

데이터베이스 설계 (Database Design) Lecture 11: 정규화

담당교수: 전강욱(컴퓨터공학부)
kw.chon@koreatech.ac.kr

NCS 정보

- 능력 단위명 : 논리 데이터베이스 설계
- 능력 단위요소 : 데이터베이스 정규화하기
- 학습목표(수행 준거):
 - 4.1 개체 내 속성 원자성과 함수 종속성을 분석하여 데이터베이스 정규화 대상 개체를 도출할 수 있다.
 - 4.2 도출된 데이터베이스 정규화 대상 개체에 대해 정규화가 반드시 필요한 지를 판단하여, 그 경우에 한하여 데이터베이스 정규화를 수행할 수 있다

NCS 정보(계속)

■ 지식

- 갱신 이상과 함수적 종속성
- 1NF ~ 3NF, BCNF 정규화 방법
- 역정규화 방법

■ 기술

- 갱신 이상 현상 분석 능력
- 함수적 종속 다이어그램(FDD) 작성 능력
- 역정규화의 필요성 판단 능력

■ 태도

- 갱신 이상 현상을 정확히 판단하려는 태도
- 함수적 종속성을 정확히 분석하려는 자세
- 정규화 과정을 정확히 문서화하려는 태도

세부 학습목표

- 1. 갱신 이상이 무엇인지 설명할 수 있다.
- 2. 함수적 종속에 대해 설명할 수 있다.
- 3. 정규화가 무엇인지 설명할 수 있다.
- 4. 무손실 분해를 통해서 릴레이션을 정규화할 수 있다.
- 5. 역정규화의 필요성에 대해 설명할 수 있다.

개요

- 정규화 필요성
- 함수종속성
- 정규화 과정

릴레이션 스키마 설계

- 관계형 데이터베이스 설계
 - 좋은 릴레이션 스키마를 생성하기 위해 애트리뷰트들을 그룹핑(grouping)하는 과정
 - 좋은 릴레이션의 기준?
- 릴레이션 스키마의 두 가지 수준
 - 논리적인 "사용자 뷰" 수준
 - 저장이 되는 "기본 릴레이션" 수준
- 데이터베이스 설계는 주로 기본 릴레이션을 대상으로 함

정규화 개념

■ 관계 데이터베이스의 대표적인 설계 방법

- ER 모델과 릴레이션 변환 규칙을 적용한 데이터베이스 설계
- 정규화를 이용한 데이터베이스 설계

■ 이상(anomaly) 현상

- 불필요한 데이터 중복으로 인하여 릴레이션에 대한 데이터 삽입·수정·삭제 연산을 수행할 때 발생할 수 있는 부작용

■ 정규화

- 이상 현상을 제거하면서 데이터베이스를 올바르게 설계해 나가는 과정
- 설계한 후 설계 결과물을 검증하기 위해 사용

정규화의 개념과 이상 현상

■ 이상 현상의 종류

- 삽입 이상: 새 데이터를 삽입하기 위해 불필요한 데이터도 함께 삽입해야 하는 문제
- 갱신 이상: 중복 튜플 중 일부만 변경하여 데이터가 불일치하게 되는 모순의 문제
- 삭제 이상: 튜플을 삭제하면 꼭 필요한 데이터까지 함께 삭제되는 데이터 손실의 문제

이상현상 예제

■ 수강 릴레이션 예제(기본키: 학번, 과목번호)

■ 삭제이상(deletion anomaly)

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

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	C412	C	4
500	C312	B	2

■ 삽입이상(insertion anomaly)

- 600번 학생이 2학년이라는 사실을 삽입
 - 어떤 과목을 등록하지 않는 한 삽입이 불가능
 - ∴ 과목 번호가 기본키
- 원하지 않는 정보의 강제 삽입

■ 갱신이상(update anomaly)

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

이상의 원인과 해결책

■ 이상의 원인

- 애트리뷰트들 간에 존재하는 여러 종속관계를 하나의 릴레이션에 표현

■ 이상의 해결

- 애트리뷰트들 간의 종속관계를 분석하여 여러 개의 릴레이션으로 분해(decomposition)
 - 정규화(normalization)

스키마 설계와 변환

- 스키마 설계 : 데이터베이스의 논리적 설계
 - 애트리뷰트들과 이들의 제약 조건 (종속성)들을 수집
 - 수집된 결과를 명시된 제약 조건에 따라 여러 개의 릴레이션으로 분할
 - 스키마 변환 (schema transformation)
- 스키마 변환의 원리
 - 정보의 무손실
 - 데이터의 중복성 감소
 - 분리의 원칙

함수 종속 (FD)

■ 함수 종속 정의

- 어떤 릴레이션 R에서, 애트리뷰트 X의 값 각각에 대해 애트리뷰트 Y의 값이 하나만 연관 \Leftrightarrow 애트리뷰트 Y는 애트리뷰트 X에 함수 종속 $X \rightarrow Y$
- 애트리뷰트 X는 Y를 (함수적으로) 결정
 - 즉, X는 Y의 결정자
 - 결정자(determinant): 주어진 릴레이션에서 다른 애트리뷰트(또는 애트리뷰트들의 집합)을 고유하게 결정하는 하나 이상의 속성
- X, Y는 복합 애트리뷰트일 수 있음

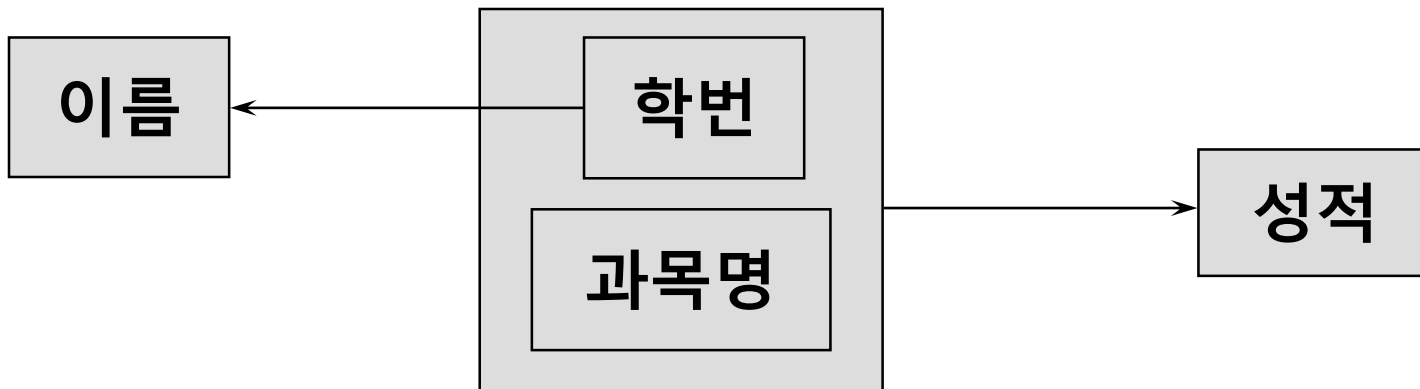
함수 종속 (계속)

- 릴레이션 R에서 애트리뷰트 X가 키이면, R의 모든 애트리뷰트 Y에 대해 $X \rightarrow Y$ 성립
- 함수종속 $X \rightarrow Y$ 의 경우
 - 애트리뷰트 X가 반드시 키(유일 값)라는 것을 요건으로 하지 않음
 - 즉, $x(X)$ 값이 하나 이상의 튜플 값으로 존재가능

함수 종속 다이어그램

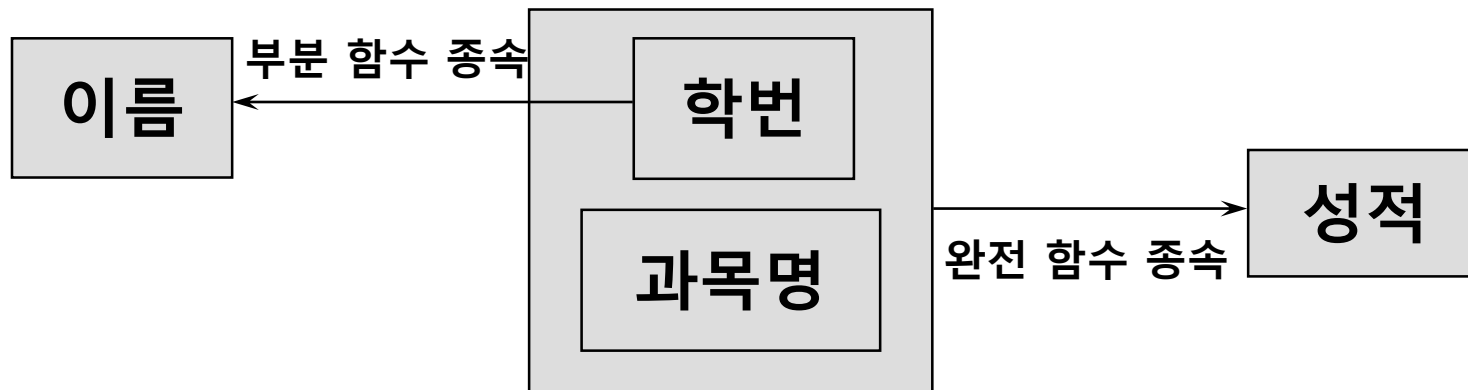
■ 수강 릴레이션에서 결정자는? 수강

학번	과목명	성적	이름
100	전자계산기구조	92	김사랑
101	데이터베이스	82	오지호
100	운영체제	90	김사랑
101	데이터통신	76	오지호
102	운영체제	82	이선균



완전 함수 종속과 부분 함수 종속

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



완전 함수 종속과 부분 함수 종속 (계속)

■ 예: 수강 릴레이션의 함수 종속

학번 \rightarrow 이름

(학번, 과목번호) \rightarrow 성적

(이름)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 (계속)

■ 추론 규칙

R1: (반사) $A \rightarrow B$ 이면 $A \rightarrow B$ 이다.

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

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

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

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

■ 함수 종속은 데이터의 의미(data semantics) 를 표현

□ 예: "학번 \rightarrow 학년"의 의미는 "학생은 학년에 속한다"

□ 의미적 제약 조건

■ DBMS는 함수 종속을 유지하기 위하여

□ 함수 종속을 스키마에 명세하는 방법과 함수 종속을 보장하는 방법을 제공하여야 함

기본 정규형

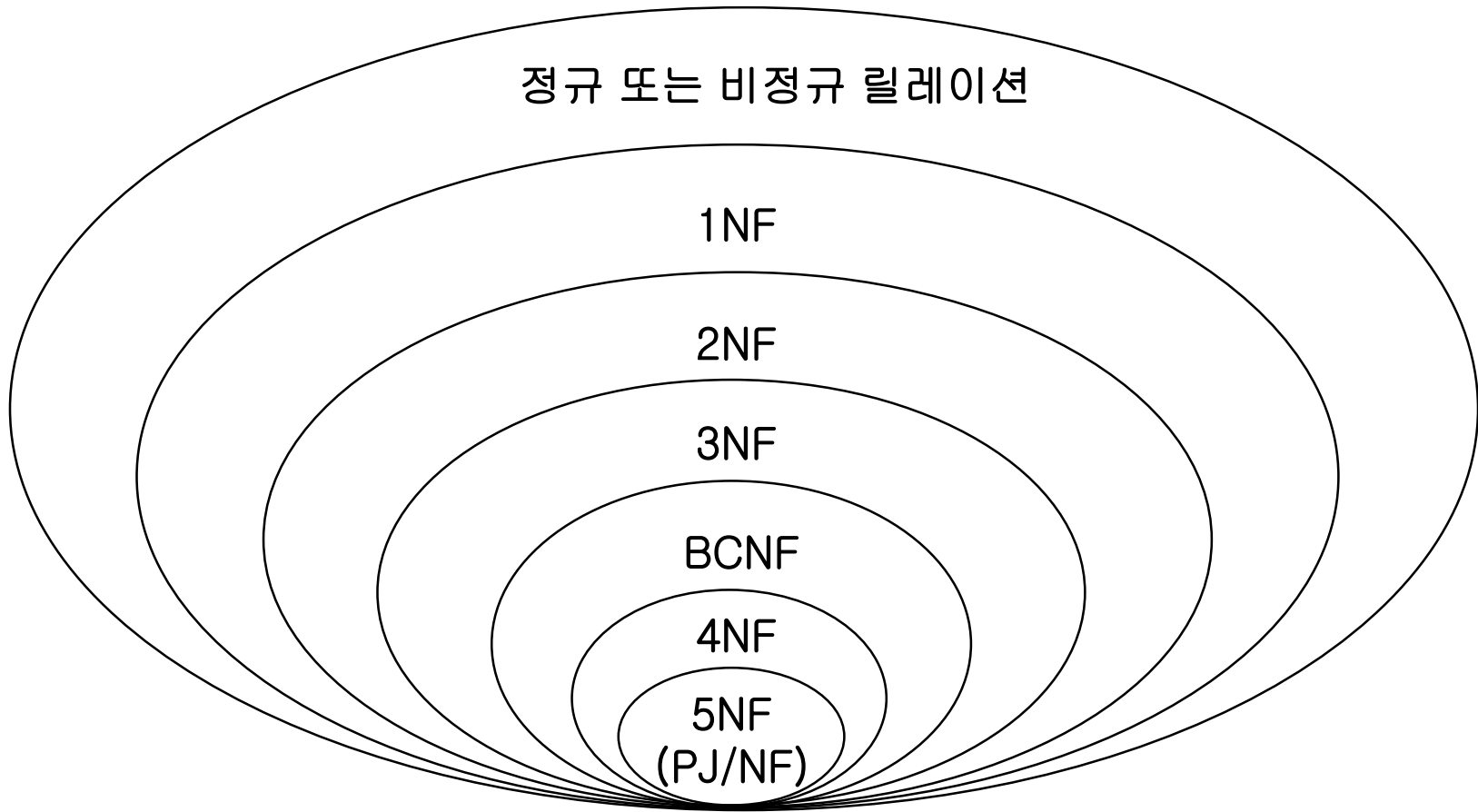
■ 정규형(Normal Form)

- 어떤 일련의 제약 조건을 만족하는 릴레이션

■ 정규화(Normalization)의 원칙

- 정규화 = 스키마 변환 ($S \rightarrow S'$)
- 무손실 표현
 - 같은 의미의 정보 유지
 - 그러나 더 바람직한 구조
- 데이터의 중복성 감소
- 분리의 원칙
 - 독립적인 관계는 별개의 릴레이션으로 표현
 - 릴레이션 각각에 대해 독립적 조작이 가능

정규형들 간의 관계



제1정규형(1NF)

■ 제1정규형에 만족하는 릴레이션

- 어떤 릴레이션 R에 속한 모든 도메인이 원자값(atomic value)
- 릴레이션의 속성값이 반복 집단이 없는 즉, 더 이상 분해될 수 없는 원자값으로만 구성

■ 반복 집합이 있는 비정규 릴레이션

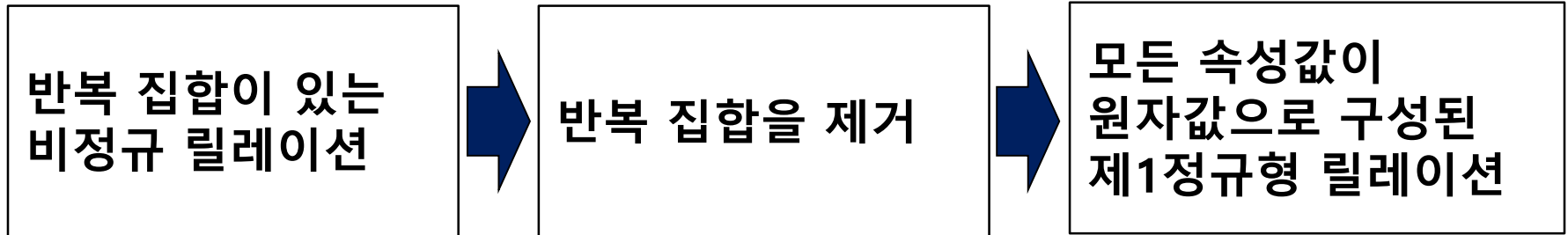
- 반복 집합은 한 개의 기본키 값에 대해서 두 개 이상의 값을 가질 수 있는 속성

수강

학번	과목명	성적	이름
100	{전자계산기구조, 운영체제}	{92,90}	김사랑
101	{데이터베이스, 데이터통신}	{82,76}	오지호
102	운영체제	82	이선균

제1정규형(계속)

■ 제1정규화 과정의 개념



■ 제1정규화 과정

수강

학번	과목명	성적	이름
100	{전자계산기구조, 운영체제}	{92,90}	김사랑
101	{데이터베이스, 데이터통신}	{82,76}	오지호
102	운영체제	82	이선균



학번	과목명	성적	이름
100	전자계산기구조	92	김사랑
101	데이터베이스	82	오지호
100	운영체제	90	김사랑
101	데이터통신	76	오지호
102	운영체제	82	이선균

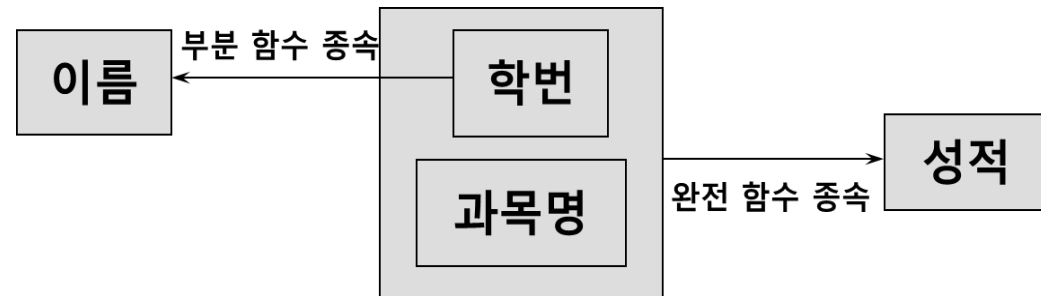
제2정규형(2NF)

■ 제2정규형

- 어떤 릴레이션 R이 제1정규형이고, 키에 속하지 않는 속성 모두가 키에 완전 함수 종속

수강

학번	과목명	성적	이름
100	전자계산기구조	92	김사랑
101	데이터베이스	82	오지호
100	운영체제	90	김사랑
101	데이터통신	76	오지호
102	운영체제	82	이선균



제2정규형(계속)

■ 제1정규화 과정의 개념

모든 속성값이
원자값으로
구성된 제1정규형
릴레이션



부분 함수의
종속성을 제거



모든 속성이 키에
완전 함수 종속인
제2정규형 릴레이션

■ 제2정규화 과정

학번	과목명	성적	이름
100	전자계산기구조	92	김사랑
101	데이터베이스	82	오지호
100	운영체제	90	김사랑
101	데이터통신	76	오지호
102	운영체제	82	이선균

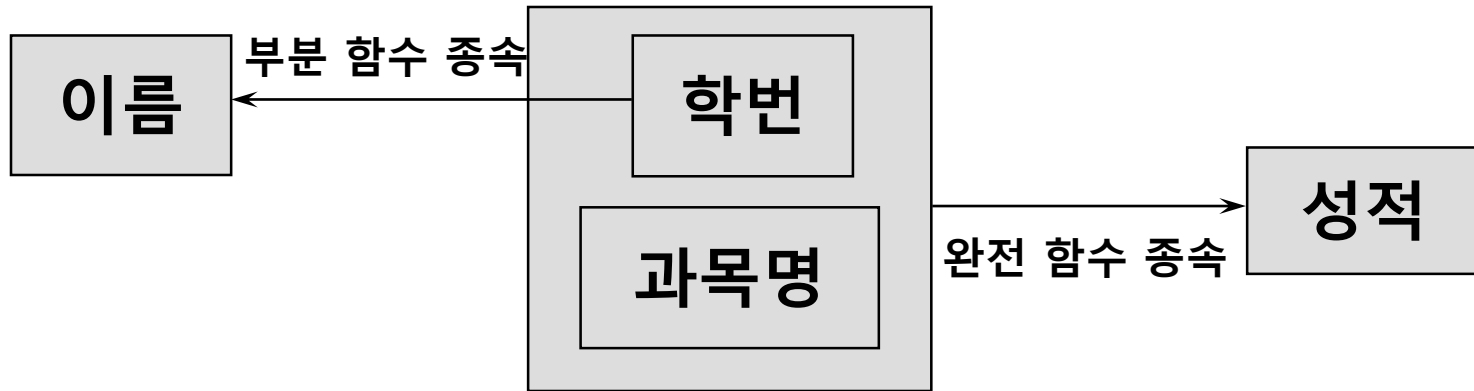


학번	과목명	성적
100	전자계산기구조	92
101	데이터베이스	82
100	운영체제	90
101	데이터통신	76
102	운영체제	82

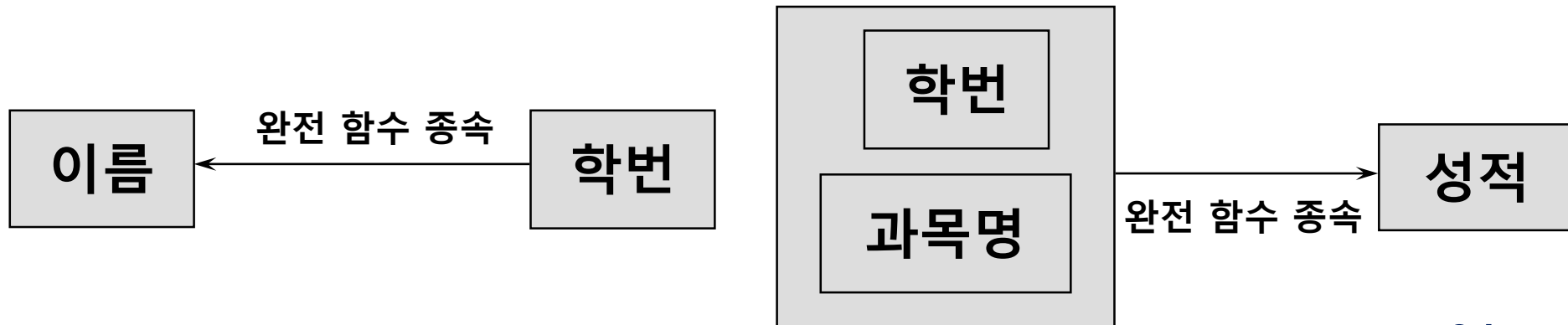
학번	이름
100	김사랑
101	오지호
102	이선균

제2정규형(계속)

■ 제2정규화 과정의 함수 종속 다이어그램



부분 함수의 종속성을 제거



제2정규형(계속)

■ 무손실분해(nonloss decomposition)

- 자연 조인하였을 때 아무런 정보 손실 없이 다시 원래의 릴레이션으로 복귀된다면 2NF 두 개의 릴레이션으로 분해하는 것

제3정규형(3NF)

■ 제3정규형

- 어떤 릴레이션 R이 2NF이고, 모든 속성들이 기본키에 이행적 함수 종속(transitive FD)을 제외

■ 이행적 함수 종속성

- 3개의 속성에 존재하는 함수의 종속성을 의미
- $A \rightarrow B$ 와 $B \rightarrow C$ 이면, $A \rightarrow C$
 - (즉, 애트리뷰트 C는 애트리뷰트 A에 이행적 함수 종속)
- 예: 학번 \rightarrow 지도교수 & 지도교수 \rightarrow 학과이면,
학번 \rightarrow 학과

제3정규형(계속)

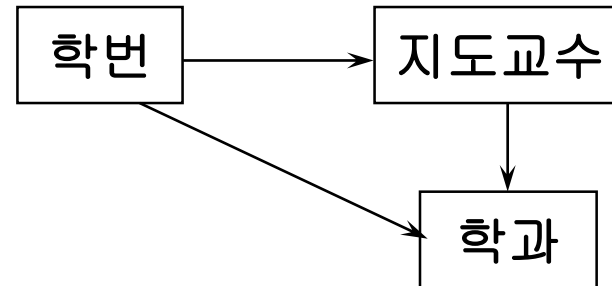
■ 이행적 함수 종속하는 지도 릴레이션

□ 학번 → 지도교수 \wedge 지도교수 → 학과 ➡ 학번 → 학과

지도

학번	지도교수	학과
100	이문세	컴퓨터 소프트웨어학과
101	김연아	멀티미디어학과
102	이문세	컴퓨터 소프트웨어학과
103	강승범	경영 정보학과
104	이문세	컴퓨터 소프트웨어학과
105	김연아	멀티미디어학과

이행적 함수 종속이 존재



제3정규형(계속)

- 이행적 함수 종속으로 인한 갱신 이상
 - 수정이상

지도

학번	지도교수	학과
100	이문세	컴퓨터 소프트웨어학과
101	김연아	멀티미디어학과
102	이문세	컴퓨터 소프트웨어학과 정보처리학과
103	강승범	경영 정보학과
104	이문세	컴퓨터 소프트웨어학과
105	김연아	멀티미디어학과

정보의 모순성이 발생

제3정규형(계속)

- 이행적 함수 종속으로 인한 갱신 이상
 - 삽입 이상

지도

학번	지도교수	학과
100	이문세	컴퓨터 소프트웨어학과
101	김연아	멀티미디어학과
102	이문세	컴퓨터 소프트웨어학과
103	강승범	경영 정보학과
104	이문세	컴퓨터 소프트웨어학과
105	김연아	멀티미디어학과
?	전혜영	인공지능학과

전혜영이란 교수가 인공지능학과에 속한다는 사실만 기록해 놓으려고 하는데 해당 교수의 지도를 받는 학생이 없다면 지도 릴레이션에 이 교수의 정보를 삽입할 수 없음

제3정규형(계속)

- 이행적 함수 종속으로 인한 갱신 이상
 - 삭제 이상

지도

학번	지도교수	학과
100	이문세	컴퓨터 소프트웨어학과
101	김연아	멀티미디어학과
102	이문세	컴퓨터 소프트웨어학과
103	강승범	경영 정보학과
104	이문세	컴퓨터 소프트웨어학과
105	김연아	멀티미디어학과

103번 학생이 강승범 교수의 지도를 받지 않겠다고 지도 관계를 취소하면 강승범 교수가 경영 정보학과 소속이라는 정보까지 지도 연쇄적으로 삭제됨

제3정규형(계속)

■ 제3정규화 과정의 개념



■ 제3정규화 과정

학번	지도교수	학과	학번	지도교수	지도교수	학과
100	이문세	컴퓨터 소프트웨어학과	100	이문세	이문세	컴퓨터 소프트웨어학과
101	김연아	멀티미디어학과	101	김연아	김연아	멀티미디어학과
102	이문세	컴퓨터 소프트웨어학과	102	이문세	강승범	경영 정보학과
103	강승범	경영 정보학과	103	강승범		
104	이문세	컴퓨터 소프트웨어학과	104	이문세		
105	김연아	멀티미디어학과	105	김연아		

제3정규형(계속)

■ 3NF의 약점

- 복수의 후보키를 가지고 있고,
- 후보키들이 복합 애트리뷰트들로 구성되며,
- 후보키들이 서로 중첩되는 경우

■ 적용 불가능 BCNF \Rightarrow (strong 3NF)

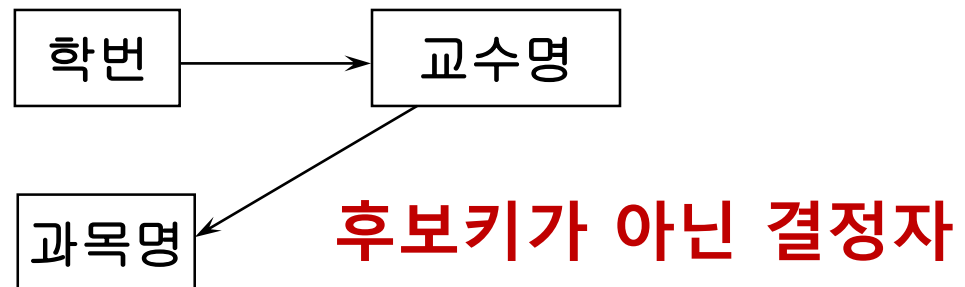
보이스/코드 정규형(BCNF)

■ 보이스/코드 정규형

- 복잡한 식별자 관계에 의해 발생하는 문제를 해결하기 위해서 제3정규형을 보완한 것
- 릴레이션 R이 제3정규형을 만족하고, 모든 결정자가 후보키

수강

학번	과목명	교수명
100	전자계산기구조	이문세
100	데이터베이스	김연아
100	운영체제	강승범
101	데이터베이스	김연아
101	운영체제	전혜영



보이스/코드 정규형(계속)

- 결정자가 후보키가 아닌 릴레이션에서의 갱신 이상
 - 수정이상

학번	과목명	교수명
100	전자계산기구조	이문세
100	데이터베이스	김연아 김주현
100	운영체제	강승범
101	데이터베이스	김연아
101	운영체제	전혜영

정보의 모순성 발생

보이스/코드 정규형(계속)

- 결정자가 후보키가 아닌 릴레이션에서의 갱신 이상
 - 삽입 이상

학번	과목명	교수명
100	전자계산기구조	이문세
100	데이터베이스	김연아
100	운영체제	강승범
101	데이터베이스	김연아
101	운영체제	전혜영
?	딥러닝실습	서경진

서경진이란 교수가 딥러닝실습을 가르친다는 사실만 기록하려하는데, 딥러닝실습을 수강 신청한 학생이 없다면 수강 릴레이션에 이 교수의 정보를 삽입할 수 없음

보이스/코드 정규형(계속)

- 결정자가 후보키가 아닌 릴레이션에서의 갱신 이상
 - 삭제 이상

학번	과목명	교수명
100	전자계산기구조	이문세
100	데이터베이스	김연아
100	운영체제	강승범
101	데이터베이스	김연아
101	운영체제	전해영

101번 학생이 전해영 교수의 운영체제 수업을 수강 취소하기로 하였다면 한 명의 학생만이 수강 신청한 과목을 삭제한 것이기에 운영체제를 전해영 교수가 가르친다는 정보마저 삭제되는 연쇄삭제 현상이 일어나게 되어 정보 손실이 발생

보이스/코드 정규형(계속)

■ 보이스/코드 정규화 과정의 개념



■ 보이스/코드 정규화 과정

학번	과목명	교수명
100	전자계산기구조	이문세
100	데이터베이스	김연아
100	운영체제	강승범
101	데이터베이스	김연아
101	운영체제	전혜영

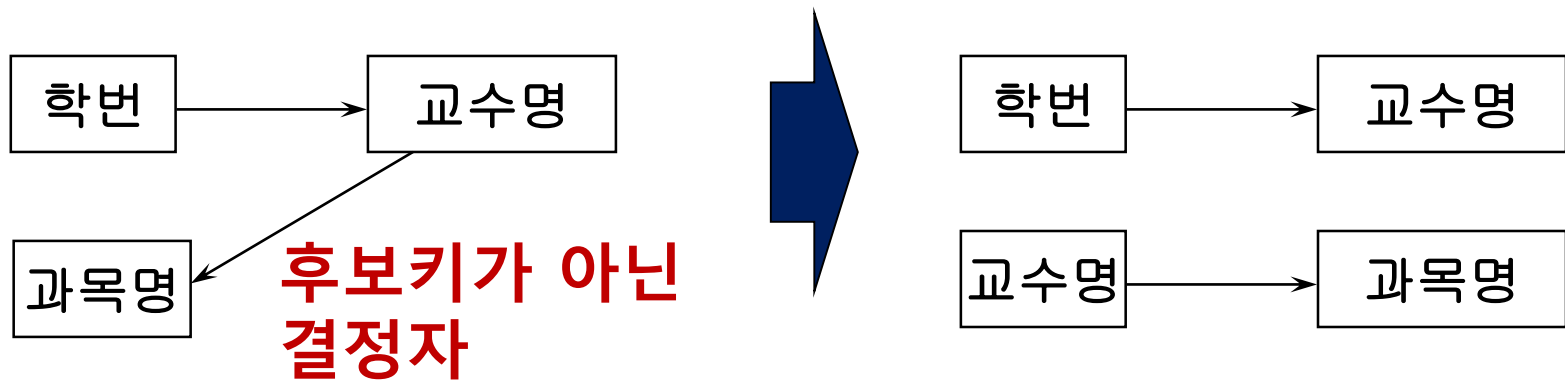


학번	교수명
100	이문세
100	김연아
100	강승범
101	김연아
101	전혜영

교수명	과목명
이문세	전자계산기구조
김연아	데이터베이스
강승범	운영체제
전혜영	운영체제

보이스/코드 정규형(계속)

- 보이스/코드 정규화 과정의 함수 종속 다이어그램
 - 후보키가 아닌 결정자를 제거함



더 많은 종속성과 정규형

- 좋은 데이터베이스 스키마 설계를 위해서는 정규형만으로는 불충분함
 - 예제: 두 개 애트리뷰트를 갖는 릴레이션은 BCNF이다. 그러면, 모든 릴레이션을 두 개 애트리뷰트로 설계하면 좋은 설계인가?
- 좋은 데이터베이스 설계를 보장하기 위해서는 다음의 추가적인 조건들이 필요함
 - 종속성 보존 특성 (dependency preservation property)
 - 무손실 조인 특성 (lossless join property)

다치 종속성과 제4정규형(4NF)

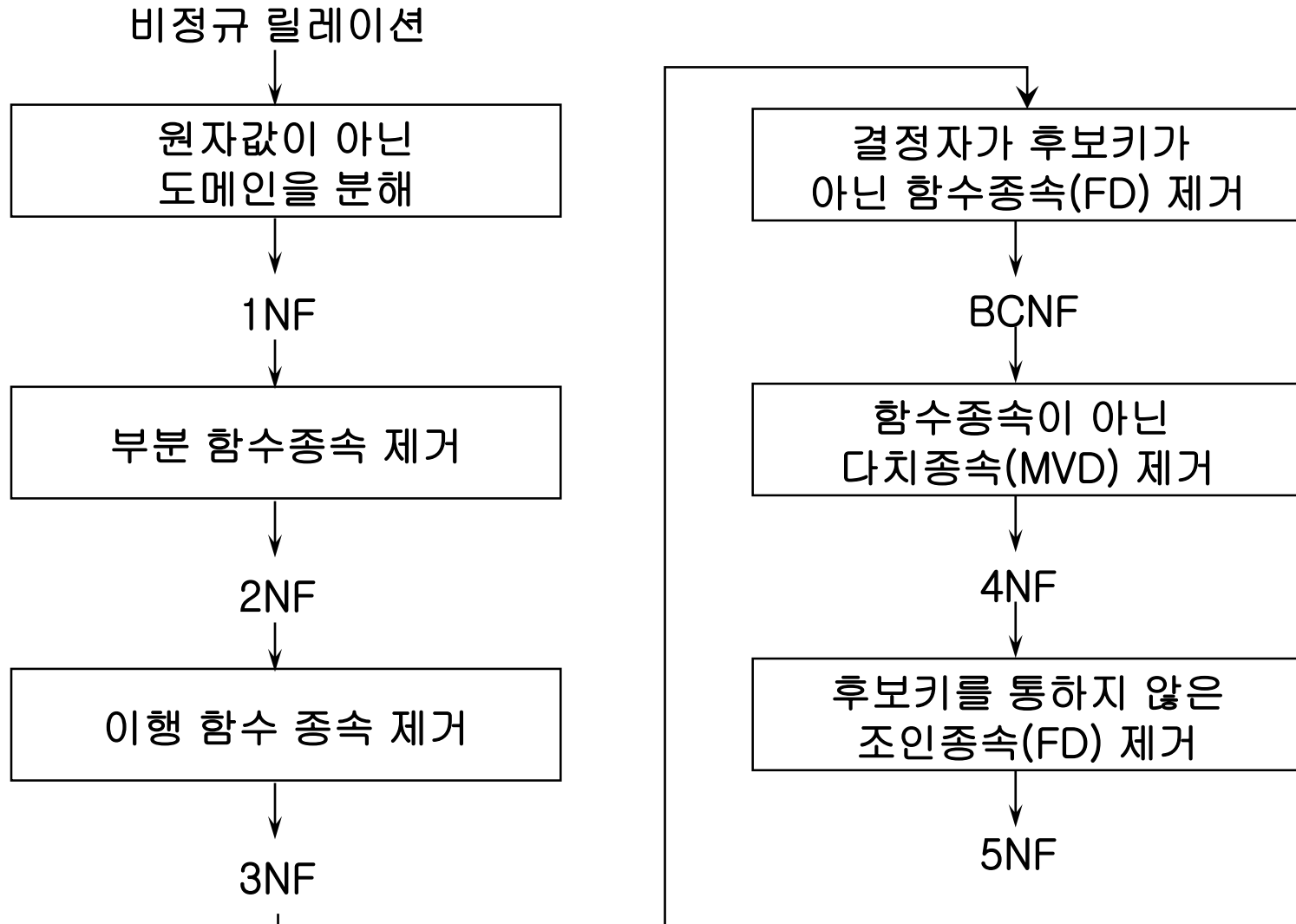
- 함수적 종속성은 하나의 공통된 형태의 제약조건을 명기하기 위해서 사용되며, 함수적 종속성 만에 의해서 명기될 수 없는 다른 형태의 제약조건들이 존재
- 추가적인 종속성에는 다치 종속성(multi-valued dependency)이 있으며, 이에 기반한 정규형이 제4정규형(4NF)임
- 제4정규형의 특성
 - 3NF와 BCNF는 다치 종속성을 다루지 않음
 - 비단순 다치 종속성을 가지는 릴레이션 스키마는 좋은 디자인이 아닐 수 있음
 - 제 4 정규형은 위와 같은 문제를 다루며, BCNF 정규형임
 - 제 4 정규형에 속하는 모든 릴레이션은 BCNF 정규형에 속함

조인 종속성과 제5정규형(5NF)

- 또 다른 추가적 종속성으로 조인 종속성(Join Dependency)이 있으며, 이에 기반한 정규형이 제5정규형임
- 제5정규형(5NF)
 - 함수적 종속성, 다치 종속성, 조인 종속성을 모두 고려하는 정규형으로, 프로젝트-조인 정규형(Project-Join NF: PJNF)이라고도 함
 - 조인 종속성을 발견하는 것은 매우 어려운 일로, 실제로 제5정규형은 거의 쓰이지 않음

보이스/코드 정규형(계속)

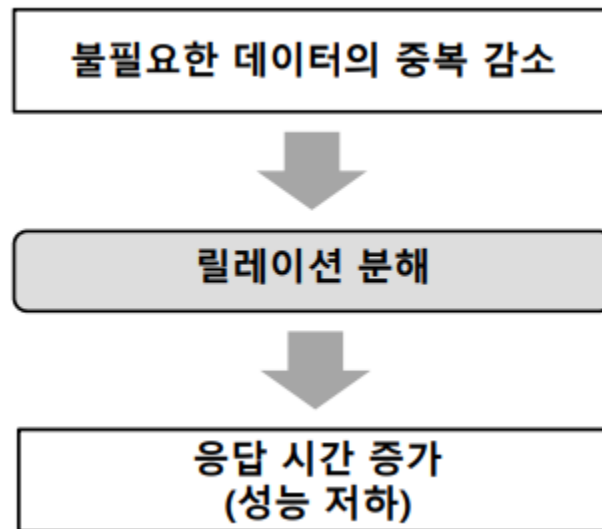
■ 정규형의 특징과 정규화 과정



역정규화

■ 정규화의 문제점

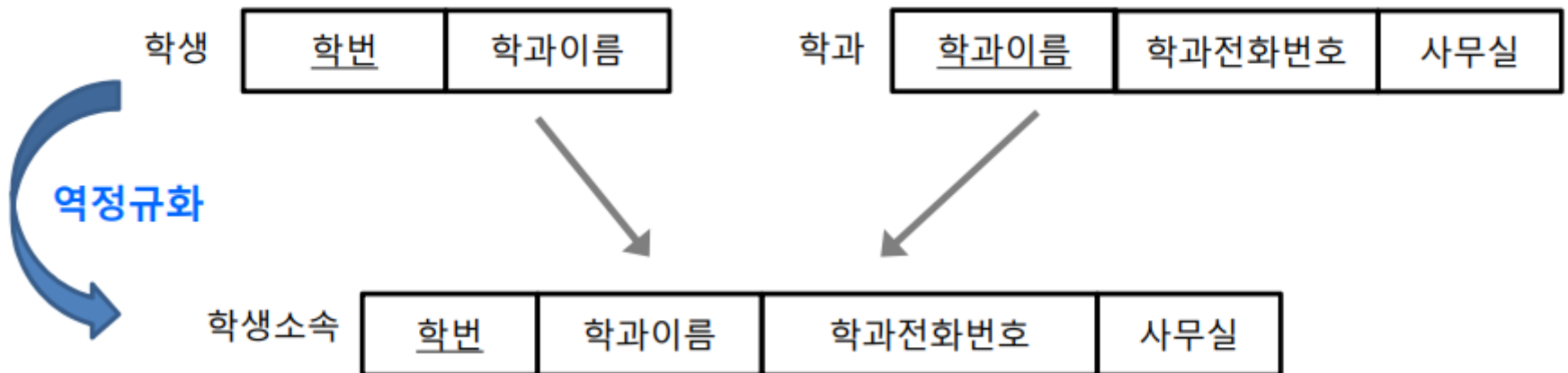
- 정규화 단계가 진행될수록 릴레이션이 분해되므로 원하는 정보를 얻기 위하여 조인의 필요성이 증가
- 즉, 불필요한 데이터의 중복은 감소하지만 질의에 대한 응답 시간은 증가할 수 있음



역정규화(계속)

■ 역정규화(Denormalization)란?

- 성능에 대한 요구를 만족시키기 위해 데이터 중복과 갱신 이상을 대가로 치르면서 보다 낮은 정규형으로 되돌아가는 것임
- 즉, 빈번하게 수행되는 검색 질의의 수행 속도를 높이기 위해서 이미 분해 된 두 개 이상의 릴레이션을 합쳐서 하나의 릴레이션으로 만드는 작업



역정규화(계속)

- 역정규화 하면 중복된 데이터를 관리해야 하므로, 응용 프로그램에서 별도로 데이터 무결성을 보장하기 위한 비용이 발생할 수 있음

데이터의 불필요한 중복 감소 : 응답시간 단축



정규화

역정규화

감사합니다!

담당교수: 전강욱(컴퓨터공학부)

kw.chon@koreatech.ac.kr