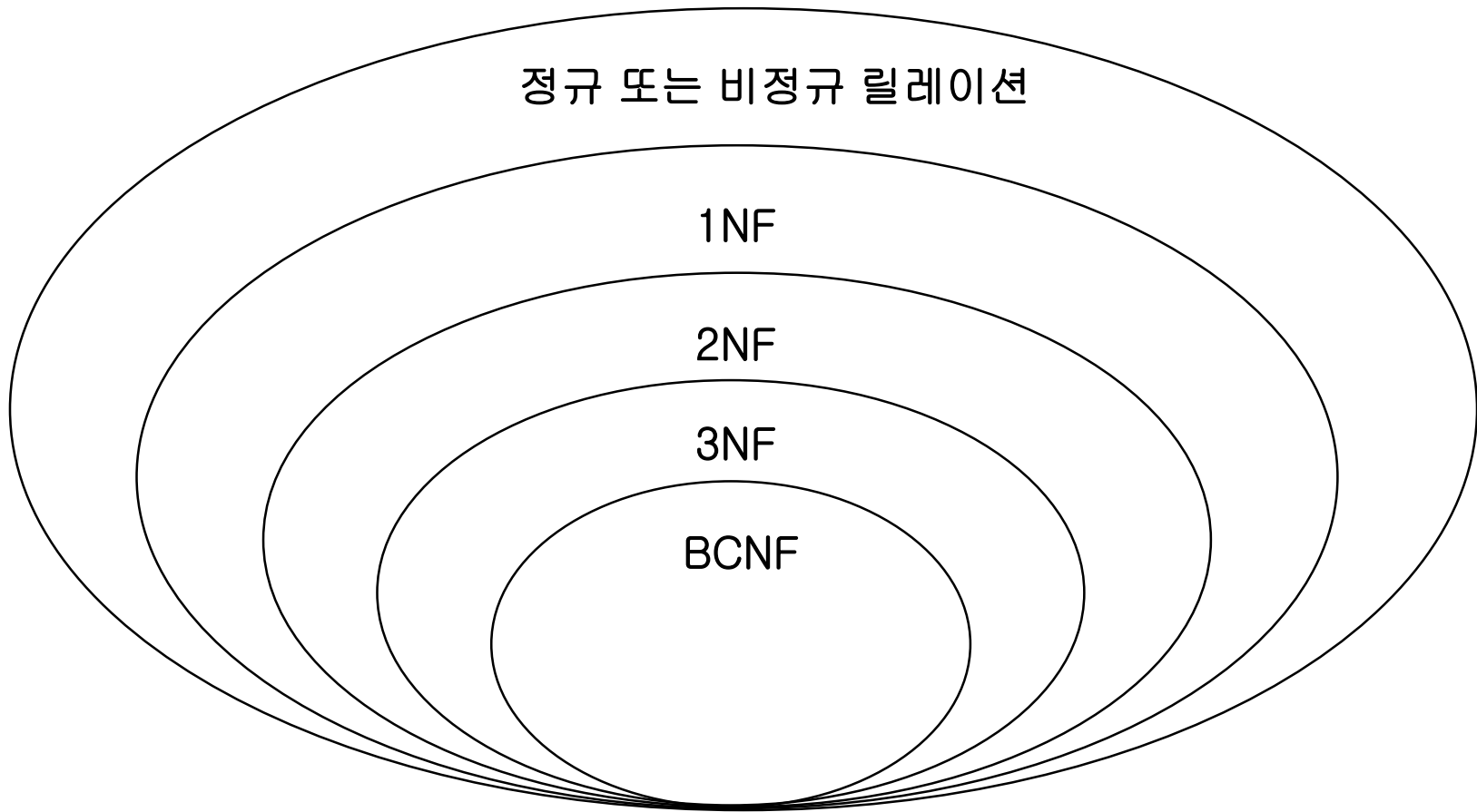
## ◆ 정규화 과정 (무손실 분해)





## ❖ 정규형들 간의 관계(2)

### ◆ 정규형들 간의 포함 관계



## ❖ 정규형들 간의 관계(3)

### 👉 Note

- 릴레이션의 정규화는 실제 데이터 값이 아니라 개념적인 측면에서 다루어져야 함
- 실제 정규화 과정은 정규형의 순서와 다를 수 있음

## ❖ 정규형들 간의 관계(4)

### 👉 Note

- 현실적으로 모든 릴레이션을 반드시 BCNF에 속하도록 분해할 필요는 없음

- 학생주소(학번,이름,주소,전화번호) : BCNF이 아님

기본키 : 학번

FD : 전화번호  $\rightarrow$  주소



학생전화(학번,이름,전화번호) : BCNF

전화주소(전화번호,주소) : BCNF

⇒ 이름, 전화번호, 주소는 분리하지 않고 사용하는 것이 편리하므로 위의 BCNF으로의 분해는 무의미함