

3장. 관계 데이터 모델과 관계 무결성 제약조건

1 관계형 데이터베이스의 구조

1-1 릴레이션(Relation) = 테이블

- 관계형 데이터베이스 (relational database)

- 현재 가장 많이 사용되고 있는 데이터베이스
- 엔티티(Entity)나 관계성(Relationship)을 모두 행과 열로 구성된 표 형식으로 데이터를 관리

- 릴레이션(Relation) = 테이블

- 관계형 데이터베이스에서 정보를 구분하여 저장하는 기본 단위



57 page

1 관계형 데이터베이스의 구조

1-1 릴레이션(Relation) = 테이블

- 릴레이션 스키마(relation schema)

- 릴레이션 이름과 일정수의 속성들의 집합으로 구성

- 릴레이션 인스턴스(relation instance)

- 어느 한 시점에서 릴레이션에 포함되어 있는 튜플의 집합



1 관계형 데이터베이스의 구조

1-2 속성(Attribute) = 열

● 속성(attribute)

- 릴레이션에서 관리하는 구체적인 정보 항목에 해당하며 엔티티의 특성을 기술

사원번호	이름	직급	주민번호	급여	부서명
101	김사랑	사원	831212-2212112	250	인사부
102	한예슬	대리	771227-2323123	300	영업부
103	오지호	과장	720224-1013112	500	영업부
104	이병헌	부장	710509-1934142	600	인사부

58 page

1 관계형 데이터베이스의 구조

1-3 튜플(Tuple) = 행

- 튜플(tuple)

- 속성의 모임으로 구성
- 릴레이션에서 각각의 행에 해당
- 파일 구조에서 레코드와 같은 의미

예

'사원'은 릴레이션을 나타내는 이름이고 '사원번호 : 101, 이름 : 김사랑, 직급 : 사원, 급여 : 250, 부서명 : 인사부'는 '사원' 릴레이션을 구성하는 튜플이다.

1 관계형 데이터베이스의 구조

1-3 튜플(Tuple) = 행

- 튜플의 수를 카디널리티(Cardinality, 기수, 대응수)라고 함.

- <사원> 릴레이션의 카디널리티는 4이다.

사원번호	이름	직급	주민번호	급여	부서명
101	김사랑	사원	831212-2212112	250	인사부
102	한예슬	대리	771227-2323123	300	영업부
103	오지호	과장	720224-1013112	500	영업부
104	이병헌	부장	710509-1934142	600	인사부

튜플 = 행 = 레코드

튜플의
개수 = 카디널리티 = 4

59 page

1 관계형 데이터베이스의 구조

1-4 도메인(Domain)

- 도메인(domain)

- 릴레이션에 포함된 각각의 속성들이 취할 수 있는 같은 타입의 원자(atomic) 값들의 집합

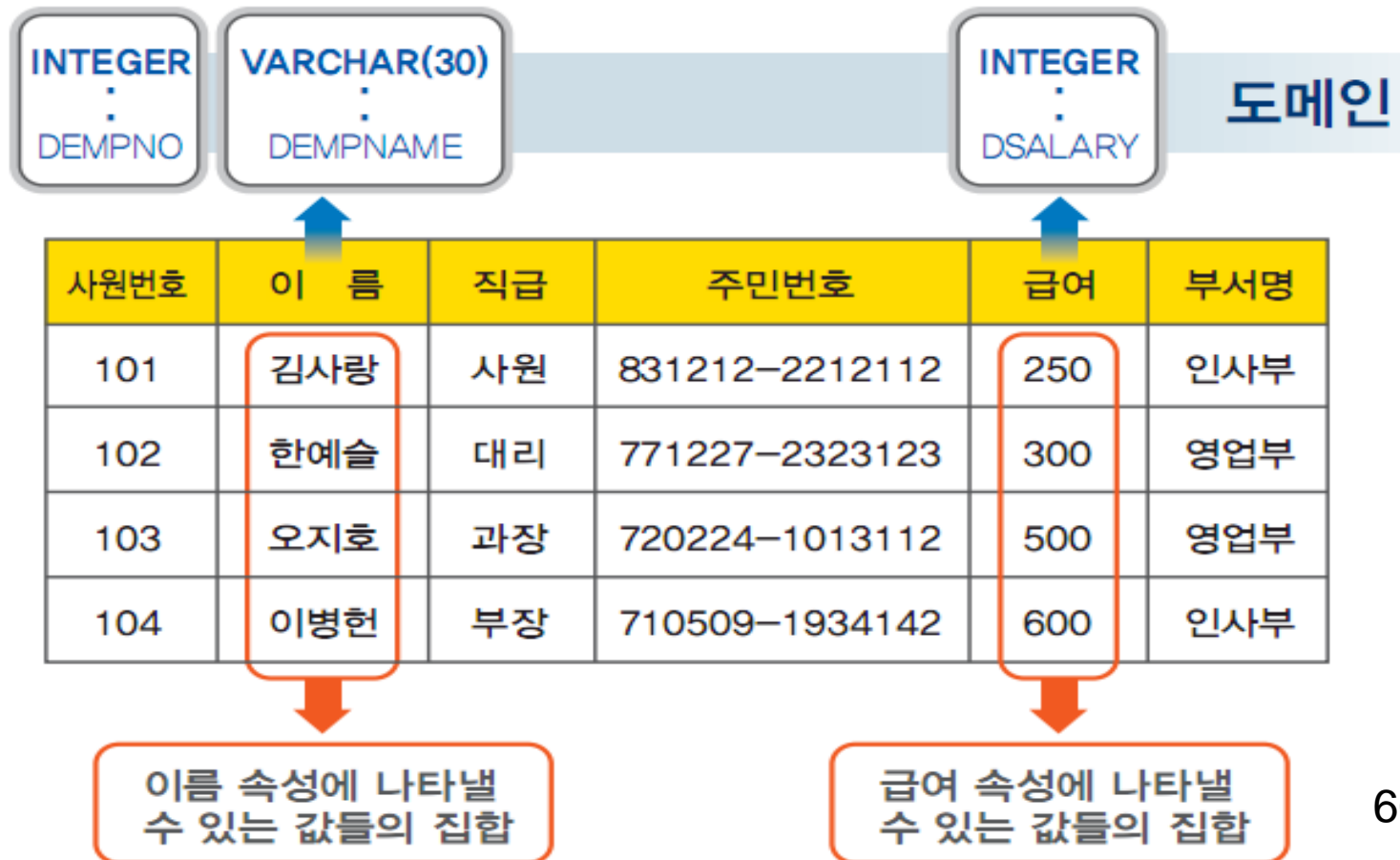
- 도메인의 정의에 대한 예

- CREATE DOMAIN DEMPNO INTEGER
- CREATE DOMAIN DEMPNAME VARCHAR(30)
- CREATE DOMAIN DSALARY INTEGER

1 관계형 데이터베이스의 구조

1-4 도메인(Domain)

- 도메인



60 page

1 관계형 데이터베이스의 구조

1-4 도메인(Domain)

- 도메인이라는 개념이 필요한 이유

- 릴레이션에 저장되는 데이터 값들이 본래 의도했던 값들만 저장되고 관리되도록 하는데 있음.

예

'성별'이라는 속성이 있다면 이 속성이 가질 수 있는 값은 {남, 여}

1 관계형 데이터베이스의 구조

1-5 릴레이션 스키마와 인스턴스

● 릴레이션(relation)

- 릴레이션 스키마(릴레이션 타입) + 릴레이션 인스턴스(릴레이션 어커런스)



61 page

정의

릴레이션이름(기본키, 속성이름1, 속성이름2, ... 속성이름n)

1 관계형 데이터베이스의 구조

1-5 릴레이션 스키마와 인스턴스

- 문제(3.1)

- 사원(사원번호, 이름, 직급, 주민번호, 급여, 부서명)

- 사원 릴레이션 스키마에서 릴레이션의 이름, 속성, 기본키를 구별해 보자.

릴레이션 이름	
기본키	
속성	

2 릴레이션의 특징

2-1 속성 간의 순서

- 릴레이션을 구성하는 속성 간의 순서는 중요하지 않다.

- 한 릴레이션 내의 속성들의 순서를 바꾼다고 해서 다른 릴레이션이 되지는 않음
- 관계 데이터베이스에서 속성값은 릴레이션 내에서의 속성의 위치에 의해서가 아니고 속성 이름에 의해서 참조되기 때문

〈부서〉 릴레이션

위치	부서명	부서번호
서울	경리부	10
인천	인사부	20
용인	영업부	30
수원	전산부	40

〈부서〉 릴레이션

부서번호	부서명	위치
10	경리부	서울
20	인사부	인천
30	영업부	용인
40	전산부	수원

63 page

2 릴레이션의 특징

2-2 상이한 튜플

- 한 릴레이션에 포함된 튜플들은 모두 다르다.

- 릴레이션의 각 튜플을 살펴보면 어떠한 튜플도 정확하게 동일한 값을 갖지 않음
- 릴레이션은 튜플의 집합이기에 튜플을 유일하게 식별하기 위해 고유한 값을 저장하는 속성이 적어도 한 개 있어야 함
- 키는 속성 혹은 속성의 집합으로 키 덕분에 테이블에 중복된 튜플이 존재할 수 없게 됨

예

<부서> 테이블의 부서 번호나

<사원> 테이블의 사원번호 혹은 주민번호가 키에 해당한다.

2 릴레이션의 특징

2-3 튜플의 순서

- 릴레이션에 포함된 튜플의 순서는 중요하지 않다.

- 튜플이 물리적으로 저장될 경우에는 저장하는 순서에 따라서 튜플의 순서가 정해지고 출력을 하게 되면 먼저 저장된 것이 위에, 나중에 저장된 것이 아래에 출력됨
- 대규모 릴레이션일 경우 튜플의 위치를 일일이 기억한다는 것은 불가능하고 그럴 이유도 없음
- 실질적으로 상용되는 DBMS는 효율성을 위해서 튜플의 순서를 바꿀 수 있는 여러 가지 방법을 제공하여 릴레이션에 포함된 튜플의 순서는 무의미

〈부서〉 릴레이션

부서번호	부서명	위치
10	경리부	서울
20	인사부	인천
30	영업부	용인
40	전산부	수원

〈부서〉 릴레이션

부서번호	부서명	위치
40	경리부	서울
30	인사부	인천
20	영업부	용인
10	전산부	수원

64 page

2 릴레이션의 특징

2-4 릴레이션의 변동성

- 튜플들의 삽입, 삭제 등의 작업으로 인해 릴레이션은 시간에 따라 변한다.

2 릴레이션의 특징

2-5 속성의 명칭과 값

- 속성의 유일한 식별을 위해 속성의 명칭은 유일해야 하지만 속성이 구성하는 값은 동일할 수 있다.

Figure 5.6

One possible database state for the COMPANY relational database schema.

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Location
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

WORKS_ON

Essn	Pno	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

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

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

2 릴레이선의 특징

2-6 속성의 명칭과 값

- 속성은 더 이상 쪼갤 수 없는 원자값(atomic value)만을 저장해야 한다.
 - 속성에는 여러 개의 값들을 리스트나 집합 형태로 저장할 수 없음
 - 하나의 속성이 두 개 이상의 값을 갖게 되면 릴레이선을 하나 추가한 후에 이를 참조할 수 있도록 해야 함



3 키의 개념 및 종류

- 릴레이션의 특징 중의 하나가 한 릴레이션에 포함된 튜플들은 모두 달라야 한다는 것이었다.
- 이상(anomaly) 현상
 - 데이터가 중복 저장되면 릴레이션 조작 시 예상하지 못한 곤란한 현상이 발생하는 현상



한예슬	과장	761227-2313123	300	경리부
한예슬	부장	750121-2013321	700	인사부

사원번호	이름	직급	주민번호	급여	부서명
101	김사랑	사원	831212-2212112	250	인사부
102	한예슬	대리	771227-2323123	300	영업부
103	오지호	과장	720224-1013112	500	영업부
104	이병헌	부장	710509-1934142	600	인사부

66 page

3 키의 개념 및 종류

● 키(key)

- 속성 혹은 속성의 집합으로 튜플을 고유하게 식별할 수 있도록 하는 속성 혹은 속성의 집합
- 데이터를 검색하거나 정렬할 때 기준이 되는 유일하게 구분되는 속성으로 **수퍼키**, **후보키**, **기본키**, **외래키** 등이 있음



3 키의 개념 및 종류

3-1 수퍼키와 후보키

● 수퍼키(Super Key)

- 릴레이션을 구성하는 속성들 중에서 각 튜플을 **유일하게 식별**하기 위해 사용하는 하나 혹은 그 이상의 속성들의 집합
- **유일성**: 하나의 키 값으로 하나의 튜플만을 유일하게 식별할 수 있는 성격

● 후보키(candidate key)

- 릴레이션을 구성하는 속성들 중에서 각 튜플을 유일하게 식별하기 위해 사용하는 하나 혹은 그 이상의 속성들의 집합
- 수퍼키와의 공통점 : 릴레이션에 있는 각 튜플을 고유하게 식별할 수 있어야 한다는 점(유일성)
- **최소성**: 모든 레코드들을 유일하게 식별하는데 꼭 필요한 속성으로만 구성되어야 하는 성격
- 수퍼키와의 차이점 : **최소한의 속성으로 구성된** 유일성을 갖는 속성의 집합

3 키의 개념 및 종류

3-2 기본키

● 기본키(Primary key : PK)

- 후보키 중 튜플을 식별하는데 기준으로 사용하는 키
- 후보키 중에서 선택한 것이기에 한 릴레이션에서 특정 튜플을 유일하게 구별할 수 있는 속성
- Null 값을 가질 수 없음
- 기본키로 정의된 속성에는 동일한 값이 중복되어 저장될 수 없음

Tip

Null 값은 데이터베이스에서 아직 알려지지 않거나 모르는 값으로서 “해당없음” 등의 이유로 정보 부재를 나타내기 위해 사용하는, 이론적으로 아무것도 없는 특수한 데이터를 말한다.

3 키의 개념 및 종류

3-2 대체키

● 대체키(Alternate key)

- 후보키가 둘 이상일 때 기본키로 선택되지 않은 나머지 후보키
- 보조키라고도 함
- 사원번호를 기본키로 정의하였다면 주민번호가 대체키가 됨

3 키의 개념 및 종류

3-2 대체키

- 문제(3.2)
 - 주민등록번호, 이름, 이메일주소
- 이 데이터 중 후보키는 어떤 것인가?

3 키의 개념 및 종류

3-2 복합키

● 복합키(composite key)

- 하나의 칼럼이 후보키의 역할을 하지 못함
- 두 개 이상의 칼럼이 합쳐져야 후보키의 역할을 하는 경우를 말함

〈사원이 속한 클럽〉 릴레이션

EMPNO	CNAME	ENAME
101	영어회화반	김사랑
102	낚시반	한예슬
101	영화사랑반	김사랑
103	영어회화반	오지호
102	영화사랑반	한예슬
104	낚시반	이병헌

기본키

70 page

3 키의 개념 및 종류

3-4 왜래키

● 외래키(Foreign Key)

- 상호 관련이 있는 테이블들 사이에서 관계성을 표현해 주는 수단

● 외래키의 필요성을 설명하기 위해서 다음과 같은 질문

- CASE1 부서 릴레이션의 첫 번째 튜플이 삭제된다면 어떻게 될까?
- CASE2 부서 릴레이션의 20번 부서 번호가 50으로 변경된다면 어떻게 될까?
- CASE3 사원 릴레이션에 부서 번호 60인 사원이 삽입된다면 어떻게 될까?

CASE 3

105	김태희	과장	761227-2312123	300	60
-----	-----	----	----------------	-----	----

사원번호	이름	직급	주민번호	급여	부서번호
101	김사랑	사원	831212-2212112	250	20
102	한예슬	대리	771227-2323123	300	30
103	오지호	과장	720224-1013112	500	30
104	이병헌	부장	710509-1934142	600	20

〈사원〉 릴레이션

부서번호	부서명	위치
10	경리부	서울
20	50	인사부
30	영업부	용인
40	전산부	수원

CASE 1

〈부서〉 릴레이션

⇒ 사원 릴레이션과 부서 릴레이션에 있는 데이터 사이에 불일치 발생!

3 키의 개념 및 종류

3-4 외래키

- 참조 관계



4 무결성 제약조건

4-1 개체 무결성

● 개체 무결성(entity integrity)

- 개체가 결점이 없음을 의미
- 결점이 없는 무결한 개체가 되려면 데이터베이스에 저장되어 관리될 때 본질적으로 서로 구별될 수 있어야 함
- 개체를 유일하게 식별하기 위해서는 당연히 식별자가 반드시 있어야 함

릴레이션을 구성하는 속성 중의 하나를 기본키로 지정하여 널(NULL) 값이나 중복된 값을 가질 수 없는 속성을 갖도록 하여 언제 어느 때고 개체가 유일하게 식별 가능하도록 하는 것이 '개체 무결성 제약조건'이다.

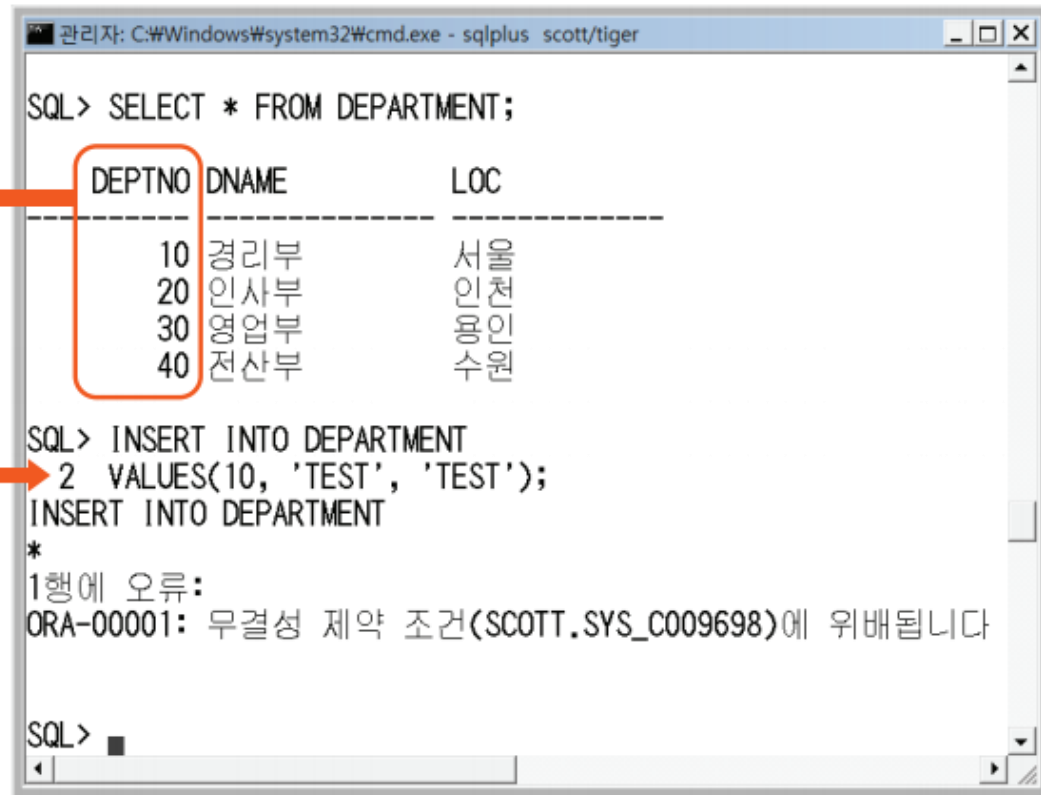
4 무결성 제약조건

4-1 개체 무결성

● 개체 무결성

부서번호(DEPTNO) 컬럼
은 중복된 값을 저장할
수 없는 기본키이다.

DEPARTMENT 테이블
에 10번 부서가 이미
존재하고 있기 때문에
10번 부서를 추가하려
고 했을 때 에러가 발생
한다.



The screenshot shows a SQL*Plus window with the following content:

```
관리자: C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT * FROM DEPARTMENT;

DEPTNO DNAME          LOC
-----
10     경리부          서울
20     인사부          인천
30     영업부          용인
40     전산부          수원

SQL> INSERT INTO DEPARTMENT
2 VALUES(10, 'TEST', 'TEST');
INSERT INTO DEPARTMENT
*
1행에 오류:
ORA-00001: 무결성 제약 조건(SCOTT.SYS_C009698)에 위배됩니다

SQL>
```

Annotations in the image:

- An orange box highlights the **DEPTNO** column header and its values (10, 20, 30, 40) in the query result. An arrow points from the text box on the left to this box.
- Another orange box highlights the **VALUES(10, 'TEST', 'TEST')** part of the insert statement. An arrow points from the text box on the left to this box.

73 page

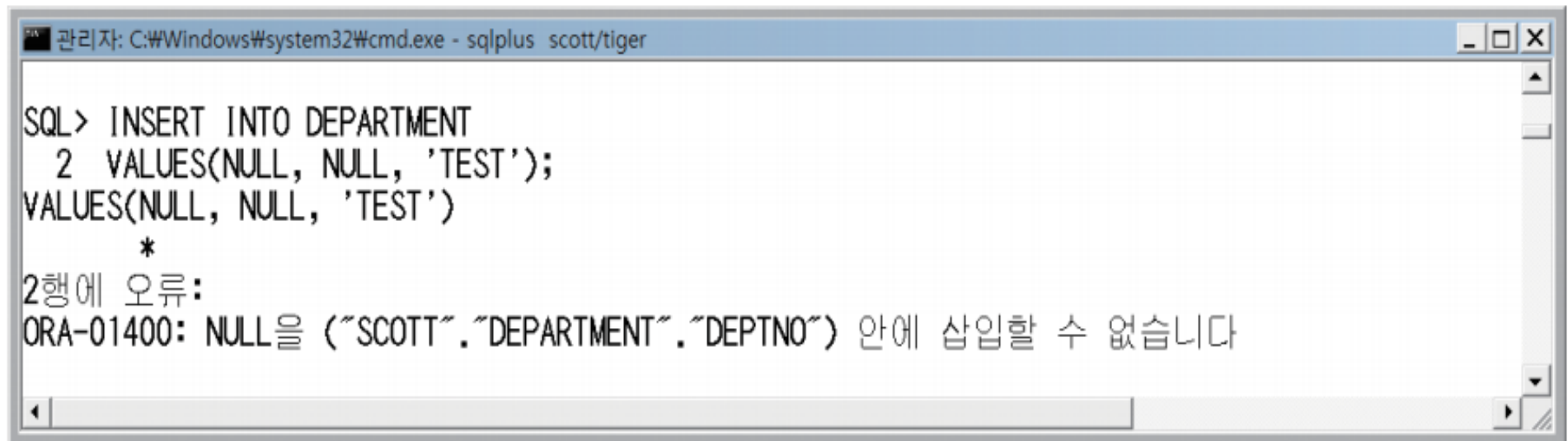
- 10번 부서가 이미 존재하는데 10번 부서를 추가하려고 시도하면 무결성 제약조건에 위배된다는 오류 메시지와 함께 10번 부서를 추가하지 않게 됨

4 무결성 제약조건

4-1 개체 무결성

● 개체 무결성

- 기본키 값이 널 값을 갖게 된다면 튜플을 유일하게 식별할 수 없게 됨
- 튜플을 유일하게 식별할 수 없게 되면 서로 구별할 수 없는 개체가 존재하게 된다는 의미가 되기에 개체 무결성에 위배됨
- 기본키의 유일한 식별성을 잃지 않도록 하기 위해서는 중복되는 값을 갖지 말아야 함
- 동시에 널 값 역시 포함되어서는 안 된다.



```
관리자: C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> INSERT INTO DEPARTMENT
  2  VALUES(NULL, NULL, 'TEST');
VALUES(NULL, NULL, 'TEST')
      *
2행에 오류:
ORA-01400: NULL을 ("SCOTT"."DEPARTMENT"."DEPTNO") 안에 삽입할 수 없습니다
```

4 무결성 제약조건

4-1 개체 무결성

● 개체 무결성

- 특정 개체가 무결성을 유지하도록 하기 위해서 릴레이션을 생성하는 데이터 정의문에서 어떤 속성이 릴레이션의 기본키인지를 알려주어야 함
- 기본키로 제약조건을 지정한 속성은 널(NULL) 값이나 중복된 값을 가질 수 없음

예

'사원' 릴레이션에서 '사원번호'가 기본키로 정의되면 튜플을 추가할 때 '주민번호'나 '성명' 필드에는 값을 입력하지 않아도 되지만 '사원번호' 속성에는 반드시 값이 입력되어야 한다. 또한 '사원번호' 속성에는 이미 한 번 입력한 속성값을 중복하여 입력할 수 없다.

4 무결성 제약조건

4-2 참조 무결성

● 참조 무결성(referential integrity)

- 참조 무결성은 릴레이션 간에 적용되는 제약조건
- 릴레이션 모델에서는 두 객체 간의 관계를 외래키 혹은 릴레이션으로 표현함

부서 테이블에 50번
부서가 없으므로
삽입이 불가능하다.

DEPTNO	DNAME	LOC
10	경리부	서울
20	인사부	인천
30	영업부	용인
40	전산부	수원

```
SQL> INSERT INTO EMPLOYEE  
2 (EMPNO, ENAME, DEPTNO)  
3 VALUES(115, '성운정', 50);  
INSERT INTO EMPLOYEE  
*  
1행에 오류:  
ORA-02291: 무결성 제약조건(SCOTT.SYS_C009740)이 위배되었습니다- 부모 키가  
없습니다
```

75 page

4 무결성 제약조건

4-2 참조 무결성

- 참조 무결성

참조할 수 없는 외래키 값을 가질 수 없도록 함으로써 두 테이블 간의 데이터 무결성을 유지하는 것을 '참조 무결성 제약조건'이라고 한다.

예

'사원' 릴레이션의 '부서 번호' 속성에는 '부서' 릴레이션의 '부서 번호' 속성에 없는 값은 입력할 수 없다.

4 무결성 제약조건

4-2 참조 무결성

● 참조 무결성 제약조건의 유지

- 두 테이블 간에 외래키에 의한 참조 관계에 있고 두 테이블 간 데이터 불일치가 발생하는 상황이 될 때, DBMS는 제한(restrict), 연쇄(cascade), 널 값으로 대체(nullify)와 같은 조치를 취할 수 있다.

● 부서테이블의 첫 번째 튜플을 삭제하려 할 때

- 제한(restrict)은 삭제하려는 튜플의 부서 번호 값을 사원 테이블에서 가지고 있는 튜플이 있으므로 삭제 연산을 거절
- 연쇄(cascade)는 삭제된 부서 번호 값을 갖는 사원 테이블의 튜플도 함께 삭제
- 널 값으로 대체(nullify)는 삭제연산을 수행한 뒤 삭제된 부서 번호 값을 갖는 사원 테이블의 튜플에서 부서 번호를 null 값으로 대체