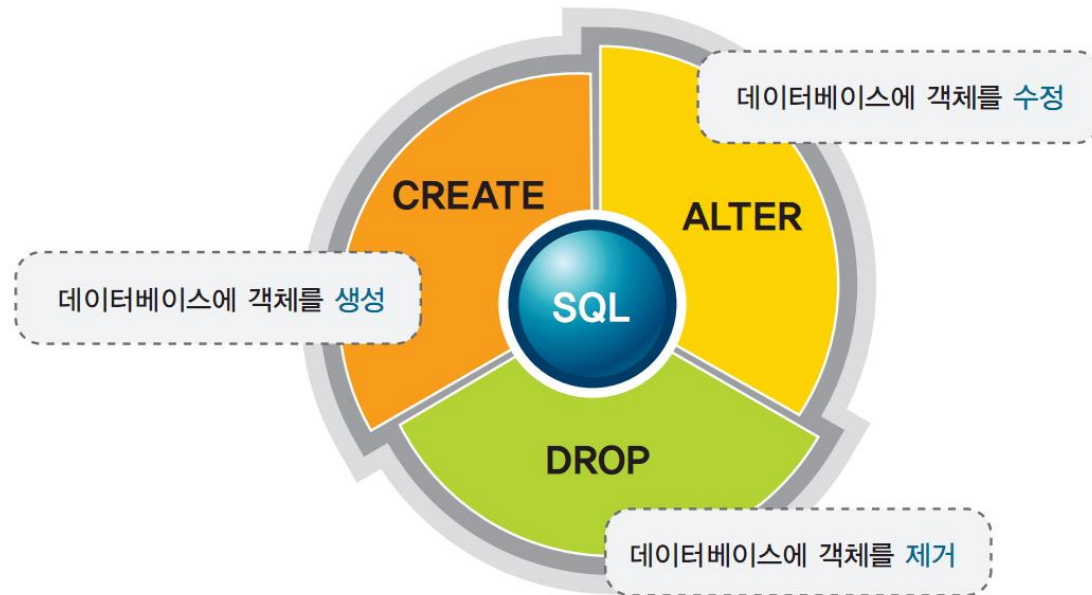


6장. SQL

2 데이터 정의어(DDL)

● 데이터 정의어(DDL)



133 Page

● DDL 명령문의 종류

SQL문	내용
CREATE	데이터베이스 및 객체 생성
DROP	데이터베이스 및 객체 삭제
ALTER	기존에 존재하는 데이터베이스 객체를 변경

2 데이터 정의어(DDL)

2-1 스키마

- 스키마

- 하나의 응용(사용자)에 속하는 테이블과 기타 구성요소(릴레이션, 도메인, 제약조건, 뷰, 권한) 등을 그룹 짓기 위해서 나온 개념
- 스키마는 CREATE SCHEMA 명령문으로 생성

- 이름이 SCOTT인 사용자가 MY_SCHEMA를 생성한 예

예

```
CREATE SCHEMA MY_SCHEMA AUTHORIZATION SCOTT;
```

- 사용하던 스키마를 제거

예

```
DROP SCHEMA MY_SCHEMA;
```

2 데이터 정의어(DDL)

2-2 테이블 생성(CREATE TABLE)

● CREATE TABLE 형식

```
CREATE TABLE 테이블이름  
( {열이름 데이터타입 [NOT NULL] [DEFAULT 값], }+  
  [PRIMARY KEY (열이름_리스트),]  
  {[UNIQUE (열이름_리스트),] }*  
  {[FOREIGN KEY(열이름_리스트)  
    REFERENCES 기본테이블[(열이름_리스트)]  
    [ON DELETE 옵션]  
    [ON UPDATE 옵션] ,] } *  
  [CONSTRAINT 이름] [CHECK(조건식)]);
```

- “[]”로 묶인 부분은 생략 가능한 부분이고, “{ }”로 묶인 부분은 중복 가능 부분을 나타냄
- +는 1번 이상, *는 0번 이상 반복을 나타냄
- +나 * 앞에 ,는 요소들이 여러 번 반복되어 형식 될 수 있음을 나타냄

2 데이터 정의어(DDL)

2-2 테이블 생성(CREATE TABLE)

● CREATE TABLE

- CREATE TABLE문 다음에는 테이블 이름을 기술
- 칼럼에 대해서 칼럼 이름과 함께 데이터 타입과 제약조건을 명시
- CREATE TABLE 문은 세미콜론 ";"으로 끝나야 함

● 이름을 주는 기본 규칙

1. 테이블 이름과 칼럼은 A~Z까지의 문자, 0~9까지의 숫자, 그리고 \$, #, _ (Under Bar)를 사용할 수 있다. 그러나 공백은 사용할 수 없다.
 2. 테이블의 칼럼은 30자를 초과할 수 없고, 예약어를 사용할 수 없다.
 3. 한 테이블 안에서 칼럼 이름은 같을 수 없으며 다른 테이블에서의 칼럼 이름과는 같을 수 있다.
- 1개의 테이블에는 여러 개의 칼럼이 존재하므로 각 칼럼들은 콤마","로 구분되고, 각 칼럼의 데이터 타입은 반드시 지정되어야 함

2 데이터 정의어(DDL)

2-1 스키마

- QL Server의 대표 데이터 타입의 종류

〈표 4.5〉 릴레이션의 정의에 사용되는 오라클의 데이터 타입

데이터 타입	의미
INTEGER 또는 INT	정수형
NUMBER(n, s)	소수점을 포함한 n개의 숫자에서 소수 아래 숫자가 s개인 십진수
CHAR (n) 또는 CHARACTER (n)	n바이트 문자열. n을 생략하면 1
VARCHAR (n) , VARCHAR2 (n) 또는 CHARACTER VARYING (N)	최대 n바이트까지의 가변 길이 문자열
BIT (n) 또는 BIT VARYING (n)	n개의 비트열 또는 최대 n개까지의 가변 비트열
DATE	날짜형. 날짜와 시간을 저장
BINARY_FLOAT	오라클 10g부터 도입되었는데, 32비트에 실수를 저장
BINARY_DOUBLE	오라클 10g부터 도입되었는데, 64비트에 실수를 저장
BLOB	Binary Large Object. 멀티미디어 데이터 등을 저장

2 데이터 정의어(DDL)

2-2 테이블 생성(CREATE TABLE)

- 과정 정보를 저장하는 COURSE 테이블을 생성

```
1: CREATE TABLE COURSE (  
2:   COU_ID      INT      NOT NULL, -- 과정 번호  
3:   COU_NAME    VARCHAR(20) NULL, -- 과정명  
4:   TEA_NAME    VARCHAR(20) NULL, -- 강사명  
5:   PRIMARY KEY(COU_ID)           -- 과정 번호를 기본키로 지정  
6: )
```

- 과정 정보를 저장하는 COURSE 테이블을 생성

```
1: CREATE TABLE STUDENT (  
2:   STU_ID      INTEGER      NOT NULL, -- 학생번호, 널 값 허용 안 함  
3:   STU_NAME    VARCHAR(20) NULL,      -- 학생명  
4:   AGE        INTEGER      NULL,      -- 나이  
5:   STU_EMAIL   VARCHAR(20) NULL,      -- 이메일  
6:   COU_ID      INTEGER      NULL,      -- 소속된 과정의 과정 번호  
7:   PRIMARY KEY(STU_ID),                -- 학생번호를 기본키로 지정  
8:   FOREIGN KEY(COU_ID) REFERENCES COURSE(COU_ID)  
9: )  
10:
```

2 데이터 정의어(DDL)

2-2 테이블 생성(CREATE TABLE)

● Default 값 설정

```
CREATE TABLE STUDENT (  
  STU_ID  INTEGER          NOT NULL,  -- 학생번호, 널 값 허용 안 함  
  STU_NAME  VARCHAR(20)    NULL,      -- 학생명  
  AGE       INTEGER        NULL      CHECK(AGE < 30),  -- 나이  
  -- AGE       INTEGER        NULL      CONSTRAINT AGE_CK CHECK(AGE < 30),  
  -- 나이  
  STU_EMAIL  VARCHAR(20)    NULL,      -- 이메일  
  COU_ID     INTEGER        NULL,      -- 소속된 과정의 과정 번호  
  SEX        VARCHAR        NULL      DEFAULT 'M',  -- 성별 )
```

● SQL 주석문

- 각 줄에 기술한 문장이 무엇인지를 개발자에게 설명하기 위해서는 주석문을 작성해야 한다. 주석문은 -- 로 시작하며 실행을 시켰을 경우 주석 처리한 문장은 실행에서 제외된다.

2 데이터 정의어(DDL)

2-5 제약조건

- 데이터 무결성 제약 조건(Data Integrity Constraint Rule)

데이터 무결성
제약 조건
(Data Integrity
Constraint
Rule)

- 테이블에 부적절한 자료가 입력되는 것을 방지하기 위해서
- 테이블을 생성할 때 각 칼럼에 대해서 형식 하는 여러 가지 규칙

- 정확성을 유지

무결점

- 데이터베이스 내에 데이터의 정확성을 유지하는 것을 의미한다.

2-5 제약조건

- 데이터 무결성 제약 조건(Data Integrity Constraint Rule)

- NOT NULL
- UNIQUE
- PRIMARY KEY
- FOREIGN KEY
- CHECK

2 데이터 정의어(DDL)

2-5 제약조건

- 제약 조건

NOT NULL	NULL을 허용하지 않음
UNIQUE	중복된 값을 허용하지 않고, 항상 유일한 값을 갖도록 함
PRIMARY KEY	NULL을 허용하지 않음. 중복된 값을 허용하지 않음 NOT NULL 조건과 UNIQUE 조건을 결합한 형태임
FOREIGN KEY	참조되는 테이블의 칼럼의 값이 존재하면 허용함
CHECK	저장 가능한 데이터 값의 범위나 조건을 지정해 설정한 값만을 허용함

- 제약조건은 생성은 다음 시기에 생성할 수 있다.

- 테이블 생성할 때
- 테이블을 생성한 후

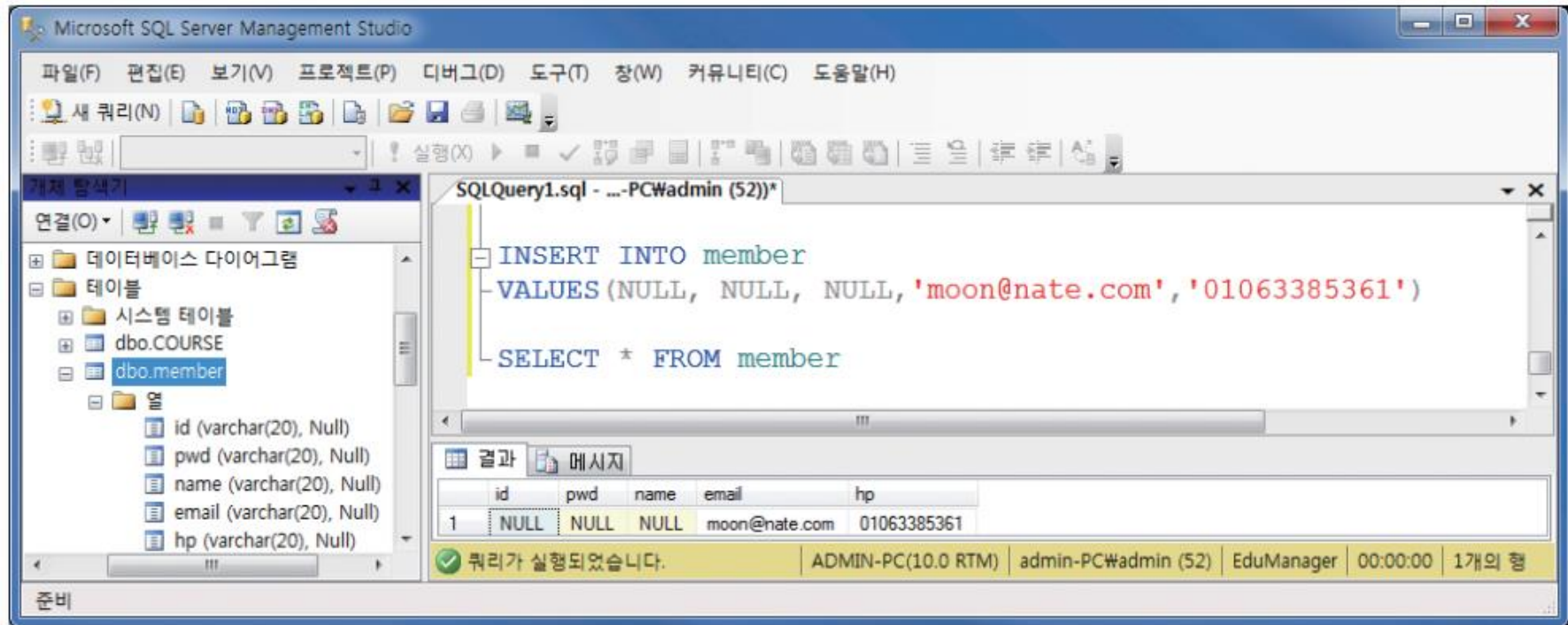
- 제약조건 생성 방법

- COLUMN 레벨
- TABLE 레벨

2 데이터 정의어(DDL)

2-5 제약조건

- 필수 입력을 위한 NOT NULL 제약 조건
 - 널 값 허용



2 데이터 정의어(DDL)

2-5 제약조건

- 필수 입력을 위한 NOT NULL 제약 조건

- 부정확한 회원 정보



147 Page

- NOT NULL 제약 조건

- 해당 칼럼에 데이터를 추가하거나 수정할 때 NULL 값이 저장되지 않게 제약을 걸어주는 것
- 주민번호와 이름과 같이 자료가 꼭 입력되게 하고 싶을 때 사용

NOT NULL
제약 조건

- 해당 칼럼에 데이터를 추가하거나 수정할 때 NULL 값이 저장되지 않게 제약을 걸어주는 것
- 아이디나 이름과 같이 자료가 꼭 입력되게 하고 싶을 때 사용

2 데이터 정의어(DDL)

2-5 제약조건

- NOT NULL 제약조건을 설정하지 않고 테이블 생성하기

- 회원 테이블을 생성

```
CREATE TABLE member(  
    id          varchar(20),  
    pwd         varchar(20),  
    name        varchar(20),  
    email       varchar(20),  
    hp          varchar(20)  
)
```



따라 하기

2 데이터 정의어(DDL)

2-5 제약조건

- **NOT NULL 제약조건을 설정하지 않고 테이블 생성하기**

- 제약조건이 불충분한 테이블로 의미 없는 정보 입력



따라 하기

INSERT INTO member



INSERT문에 자료가 삽입될 칼럼을 지정하지 않았으므로
기본 칼럼 순서대로 **VALUES** 절의 자료가 삽입됨. 칼럼에
NULL 값이 저장됨

VALUES(NULL,NULL,NULL, 'moon@nate.com', '1063385361');

2 데이터 정의어(DDL)

2-5 제약조건

- 제약조건 지정형식

```
column_name  data_type  constraint_type
```



따라 하기

- NOT NULL 제약조건을 설정하여 테이블 생성하기

- NOT NULL 제약 조건을 설정

```
DROP TABLE member ← member 테이블이 존재하면 다시 생성되지 않기에 일단 member 테이블 제거
```

```
CREATE TABLE member(  
  id   varchar(20)  NOT NULL, ← 테이블 생성 시 칼럼에 NOT NULL 제약조건 설정  
  pwd  varchar(20)  NOT NULL,  
  name varchar(20)  NOT NULL,  
  email varchar(20) NULL,  
  hp   varchar(20)  NULL  
)
```


2 데이터 정의어(DDL)

2-5 제약조건

- 필수 입력을 위한 NOT NULL 조건 지정

- 성한 member 테이블에 데이터를 추가 (오류)

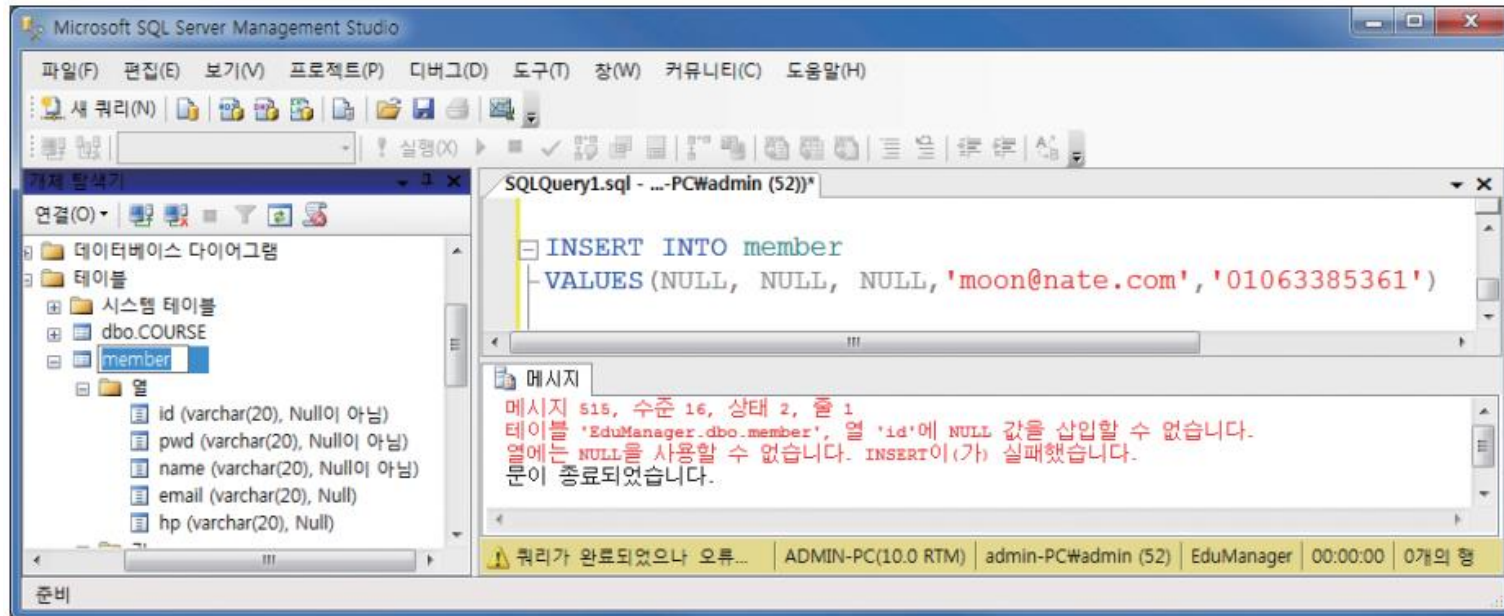
INSERT INTO member ← INSERT문에 자료가 삽입될 칼럼을 지정하지 않았으므로 기본 칼럼 순서대로 VALUES 절 다음에 기술한 값을 저장.
VALUES(NULL, NULL, NULL, 'moon@nate.com', '01063385361')



따라 하기

- 필수 입력을 위한 NOT NULL 조건 지정

- 필수 입력 요구



2 데이터 정의어(DDL)

2-5 제약조건

- 필수 입력을 위한 NOT NULL 조건 지정

- member 테이블에 아이디와 패스워드, 이름에 NULL 값이 아닌 값을 지정

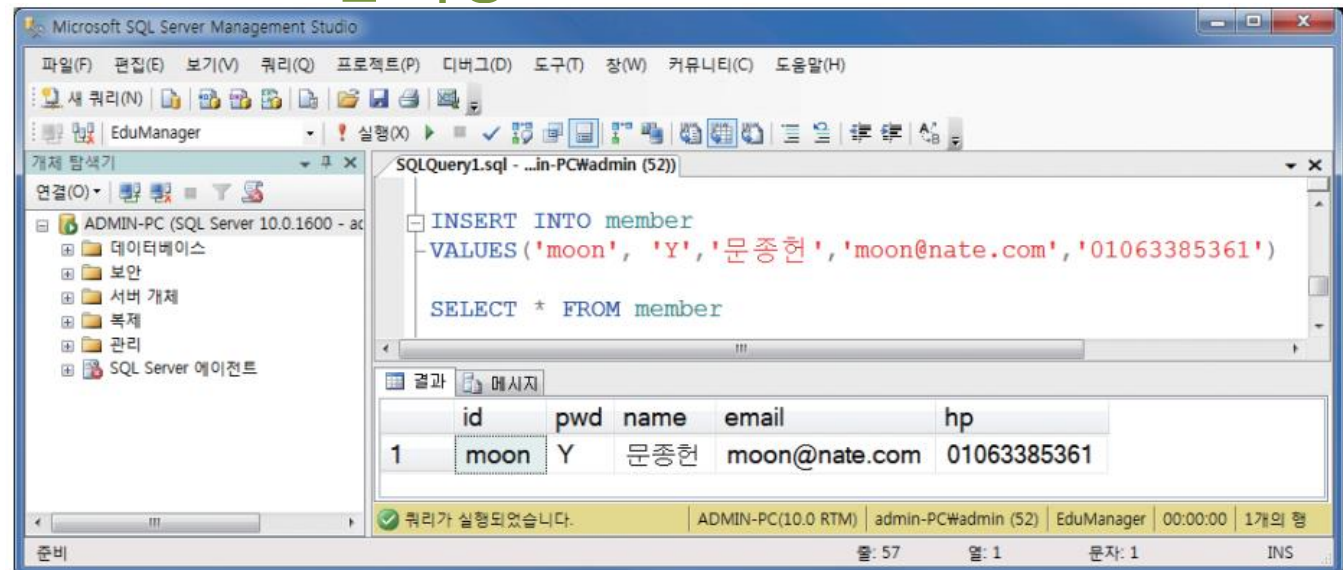
```
INSERT INTO member  
VALUES('moon', 'Y', '문종헌', 'moon@nate.com', '01063385361')  
  
SELECT * FROM member
```



따라 하기

- 필수 입력을 위한 NOT NULL 조건 지정

- 정확한 정보 입력



2 데이터 정의어(DDL)

2-5 제약조건

- 유일한 값을 저장하기 위한 **UNIQUE** 제약 조건

- UNIQUE 제약 조건이란 특정 칼럼에 대해 자료가 중복되지 않게 하는 것

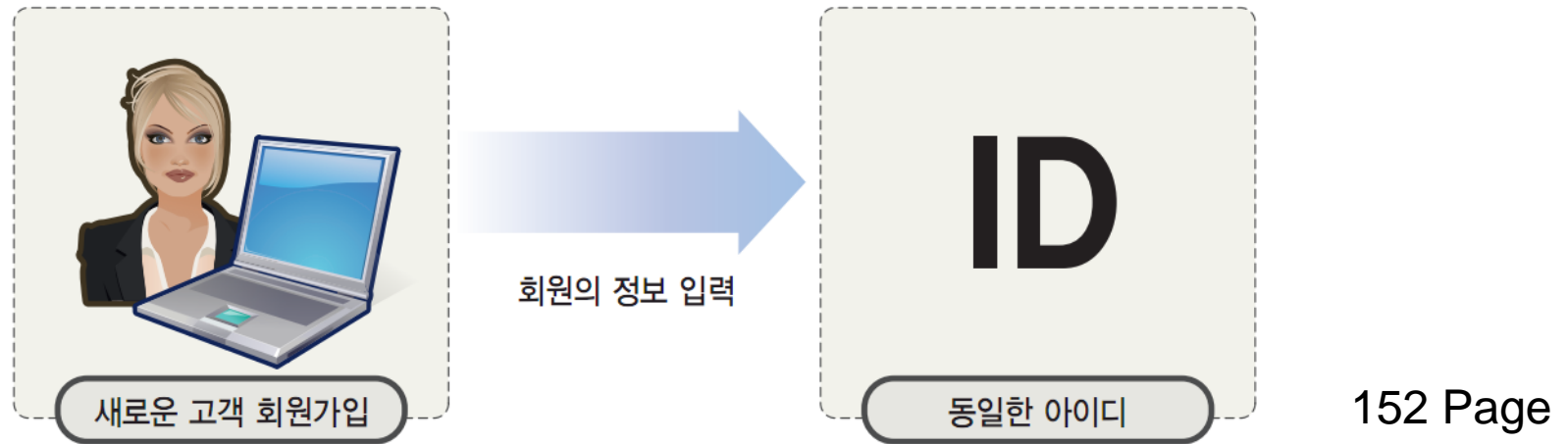
UNIQUE 제약 조건

- 특정 칼럼에 대해 자료가 중복되지 않게 하는 것
- 지정된 칼럼에는 유일한 값이 수록되게 하는 것
- 널값 허용

2 데이터 정의어(DDL)

2-5 제약조건

- 동일한 아이디로 구분 불가능



	id	pwd	name	email	hp
1	moon	Y	문종헌	moon@nate.com	01063385361
2	moon	SOL	오한솔	five@nate.com	01011113121

- 아이디 칼럼에 UNIQUE KEY 제약 조건을 지정하면 중복된 값을 저장할 수 없음

2 데이터 정의어(DDL)

2-5 제약조건

● UNIQUE 제약조건을 설정하여 테이블 생성하기

- member 테이블을 생성하되 아이디를 유일키로 지정
- COLUMN 레벨 - 제약 조건은 칼럼명과 자료형을 기술한 후에 연이어서 UNIQUE를 기술



따라 하기

```
CREATE TABLE member(  
  id varchar(20) UNIQUE,  
  -- id varchar(20) CONSTRAINT mem_id_uk UNIQUE,  
  pwd varchar(20) NOT NULL,  
  name varchar(20) NOT NULL,  
  email varchar(20) NULL,  
  hp varchar(20) NULL );
```

- TABLE 레벨 - 테이블 마지막에 기술

```
CREATE TABLE member(  
  id varchar(20),  
  pwd varchar(20) NOT NULL,  
  name varchar(20) NOT NULL,  
  email varchar(20) NULL,  
  hp varchar(20) NULL  
  CONSTRAINT mem_id_uk UNIQUE(id))
```

2 데이터 정의어(DDL)

2-5 제약조건

- UNIQUE 제약조건을 설정하여 테이블 생성하기

- 생성한 member 테이블에 데이터를 추가

```
INSERT INTO member  
VALUES('moon', 'Y', '문종현', 'moon@nate.com', '01063385361')
```

- 'moon'이란 자료를 입력하였는데 다시 동일한 아이디를 입력(오류)

```
INSERT INTO member  
VALUES('moon', 'SOL', '오한솔', 'five@nate.com', '01011113121')
```

- 아이디 중복 허용 안 함



따라 하기

2 데이터 정의어(DDL)

2-5 제약조건

- 개체의 무결성을 위한 PRIMARY KEY 제약 조건

- 유일키 제약 조건의 널 값 허용

유일키 (PRIMARY KEY) 제약 조건을
지정한 컬럼

- 중복된 데이터 저장 못 함
- NULL 값 저장 허용

	id	pwd	name	email	hp
1	NULL	Y	문종헌	moon@nate.com	01063385361
2	NULL	SOL	오한솔	five@nate.com	01011113121

테이블 내의 해당 행을 다른 행과 구분할 수 있도록 하는 컬럼은 반드시 존재해야 함

유일하면서도 **NULL** 값을 허용하지 말아야 함

2 데이터 정의어(DDL)

2-5 제약조건

- 개체의 무결성을 위한 PRIMARY KEY 제약 조건

기본키
(PRIMARY KEY)
제약 조건

- UNIQUE 제약 조건과 NOT NULL 제약 조건을 모두 갖는 것

PRIMARY KEY는 NOT NULL 조건과 UNIQUE 조건을 결합한 형태로서 NULL을 허용하지 않고 중복된 값을 허용하지 않는다. 테이블에 하나만 선언할 수 있다.

2-5 제약조건

● 기본키 제약 조건

- 과정 번호 칼럼이 반드시 입력되고 유일한 값만을 가질 수 있도록 하는 다양한 무결성 제약 조건들 중의 한 가지
- 중복된 값을 저장할 수 없도록 하는 개체 무결성을 유지하기 위해서 과정 번호(COU_ID)

과정 번호(COU_ID) 칼럼은 중복된 값을 저장할 수 없는 기본키이다.

결과		메시지	
	COU_ID	COU_NAME	TEA_NAME
1	10	모바일	성윤정
2	20	자바	김혜경
3	30	윈도우	활연주

COURSE 테이블에 10번 과정이 이미 존재하고 있기 때문에 10번 과정을 추가하려고 했을 때 에러가 발생한다.

```
SQLQuery1.sql - ...-PCWadmin (52))*  
INSERT INTO course  
VALUES (10, '웹 표준', '전혜영')
```

메시지
메시지 2627, 수준 14, 상태 1, 줄 2
PRIMARY KEY 제약 조건 'PK COURSE AEC69E201273C1CD'을(를) 위반했습니다.
개체 'dbo.COURSE'에 중복 키를 삽입할 수 없습니다.
문이 종료되었습니다.

2 데이터 정의어(DDL)

2-5 제약조건

● PRIMARY KEY 제약조건 설정하기

- COLUMN 레벨-칼럼명과 자료형을 기술한 후에 연이어서 PRIMARY KEY를



따라 하기

```
CREATE TABLE member(  
  id varchar(20) NOT NULL PRIMARY KEY ,  
  -- id varchar(20) NOT NULL CONSTRAINT mem_id_pk PRIMARY KEY ,  
  pwd varchar(20) NOT NULL,  
  name varchar(20) NOT NULL,  
  email varchar(20) NULL,  
  hp varchar(20) NULL,  
  PRIMARY KEY (id) )
```

- TABLE 레벨

```
CREATE TABLE member(  
  id varchar(20) NOT NULL ,  
  pwd varchar(20) NOT NULL,  
  name varchar(20) NOT NULL,  
  email varchar(20) NULL,  
  hp varchar(20) NULL,  
  CONSTRAINT mem_id_pk PRIMARY KEY (id) )
```

2-5 제약조건

● PRIMARY KEY 제약조건 설정하기

- 생성한 member 테이블에 데이터를 추가

```
INSERT INTO member  
VALUES('moon', 'Y', '문종헌', 'moon@nate.com', '01063385361')
```

- 'moon'이란 자료를 입력하였는데 다시 동일한 아이디를 입력(오류)

```
INSERT INTO member  
VALUES('moon', 'SOL', '오한솔', 'five@nate.com', '01011113121')
```

- 아이디 중복 허용 안 함



따라 하기

2 데이터 정의어(DDL)

2-5 제약조건

● PRIMARY KEY 제약조건 설정하기

- 본키로 지정된 아이디에 NULL 값을 저장

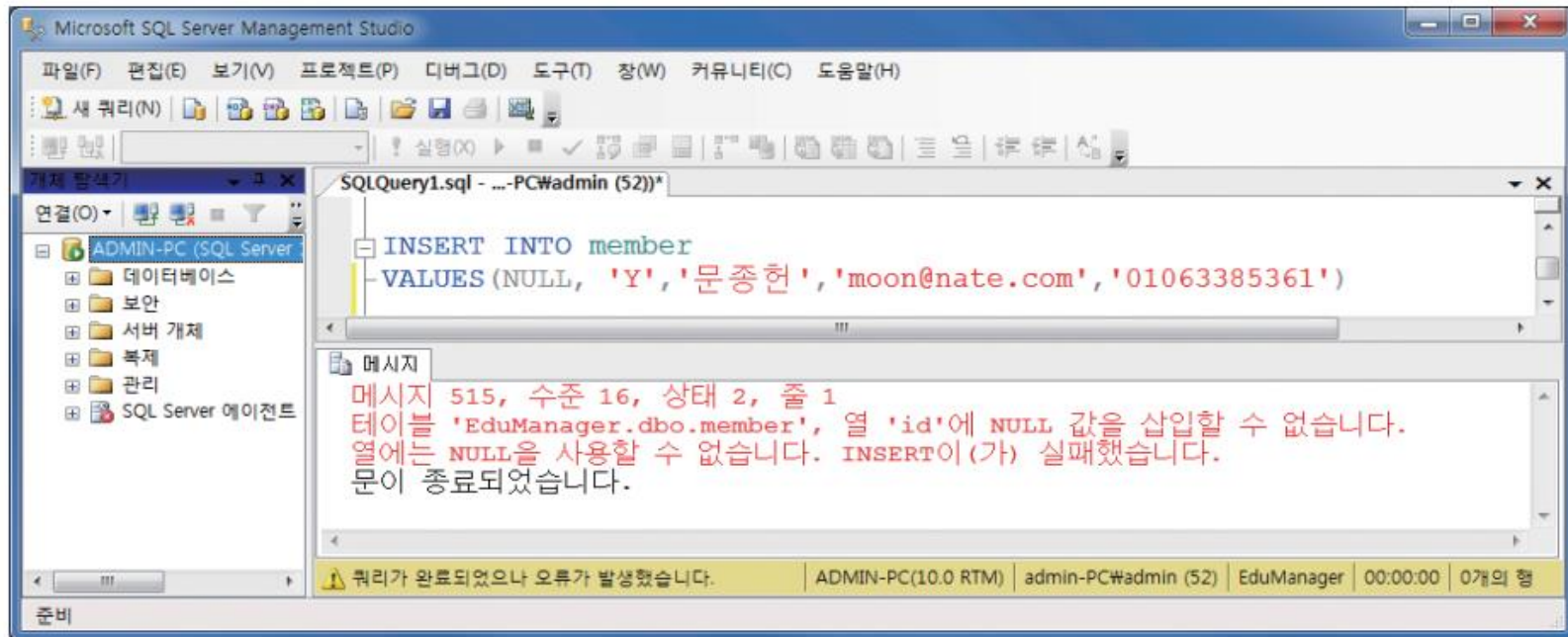
```
INSERT INTO member  
VALUES(NULL, 'Y', '문종헌', 'moon@nate.com', '01063385361')
```



따라 하기

● PRIMARY KEY 제약조건 설정하기

- 널 값 허용하지 않는 기본키 제약 조건



2 데이터 정의어(DDL)

2-5 제약조건

- 참조의 무결성을 위한 FOREIGN KEY 제약 조건

참조의 무결성

- 테이블 사이의 관계에서 발생하는 개념

FOREIGN KEY

- 일반적으로 업무 규칙에서 주종 관계가 있는 두 테이블 간에 사용되며 종속되는 테이블의 키 칼럼이 주가 되는 테이블의 PRIMARY KEY 또는 UNIQUE 칼럼을 참조함

- 참조의 무결성을 위한 FOREIGN KEY 제약 조건

- 과정 테이블

결과		메시지	
	COU_ID	COU_NAME	TEA_NAME
1	10	모바일	성윤정
2	20	자바	김혜경
3	30	윈도우	황연주
4	40	웹표준	전혜영

158 Page

2 데이터 정의어(DDL)

2-5 제약조건

● ERD를 보고 데이터베이스를 구현할 때

- 과정이나 학생과 같은 개체는 테이블로, 형식하고 소속이란 관계
- 조의 무결성을 위한 특정 칼럼에 외래키 제약 조건으로 정의
- "학생은 교육센터 내에 존재하는 과정에 수강해야 한다."

● 참조의 무결성을 위한 부모, 자식 테이블 관계



2 데이터 정의어(DDL)

2-5 제약조건

● 과정 테이블과 학생 테이블 관계

COU_ID가 부모 키가
되려면 기본키 혹은
UNIQUE 제한 조건으로
지정되어야 함

	COU_ID	COU_NAME	TEA_NAME
1	10	모바일	성윤정
2	20	자바	김혜경
3	30	윈도우	황연주
4	40	웹표준	전혜영

	STU_ID	STU_NAME	AGE	STU_EMAIL	COU_ID	SEX
1	101	문종현	24	moon@nate.com	10	M
2	102	오한솔	22	five@nate.com	20	M
3	103	제웅석	22	again@nate.com	20	M
4	104	정국철	22	cook@nate.com	20	M
5	105	박홍진	24	red@nate.com	10	M
6	106	김현우	22	kim@nate.com	20	M
7	107	박시준	22	season@nate.com	20	M
8	108	김준형	24	brother@nate.com	10	M
9	109	문혜진	22	sun@nate.com	20	F
10	110	박기석	24	flag@nate.com	10	M
11	111	윤효선	24	good@nate.com	30	F
12	112	안창범	24	window@nate.com	30	M
13	113	공지훈	24	empty@nate.com	10	M
14	114	이봉림	24	bbong@nate.com	10	M
15	115	안창범	24	chang@nate.com	30	M
16	116	장희성	24	shine@nate.com	10	M

학생 테이블에서
외래키인 COU_ID는
과정 테이블의
기본키인 COU_ID를
참조하므로 과정
테이블에 존재하는
과정 번호만 입력
가능함. 과정 테이블
내의 COU_ID를
부모 키라고 함

2 데이터 정의어(DDL)

2-5 제약조건

- 무결성 제약 조건에 위배

결과

메시지

	COU_ID	COU_NAME	TEA_NAME
1	10	모바일	성윤정
2	20	자바	김혜경
3	30	윈도우	황연주
4	40	웹표준	전혜영

COURSE 테이블에는 10, 20, 30, 40만 존재하기에 50번 과정에 새로운 학생을 추가할 수 없다.

```
SQLQuery1.sql - ...-PCWadmin (52))*  
INSERT INTO STUDENT VALUES  
(117, '김희선', 24, 'shine@nate.com', 50, F')
```

메시지
메시지 547, 수준 16, 상태 0, 줄 1
INSERT 문이 FOREIGN KEY 제약 조건 "FK__STUDENT__COU_ID__2B3F6F97"과(와) 충돌했습니다.
데이터베이스 "EduManager", 테이블 "dbo.COURSE", column 'COU_ID'에서 충돌이 발생했습니다.
문이 종료되었습니다.

2 데이터 정의어(DDL)

2-5 제약조건

- 참조 무결성을 유지하기 위해서 학생(STUDENT) 테이블의 과정 번호(COU_ID)를 외래키로 지정

```
CREATE TABLE STUDENT (  
  STU_ID      INTEGER      NOT NULL,    -- 학생번호, 널 값 허용 안 함  
  STU_NAME    VARCHAR(20)   NULL,       -- 학생명  
  AGE         INTEGER      NULL,       -- 나이  
  STU_EMAIL   VARCHAR(20)   NULL,       -- 이메일  
  COU_ID      INTEGER      NULL,       -- 소속된 과정의 과정 번호  
  PRIMARY KEY(STU_ID),              -- 학생번호를 기본키로 지정  
  -- -- 과정 번호를 외래키로 지정  
  FOREIGN KEY(COU_ID) REFERENCES COURSE(COU_ID)  
  -- CONSTRAINT stu_cou_fk FOREIGN KEY(COU_ID) REFERENCES COURSE(COU_ID)  
  -- ON DELETE CASCADE  
  -- ON DELETE SET NULL  
)
```

- ON DELETE CASCADE : 해당 튜플을 삭제
- ON DELETE SET NULL : 해당 튜플에 null 값 입력

2-5 제약조건

● CHECK 제약조건 설정하기

- 입력되는 값을 체크하여 설정된 값 이외의 값이 들어오면 오류
- 조건: 데이터의 값의 범위나 특정 패턴의 숫자나 문자값을 설정



따라 하기

```
CREATE TABLE STUDENT (  
  STU_ID  INTEGER          NOT NULL,  -- 학생번호, 널 값 허용 안 함  
  STU_NAME  VARCHAR(20)  NULL,        -- 학생명  
  AGE      INTEGER        NULL  CHECK(AGE < 30),  -- 나이  
  -- AGE      INTEGER        NULL  CONSTRAINT AGE_CK CHECK(AGE < 30),  -- 나이  
  STU_EMAIL  VARCHAR(20)  NULL,        -- 이메일  
  COU_ID     INTEGER      NULL,        -- 소속된 과정의 과정 번호  
  SEX       VARCHAR      NULL  DEFAULT 'M',  -- 성별 )
```

- TABLE 레벨

```
CREATE TABLE STUDENT (  
  STU_ID  INTEGER          NOT NULL,  -- 학생번호, 널 값 허용 안 함  
  STU_NAME  VARCHAR(20)  NULL,        -- 학생명  
  AGE      INTEGER        NULL,        -- 나이  
  STU_EMAIL  VARCHAR(20)  NULL,        -- 이메일  
  COU_ID     INTEGER      NULL,        -- 소속된 과정의 과정 번호  
  SEX       VARCHAR      NULL  DEFAULT 'M',  -- 성별  
  CONSTRAINT AGE_CK CHECK(AGE < 30) )
```

2 데이터 정의어(DDL)

2-5 제약조건

● CHECK 제약조건 설정하기

- 생성한 테이블에 데이터를 추가

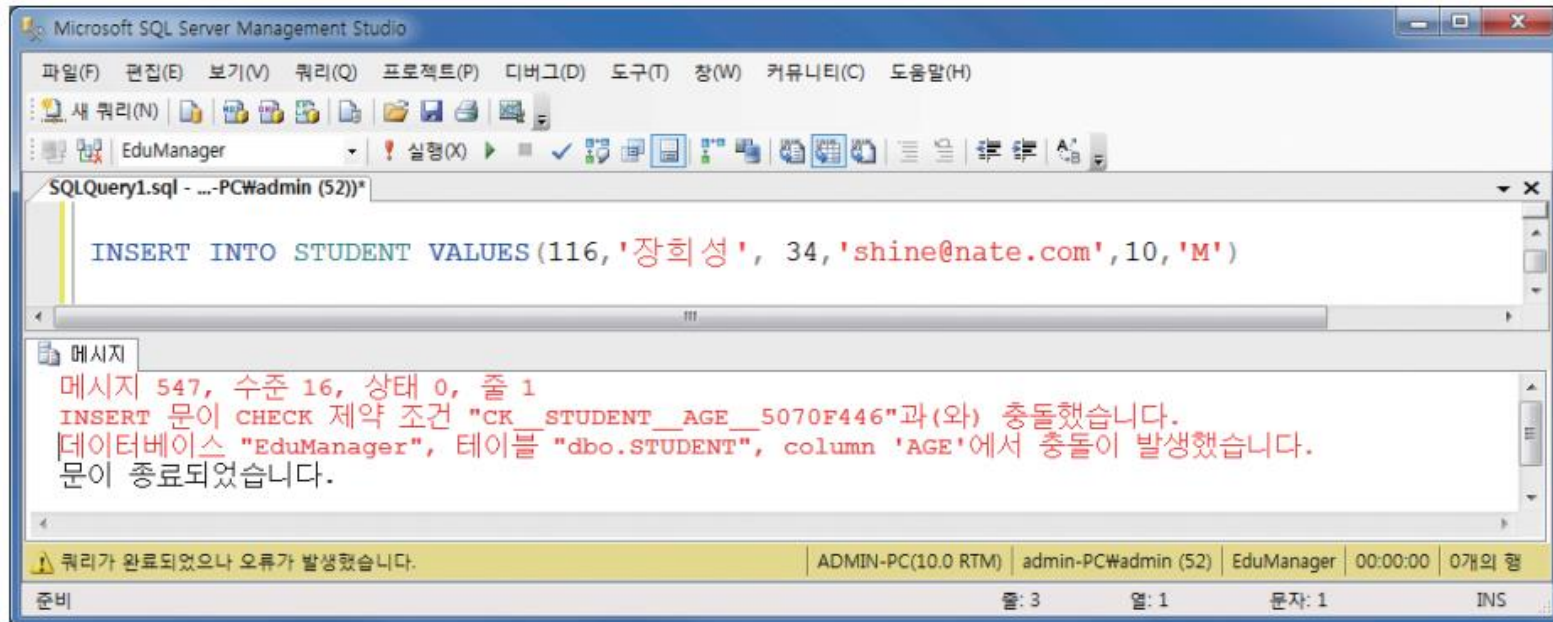
```
INSERT INTO STUDENT VALUES(116,'장희성', 34,'shine@nate.com',10,'M')
```



따라 하기

● CHECK 제약조건 설정하기

- check 제약 조건 위배-1



164 Page

2-5 제약조건

- 제약조건 보기

```
SELECT constraint_name, constraint_type, search_condition  
FROM user_constraints  
WHERE table_name = 'EMPLOYEES';
```

```
SELECT constraint_name, column_name  
FROM user_cons_columns  
WHERE table_name = 'EMPLOYEES';
```

2-3 테이블 변경(ALTER TABLE)

● ALTER TABLE 명령

```
ALTER TABLE 테이블명  
  ([ADD 칼럼명 데이터_타입] [DEFAULT 값] |  
   [DROP 칼럼명] [CASCADE] |  
   [ALTER 칼럼명 (DROP DEFAULT | SET DEFAULT 값)] |  
   [RENAME 칼럼명 TO 새로운 칼럼명] |  
  
  [ADD CONSTRAINT 제약조건] |  
  [DROP CONSTRAINT 제약조건명] |  
);
```

2-3 테이블 변경(ALTER TABLE)

● 테이블 컬럼 추가하기(ALTER TABLE ADD)

- [문법] ALTER TABLE 테이블명 ADD(컬럼명 데이터타입(사이즈));
- (EX) STUDENT라는 테이블에 sex이라는 컬럼을 CHAR(1) 타입으로 추가할 때
ALTER TABLE STUDENT ADD(sex CAHR(1));

● 테이블 컬럼 수정하기(ALTER TABLE MODIFY)

- [문법] ALTER TABLE 테이블명 MODIFY(컬럼명 데이터타입(사이즈));
- (EX) STUDENT 라는 테이블에 AGE 라는 컬럼을 NUNBER(3) 타입으로 수정할 때
ALTER TABLE STUDENT MODIFY(AGE NUMBER(3));

● 테이블 컬럼 삭제하기(ALTER TABLE DROP)

- [문법] ALTER TABLE 테이블명 DROP COLUMN 컬럼명
- (EX) STUDENT 라는 테이블에 SEX 이라는 컬럼을 삭제할 때
ALTER TABLE STUDENT DROP COLUMN SEX;

● 테이블 컬럼 이름 변경하기(ALTER TABLE RENAME)

- [문법] ALTER TABLE 테이블명 RENAME COLUMN 원래컬럼명 TO 바꿀컬럼명;
- (EX) STUDENT 라는 테이블에 STU_NAME 이라는 컬럼을 STU_FIRST_NAME으로 변경할 때
ALTER TABLE STUDENT RENAME COLUMN STU_NAME TO STU_FIRST_NAME;

2-3 테이블 변경(ALTER TABLE)

● 테이블 제약조건 추가하기(ALTER TABLE ADD CONSTRAINT)

- [문법] ALTER TABLE 테이블명 ADD CONSTRAINT (타입 컬럼명);
- (EX) STUDENT라는 테이블에 COU_ID이라는 컬럼에 외래키 추가할 때
ALTER TABLE STUDENT ADD(sex CAHR(1));
ALTER TABLE STUDENT ADD CONSTRAINT stu_cou_fk FOREIGN KEY(manager_id)
REFERENCES employees(employee_id)

● 테이블 컬럼 삭제하기(ALTER TABLE DROP CONSTRAINT)

- [문법] ALTER TABLE 테이블명 DROP CONSTRAINT 제약조건명
- (EX) STUDENT 라는 테이블에 COU_ID에 외래키 삭제할 때
ALTER TABLE STUDENT DROP CONSTRAINT stu_cou_fk

2 데이터 정의어(DDL)

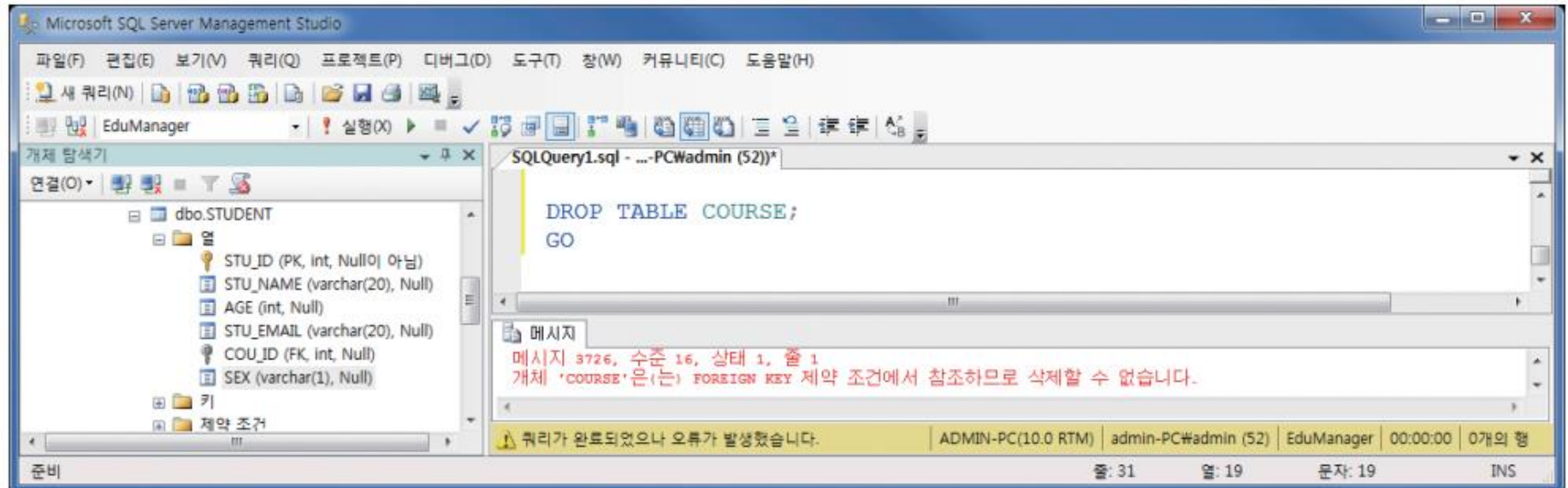
2-4 테이블 제거(DROP TABLE)

● DROP TABLE 명령

예

DROP TABLE COURSE;

● 테이블 제거 실패 (외래키 제약조건 위반)



2 데이터 정의어(DDL)

2-4 테이블 제거(DROP TABLE)

● DROP TABLE 명령

- 1: DROP TABLE STUDENT
- 2: DROP TABLE COURSE

● 테이블 제거 성공

