


1


## 관계 스키마 설계 시의 지침

# 01 관계 스키마 설계 시의 지침

## 1 설계 지침


 어떤 릴레이션 스키마가 좋고 어떤 릴레이션 스키마가 나쁜지를 판단하기 위한 개략적 지침


- 한 릴레이션에 포함되는 애트리뷰트들
- 튜플들에서 중복되는 값들을 줄임
- 튜플들에서 널 값들을 줄임
- 가짜 튜플(Spurious Tuple)을 허용하지 않음

 여기서는 좋은 릴레이션 설계에 관한 개괄적인 지침을 논한 후, 함수적 종속성과 정규형 개념에 관해 논의

## 01 관계 스키마 설계 시의 지침





### 2 한 릴레이션에 포함되는 애트리뷰트들

 설계자는 데이터베이스에 포함될 전체 애트리뷰트들을 대상으로 실세계에서 어떤 연관성이 있는 애트리뷰트들을 묶어서 하나의 릴레이션 스키마를 만들어 나감

 한 릴레이션에 속하는 애트리뷰트들은 실세계에서 어떤 의미를 가져야 하며, 그에 합당하는 적절한 해석을 가져야 함

## 01 관계 스키마 설계 시의 지침

### 2 한 릴레이션에 포함되는 애트리뷰트들

-  각 릴레이션의 의미가 쉽게 설명될수록 좋은 스키마임
-  릴레이션에 속하는 애트리뷰트들의 의미를 얼마나 잘 쉽게 설명할 수 있는가 하는 것은 릴레이션이 얼마나 잘 설계되었는가를 나타내는 척도임
-  다른 엔티티를 참조하기 위해서는 외래키 만을 사용해야 함
-  여러 엔티티(EMPLOYEE, DEPARTMENT, PROJECT)의 애트리뷰트들이 하나의 릴레이션에 혼합되면 안됨

# 01 관계 스키마 설계 시의 지침

## 2 한 릴레이션에 포함되는 애트리뷰트들

🔍 좋은 스키마의 예

[COMPANY 데이터베이스  
(좋은 스키마)]

EMPLOYEE					외래키
ENAME	<u>SSN</u>	BDATE	ADDRESS	DNUMBER	
기본키					

DEPARTMENT				외래키
DNAME	<u>DNUMBER</u>	DMGRSSN	DLOCATIONS	
기본키				

DEPT_LOCATIONS		외래키
<u>DNUMBER</u>	<u>DLOCATIONS</u>	
기본키		


WORKS_ON			외래키	외래키
<u>SSN</u>	<u>PNUMBER</u>	HOURS		
기본키				

PROJECT				외래키
PNAME	<u>PNUMBER</u>	PLOCATIONS	DNUM	
기본키				

※ 출처 : 데이터베이스 시스템 6판, Elmasri, Navathe 저, 황규영 외 역, 홍릉과학출판사, 2016년

# 01 관계 스키마 설계 시의 지침

## 2 한 릴레이션에 포함되는 애트리뷰트들

 좋은 설계 지침-1 (Guide line-1)

- 의미가 쉽게 전달되도록 릴레이션 스키마를 설계함, 여러 개의 엔티티 타입과 관계 타입의 애트리뷰트들을 섞어서 하나의 릴레이션을 구성하면 안됨
- 직관적으로 하나의 릴레이션 스키마가 하나의 엔티티 타입이나 하나의 관계 타입에 대응된다면, 그 의미를 해석하고 설명하기가 쉬움
- 그렇지 않고, 여러 개의 엔티티 타입과 관계 타입이 한 릴레이션에 섞이게 되면, 의미가 모호하게 되고 릴레이션을 쉽게 설명할 수 없게 됨

# 01 관계 스키마 설계 시의 지침

## 2 한 릴레이션에 포함되는 애트리뷰트들

🔍 나쁜 스키마의 예

- EMP\_DEPT의 각 튜플은 사원에 대한 정보뿐만 아니라, 사원이 일하는 부서의 정보까지 포함함

[COMPANY 데이터베이스 (나쁜 스키마)]

EMP\_DEPT

ENAME	<u>SSN</u>	BDATE	ADDRESS	DNUMBER	DNAME	DMGRSSN
-------	------------	-------	---------	---------	-------	---------

EMP\_PROJ

<u>SSN</u>	<u>PNUMBER</u>	HOURS	ENAME	PNAME	PLOCATIONS
------------	----------------	-------	-------	-------	------------

※ 출처 : 데이터베이스 시스템 6판, Elmasri, Navathe 저, 황규영 외 역, 홍릉과학출판사, 2016년

## 01 관계 스키마 설계 시의 지침

### 2 한 릴레이션에 포함되는 애트리뷰트들

#### 나쁜 스키마의 예

- EMP\_PROJ의 각 튜플은 사원과 프로젝트 사이의 관계를 맺어주는 정보뿐만 아니라 사원과 프로젝트의 정보까지 포함함
- 논리적으로 잘못된 설계는 아니지만 여러 개 엔티티로부터의 애트리뷰트를 섞어 놓았기 때문에 지침-1을 위반한 잘못된 설계라고 판단함



## 01 관계 스키마 설계 시의 지침


### 3 중복된 정보와 갱신 이상

🔍 하나의 릴레이션에 하나 이상의 엔티티의 애트리뷰트들을 혼합하는 것은 여러 가지 문제를 일으킴

🔍 저장공간의 낭비  
: EMPLOYEE, DEPARTMENT 와 EMP\_DEPT의 인스턴스를 비교해 보면, EMP\_DEPT에 정보가 중복 저장되어 저장 공간이 낭비되고 있음을 알 수 있음

## 01 관계 스키마 설계 시의 지침

### 3 중복된 정보와 갱신 이상


 중복의 예

#### EMPLOYEE

ENAME	<u>SSN</u>	BDATE	ADDRESS	DNUMBER
Smith, John B.	123456789	09-JAN-55	731 Fondren, Houston, TX	5
Wong, Franklin T.	333445555	08-DEC-45	638 Voss, Houston, TX	5
Zelaya, Alicia J.	999887777	19-JUL-58	3321 Castle, Spring, TX	4
Wallace, Jennifer S.	987654321	20-JUN-31	291 Berry, Bellaire, TX	4
Narayan, Ramesh K.	666884444	15-SEP-52	975 Fire Oak, Humble, TX	5
English, Joyce A.	453453453	31-JUL-62	5631 Rice, Houston, TX	5
Jabbar, Ahmad V.	987987987	29-MAR-59	980 Dallas, Houston, TX	4
Bong, James E.	888665555	10-NOV-27	731 Stone, Houston, TX	1

## 01 관계 스키마 설계 시의 지침

### 3 중복된 정보와 갱신 이상

 중복의 예

#### DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN
Research	5	333445555
Administration	4	987654321
Headquarters	1	888665555

# 01 관계 스키마 설계 시의 지침

## 3 중복된 정보와 갱신 이상

중복의 예

EMP\_DEPT

ENAME	<u>SSN</u>	BDATE	ADDRESS	DNUMBER	DNAME	DMGRSSN
Smith, John B.	123456789	09-JAN-55	731 Fondren, Houston, TX	5	Research	333445555
Wong, Franklin T.	333445555	08-DEC-45	638 Voss, Houston, TX	5	Research	333445555
Zelaya, Alicia J.	999887777	19-JUL-58	3321 Castle, Spring, TX	4	Administration	987654321
Wallace, Jennifer S.	987654321	20-JUN-31	291 Berry, Bellaire, TX	4	Administration	987654321
Narayan, Ramesh K.	666884444	15-SEP-52	975 Fire Oak, Humble, TX	5	Research	333445555
English, Joyce A.	453453453	31-JUL-62	5631 Rice, Houston, TX	5	Research	333445555
Jabbar, Ahmad V.	987987987	29-MAR-59	980 Dallas, Houston, TX	4	Administration	987654321
Bong, James E.	888665555	10-NOV-27	731 Stone, Houston, TX	1	Headquarters	888665555

## 01 관계 스키마 설계 시의 지침

### 3 중복된 정보와 갱신 이상



갱신 이상의 발생

: 동일한 정보를 한 릴레이션에는 변경하고  
나머지 릴레이션에서는 변경하지 않은 경우  
어느 것이 정확한지 알 수 없게 됨

## 01 관계 스키마 설계 시의 지침

### 3 중복된 정보와 갱신 이상

 갱신 이상(Update anomaly)의 종류

- 삽입 이상 (Insertion anomaly)  
: EMP\_DEPT에 객체를 삽입할 때 부서가 정해지지 않은 직원이나 직원이 없는 부서를 insert 하는데 문제가 발생함
- 삭제 이상 (Deletion anomaly)  
: 부서의 유일한 직원을 삭제하면 부서 정보도 없어짐
- 수정 이상 (Modification anomaly)  
: 부서 정보를 변경하면 부서의 모든 직원 튜플에서 동일하게 변경해야 함

## 01 관계 스키마 설계 시의 지침

### 3 중복된 정보와 갱신 이상

#### 좋은 설계 지침-2(Guide line-2)

- 릴레이션에서 삽입, 삭제, 그리고 수정 이상이 생기지 않도록 기본 릴레이션 스키마를 설계함
- 질의 수행의 효율성을 위해 중복을 허용할 수 밖에 없을 때, 어떤 이상이 존재하면 그것을 잘 이해하여 데이터베이스를 수정하는 프로그램들이 트리거나 자동변경을 수행하는 저장 프로시저를 이용하여 올바르게 동작할 수 있도록 함

# 01 관계 스키마 설계 시의 지침

## 4 튜플의 널 값

- 릴레이션의 튜플들이 널 값을 가지지 않도록 설계해야 함
- 널 값은 저장 단계에서 공간을 낭비하게 하고
- 논리적 차원에서는 조인 연산들을 지정하기 힘들게 하고
- 애트리뷰트들의 의미를 이해하기 어렵게 만듦
- COUNT나 AVG와 같은 집단 함수들이 적용되었을 때 널 값의 처리 방법이 모호해짐




## 01 관계 스키마 설계 시의 지침

### 4 튜플의 널 값

- 🔍 널 값은 다음과 같이 여러 가지로 해석이 가능함
  - 그 애트리뷰트가 이 튜플에는 적용되지 않음
  - 이 튜플에서 애트리뷰트의 값이 아직 알려져 있지 않았음
  - 애트리뷰트 값을 알고 있지만 DB에 기록되지는 않았음
- 🔍 모든 널 값을 동일하게 표현하면 널 값이 갖는 여러 의미를 훼손하게 됨

## 01 관계 스키마 설계 시의 지침

### 4 튜플의 널 값

 널 값의 방지 기법 - 릴레이션의 분리

- 널 값이 많이 나타나는 애트리뷰트들은 별개의 릴레이션으로 분리함
- 예) 사원들 중 10%만 이 개인사무실을 가지고 있는 경우 office\_no 속성을 Employee에 추가하지 말고, Emp\_Office라는 새로운 릴레이션을 만들어서 개인 사무실을 가지고 있는 사원에 대해서만 이 릴레이션에 튜플들을 삽입함

# 01 관계 스키마 설계 시의 지침

## 4 튜플의 널 값

🔍 널 값의 방지 기법 - 릴레이션의 분리

Employee			
ssn	ename	age	office_no

분할

Employee		
ssn	ename	age

Emp_Office	
ssn	office_no

# 01 관계 스키마 설계 시의 지침


## 4 튜플의 널 값


### 좋은 설계 지침-3(Guide line-3)

- 널값을 자주 가질 수 있는 애트리뷰트를 가능한 한 기본 릴레이션의 애트리뷰트로 포함하지 않음
- 만약, 널값을 포함할 수밖에 없다면 예외적인 경우에만 포함하고 대다수 튜플들에 대해서는 널값이 없도록 함

## 01 관계 스키마 설계 시의 지침

### 5 가짜 튜플의 생성

 관계 데이터베이스 설계를 잘못하게 되면, 조인 연산들이 가짜 튜플들을 생성할 수 있음

 조인 연산의 결과가 올바르기 위해서는 릴레이션들이 “무손실 조인(Lossless join) / 비 부가 조인(Non additive join)” 조건을 만족하도록 설계되어야 함

# 01 관계 스키마 설계 시의 지침

## 5 가짜 튜플의 생성

가짜 튜플의 생성 예

학번	과목번호	성적
1	100	A
1	300	B
2	500	B
2	100	C

분할

학번	과목번호
1	100
1	300
2	500
2	100

학번	성적
1	A
1	B
2	B
2	C

조인

학번	과목번호	성적
1	100	A
1	100	B
1	300	A
1	300	B
2	500	B
2	500	C
2	100	B
2	100	C

## 01 관계 스키마 설계 시의 지침

### 5 가짜 튜플의 생성

#### 좋은 설계 지침-4(Guide line-4)

- 가짜 튜플들이 생성되지 않도록 하기 위해 (기본키, 외래키) 애트리뷰트를 가지고 동등 조인할 수 있는 릴레이션 스키마를 설계함
- (외래키, 기본키) 조합을 제외하고 릴레이션들이 대응하는 애트리뷰트들을 가지지 않도록 함, 왜냐하면, 그러한 애트리뷰트에 대한 조인은 가짜 튜플들을 생성하기 때문임

## 2 함수 종속성



## 02 함수 종속성

### 1 개요

- 🔍 함수적 종속성(FD : Functional Dependency)은 좋은 릴레이션 설계의 정형적 기준으로 사용
- 🔍 FD와 키는 릴레이션의 정규형(NF : Normal Form)을 정의하기 위해 사용
- 🔍 FD는 데이터 애트리뷰트들의 의미와 애트리뷰트들 간의 상호 관계로부터 유도되는 제약조건의 일종

## 02 함수 종속성

### 2 FD의 정의 및 검사 방법

- 🔍 X와 Y를 임의의 애트리뷰트들의 집합이라고 할 때, X의 값이 Y의 값을 유일하게 결정한다면 “X는 Y를 함수적으로 결정한다(Functionally Determine)”라고 함
- 🔍 이를  $X \rightarrow Y$ 로 표기하고, “Y는 X에 함수적으로 종속된다” 라고 함
- 🔍 FD는 특정 릴레이션 인스턴스보다는 실세계에서 존재하는 애트리뷰트들 사이의 제약조건으로부터 유도됨

## 02 함수 종속성

### 2 FD의 정의 및 검사 방법

🔍 함수적 종속성은 모든 릴레이션 인스턴스  $r(R)$ 에 대하여 성립해야 함

🔍 함수적 종속성의 검사 방법

- 릴레이션 인스턴스  $r(R)$ 에 속하는 어떠한 임의의 두 튜플에 대해서도 속성들의 집합  $X$ 에 대해 동일한 값을 가질 때마다  $Y$ 에 대해서도 동일한 값을 가진다면  $X \rightarrow Y$ 라는 함수적 종속성이 성립함
- 즉,  $r(R)$ 에서의 임의의 두 튜플  $t_1$ 과  $t_2$ 에 대해  $t_1[X] = t_2[X]$ 이면,  $t_1[Y] = t_2[Y]$ 임

## 02 함수 종속성

### 2 FD의 정의 및 검사 방법

🔍 FD의 예 1)

▪  $R = \{A, B, C, D, E\}$

FD : ①  $A \rightarrow B, D$

②  $B, C \rightarrow E$

A	B	C	D	E
a1	b1	c1	d2	e1
a1	?	c2	?	e2
a2	b2	c4	d4	e3
a1	?	c3	?	e4
a3	b2	c4	d2	?

## 02 함수 종속성

### 2 FD의 정의 및 검사 방법

 FD의 예 2)

- 주민등록번호는 사원의 이름을 결정함
  - $SSN \rightarrow ENAME$
- 프로젝트 번호는 프로젝트 이름과 위치를 결정함
  - $PNUMBER \rightarrow \{PNAME, PLOCATION\}$
- 사원 번호와 프로젝트 번호는 그 사원이 일주일동안 그 프로젝트를 위해서 일하는 시간을 결정함
  - $\{SSN, PNUMBER\} \rightarrow HOURS$

## 02 함수 종속성

### 3 FD의 추론 규칙

- 설계자는 주어진(알려진) FD의 집합  $F$ 를 가지고, 추가로 성립하는 FD들을 추론할 수 있음
- 암스트롱의 추론 규칙들
  - A1. (재귀성 규칙)  $Y \subseteq X$ 이면,  $X \rightarrow Y$
  - A2. (부가성 규칙)  $X \rightarrow Y$ 이면,  $XZ \rightarrow YZ$
  - A3. (이행성 규칙)  $X \rightarrow Y$ 이고  $Y \rightarrow Z$ 이면,  $X \rightarrow Z$

## 02 함수 종속성

### 3 FD의 추론 규칙

🔍 추가적으로 유용한 추론 규칙들

- (분해 규칙)  $X \rightarrow YZ$ 이면,  $X \rightarrow Y$ 이고  $X \rightarrow Z$
- (합집합 규칙)  $X \rightarrow Y$ 이고  $X \rightarrow Z$ 이면,  $X \rightarrow YZ$
- (의사 이행성 규칙)  
 $X \rightarrow Y$ 이고  $WY \rightarrow Z$ 이면,  $WX \rightarrow Z$

🔍  $F^+$  FD 집합  $F$ 의 폐포(Closure)  
:  $F$ 로부터 추론할 수 있는 모든 가능한  
함수적 종속성들의 집합

## 02 함수 종속성

### 3 FD의 추론 규칙



$X^+$  F 하에서 속성 집합

X의 폐포(Closure of X under F)

: 함수적 종속성 집합 F를 사용하여 속성 X에 의해  
함수적으로 결정되는 모든 속성들의 집합



## 02 함수 종속성

### 3 FD의 추론 규칙



알고리즘을 사용하여 F 하에서  
다음과 같은 폐포 집합들을 구할 수 있음

〈알고리즘〉 F 하의 X 의 폐포  $X^+$  를 구하는 알고리즘

$X^+ := X;$

repeat

$\text{old}X^+ := X^+;$

    for each functional dependency  $Y \rightarrow Z$  in F do

        if  $Y \subseteq X^+$  then  $X^+ := X^+ \cup Z;$

until ( $\text{old}X^+ = X^+$ ) ;

## 02 함수 종속성

### 3 FD의 추론 규칙

🔍  $X^+$ 의 예)

FD =  $SSN \rightarrow ENAME$   
 $PNUMBER \rightarrow (PNAME, PLOCATION)$   
 $(SSN, PNUMBER) \rightarrow HOURS$  일 때,

$SSN^+ = \{SSN, ENAME\}$

$PNUMBER^+ = \{PNUMBER, PNAME, PLOCATION\}$



$(SSN, PNUMBER)^+ = \{SSN, PNUMBER, ENAME, PNAME, PLOCATION, HOURS\}$

FD와 Key와의 관계  
: 어떤  $K^+$ 가 R의 모든 속성들을 포함하면 K는 R의 키임

### 3 기본 키 기반의 정규형

## 03 기본 키 기반의 정규형

### 1 정규화(Normalization)

-  불만족스러운 “나쁜” 릴레이션의 애트리뷰트들을 나누어서 더 작은 “좋은” 릴레이션으로 분해하는 과정
-  중복과 갱신 이상을 최소화하기 위해 FD와 기본키를 기반으로, 릴레이션 스키마를 분석하여 더 작은 릴레이션들로 분해하는 과정

## 03 기본 키 기반의 정규형

### 1 정규화(Normalization)



정규형(Normal form)

: 특정 조건을 만족하는 릴레이션 스키마의 형태

- 제 2 정규형, 제 3 정규형, BCNF  
: 한 릴레이션의 속성들 사이의  
함수적 종속성과 기본키를 기반으로 정의
- 제 4 정규형 : 다치 종속성을 기반으로 정의
- 제 5 정규형 : 조인 종속성을 기반으로 정의

## 03 기본 키 기반의 정규형

### 2 제 1 정규형 (1-NF: First Normal Form)

- 🔍 관계 모델(Relational Model)의 엄격한 정의의 일부분
- 🔍 다치 속성, 복합 속성, 그리고 이들의 조합을 금지함
- 🔍 1-NF는 속성의 도메인이 오직 원자값 만을 포함하고, 튜플의 모든 속성 값은 그 속성 도메인에 속하는 하나의 값이어야 함을 의미함

## 03 기본 키 기반의 정규형

### 2 제 1 정규형 (1-NF: First Normal Form)

🔍 1-NF의 예

스키마 : 1NF가 아님

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	{ DLOCATIONS }
-------	----------------	---------	----------------

관계 모델의 정의에 위배

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	{ DLOCATIONS }
Research	4	333445555	{Bellaire, Sugarland, Houston}
Administration	5	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

## 03 기본 키 기반의 정규형

### 2 제 1 정규형 (1-NF: First Normal Form)

🔍 1-NF의 예

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	DLOCATIONS
Research	4	333445555	Bellaire
Research	4	333445555	Sugarland
Research	4	333445555	Houston
Administration	5	987654321	Stafford
Headquarters	1	888665555	Houston

기본키 제약조건 위배



### 03 기본 키 기반의 정규형

## 2 제 1 정규형 (1-NF: First Normal Form)

🔍 1-NF의 예

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	{ DLOCATIONS }
-------	----------------	---------	----------------

분할

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN
-------	----------------	---------

Research	4	333445555
Administration	5	987654321
Headquarters	1	888665555

PK=-{DNUMBER}

DEPARTMENT\_LOCATIONS

<u>DNUMBER</u>	<u>DLOCATION</u>
----------------	------------------

4	Bellaire
4	Sugarland
4	Houston
5	Stafford
1	Houston

PK=-{DNUMBER, DLOCATION}, FK={DNUMBER}