

# 7장 정규화

데이터베이스 연구실

---

## [목 차]

- 7.1 관계 데이터베이스 설계
  - 7.1.1 설계 기준
  - 7.1.2 이상현상(anomaly)
- 7.2 설계 이론
  - 7.2.1 함수적 종속성
- 7.3 정규형
  - 7.3.1 정규형
  - 7.3.2 무손실 분해

# 7.1 관계 데이터베이스 설계

## 7.1.1 설계 기준

데이터베이스 설계 예

ex) 수강 - 신청 (학번, 이름, 학년, 수강과목, 과목시간) 정보 표현 방법

-> 다음 방법1과 방법2 중 어느 방법이 좋은가?

-> 어떤 문제가 있는가?

**방법 1**  
수강-신청

학번	이름	학년	수강과목	과목시간
100	이	4	PASCAL	월2,3
200	조	3	FORTRAN	화3,4
100	이	4	C	수1,2
150	김	3	PASCAL	월2,3
200	조	3	COBOL	금3,4

**방법 2**  
학생

학번	이름	학년
100	이	4
150	김	3
200	조	3

수강

학번	수강과목	과목시간
100	PASCAL	월2,3
200	FORTRAN	화3,4
100	C	수1,2
150	PASCAL	월2,3
200	COBOL	금3,4

## 7.1.2 이상현상(anomaly)

-좋은 설계는 데이터 조작 시 **anomaly**가 없도록 해야 한다.

ex) 계절 강좌 신청(학번, 과목, 수강료) – 잘못된 설계의 예

- 삭제이상(deletion anomaly) -> **triggered deletion** 현상
  - 150번 학생의 삭제(PASCAL 수강료의 삭제)
- 삽입이상(insertion anomaly) -> **null value** 문제
  - COBOL의 수강료(10,000원)의 삽입 (가상의 학생 삽입 필요)
- 갱신이상(update anomaly) -> **inconsistency** 발생
  - FORTRAN 수강료 변화 (모든 FORTRAN 수강료 data를 찾아 갱신)

학번	과목	수강료
100	FORTRAN	20,000
150	PASCAL	15,000
200	C	10,000
250	FORTRAN	20,000

릴레이션 : **계절강좌신청**

## 7.2 설계 이론

### 7.2.1 함수적 종속성

#### -함수적 종속성(functional dependency) ( $R.X \rightarrow R.Y$ )

(정의)  $X \subseteq R, Y \subseteq R$  일 때 relation R의 tuple t1, t2 에 대하여

(if  $(t1[X] = t2[X])$  이면  $\rightarrow (t1[Y] = t2[Y])$ ) 일 때, "Y는 X에 함수적으로 종속되었다"

- 즉,  $f(X)=Y$  에서 X가 같으면 Y는 항상 같다.

- 애트리뷰트 X는 Y를 (함수적으로) 결정, 즉, X는 Y의 결정자(determinant)-

- X, Y는 복합 애트리뷰트일 수 있음

- 속성 X가 키이면, R의 모든 애트리뷰트 Y에 대해  $X \rightarrow Y$  성립

#### ▶ 함수 종속성에 대한 **추론 규칙**

(기본규칙) R1: (재귀)  $A \supseteq B$  이면  $A \rightarrow B$  이다.

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

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

(추가규칙) R4: (연합)  $A \rightarrow B$  이고  $A \rightarrow C$  이면  $A \rightarrow BC$  이다.

R5: (분해)  $A \rightarrow BC$  이면  $A \rightarrow B, A \rightarrow C$  이다.

R6: (가이행)  $A \rightarrow B, CB \rightarrow D$  이면  $AC \rightarrow D$  이다.

#### -완전 함수 종속성 (fully functionally dependency) $\rightarrow$ 부분 함수 종속

(정의) Y가 X에 종속되어 있으면서 X의 어떤 진부분 집합에도 종속되지 않을 때

---

- 함수종속성 연습

□ 다음 함수적 종속성이 설립하는지 아닌지를 답하라.

A	B	C
2	3	8
5	9	6
7	9	6
5	2	2

(1)  $A \rightarrow B$

(2)  $B \rightarrow C$

(3)  $(B, C) \rightarrow A$

(4)  $(A, B) \rightarrow C$

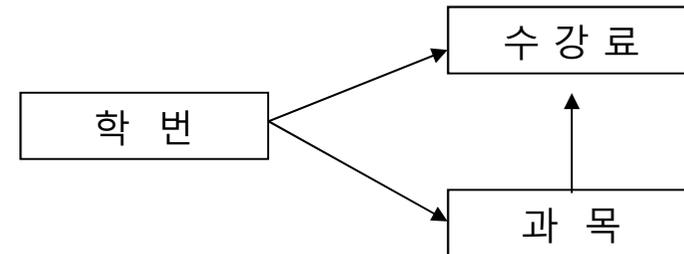
□ 다음 릴레이션에 존재하는 함수적 종속성을 모두 열거하라.

A	B	C	D
a1	b4	c1	d6
a1	b2	c4	d5
a2	b4	c1	d4
a2	b2	c4	d3
a2	b3	c2	d2

- 함수적 종속성 다이어그램

ex) 계절강좌신청 relation (**functionally dependency diagram**) 함수적 종속성 다이어그램

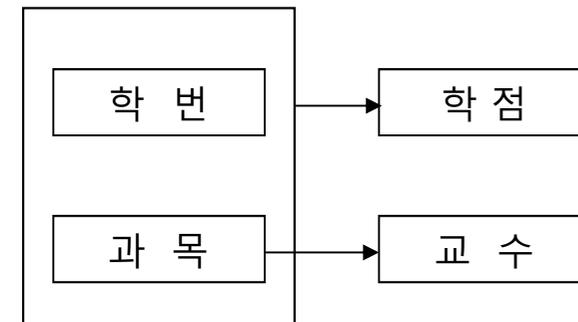
학번	과목	수강료
100	FORTTRAN	20,000
150	PASCAL	15,000
200	C	10,000
250	FORTTRAN	20,000



ex) 성적 relation 함수적 종속성 다이어그램

학번	과목	교수	학점
100	FORTTRAN	김	A
100	PASCAL	이	A
150	PASCAL	이	B
175	C	이	A
175	FORTTRAN	김	C

(a)



(b)

## 7.3 정규형

### 7.3.1 정규형 (Normal Form, NF)

---

#### (1) 제 1 정규형 (1NF)

정의 (1NF) : 어떤 릴레이션 R이 모든 도메인들의 값이 오직 원자값(atomic value)만을 가지며,  
릴레이션 R은 제 1정규형 (1NF)에 속한다

- 관계 데이터베이스 정의에 의하면 모든 속성의 값은 원자 값을 갖도록 정의하고 있다. 즉 모든 릴레이션은 1 정규형을 만족한다.

## (2) 제 2 정규형 (2NF)

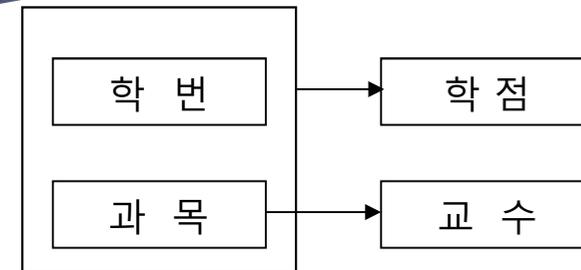
- 1 정규형의 불완전 함수적 종속성 (non-full FD)를 없앤다.

정의 (2NF) : 어떤 릴레이션 R이 1NF이고, 또 키가 아닌 모든 애트리뷰트 (non-key attribute)들이 기본 키에 완전 함수적 종속일 때 이 릴레이션 R은 제 2정규형(2NF)에 속한다.

GRAD 릴레이션 - 1정규형이나 2정규형은 아님

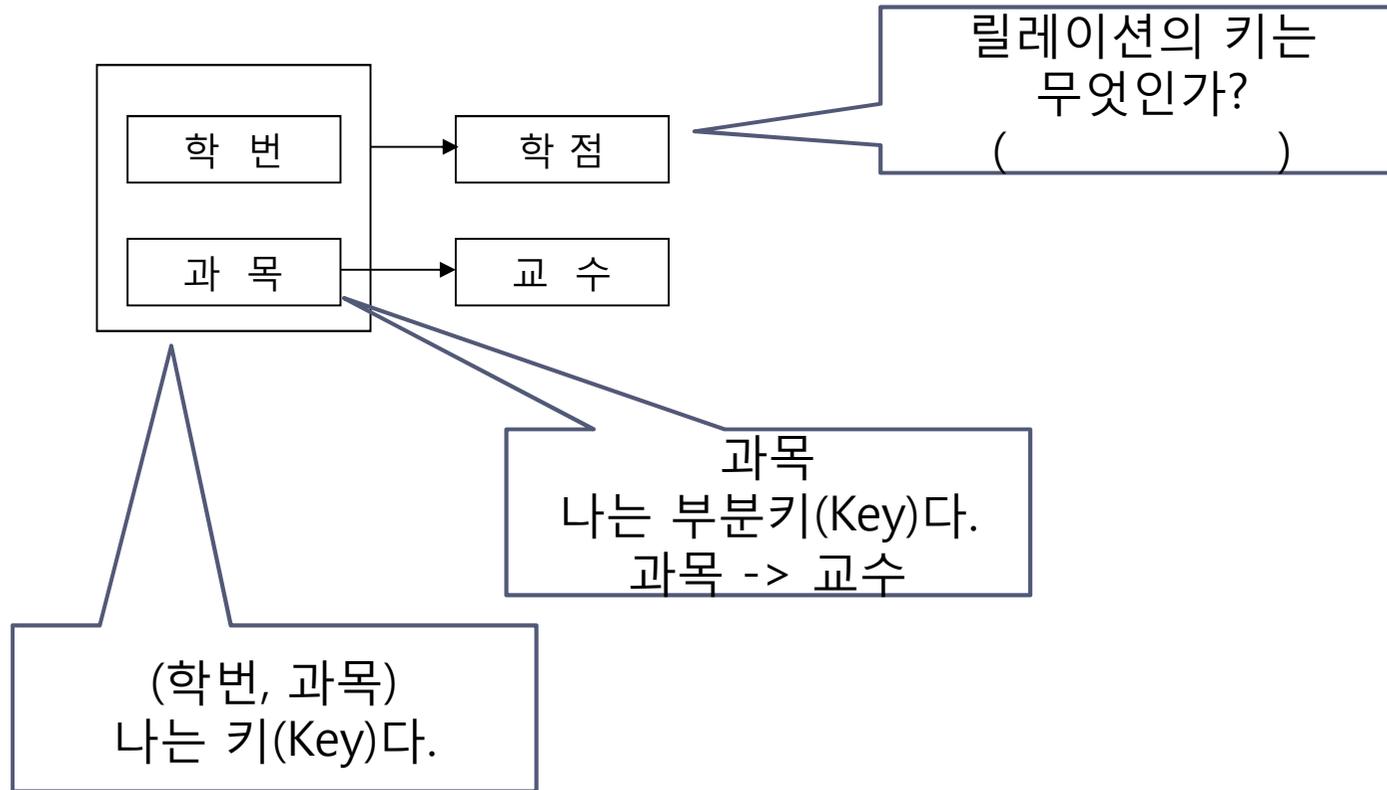
학 번	과 목	교 수	학 점
100	FORTTRAN	김	A
100	PASCAL	이	A
150	PASCAL	이	B
175	C	이	A
175	FORTTRAN	김	C

릴레이션의 키는 무엇인가?



릴레이션 GRAD와 종속성 다이어그램(1NF but NOT 1NF)

- 정규화가 안되는 경우 문제 발생 이유



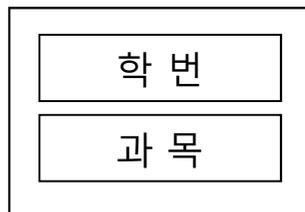
학 번	과 목	교 수	학 점
100	FORTRAN	김	A
100	PASCAL	이	A
150	PASCAL	이	B
175	C	이	A
175	FORTRAN	김	c

성적

학 번	과 목	학 점
100	FORTRAN	A
100	PASCAL	A
150	PASCAL	B
175	C	A
175	FORTRAN	C

강좌

과 목	교 수
FORTRAN	김
PASCAL	이
C	이
COBOL	조



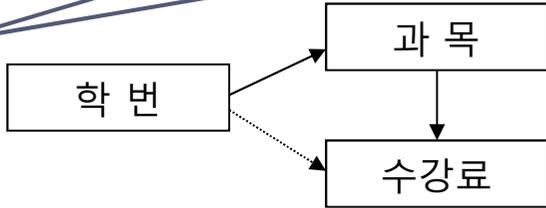
**(3) 제 3 정규형 (3NF) – 2NF에 있는 이행적 (transitive) 종속성을 분해한다.**

정의 (3NF) : 어떤 릴레이션 R이 2NF이고, 또 키가 아닌 모든 애트리뷰트들이 비 이행적(non-transitive)으로 기본 키에 종속되어 있을 때 이 릴레이션은 제 3정규형 (3NF)에 속한다

-> 릴레이션 PL(not 3NF) - 2NF but have anomaly

릴레이션의 키는 무엇인가?

학 번	과 목	수강료
100	FORTTRAN	20,000
150	PASCAL	15,000
200	C	10,000
250	FORTTRAN	20,000

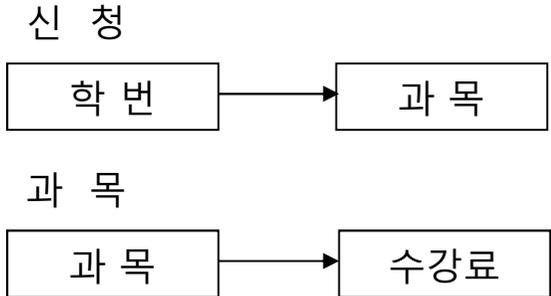


↓

학 번	과 목
100	FORTTRAN
150	PASCAL
200	C
250	FORTTRAN

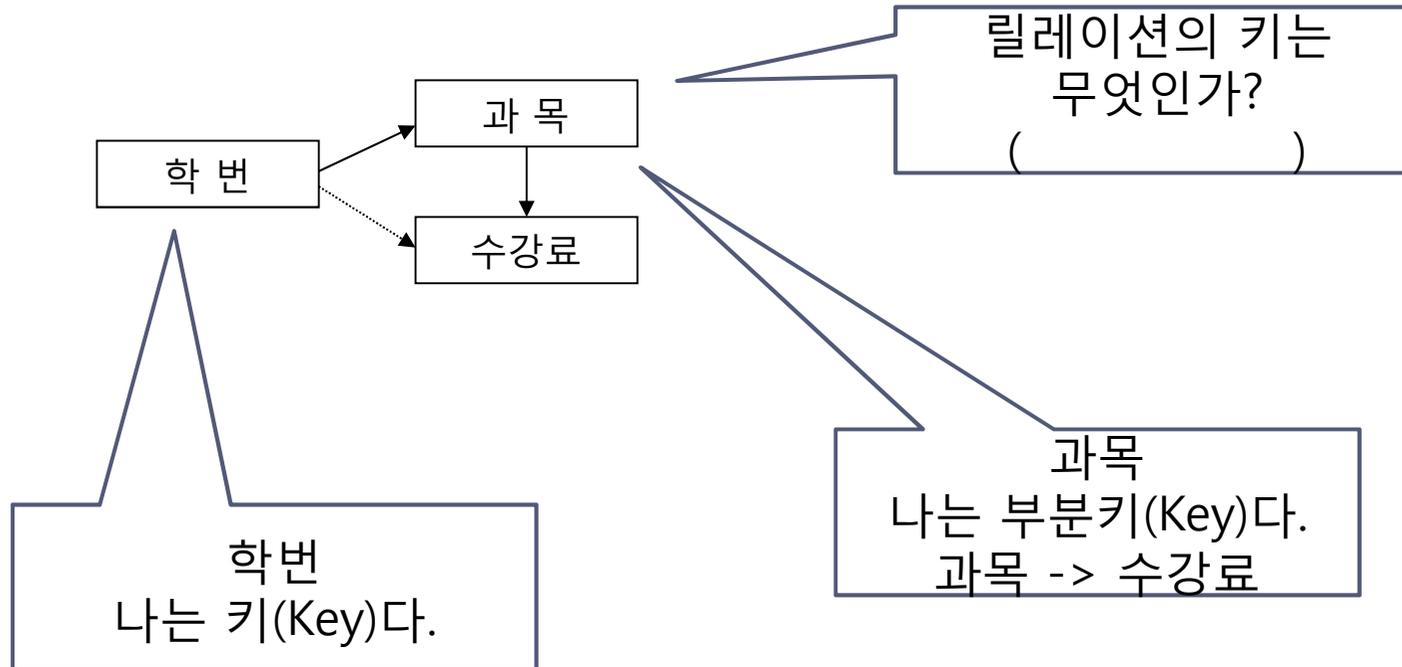
↓

과 목	수강료
FORTTRAN	20,000
PASCAL	15,000
C	10,000



▶ 12 -> 릴레이션이 PL을 두 개의 릴레이션 신청, 과목으로 분해

- 정규화가 안되는 경우 문제 발생 이유

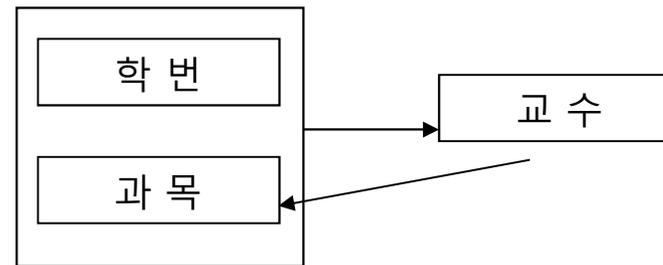


#### (4) Boyce / Codd Normal Form (BCNF)

정의 (BCNF) : 릴레이션 R의 모든 결정자(determinant)가 후보키(candidate key) 이면, 릴레이션 R은 Boyce/Codd 정규형에 속한다.

수강 현황 릴레이션 - (1 교수가 1과목을 가르친다.) - 3NF but anomaly

학 번	과 목	교 수
100	FORTTRAN	정
100	PASCAL	이
150	PASCAL	이
175	C	조
175	FORTTRAN	황
200	C	조
200	PASCAL	김

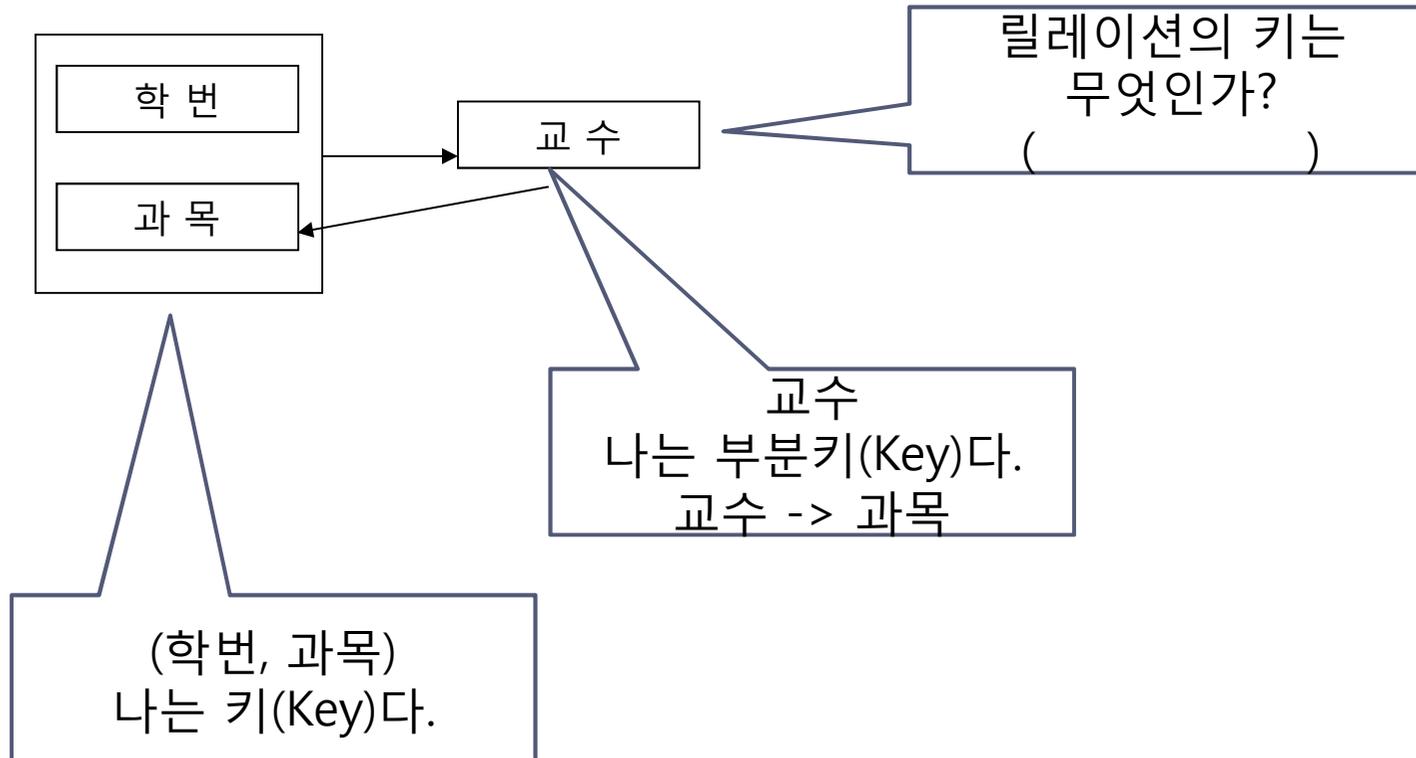


(학번,과목)을 기본키로 했을 때  
함수 종속성 다이어그램

릴레이션의 키는  
무엇인가?

( )

- 정규화가 안되는 경우 문제 발생 이유



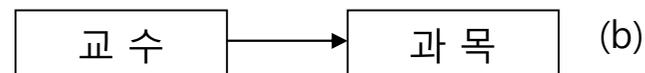
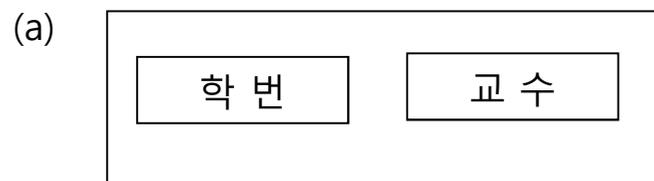
- 릴레이션 수강-현황을 두 개의 릴레이션으로 분해

학번 - 교수

학 번	교 수
100	정
100	이
150	이
175	조
175	황
200	조
200	김

과목 - 교수

과 목	교 수
FORTTRAN	정
PASCAL	이
C	조
FORTTRAN	황
PASCAL	김



릴레이션 분해 결과

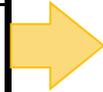
## (5) 제 4 정규형

정의 (4NF) : 릴레이션 R이 MVD A->->B가 성립하는 경우에 A가 슈퍼키이면 그 릴레이션 P는 4NF에 속한다.  
 즉,  $B \subseteq A$  or  $(AB) = R$  (trivial) 이거나 아니면 A가 슈퍼키여야 한다.

(다치종속성, MVD) : (정의) A,B,C 세 개의 애트리뷰트를 가진 릴레이션 R에서 속성에 대하여  $t1[A]=t2[a]$  튜플 t1과 t2가 존재한다면 다음 튜플 t3와 t4도 반드시 존재한다.  
 (A,B,C는 물론 복합 애트리뷰트이어도 상관 없다.)  
 $t1[A]=t2[A]=t3[A]=t4[A]$ 에 대하여  
 $t3[B]=t1[B]$  and  $t4[B]=t2[B]$ ,  $t3[C]=t2[C]$  and  $t4[C]=t1[C]$

개설-교과목

과 목	교 수	교 재
FORTTRAN	{정, 황}	{전산기 개론 및 프로그래밍, FORTRAN 77}
PASCAL	{이, 조}	{PASCAL REPORT 자료구조론}
C	김	C프로그래밍



개설-교과목 - (3NF but not 4NF)

과 목	교 수	교 재
FORTTRAN	정	전산기개론 및 프로그래밍
FORTTRAN	정	FORTRAN 77
FORTTRAN	황	전산기개론 및 프로그래밍
FORTTRAN	황	FORTRAN 77
PASCAL	이	PASCAL REPORT
PASCAL	이	자료구조론
PASCAL	조	PASCAL REPORT
PASCAL	조	자료구조론
C	김	C프로그래밍

- { FORTRAN 담당교수로 "정"과 "홍"이며 지정교재는 "T1"과 "T2" 중 어느 것도 (어느 교수, 어느 교재) 될 수 있다. }
- (함수종속성) 과목->교수, 과목->교재  
PASCAL -> {이,조}, PASCAL -> {PASCAL REPORT, 자료구조론}



과 목	교 수	과 목	교 재
FORTRAN	정	FORTRAN	전산기개론 및 프로그래밍
FORTRAN	홍	FORTRAN	FORTRAN 77
PASCAL	이	PASCAL	PASCAL REPORT
PASCAL	조	PASCAL	자료구조론
C	김	C	C프로그래밍

분해된 릴레이션(4NF) – 둘 다 4NF이다.

## (6) 제 5 정규형

---

정의 (5NF) : 릴레이션 R의 모든 조인 종속성(JD)의 만족이 R의 후보키로 암시될 수 있을 때 그 릴레이션은 제 5 정규형(5NF), 또는 프로젝션-조인 정규형(projection-join normal form; PJ/NF)에 속한다고 한다.

(정의) : 조인종속성(JD) – 일반적으로, 릴레이션 R이 그이 프로젝션 X,Y,...,Z의 조인으로 만들어 질 수 있을 때 조인 종속성(JD) (X,Y,...Z)를 만족한다고 한다. 여기서 X,Y,...,Z의 애트리뷰트 집합이 부분 집합이다.

예) **SPJ** (supplier supplies the part to the specified project)

SPJ 릴레이션

Supplier Part Project

S#	P#	J#
S1	P1	J2
S1	P2	J1
S2	P1	J1
S1	P1	J1

If this is inserted So is this tuple  
(그래야 project-join = original)

SP

부적 튜플

S#	P#
S1	P1
S1	P2
S2	P1

P#	J#
P1	J2
P2	J1
P1	J1

J#	S#
J2	S1
J1	S1
J1	S2

P#으로 조인

S#	P#	J#
S1	P1	J2
S1	P1	J1
S1	P2	J1
S2	P1	J2
S2	P1	J1

JP

(J#,S#)로 조인

원래의 릴레이션 SPJ

예) SPJ의 **anomaly** 발생 예 (앞 예에서 (S2,P1,J1)만 삽입했을 때)

SPJ 릴레이션

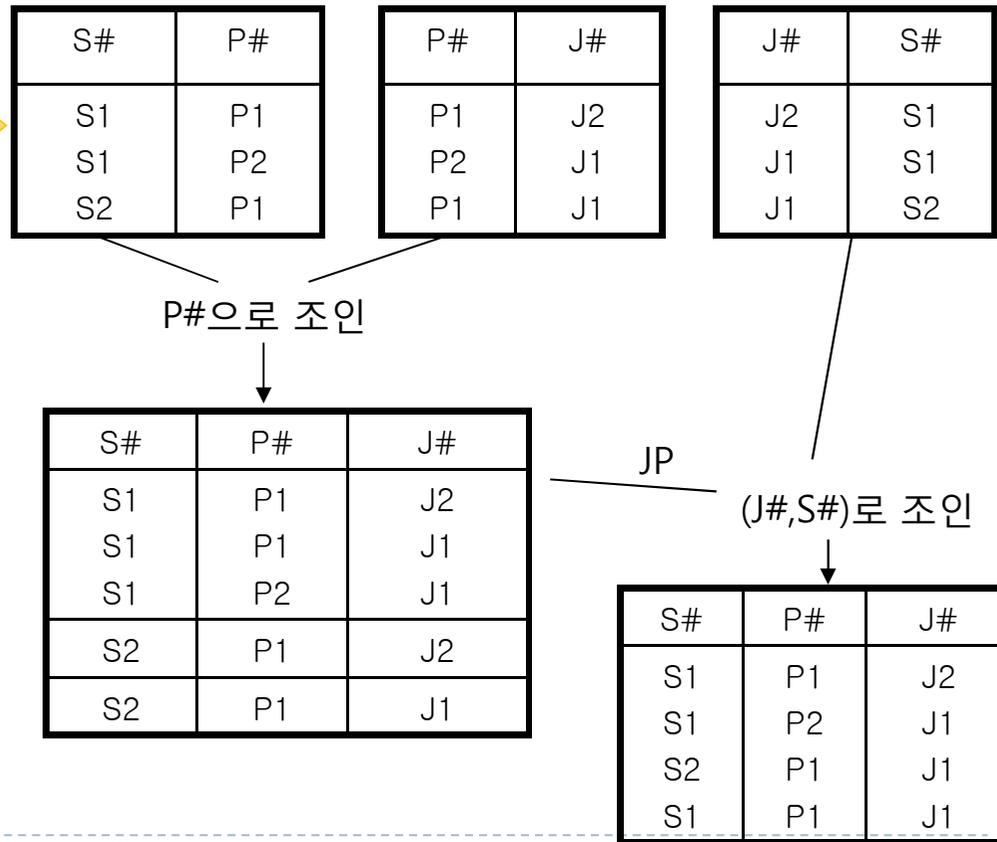
Supplier Part Project

S#	P#	J#
S1	P1	J2
S1	P2	J1
S2	P1	J1

If this is inserted

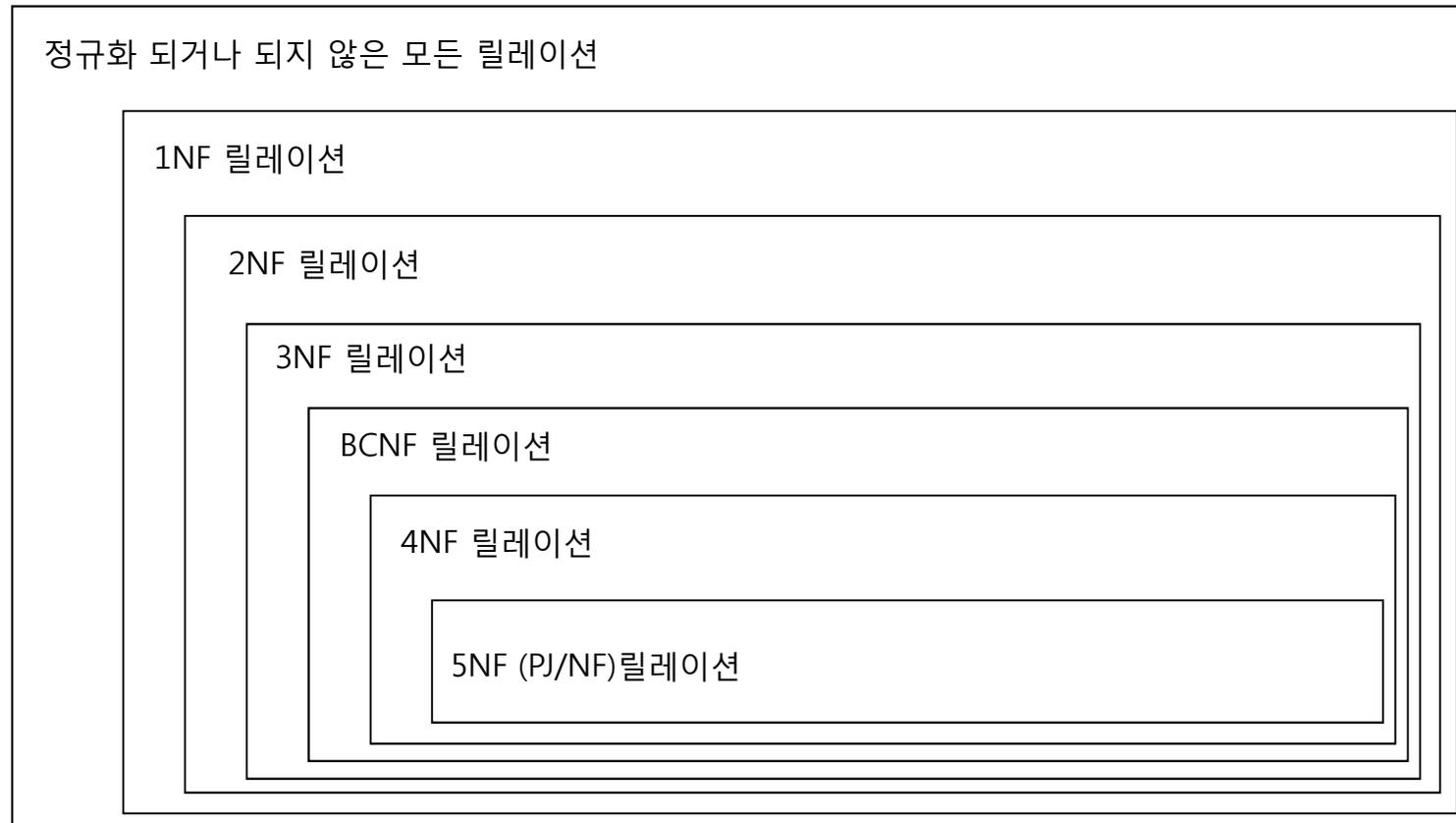
부적 튜플

SP



## << 정규형들의 포함 관계 >>

---



## - 정규화 정리

---

- 데이터베이스 설계의 **문제점**은 다음과 같다 - 잘못된 설계의 결과
  1. 정보의 중복 ( repetition of information -> inconsistency )
  2. NULL 값 ( inability to represent certain information -> null value )
  3. 정보의 손실 (loss of information )

-잘못된 설계의 **해결책**은 분해(decomposition)

(anomaly의 해결책은 분해이다.)

relation R의 분해 ->  $R_1 R_2, \dots, R_n$

$r \subseteq r_1 \bowtie r_2, \dots, \bowtie r_n$  이 성립 => ( $r = r_1 \bowtie r_2, \dots, \bowtie r_n$  이면 무손실 분해)

- **분해의 바람직한 조건**  $R \rightarrow R_1$  and  $R_2$

1. 무손실 분해(lossless-join decomposition,  $r = r_1 \bowtie r_2, \dots, \bowtie r_n$  )

if at least  $R_1 \cap R_2 \rightarrow R_1$  or  $R_1 \cap R_2 \rightarrow R_2$

2. 종속성 보존(dependency preservation)

if  $F_i^+ = F$

3. 정보 중복의 해소(avoid repetition of information)

-> 분해하면 자동으로 해결

- 함수적 종속성에 의한 분해

(normalization using functional dependencies)

- **BCNF** 함수종속성  $\alpha \rightarrow \beta$ 에 대하여 at least one of 1,2 holds

1.  $\alpha \rightarrow \beta$  is trivial FD
2.  $\alpha$  is a superkey for scheme R

(BCNF 정규화 :  $\alpha \rightarrow \beta$  when  $\alpha$  is not a key, decomposition of R : (R -  $\beta$ ), ( $\alpha \beta$ )

(예 R(A,B,C) with  $AB \rightarrow C$ ,  $C \rightarrow B \Rightarrow R(A,C)$ ,  $R(C,B)$  with  $C \rightarrow B$

단점 : dependency  $AB \rightarrow C$ 가 없어진다.

dependency를 보존하기 위하여 그대로 두면 데이터 중복이 있다)

- **3NF** 함수종속성  $\alpha \rightarrow \beta$ 에 대하여 at least one of 1,2,3 holds

1.  $\alpha \rightarrow \beta$  is trivial FD
2.  $\alpha$  is a superkey for scheme R
3. every attribute A in  $\beta$  is contained in a candidate key for R  
( 3NF allow transitive dependencies – eg  $AB \rightarrow C$ ,  $C \rightarrow A$  )

- BCNF와 3NF 비교

	3NF	BCNF	MVD
Lossless decomposition	O	O	O
Dependency preserving	O	X	X
No repetition of info	X	O	O

### 7.3.2 무손실 분해

- 무손실 분해(lossless decomposition)/종속성 보존(dependency preservation)

□ **손실 분해**의 예 - R을 R1과 R2로 분해하면 어떻게 될까?

A	B	C
a1	b1	c1
a1	b1	c2
a2	b1	c1
a3	b2	c2

A	B
a1	b1
a2	b1
a3	b2

B	C
b1	c1
b1	c2
b2	c2

□ **종속성 보존**이 되지 않는 경우의 문제의 예 - R을 R1과 R2로 분해하면 어떻게 될까?

(R 에서 (100,FORTRAN,최)는 삽입이 불가능하지만 분해된 R1, R2에서는 삽입이 가능함)

학 번	과 목	교 수
100	FORTRAN	정
100	PASCAL	이
150	PASCAL	이
175	C	조
175	FORTRAN	황
200	C	조
200	PASCAL	김

학 번	교 수
100	정
100	이
150	이
175	조
175	황
200	조
200	김
100	최

과 목	교 수
FORTRAN	정
PASCAL	이
C	조
FORTRAN	황
PASCAL	김
FORTRAN	최

---

## - 분해 연습

- 수강과목(학번,과목,교수) 릴레이션(3NF)의 분해 예  
키                               :(학번,과목), (학번,교수)  
종속성                         :(학번,과목)->교수, 교수->과목

- 아래 3가지로 분해하는 경우 각각 무슨 문제가 생길 것인가?

1. R1(학번,과목), R2(과목,교수)
2. R1(학번,과목), R2(학번,교수)
3. R1(학번,교수), R2(과목,교수)