

**CSE235**

**데이터베이스 시스템  
(Database Systems)**

**Lecture 06: 관계데이터 모델**

**담당교수: 전강욱(컴퓨터공학부)**

**kw.chon@koreatech.ac.kr**

# 지난시간 복습

## ■ 데이터 모델링의 개념 학습

- 현실 세계에 존재하는 데이터를 컴퓨터의 DB로 옮기는 변환과정
- 개념적 구조와 논리적 구조를 거쳐서 실제로 디스크에 저장할 수 있는 물리적 구조로 변환하는 과정을 의미

## ■ 개념적 데이터 모델링 학습 및 대표적인 모델 확인

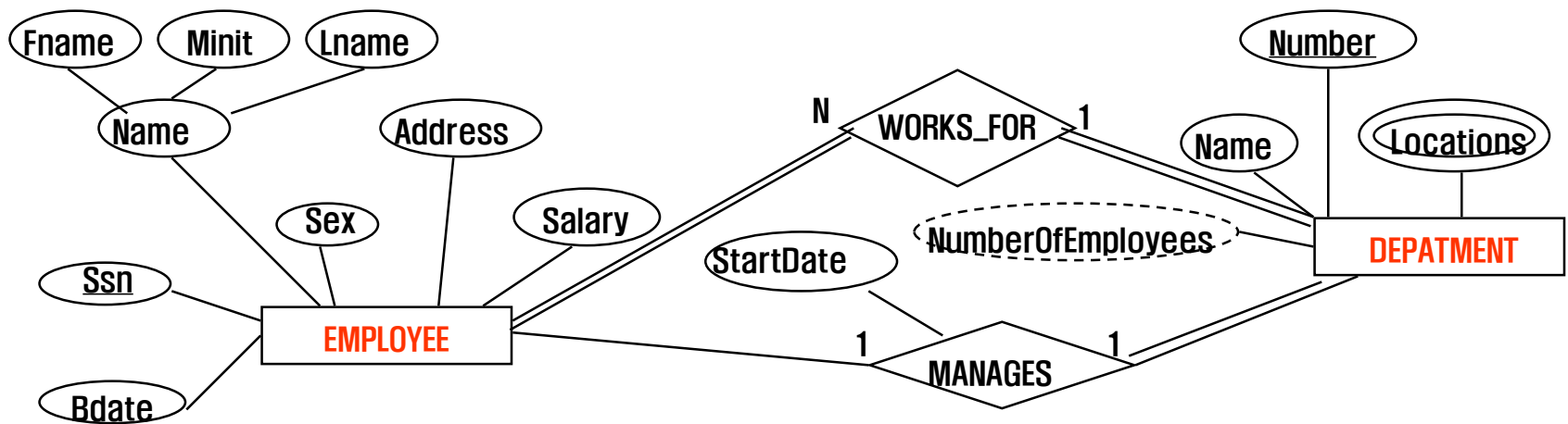
- DB를 개념적 구조로 표현하는 도구
- 대표적인 모델: ER 모델

## ■ 논리적 데이터 모델링 학습 및 대표적인 모델 확인

- DB를 논리적 도구로 표현하는 도구이며, "레코드"와 "관계"라는 논리적 개념으로 구성
- 대표적인 모델: 관계 데이터 모델, 네트워크 데이터 모델, 계층형 데이터 모델

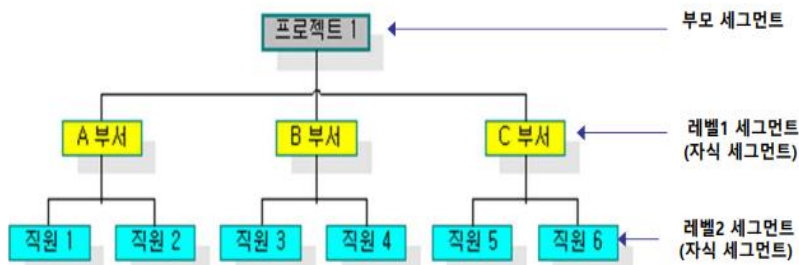
# 지난시간 복습 (계속)

- 개체관계 (Entity-Relationship Model, ER Model) 모델과 개체관계 다이어그램(ER Diagram) 학습
  - 개체관계 모델: 개체와 개체 간 관계를 이용해 현실 세계를 개념적 구조로 표현
  - 개체관계 다이어그램: ER 모델의 결과물을 그림으로 표현한 것

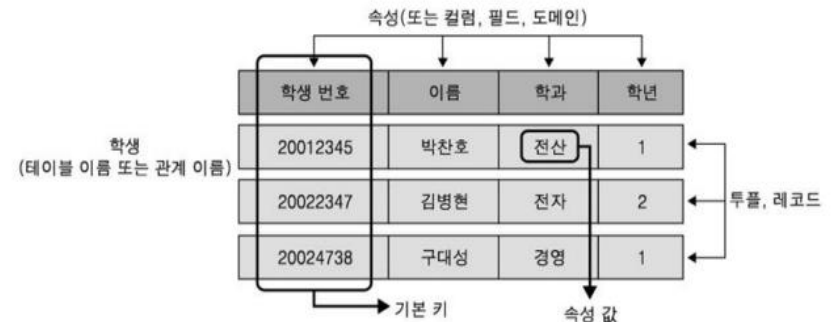


# 지난시간 복습 (계속)

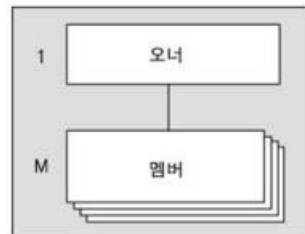
- 논리적 데이터 모델은 개념적 모델을 DB로 구현하기 위한 중간 단계로 "레코드"와 "관계"라는 논리적인 개념
- DBMS는 하나의 논리적 데이터 모델을 기반으로 개발
  - DBMS는 논리적 데이터 모델을 이해하는 것이 가능



계층 데이터 모델: 논리적 구조가 Tree 형태.



관계형 데이터 모델: 논리적 구조가 테이블 형태.



네트워크 데이터 모델: 논리적 구조가 그래프 형태.

# 복습 문제

- 데이터 모델의 구성 요소가 아닌 것은?
  - 논리적으로 표현된 데이터 구조
  - 구성 요소의 연산
  - 구성 요소의 제약 조건
  - 물리적 저장 구조
- 데이터 모델은 현실 세계를 데이터베이스로 표현하는 과정에서 개념적인 구조, 논리적인 구조, 물리적인 구조를 표현하기 위해 사용된다. ( )
- 개념적 데이터 모델은 속성들로 기술된 개체 타입과 이 개체 타입들 간의 관계를 이용하여 현실 세계를 표현하는 방법이다. ( )
- 논리적 데이터 모델은 필드로 기술된 데이터 타입과 이 데이터 타입들 간의 관계를 이용하여 현실 세계를 표현하는 방법이다. ( )

# 복습 문제 (계속)

- 데이터베이스의 논리적 구조가 트리 형태로 이루어진 데이터 모델은 (    ) 이다.
- 일반적으로 많이 사용되는 논리적 데이터 모델로 개체 집합에 대한 속성 관계를 표현하기 위해 개체를 2차원 테이블 형태로 사용하는 데이터 모델은 이다.
- 데이터베이스 설계 순서를 기술하시오.
- 속성(attribute)의 수를 cardinality 라고 한다. (    )
- ER 모델에서 속성을 나타내는 방법은?
- 데이터베이스 설계 단계 중 아래는 무엇과 관련이 있는가? (    )
  - 파일 조작 방법, 저장 방법, 파일 접근 방법 등을 선정

# 복습 문제 (계속)

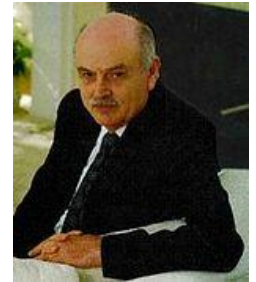
- P. Chen이 제안한 현실 세계에 존재하는 객체들과 그들의 관계를 사람이 이해하기 쉽게 표현한 모델은 (    ) 이다.
- 데이터의 가장 작은 논리적 단위로서 데이터 항목 또는 데이터 필드에 해당하는 것은 (    ) 이다.
- 데이터 모델을 구성하는 3가지는 (    ) 이다.

# 개요

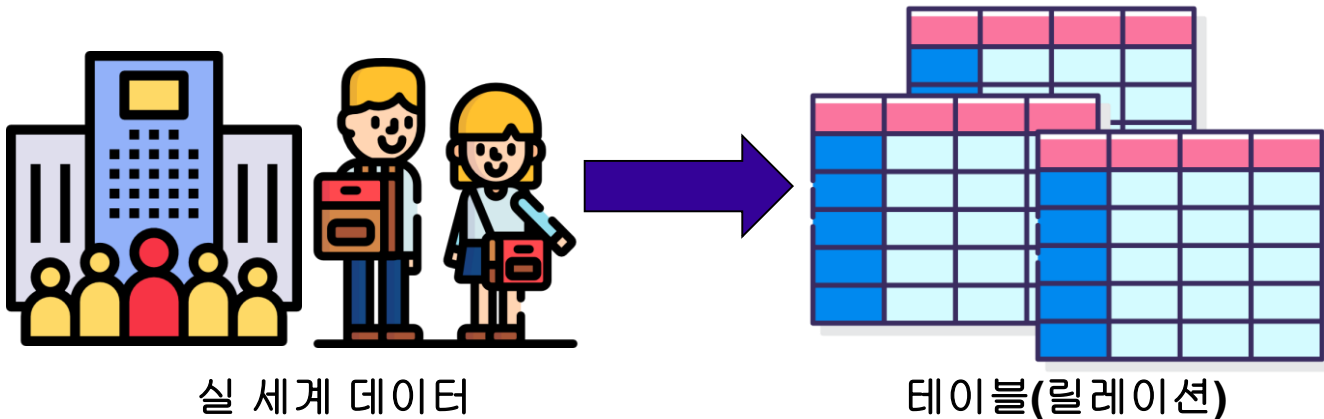
- 관계 데이터 모델
  - 정의, 용어 등
- 릴레이션
- 관계 데이터 제약조건
- 요약



# 관계 데이터 모델이란?



- 개념적 구조를 논리적 구조로 표현하는 논리적 모델
  - IBM의 E.F. Codd가 1972년 제안한 데이터 모델
  - 대부분 DBMS에서 지원하는 데이터 모델
- 실세계 데이터를 행과 열로 구성된 표(테이블, 릴레이션) 형태로 저장
  - 하나의 개체에 대한 데이터를 하나의 릴레이션에 저장
- 선언적인 질의어(query)를 통한 데이터 접근을 제공
  - 사용자는 원하는 데이터만을 명시하여 데이터 접근
  - 한 테이블의 필드 값을 이용하여 다른 테이블에 관련된 데이터를 찾는 방식
    - 링크나 포인터 불필요



# 관계 데이터 모델의 기본 용어

## ■ 테이블 → 릴레이션(relation)

- 하나의 개체에 대한 데이터를 2차원 테이블 구조로 저장
- 파일 관리 시스템 관점에서 파일(file)에 대응

## ■ 행 → 튜플

- 릴레이션의 행
- 파일 관리 시스템 관점에서 레코드(record)에 대응

## ■ 열 → 애트리뷰트

- 릴레이션의 열
- 파일 관리 시스템 관점에서 필드(field)에 대응

테이블

<i>customer-name</i>	<i>customer-street</i>	<i>customer-city</i>
<i>Jones</i>	Main	Harrison
<i>Smith</i>	North	Rye
<i>Curry</i>	North	Rye
<i>Lindsay</i>	Park	Pittsfield

열(속성, 애트리뷰트)

행(튜플)

*Customer* 릴레이션

# 관계 데이터 모델의 기본 용어 (계속)

## ■ 차수(degree)

- 하나의 릴레이션에서 속성의 전체 개수
  - 최소 차수: 1
- 릴레이션의 차수는 자주 바뀌지 않음

## ■ 카디널리티(cardinality)

- 하나의 릴레이션에서 튜플 전체 개수
  - 유효 카디널리티: 0부터

## ■ 널(null)

- 속성 값을 아직 모르거나 해당되는 값이 없음

# 관계 데이터 모델의 기본 용어 (계속)

## ■ 도메인

- 이름, 데이터 타입 또는 포맷, 값 해석을 위한 추가정보(e.g., 측정단위)로 정의
- 원자 값들(atomic values)의 집합
- 도메인의 예
  - USA\_phone\_numbers: 미국에서 사용하는 10자리 전화번호들의 집합
  - Names: 개인의 이름들의 집합
  - Employee\_ages: 15~80 사이의 사원들의 나이
- 각 도메인을 위해 데이터 타입 또는 포맷을 명시하기도 함
  - e.g., USA\_phone\_numbers 도메인을 위한 데이터 타입은 (ddd)ddd-dddd로 명시

## ■ 데이터 타입

- 도메인은 실제 데이터 타입으로 명시함
- 데이터 타입의 예: string, integer, real

# 릴레이션의 구성

## ■ 릴레이션 스키마(relation schema)

- 릴레이션의 논리적 구조
- 릴레이션 이름  $R$ 과 속성  $A_i$ 들의 집합
  - $R(A_1, A_2, \dots, A_n)$ 로 표기
- 릴레이션의 차수(degree): 릴레이션의 속성의 수
- e.g., EMPLOYEE(Name, BirthDate, Address)의 차수=3

## ■ 릴레이션 인스턴스(relation instance)

- 어느 한 시점에 릴레이션에 존재하는 튜플들의 집합
  - $r = r(R) = \{t_1, t_2, \dots, t_m\}$

# 릴레이션 구성(계속)

속성(애트리뷰트)

<i>customer-name</i>	<i>customer-street</i>	<i>customer-city</i>
<i>Jones</i>	Main	Harrison
<i>Smith</i>	North	Rye
<i>Curry</i>	North	Rye
<i>Lindsay</i>	Park	Pittsfield

릴레이션  
스키마

릴레이션  
인스턴스  
(튜플)

# 데이터베이스의 구성

- **데이터베이스 스키마(database schema)**
  - 데이터베이스의 전체 구조
  - 데이터베이스를 구성하는 릴레이션 스키마의 모음
- **데이터베이스 인스턴스(database instance)**
  - 데이터베이스를 구성하는 릴레이션 인스턴스의 모음

# 릴레이션의 특성

## ■ 릴레이션은 튜플들의 집합

- 튜플의 유일성: 하나의 릴레이션에는 **동일한 튜플이 존재할 수 없음**
- 튜플의 무순서: 하나의 릴레이션에서 **튜플 사이의 순서는 무의미함**
- 속성의 무순서: 하나의 릴레이션에서 **속성 사이의 순서는 무의미함**
- 속성의 원자성: 속성 값으로 **원자 값만 사용 가능함**

FNAME	MINT	LNAME	SSN	BDATE	ADDRESS	SEX
John	B	Smith	123456789	09-Jan-55	Houston	M
Franklin	T	Wong	333445555	08-Dec-45	Houston	M
Alicia	J	Zelaya	999887777	19-Jul-58	Spring	F
Jennifer	S	Wallace	987654321	20-Jun-31	Bellaire	F
Ramesh	K	Narayn	666884444	15-Sep-52	Oak	M
Joyce	A	English	453453453	31-Jul-62	Houston	F
Ahmad	V	Jabbar	987987987	29-Mar-59	Houston	M
James	E	Borg	888665555	10-Nov-27	Houston	M

**튜플의 순서 상이 → 동일한 릴레이션**

FNAME	MINT	LNAME	SSN	BDATE	ADDRESS	SEX
Ramesh	K	Narayn	666884444	15-Sep-52	Oak	M
Joyce	A	English	453453453	31-Jul-62	Houston	F
Ahmad	V	Jabbar	987987987	29-Mar-59	Houston	M
James	E	Borg	888665555	10-Nov-27	Houston	M
John	B	Smith	123456789	09-Jan-55	Houston	M
Franklin	T	Wong	333445555	08-Dec-45	Houston	M
Alicia	J	Zelaya	999887777	19-Jul-58	Spring	F
Jennifer	S	Wallace	987654321	20-Jun-31	Bellaire	F



# 릴레이션의 특성 (계속)

- 다중 값 속성을 포함하는 릴레이션 예제
  - 나눌 수 없는 원자 값들(atomic values)이 아님
  - ER 모델에서의 **다치 속성이나 복합 속성은 관계모델에서 허용되지 않음** → 제1정규형의 만족
    - ER 모델에서의 다치 애트리뷰트나 복합 애트리뷰트는 관계모델에서 허용되지 않으며, 별도의 릴레이션으로 표현되어야 함

<i>customer-name</i>	<i>customer-street</i>	<i>customer-city</i>
<i>Jones</i>	Main	Harrison
<i>Smith</i>	North	Rye
<i>Curry</i>	North	Rye; Harrison
<i>Lindsay</i>	Park	Pittsfield

# 릴레이션의 키

- 릴레이션에서 튜플들을 유일하게 구별하는 속성 또는 속성들의 집합
- 키의 종류
  - 기본키(primary key)
  - 후보키(candidate key)
  - 수퍼키(super key)
  - 외래키(foreign key)
  - 대체키(alternative key)

# 릴레이션의 키 (계속)

## ■ 키의 특성

- 유일성(uniqueness): 하나의 릴레이션에서 모든 튜플들은 서로 다른 키 값을 가져야 함
- 최소성(minimality): 꼭 필요한 최소한의 속성들로만 키를 구성해야 함

# 릴레이션의 키 (계속)

## ■ 슈퍼키(super key)

- 유일성을 만족하는 속성 또는 속성들의 집합
- e.g., EMPLOYEE 릴레이션의 슈퍼키: SSN, (SSN,BDATE) 등

## ■ 후보키(candidate key)

- 유일성과 최소성을 만족하는 속성 또는 속성들의 집합
- e.g., EMPLOYEE 릴레이션의 후보키: SSN, FNAME 등

FNAME	MINT	LNAME	<u>SSN</u>	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
John	B	Smith	123456789	09-Jan-55	Houston	M	30000	333445555	5
Franklin	T	Wong	333445555	08-Dec-45	Houston	M	40000	888665555	5
Alicia	J	Zelaya	999887777	19-Jul-58	Spring	F	25000	987654321	4
Jennifer	S	Wallace	987654321	20-Jun-31	Bellaire	F	43000	888665555	4
Ramesh	K	Narayn	666884444	15-Sep-52	Oak	M	38000	333445555	5
Joyce	A	English	453453453	31-Jul-62	Houston	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	29-Mar-59	Houston	M	25000	987654321	4
James	E	Borg	888665555	10-Nov-27	Houston	M	55000	NULL	1

EMPLOYEE 릴레이션.

# 릴레이션의 키 (계속)

- 기본키(primary key)

- 후보키 중에서 기본적으로 사용하기 위해 선택한 키
- e.g., EMPLOYEE 릴레이션의 기본키: SSN

- 대체키(alternate key)

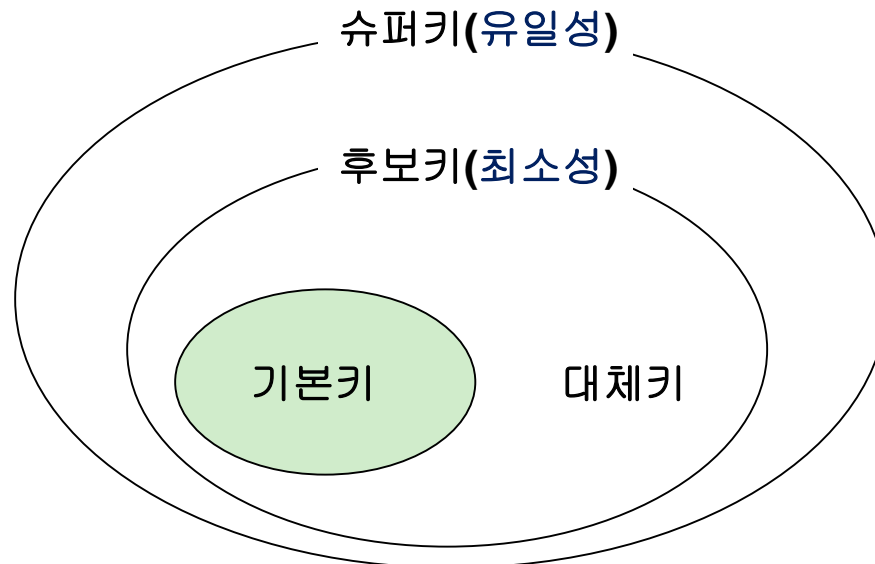
- 기본키로 선택되지 못한 후보키
- e.g., EMPLOYEE 릴레이션의 후보키: FNAME

# 릴레이선의 키 (계속)

## ■ 기본키 선택시 고려사항

- 기본키는 널(NULL) 값을 불허함
- 기본키는 자주 변경되지 않는 속성이 적합함
- 단순한 후보키가 기본키로 적합함

## ■ 키들의 포함관계



# 릴레이션의 키 (계속)

## ■ 키의 종류

### □ 외래키(foreign key)

- 다른 릴레이션의 기본키를 참조하는 속성 또는 속성들의 집합
- 릴레이션들 간의 관계를 표현
  - 참조하는 릴레이션: 외래키를 포함하는 릴레이션
  - 참조되는 릴레이션: 외래키가 참조하는 기본키를 포함하는 릴레이션

참조

외래키

기본키

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE
Research	5	333445555	22-May-78
Administration	4	987654321	01-Jan-85
Headquarters	1	888665555	19-Jun-71

FNAME	MINT	LNAME	<u>SSN</u>	BDATE
John	B	Smith	123456789	09-Jan-55
Franklin	T	Wong	333445555	08-Dec-45
Alicia	J	Zelaya	999887777	19-Jul-58
Jennifer	S	Wallace	987654321	20-Jun-31
Ramesh	K	Narayn	666884444	15-Sep-52
Joyce	A	English	453453453	31-Jul-62
Ahmad	V	Jabbar	987987987	29-Mar-59
James	E	Borg	888665555	10-Nov-27

# 릴레이션의 키 (계속)

- 외래키 속성과 참조하는 기본키의 속성의 이름은 달라도 도메인은 같아야 함
  - e.g., MGRSSN과 SSN의 속성의 이름은 다르나, 동일 도메인을 포함

참조

외래키

기본키

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE
Research	5	333445555	22-May-78
Administration	4	987654321	01-Jan-85
Headquarters	1	888665555	19-Jun-71

FNAME	MINT	LNAME	SSN	BDATE
John	B	Smith	123456789	09-Jan-55
Franklin	T	Wong	333445555	08-Dec-45
Alicia	J	Zelaya	999887777	19-Jul-58
Jennifer	S	Wallace	987654321	20-Jun-31
Ramesh	K	Narayn	666884444	15-Sep-52
Joyce	A	English	453453453	31-Jul-62
Ahmad	V	Jabbar	987987987	29-Mar-59
James	E	Borg	888665555	10-Nov-27



# 릴레이션의 키 (계속)

- 하나의 릴레이션에 외래키가 여러 개 포함 가능하며, 외래키가 기본키가 될 수 있음

<u>ESSN</u>	<u>PNO</u>	HOURS
123456789	1	32.5
123456789	2	7.5
666884444	3	40
453453453	1	20
453453453	2	20
333445555	2	10
333445555	3	10
333445555	10	10
333445555	20	10
999887777	30	30
999887777	10	10
987987987	10	35
987987987	30	5
987654321	30	20
987654321	20	15
888665555	20	null

<u>FNAME</u>	<u>MINT</u>	<u>LNAME</u>	<u>SSN</u>	<u>BDATE</u>
John	B	Smith	123456789	09-Jan-55
Franklin	T	Wong	333445555	08-Dec-45
Alicia	J	Zelaya	999887777	19-Jul-58
Jennifer	S	Wallace	987654321	20-Jun-31
Ramesh	K	Narayn	666884444	15-Sep-52
Joyce	A	English	453453453	31-Jul-62
Ahmad	V	Jabbar	987987987	29-Mar-59
James	E	Borg	888665555	10-Nov-27

<u>PNAME</u>	<u>PNUMBER</u>	<u>PLOCATION</u>	<u>DNUM</u>
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

# 릴레이선의 키 (계속)

- 같은 릴레이선의 기본키를 참조하는 외래키도 정의 가능
- 외래키 속성은 널 값 허용



FNAME	MINT	LNAME	<u>SSN</u>	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
-------	------	-------	------------	-------	---------	-----	--------	----------	-----

# 관계 데이터 모델의 제약

## ■ 무결성 제약조건(integrity constraint)

- 데이터의 무결성을 보장
- 데이터베이스의 상태를 일관되게 유지하기 위한 규칙
- 일관된 데이터베이스 상태를 정의하는 규칙들을 묵시적으로 또는 명시적으로 정의
- 어느 시점에 데이터베이스의 상태 또는 인스턴스가 항상 지켜야 하는 중요한 규칙

## ■ 무결성(integrity)

- 데이터를 결함이 없는 상태로 유지

# 관계 데이터 모델의 제약 (계속)

- 개체 무결성 제약조건(entity integrity constraint)
  - 기본키를 구성하는 모든 속성은 **널 값 불허**
- 참조 무결성 제약조건(referential integrity constraint)
  - 외래키는 **참조할 수 없는 값을 가질 수 없음**

# 개체 무결성 제약조건

- 기본키를 구성하는 모든 속성은 널 값을 가질 수 없는 규칙
- 사용자는 데이터 정의문(data definition language)에서 어떤 속성이 릴레이션의 기본 키인지 미리 정함
- e.g., 개체무결성 제약조건을 위반한 릴레이션

FNAME	MINT	LNAME	<u>SSN</u>	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
John	B	Smith	123456789	09-Jan-55	Houston	M	30000	333445555	5
Franklin	T	Wong	333445555	08-Dec-45	Houston	M	40000	888665555	5
Alicia	J	Zelaya	999887777	19-Jul-58	Spring	F	25000	987654321	4
Jennifer	S	Wallace	987654321	20-Jun-31	Bellaire	F	43000	888665555	4
Ramesh	K	Narayn	666884444	15-Sep-52	Oak	M	38000	333445555	5
Joyce	A	English	NULL	31-Jul-62	Houston	F	25000	333445555	5
Ahmad	V	Jabbar	NULL	29-Mar-59	Houston	M	25000	987654321	4
James	E	Borg	888665555	10-Nov-27	Houston	M	55000	NULL	1

# 참조 무결성 제약조건

- 외래키는 참조할 수 없는 값을 가질 수 없는 규칙
- 참조 무결성 제약조건은 두 릴레이션 간 연관된 튜플들 사이의 일관성을 유지하는데 사용
- e.g., WORKS\_ON 릴레이션이 PROJECT 릴레이션을 참조하는 경우
  - PROJECT 릴레이션의 PNUMBER 속성이 기본키
  - WORKS\_ON 릴레이션의 PNO 속성이 외래키

**WORKS\_ON**

외래키(FK)

ESSN	PNO	HOURS
123456789	1	32.5
123456789	2	7.5
666884444	3	40
453453453	1	20
453453453	2	20
333445555	2	10
333445555	3	10

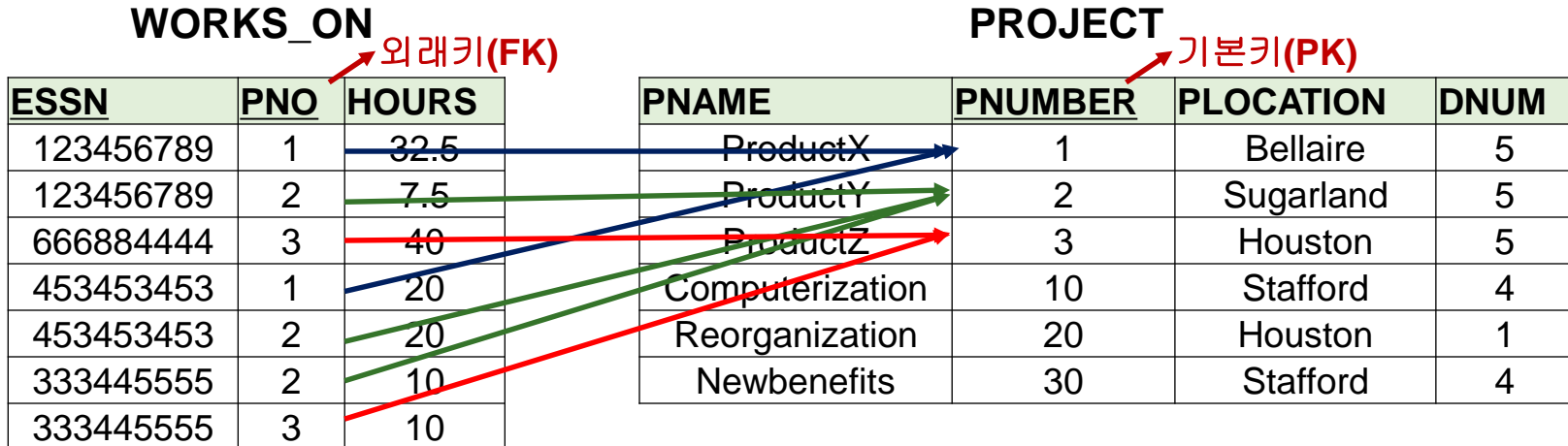
**PROJECT**

기본키(PK)

PNAME	PNUMBER	PLOCATION	DNUM
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

# 참조 무결성 제약조건

- 외래키는 참조할 수 없는 값을 가질 수 없는 규칙 (널 값은 가능)
- 참조 무결성 제약조건은 두 릴레이션 간 연관된 튜플들 사이의 일관성을 유지하는데 사용
- e.g., WORKS\_ON 릴레이션이 PROJECT 릴레이션을 참조하는 경우
  - PROJECT 릴레이션의 PNUMBER 속성이 기본키
  - WORKS\_ON 릴레이션의 PNO 속성이 외래키



# 참조 무결성 제약조건 (계속)

## ■ 외래키가 널 값인 릴레이션?

- 외래키에 널 값은 허용되므로, 참조 무결성 제약조건을 위배한 것이 아님

WORKS\_ON

외래키(FK)

ESSN	PNO	HOURS
123456789	1	32.5
123456789	2	7.5
666884444	3	40
453453453	1	20
453453453	2	20
333445555	2	10
333445555	3	10
333445555	10	10
333445555	NULL	10
999887777	NULL	30
999887777	NULL	10
987987987	10	35

PROJECT

기본키(PK)

PNAME	PNUMBER	PLOCATION	DNUM
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4



# 참조 무결성 제약조건 (계속)

## ■ 참조 무결성 제약조건 관련 유형들

### □ 새로운 튜플의 삽입

- 참조되는 릴레이션에 새로운 튜플 삽입 OK
- 참조하는 릴레이션에 새로운 튜플 삽입 시 제약조건 위배 여부 확인
- e.g., WORKS\_ON 릴레이션에 인스턴스 추가 시 PNO 속성 값은 반드시 PROJECT 릴레이션의 PNUMBER 속성에 존재하는 값이어야 함
- e.g., WORKS\_ON 릴레이션에 튜플 (333445555, 1000, 20) 삽입하면 참조 무결성 제약조건에 위배

WORKS\_ON

ESSN	PNO	HOURS
123456789	1	32.5
123456789	2	7.5
666884444	3	40
453453453	1	20
453453453	2	20
333445555	2	10
333445555	3	10
333445555	10	10

외래키(FK)

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

기본키(PK)

# 참조 무결성 제약조건 (계속)

## ■ 참조 무결성 제약조건 관련 유형들

### □ 튜플의 삭제

- 참조하는 릴레이션의 튜플을 삭제하는 경우 허용
- 참조되는 릴레이션의 튜플을 삭제하는 경우 제약조건 위배 여부 확인
- e.g., PROJECT 릴레이션에서 삭제하고자 하는 튜플을 참조하는 튜플이 WORKS\_ON에 존재하는 경우
  - 삭제하지 않거나, WORKS\_ON의 관련 튜플도 함께 삭제
  - 또는, WORKS\_ON의 관련 튜플의 외래키 속성 값을 널 값으로 설정

WORKS\_ON

외래키(FK)

ESSN	PNO	HOURS
123456789	4	32.5
123456789	2	7.5
666884444	3	40
453453453	4	20
453453453	2	20
333445555	2	10
333445555	3	10
333445555	10	10

PROJECT

기본키(PK)

PNAME	PNUMBER	PLOCATION	DNUM
ProductX	4	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

# 요약

- **관계 모델의 개념**

- 용어 정의(릴레이션, 튜플, 애트리뷰트 등)

- **관계 제약조건과 관계형 데이터베이스 스키마**

- 키 제약조건, 무결성 제약조건

- **갱신 연산과 제약조건의 위반 처리**

- 릴레이션에 튜플을 삽입하거나, 삭제할 때 제약조건을 만족하는지 검사

# 감사합니다!

담당교수: 전강욱(컴퓨터공학부)

[kw.chon@koreatech.ac.kr](mailto:kw.chon@koreatech.ac.kr)