

데이터베이스

강의 노트

제 6 회차
관계 데이터 모델

❖ 학습목표

- 관계 데이터 모델의 특징을 설명할 수 있다.
- 관계 데이터 모델의 주요 용어를 설명할 수 있다.
- 릴레이션의 특징을 나열할 수 있다.
- 키(Key)의 종류와 특징을 나열할 수 있다.
- 무결성 제약조건을 3가지로 구분해서 설명할 수 있다.

❖ 학습내용

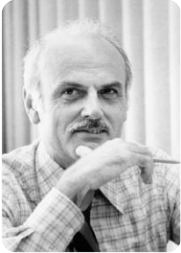
- 관계 데이터 모델 이해
- 키(Key)와 무결성 제약조건

관계 데이터 모델 이해

1. 관계 데이터 모델이란?
2. 주요 용어들
3. 릴레이션의 개념
4. 릴레이션의 특징

1. 관계 데이터 모델이란?

1) 관계 데이터 모델의 정의



관계 데이터 모델이란?

- DB를 2차원 테이블(Table) 즉, 릴레이션(Relation)의 집합으로 모델링하는 논리적 데이터 구조
- 1970년 E. F. Codd 박사가 처음 제안한 모델
- 현재 가장 많은 DBMS가 기초로 하고 있는 논리적 데이터 모델

E. F. Codd 박사

예) 사원 개체를 표현한 관계 데이터 모델

사번	이름	입사일	호봉	휴대폰
1101	김정아	1995-03-01	20	010-1111-2222
1102	이기원	1997-01-01	18	011-2222-3333
1103	박영종	2005-09-01	15	010-3333-4444
1201	최강희	1994-02-01	21	010-4444-5555
1202	조현수	2007-03-15	14	010-5555-6666
1203	박제성	1999-09-01	17	019-6666-7777
1312	김민수	2011-03-01	10	011-7777-8888
1314	이정숙	2011-03-01	10	010-8888-9999

1. 관계 데이터 모델이란?

2) 관계 데이터 모델의 특징

특징

개체와 관계가 모두 **2차원 테이블(릴레이션)**로 표현됨

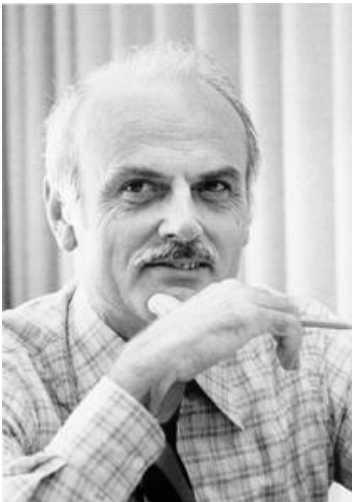
ER 모델을 관계 데이터 모델로 변형하는 것이 매우 쉬움

테이블과 릴레이션의 차이점

테이블	릴레이션
<ul style="list-style-type: none"> 릴레이션을 표현하는 구체적인 표현 방법 중복되는 행을 허용 	<ul style="list-style-type: none"> 추상적인 개념 하나의 릴레이션은 여러 가지 형태의 테이블로 표현 중복되는 행을 허용하지 않음



관계 DB 이론의 창시자인 E. F. Codd 박사 이야기



E. F. Codd 박사(1923 ~ 2003)

E. F. Codd 박사(1923 ~ 2003)는 1970년 6월 ACM(Association of Computer Machinery)의 'Communications of the ACM' 이라는 잡지에 "A Relational Model of Data for Large Shared Data Banks"라는 논문을 발표했는데, 이 논문에서 DB 시스템을 위한 관계 데이터 모델을 처음으로 제안했으며, 이 모델은 현재 관계형 데이터베이스 관리시스템(RDBMS)의 표준적인 모델로서 인정 받고 있다.

그 당시 계층 및 네트워크 데이터 모델이 더 많이 알려진 모델이었으나, Codd 박사가 제안한 관계 데이터 모델이 갖는 여러 가지 장점 때문에 짧은 시간 안에 널리 알려지게 되었고, 많은 DBMS 업체들이 이 관계 데이터 모델을 기초로 DBMS를 개발하게 되어 현재에 이르기까지 가장 널리 사용되고 있다.

2. 주요 용어들

1) 관계 데이터 모델과 관련한 주요 용어

용어	의미
튜플(Tuple)	릴레이션의 각 행(Row)으로서, 개체(Entity)에 대응함
속성(Attribute)	릴레이션의 각 열(Column)로서, 이름을 가진 정보의 가장 작은 논리적인 단위
도메인(Domain)	한 속성이 취할 수 있는 모든 원자 값의 집합 (데이터 타입, 크기, 형식, 제한 범위, Not Null 여부 등으로 정의함)
기수(Cardinality)	하나의 릴레이션이 갖는 튜플의 수
차수(Degree)	하나의 릴레이션이 갖는 속성의 수
릴레이션 스키마 (Relation Schema)	릴레이션을 구성하는 속성들의 집합
릴레이션 인스턴스 (Relation Instance)	릴레이션을 구성하는 튜플들의 집합
릴레이션(Relation)	행과 열로 구성된 2차원 구조로서, 릴레이션 스키마와 릴레이션 인스턴스가 합쳐진 개념 (릴레이션 스키마 + 릴레이션 인스턴스)
키(Key)	튜플들을 유일(Unique)하게 식별할 수 있는 속성, 또는 속성들의 집합

2. 주요 용어들

2) 사원 릴레이션에서 주요 용어 확인하기

사원 릴레이션

속성					릴레이션 스키마
사번	이름	입사일	호봉	휴대폰	
1101	김정아	1995-03-01	20	010-1111-2222	릴레이션 인스턴스
1102	이기원	1997-01-01	18	011-2222-3333	
1103	박영중	2005-09-01	15	010-3333-4444	
1201	최강희	1994-02-01	21	010-4444-5555	
1202	조현수	2007-03-15	14	010-5555-6666	
1203	박제성	1999-09-01	17	019-6666-7777	
1312	김민수	2011-03-01	10	011-7777-8888	
1314	이정숙	2011-03-01	10	010-8888-9999	

1) **사원 릴레이션 스키마** : 사원 = {사번, 이름, 직책, 부서번호, 호봉} 또는 간단히 사원(사번, 이름, 직책, 부서번호, 호봉)으로 표현한다.

2) **사원 릴레이션의 튜플 수** : 8

3) **릴레이션 인스턴스** : 8개의 튜플로 구성된다.

즉, {(1101, 김정아, 부장, 100, 20), ..., (1314, 이정숙, 사원, 200, 10)}

4) **사번의 도메인** : 4자리 정수

5) **사원 릴레이션의 기수** : 8 (튜플의 수)

6) **사원 릴레이션의 차수** : 5 (속성의 수)

7) **키(Key)** : 사번, 휴대폰

사원 릴레이션의 현재 상태에서는 '이름'도 튜플을 유일하게 식별할 수 있으므로 '키(Key)'가 될 수 있다고 생각할 수 있으나, 이름이 동일한 신입 사원이 얼마든지 추가될 수 있기 때문에 이름은 절대 키가 될 수 없습니다. 이처럼 키를 결정할 때는 현재의 데이터만으로 판단하는 것이 아니라, ∞ 식적으로 일반적인 관점에서 결정해야 합니다.

2. 주요 용어들

1) 릴레이션의 개념

릴레이션(Relation)이란?

- 행(Row)과 열(Column)로 구성된 2차원 구조로서, 각 행은 튜플이고, 각 열은 속성임
- 릴레이션 스키마와 릴레이션 인스턴스가 합쳐진 개념
(릴레이션 스키마 + 릴레이션 인스턴스)

2) 릴레이션 스키마와 릴레이션 인스턴스의 정의

릴레이션 스키마(Relation Schema)	릴레이션 인스턴스(Relation Instance)
<ul style="list-style-type: none"> • 릴레이션을 구성하는 속성들의 집합을 의미함 • 릴레이션 내포(Intension) 또는 릴레이션 스킴(Scheme)이라고도 칭함 • “릴레이션 이름 + 속성 이름”으로 표현함 <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> $R(A_1, A_2, \dots, A_n), \quad A_i \in D_i$ $= R(\{A_1, A_2, \dots, A_n\})$ </div> <p style="margin-left: 20px;">R : 릴레이션 이름 A : 속성 이름 D : 도메인</p> <ul style="list-style-type: none"> • 릴레이션의 정적인 성질임 <ol style="list-style-type: none"> ① 시간이 지나도 자주 변경되지 않음 (시간과 무관한 정적 성질) ② 릴레이션 타입과 같은 의미 	<ul style="list-style-type: none"> • 릴레이션을 구성하는 튜플들의 집합을 의미함 • 릴레이션 외포(extension)라고도 칭함 • 릴레이션의 내용, 즉 어느 한 시점에 릴레이션 R이 포함하고 있는 튜플들의 집합임 <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> $\{ \langle V_1, V_2, \dots, V_n \rangle \mid V_i \in D_i \}$ </div> <p style="margin-left: 20px;">V : 속성 값 D : 도메인</p> <ul style="list-style-type: none"> • 릴레이션의 동적인 성질임 <ol style="list-style-type: none"> ① 삽입, 삭제, 갱신으로 인해 시간에 따라 변경됨 ② 릴레이션의 값을 의미

▶ 다른 언급 없이 그냥 릴레이션이라고 하면, 릴레이션 인스턴스를 의미함

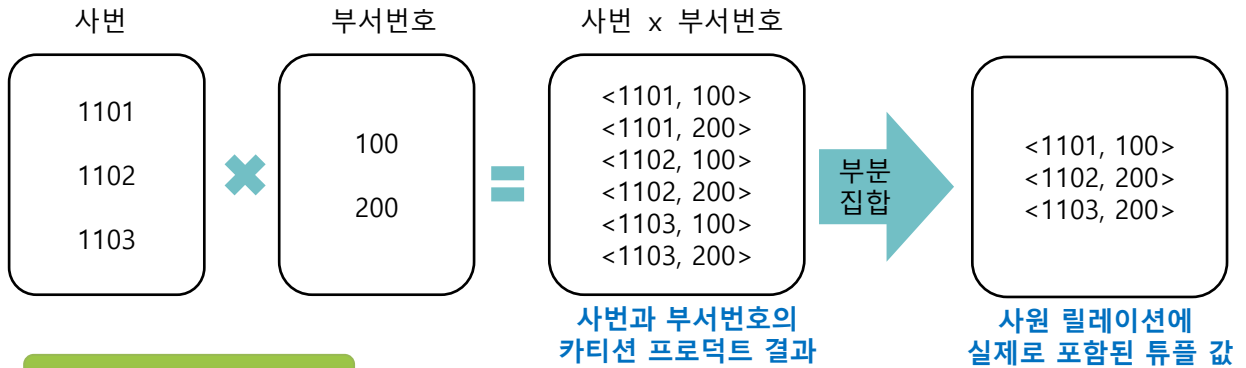
2. 주요 용어들

3) 릴레이션의 정의

수학적 정의

릴레이션 R은 다음과 같이 모든 도메인의 카티션 프로덕트(Cartesian Product)*의 부분집합으로 정의

$$R \subseteq D_1 \times D_2 \times \dots \times D_n \quad (\text{단 } D_i : i \text{ 번째 속성의 도메인})$$



개념적 정의

릴레이션 = 릴레이션 스키마 + 릴레이션 인스턴스

2. 주요 용어들

4) 릴레이션의 특성

튜플의 유일성

- 하나의 릴레이션에 포함된 튜플들은 모두 서로 달라야 함
- 중복된 튜플이 존재하지 않는 튜플들의 **집합(Set)**임

릴레이션에는 항상 튜플을 유일하게 식별할 수 있는 **키(key)**가 **존재**해야 함

[참고] 수학적 "집합"의 성질 : $\{a, b, c\} = \{c, b, a\}$

튜플의 무순서성

- 하나의 릴레이션에서 튜플들의 순서(order)는 아무런 의미가 없음

순서만 다른 튜플들로 구성된 릴레이션은 다른 릴레이션이 될 수 없음

속성의 무순서성

- 하나의 릴레이션에서 속성들의 순서는 아무런 의미가 없음

속성은 순서가 아니라 '**이름**'(예: 사번, 부서번호 등)에 의해서 참조

속성의 유일성

- 하나의 릴레이션에는 같은 이름을 갖는 속성은 존재할 수 없음

2. 주요 용어들

4) 릴레이션의 특성

속성의 유일성

- 모든 속성은 반드시 원자 값(Atomic Value)을 가져야 함



속성 값은 **논리적으로** 더 이상 분리될 수 없으며, 여러 개의 값을 갖는 속성은 직접 표현할 수 없음

- 널(Null) 값도 원자 값으로 간주함



원자성에 대한 보충 설명

성명은 성(Last name)과 명(First name)으로 분리될 수 있으나, 이런 분리가 필요하다면 성과 명이 처음부터 별개의 속성으로 설계되어야 하며, 성명이 하나의 속성으로 설계되면 그 값은 더 이상 분리될 수 없는 원자 값이 된다.

키(Key)와 무결성 제약조건

1. 관계 DB란?
2. 키(Key)
3. 무결성 제약조건

1. 관계 DB란?

1) 관계 DB의 구성과 관계 DB 스키마

관계 DB의 구성

릴레이션(테이블)들의 집합으로 구성

시간에 따라 내용이 변할 수 있는 테이블 형태로 DB를 표현

관계 DB 스키마

관계 DB 스키마는 릴레이션 스키마와 무결성 제약조건으로 정의

{릴레이션 스키마} + {무결성 제약조건*}

#3

데이터의 정확성과 일관성 유지를 위해 DB가 항상 만족시켜야 하는 조건을 의미

2) 관계 DB 사례: A라는 회사 DB를 구성하는 3개의 릴레이션

릴레이션 스키마

사원(사번, 이름, 입사일, 호봉, 휴대폰)

부서(부서번호, 부서명, 부서장, 사무실, 전화번호)

근무(사번, 부서번호, 직책)

밑줄 표시는 사번이 키(Key)임을 나타냄

1. 관계 DB란?

2) 관계 DB 사례: A라는 회사 DB를 구성하는 3개의 릴레이션

릴레이션 (릴레이션 스키마 + 릴레이션 인스턴스)

사원 릴레이션

사번	이름	입사일	호봉	휴대폰	릴레이션 스키마
1101	김정아	1995-03-01	20	010-1111-2222	릴레이션 인스턴스
1102	이기원	1997-01-01	18	011-2222-3333	
1103	박영종	2005-09-01	15	010-3333-4444	
1201	최강희	1994-02-01	21	010-4444-5555	
1202	조현수	2007-03-15	14	010-5555-6666	
1203	박제성	1999-09-01	17	019-6666-7777	
1312	김민수	2011-03-01	10	011-7777-8888	
1314	이정숙	2011-03-01	10	010-8888-9999	

부서 릴레이션

부서번호	부서명	부서장	사무실	전화번호	릴레이션 스키마
100	기획실	1101	A402	02-111-2222	릴레이션 인스턴스
200	비서실	1102	A501	02-222-3333	
300	총무부	1201	B311	041-333-4444	
400	인사부	1203	B201	041-444-5555	
500	자재부		A102	02-555-6666	

근무 릴레이션

사번	부서번호	직책	릴레이션 스키마
1101	100	부장	릴레이션 인스턴스
1102	200	과장	
1103	200	대리	
1201	300	부장	
1202	300	대리	
1203	400	과장	
1312	100	사원	
1314	400	사원	

2. 키(Key)

1) 키(Key)

키(Key)란?

릴레이션을 구성하는 튜플들을 유일(Unique)하게 식별할 수 있는 속성, 또는 속성들의 집합

종류	특징
후보 키 (Candidate Key)	키 가운데 유일성 (Uniqueness)과 최소성(Minimality)을 만족하는 속성 집합
슈퍼 키 (Super Key)	유일성은 만족하지만 최소성이 만족되지 않는 속성 집합
기본 키 (Primary Key)	<ul style="list-style-type: none"> - 후보 키 가운데 업무에 적합하게 선택된 하나의 키 (대개 DB 설계자가 지정함) - 널 값을 가질 수 없고, 중복되지 않는 유효한 값을 가져야 함
대체 키 (Alternative Key)	기본 키를 제외한 나머지 후보 키들
외래 키 (Foreign Key)	릴레이션 간의 관계를 정의하는 키로서, 하위 릴레이션(참조 릴레이션)에서 상위 릴레이션(피참조 릴레이션)의 기본 키를 외래 키로 포함하게 됨

2. 키(Key)

2) 후보 키(Candidate Key)

후보 키(Candidate Key)란?

릴레이션 $R(A_1, A_2, \dots, A_n)$ 의 전체 속성 집합 A 의 부분 집합이면서,
유일성 (Uniqueness)과 **최소성(Minimality)**을 만족하는 속성 집합 $\{A_i, A_j, \dots, A_k\}$

유일성 (Uniqueness)

각각의 튜플을 유일하게 식별할 수 있는 것

최소성(Minimality)

각각의 튜플을 유일하게 식별하는데 필요한 **최소 속성만** 포함하는 것

즉, 키가 하나 이상의 속성으로 구성되어 있을 때,
 어떤 속성을 제거해도 유일성을 만족하는 경우, 최소성이 만족되지 않는 것임

예: 사원 릴레이션의 후보 키

사번	이름	입사일	호봉	휴대폰
1101	김정아	1995-03-01	20	010-1111-2222
1102	이기원	1997-01-01	18	011-2222-3333
1103	박영종	2005-09-01	15	010-3333-4444
1201	최강희	1994-02-01	21	010-4444-5555
1202	조현수	2007-03-15	14	010-5555-6666
1203	박제성	1999-09-01	17	019-6666-7777
1312	김민수	2011-03-01	10	011-7777-8888
1314	이정숙	2011-03-01	10	010-8888-9999

사원 릴레이션의 후보 키: 사번 속성, 휴대폰 속성

이름과 입사일, 호봉 속성은 유일성을 만족하지 않고, {사번, 이름}, {사번, 호봉} 등의 속성 집합은 최소성을 만족하지 않기 때문에 후보 키가 될 수 없음

2. 키(Key)

3) 외래 키(Foreign Key)

외래 키(Foreign Key)란?

릴레이션 R의 어떤 속성 집합 FK가 릴레이션 S의 기본 키일 때, FK를 릴레이션 R의 외래 키라고 함

특징

외래 키의 도메인은 릴레이션 S의 기본 키의 도메인과 동일함

외래 키의 값은 릴레이션 S에 존재하는 값이거나 **널 값을 가질 수 있음**

하나 이상의 테이블을 연결해서 사용하는 경우 필요한 키임

릴레이션 R과 S가 동일한 릴레이션인 경우도 있음

참조 릴레이션과 피참조 릴레이션

외래 키를 포함하는 릴레이션 R : **참조 릴레이션(Referencing Relation)**

외래 키가 참조하는 기본 키를 포함하는 릴레이션 S : **피참조 릴레이션(Referenced Relation)**



릴레이션 R은 외래 키(FK)를 통해 릴레이션 S를 참조함

2. 키(Key)

3) 외래 키(Foreign Key)

다른 릴레이션의 기본 키를 참조하는 외래 키

사원_1 릴레이션

사번	이름	담당업무	부서번호
1101	김정아	기획	100
1102	이기원	비서실장	200
1103	박영종	비서	200
1201	최강희	회계	300
1203	조현수	인사	400

기본 키

부서 릴레이션의
기본 키인
부서번호를 참조함

외래 키

부서 릴레이션

부서번호	부서명	사무실
100	기획실	A402
200	비서실	A501
300	총무부	B311
400	인사부	B201

기본 키

동일한 릴레이션의 기본 키를 참조하는 외래 키

사원_2 릴레이션

사번	이름	관리자	부서번호
1101	김정아	1203	100
1102	이기원	1101	200
1103	박영종	1201	200
1201	최강희	1102	300
1203	조현수		400

기본 키

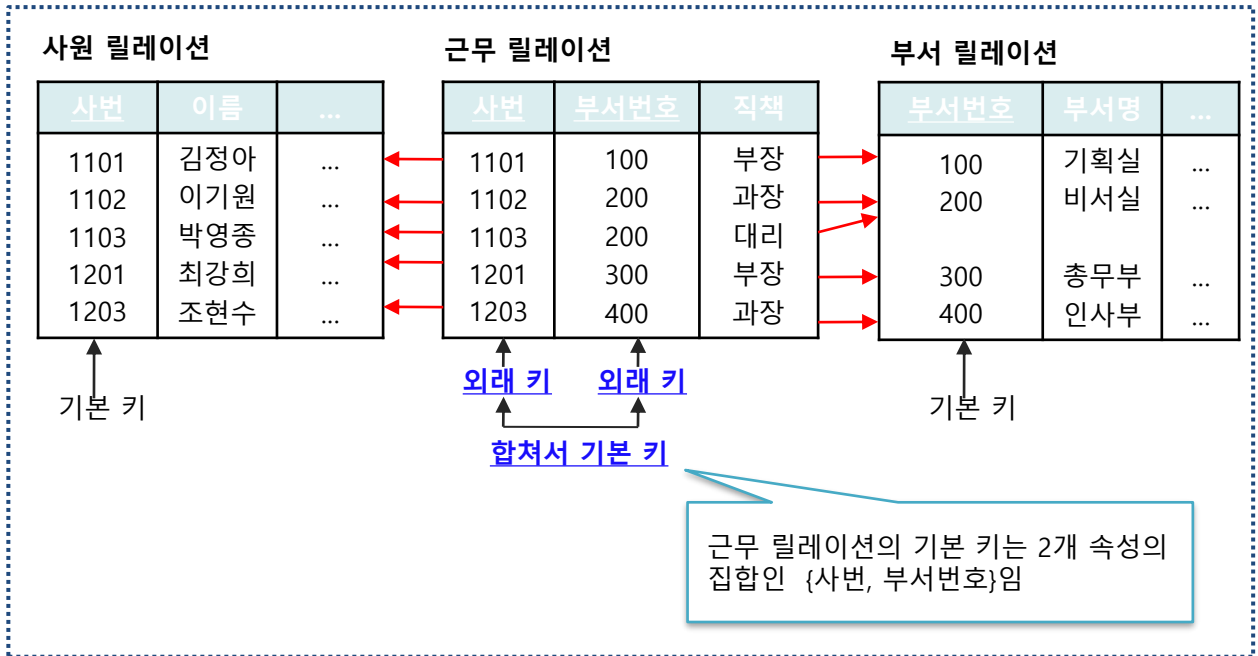
외래 키

동일한 릴레이션인 사원_2
릴레이션의 기본 키인 사번을
참조

2. 키(Key)

3) 외래 키(Foreign Key)

기본 키의 구성 요소가 되는 외래 키



2. 키(Key)

4) 예제 - 학생 릴레이션과 교수 릴레이션에서 본 여러 가지 키들

학생 릴레이션

학번	주민등록번호	이름	주소	휴대폰	지도교수
98201002	800321	김정욱	천안	010-1111-2222	01223
99202012	811123	이기원	서울	010-2222-3333	02355
98201025	810923	박문구	대전	010-3333-4444	01223

교수 릴레이션

교수번호	이름	연구실	휴대폰
01223	장경구	A112	010-1111-2222
02355	김민숙	B344	010-2222-3333
05378	이장수	A322	010-3333-4444

키

주민등록번호, 학번, 휴대폰 (주민등록번호, 이름), (학번, 이름), (휴대폰, 이름), (주민등록번호, 주소), (학번, 주소), (휴대폰, 주소), (주민등록번호, 나이) 등

후보 키

주민등록번호, 학번, 휴대폰

키 가운데 이 세가지를 제외하고는 유일성은 만족되지만 최소성이 만족되지 않음
예 : (주민등록번호, 이름) 키의 경우, '이름'을 제거해도 유일성이 만족되므로, 최소성을 만족하지 않는 것임

기본 키

주민등록번호나 학번, 휴대폰 가운데 하나를 업무의 특성에 맞게 선택함

단, 휴대폰이 없는 사람이 있는 경우, 휴대폰은 널 값을 가질 수 있으므로 기본 키가 될 수 없음

외래 키

지도교수

- 이 속성은 남학생의 지도교수 이름을 검색할 때, 교수 릴레이션의 기본 키(교수번호)를 참조하게 되므로 외래 키임
- 지도교수가 지정되지 않은 학생의 경우 널 값을 갖게 됨

3. 무결성 제약조건

1) 무결성 제약조건

무결성 제약조건이란?

DB에 저장된 데이터의 **정확성과 일관성 유지**를 위해서 **DB가 항상 만족해야 하는 조건**을 의미

중요성

사용자에 의한 DB 갱신이 DB의 일관성을 깨지 않도록 보장하는 수단

DB의 상태가 현실 세계에서 허용되는 상태만 나타낼 수 있도록 보장하는 수단

2) 관계 데이터 모델의 무결성 제약조건

개체 무결성 (Entity Integrity)

기본 키는 튜플들을 유일하게 식별하기 위해서 널 값을 가질 수 없음

튜플을 유일하게 식별하기 위해서 기본 키는 반드시 값을 가져야 함을 의미

예	회원 가입 시에 '주민등록번호'를 입력하지 않으면 회원 가입을 할 수 없는 것은 주민등록번호가 회원 개체의 기본 키이기 때문
---	---

3. 무결성 제약조건

2) 관계 데이터 모델의 무결성 제약조건

참조 무결성 (Referential Integrity)

외래 키는 반드시 피참조 릴레이션의 기본 키 값이나 널 (Null) 값을 가져야 함

외래 키는 대응하지 않는 값을 가질 수 없으며,
이를 위배하는 경우 두 릴레이션은 서로 참조될 수 없음을 의미

예

앞의 예제에서, 교수 릴레이션에 없는 '교수번호' 속성 값이 학생 릴레이션의 '지도교수' 속성 값이 될 수 없음

도메인 무결성 (Domain Integrity)

특정 속성은 반드시 미리 정해진 값을 가져야 함

데이터 타입, 길이, 디폴트 값, 널(Null) 값 허용 여부, 허용되는 값의 범위 등을
만족하는 값을 가져야 함을 의미

예

- 대학생의 학년 속성은 1부터 5 사이 값만 가질 수 있다.
- 회원의 id 속성은 최대 10자의 문자 값만 가질 수 있다.

3. 무결성 제약조건

3) 무결성 제약조건 정의 방법

묵시적 정의	명시적 정의
릴레이션을 정의할 때 기본 키와 외래 키를 정의하면, 개체 무결성과 참조 무결성이 묵시적으로 정의됨	<p>개체 무결성과 참조 무결성 이외에, 데이터의 정확성과 일관성 유지를 위해서 특정 속성이 만족해야 하는 의미 무결성 제약조건은 명시적으로 정의해야 됨</p> <div> <p>릴레이션 생성할 때 정의</p> <ul style="list-style-type: none"> 널 값 허용 여부(NOT NULL 제약조건) 고유 값 여부(UNIQUE 제약조건) 속성 값이 만족 해야 하는 특정 조건(CHECK 제약조건) </div>



DB가 갱신될 때마다 DBMS가 자동으로 제약조건의 만족 여부를 검사하므로 응용 프로그램에서 별도로 검사할 필요가 없음



DBMS가 자동으로 처리하는 제약조건 외에 특수한 제약조건은 응용 프로그램에서 별도로 처리해야 하는 경우도 있음

3. 무결성 제약조건

4) 무결성 제약조건 정의 사례 : A라는 회사의 관계 DB 스키마를 위한 명시적 무결성 제약조건 정의

사원 릴레이션

- ① 이름과 입사일, 호봉, 휴대폰 번호는 널 값을 가질 수 없음 (NOT NULL 제약조건)
- ② 휴대폰은 유일한 값을 가져야 함 (UNIQUE 제약조건)
- ③ 호봉은 1부터 30 사이의 정수 값만 가질 수 있음(CHECK 제약조건)

사번	이름	입사일	호봉	휴대폰
1101	김정아	1995-03-01	20	010-1111-2222
1102	이기원	1997-01-01	18	011-2222-3333
1103	박영종	2005-09-01	15	010-3333-4444
1201	최강희	1994-02-01	21	010-4444-5555
1202	조현수	2007-03-15	14	010-5555-6666
1203	박제성	1999-09-01	17	019-6666-7777
1312	김민수	2011-03-01	10	011-7777-8888
1314	이정숙	2011-03-01	10	010-8888-9999

부서 릴레이션

- ① 부서명과 전화번호는 널 값을 가질 수 없음 (NOT NULL 제약조건)
- ② 사무실은 유일한 값을 가져야 함 (UNIQUE 제약조건)

부서번호	부서명	부서장	사무실	전화번호
100	기획실	1101	A402	02-111-2222
200	비서실	1102	A501	02-222-3333
300	총무부	1201	B311	041-333-4444
400	인사부	1203	B201	041-444-5555
500	자재부		A102	02-555-6666

3. 무결성 제약조건

4) 무결성 제약조건 정의 사례 : A라는 회사의 관계 DB 스키마를 위한 명시적 무결성 제약조건 정의

근무 릴레이션

① 직책은 널 값을 가질 수 없음 (NOT NULL 제약조건)

사번	부서번호	직책
1101	100	부장
1102	200	과장
1103	200	대리
1201	300	부장
1202	300	대리
1203	400	과장
1312	100	사원
1314	400	사원