

# 데이터 무결성

<정리 및 실습>

# 목차

## 1. DDL(DATA DEFINITION LANGUAGE)

1-1 DB자료형

1-2 NULL의미

1-3 테이블 생성 (CREATE)

1-4 테이블 수정(ALTER) - (ADD,DROP,MODIFY)

1-5 테이블 삭제 (DROP)

## 2. DML (DATA MANIPULATION LANGUAGE)

2-1 제약조건(CONSTRAINT)의 종류

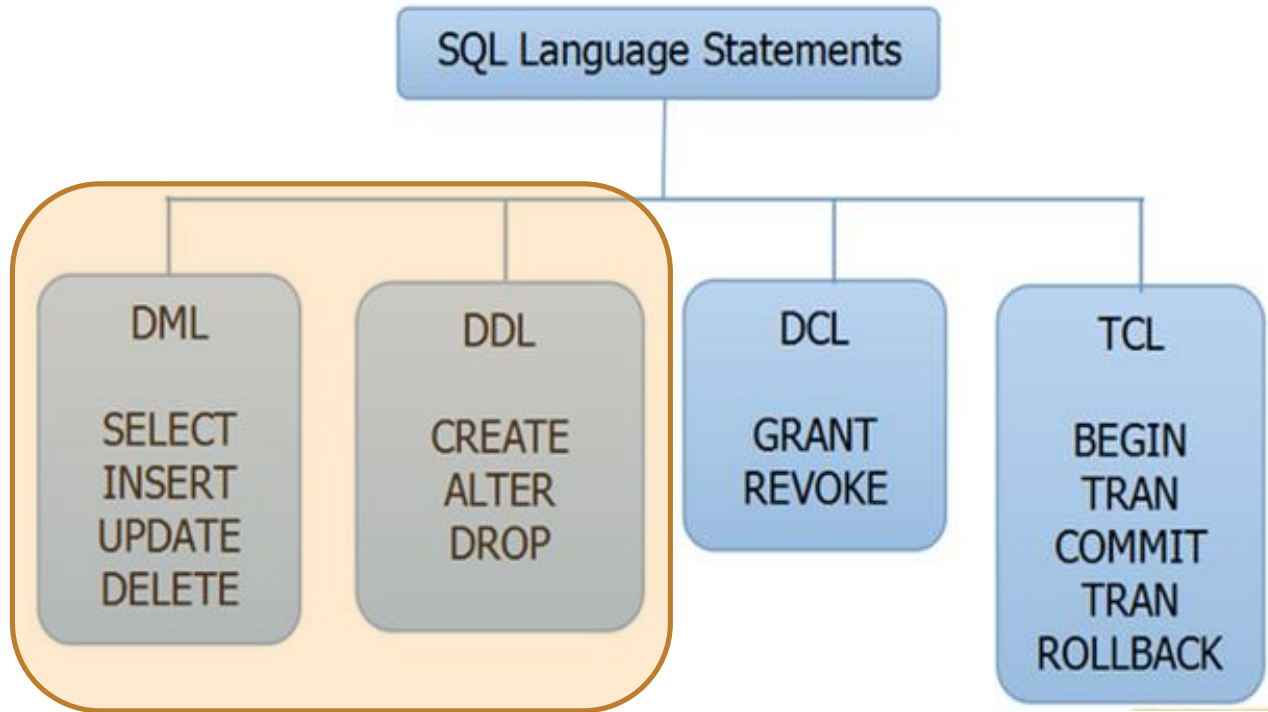
2-2 INSERT

2-3 UPDATE

2-4 DELETE

2-5 SELECT

## 3. 결론



## 1-1 DB 자료형

DATA TYPE : 자료를 받아들이는 공간 = 자료의 유형별로 나누는 기준  
( CHAR(s) , VARCHAR(s), NUMBER , DATE ,,,,,,)

O	R	A	C	L	E
---	---	---	---	---	---

O	R	A	C	L	E			
---	---	---	---	---	---	--	--	--

### VARCHAR2(s)

가변형 데이터 공간  
데이터에 의한 저장공간 자동할당  
저장공간 분배 효율

### CHAR(s)

정적형 데이터 공간  
남은공간 **NULL**이 자동입력  
남는 저장공간 생김

### DB에서 NULL이란?

공백 **X**

미확정, 알수없는 **(unknown)**

할당받지 않은 값, 아직 무엇인지 모르는 값

연산, 할당, 비교 불가능

-숫자에 **NULL**을 연산 시도 >>>> **NULL**

-‘=’ 연산자로 판단 불가 >>>> **Is null or Is not null** 사용

-**NULL**을 숫자초기값으로 지정하는 함수 >>>> **NVL(대상칼럼, 초기값)**

## 1-3 테이블 생성 (CREATE)

```
9  --테이블을 생성하자.
10 CREATE TABLE MYTABLE(
11     NAME VARCHAR2(20) NOT NULL,
12     AGE INT,
13     ADDR VARCHAR2(20)
14 )
15
16 SELECT * FROM MYTABLE;
17
18 INSERT INTO MYTABLE VALUES('홍길동',10,'서울 서초');
19 INSERT INTO MYTABLE VALUES('홍길동',10,'서울 서초');
20 INSERT INTO MYTABLE VALUES(NULL,NULL,NULL);
21
22 SELECT * FROM MYTABLE;
23
```

NAME	AGE	ADDR
------	-----	------

----	----	----
------	------	------

NAME	AGE	ADDR
------	-----	------

----	----	----
------	------	------

홍길동	10	서울 서초
-----	----	-------

홍길동	10	서울 서초
-----	----	-------

ORA-01400: NULL을 ("PLAYDATA"."MYTABLE"."NAME")  
안에 삽입할 수 없습니다  
(0 rows affected)

## 1-4 테이블수정(ALTER) - (ADD,DROP,MODIFY,,)

```
select * from address;  
select * from myaddr;
```

```
--address의 이름을 myaddr 테이블로 변경하자  
alter table address rename to myaddr;
```

```
--myaddr 테이블의 tel 컬럼이름을 phone으로 변경  
alter table myaddr rename column tel to phone;
```

```
--myaddr 테이블에 tel를 추가하자  
alter table myaddr  
add tel varchar2(20) not null;
```

```
--myaddr 테이블에 tel컬럼 삭제하자  
alter table myaddr  
drop column tel;
```

NAME	ADDR	TEL
남희영	한국	010-111-1111
남땡땡	미국	010-222-2222

NAME	ADDR	PHONE
남희영	한국	010-111-1111
남땡땡	미국	010-222-2222

NAME	ADDR	PHONE	TEL
남희영	한국	010-111-1111	NULL
남땡땡	미국	010-222-2222	NULL

NAME	ADDR	PHONE
남희영	한국	010-111-1111
남땡땡	미국	010-222-2222

ORA-01758: NOT NULL 열을 추가하기 위해서는 테이블이 비어 있어야 합니다  
(0 rows affected)

## 1-5 테이블 삭제 (DROP) & Recyclebin

--현재 있는 테이블 보기

```
select table_name from user_tables;
```

TABLE_NAME
DEPT
EMP
BONUS
SALGRADE
TEXT
MYADDR
MYTABLE

--휴지통에 넣지 않고 바로 삭제하자 >> 아예삭제

```
drop table myaddr purge;
```

```
select table_name from user_tables;
```

TABLE_NAME
DEPT
EMP
BONUS
SALGRADE
TEXT
MYTABLE

6 개의 행이 선택되었습니다.

--휴지통에 버리자

```
drop table myaddr;  
desc recyclebin;
```

OBJECT_NAME	ORIGINAL_NAME	OPERATION	TYPE	TS_NAME	CREATETIME	DROPTIME	DROPSCN	PARTITION_NAME	CAN_UNDROP	CAN_PURGE	RELATED	BASE_OBJECT	PURGE_OBJECT	SPACE
-------------	---------------	-----------	------	---------	------------	----------	---------	----------------	------------	-----------	---------	-------------	--------------	-------

--완전 삭제되지 않은 테이블 보기

```
select ORIGINAL_NAME from recyclebin;
```

MYTABLE
MYTABLE
MYTABLE
MYADDR
ADDRESS
MYADDR
MYADDR
MYADDR
MYADDR
MYADDR

```
SQL> purge recyclebin;
```

휴지통이 지워졌습니다.

```
SQL> select ORIGINAL_NAME from recyclebin;
```

선택된 레코드가 없습니다.



## 2-1 제약조건(CONSTRAINT)

: 컬럼마다 정하는 규칙 테이블에 올바른 데이터만 입력받고 잘못된 데이터는 들어오지 못하도록 함

조건 이름	의 미
NOT NULL	NULL값이 입력되지 못함
UNIQUE	중복된 값이 입력되지 못함
PRIMARY KEY	NOT NULL + UNIQUE의 의미를 가짐 테이블 내에서 데이터들끼리의 <b>유일성</b> 을 보장하는 컬럼에 설정할 수 있으며 테이블 당 <b>1개</b> 만 설정할 수 있음
FOREIGN KEY	다른 테이블의 칼럼을 참조해서 무결성 검사를 하게 됨
CHECK	이 조건으로 설정된 값만 입력을 허용하고 나머지는 거부

# 제약조건 (CONSTRAINT)

부서 테이블

DEPTNO	DNAME	COLLEGE	LOC
101	컴퓨터공학과	100	1호관
102	멀티미디어학과	100	2호관
.....	.....	...	...
100	정보미디어학부	10	
10	공과대학		

## ① 기본 키 무결성 제약조건

- STUDNO 컬럼에는 값이 반드시 존재해야 함
- STUDNO 컬럼의 모든 값은 반드시 유일해야 함

## ② 고유 키 무결성 제약조건

- USERID, IDNUM 컬럼의 모든 값은 반드시 유일해야 함

학생 테이블

STUDNO	NAME	USERID	GRADE	IDNUM	BIRTHDATE	TEL	HEIGHT	WEIGHT	DEPTNO	PROFNO
10101	전안하	Jun123	4	7904021369824	79/04/02	051)781-2158	176	72	101	9903
20101	이동훈	Dals	1	8312101128467	83/12/10	055)426-1752	172	64	201	
.....	.....	.....							.....	
10203	윤진욱	Samba7	3	7904021358671	79/04/02	053)487-2698	171	70	102	9905
10107	이광훈	huriky	4	8109131276431	81/10/13	055)736-4981	175	92	101	9903
.....	.....	.....							.....	

## ③ NOT NULL 무결성 제약조건

- NAME 컬럼에는 값이 반드시 존재해야 함
- 컬럼 값으로 NULL을 가질 수 없음

## ④ CHECK 무결성 제약조건

- GRADE 컬럼 값은 반드시 1, 2, 3, 4 중 하나여야 함

## ⑤ 참조키 무결성 제약조건

- 학생 테이블의 DEPTNO 컬럼의 값은 부서 테이블의 DEPTNO 컬럼의 값 중 하나와 일치해야 함

입력 예 (1)

30101	홍길동	gdhong	1	8605051827463	86/05/05	172	182	78	201	
-------	-----	--------	---	---------------	----------	-----	-----	----	-----	--

입력성공

입력 예 (2)

10101		guest	5	8011301898765					500	9903
-------	--	-------	---	---------------	--	--	--	--	-----	------

입력실패

기본키

NOT NULL

CHECK

고유키

참조키

CONSTRAINT\_NAME: SYS\_제약조건타입 + 식별값

```
create table MyUSER02(  
  idx number constraint PKIDX primary key,  
  id varchar2(10) constraint unid unique,  
  name varchar2(20) constraint notname not null,  
  phone varchar2(15),  
  address varchar2(50),  
  score number(6, 2) constraint CHSCORE check(score >= 0 and score <= 100),  
  subject_code number(5),  
  hire_date date default sysdate,  
  marriage char(1) default 'N' constraint CHMARR check (marriage in ('Y','N')));
```

```
select * from MyUSER02;
```



```
select constraint_name,  
constraint_type,  
SEARCH_CONDITION  
from user_constraints  
where table_name = 'MYUSER02'
```

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION
NOTNAME	C	"NAME" IS NOT NULL
CHSCORE	C	score >= 0 and score <= 100
CHMARR	C	marriage in ('Y','N')
PKIDX	P	NULL
UNID	U	NULL

```

create table MyUSER02(
  idx number constraint PKIDX primary key,
  id varchar2(10) constraint uid unique,
  name varchar2(20) constraint notname not null,
  phone varchar2(15),
  address varchar2(50),
  score number(6, 2) constraint CHSCORE check(score >= 0 and score <= 100),
  subject_code number(5),
  hire_date date default sysdate,
  marriage char(1) default 'N'constraint CHMARR check (marriage in ('Y','N')));

```

=====무결성 제약 조건 =====

```

insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');

```

```

insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');
-- bbb가 두번실행하게 되면 유일 키값(unid) 위반 >>> 이유 - bbb중복

```

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'null', '02-222-2222', '서울', 120, 100, null, 'Y');
---체크 제약조건 >>> 이유 - name은 not null이어야 하기 때문

```

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'Y');
--체크 제약조건 >>> 이유 : 점수 제약조건 위배

```

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'y');
-- 체크제약조건 >>> 이유 : marriage in(Y,N) 대문자로 짤랐기 때문...

```

# 기본키 무결성 제약 조건(PRIMARY KEY)

기본키 무결성 제약조건 존재 : 컬럼의 모든 값이 유일하고, NULL을 허용하지 않음

STUDNO	NAME	USERID	GRADE	IDNUM	BIRTHDATE	TEL	HEIGHT	WEIGHT	DEPTNO	PROFNO
10101	전인하	Jun123	4	7904021369824	79/04/02	051)781-2158	176	72	101	9903
20101	이동훈	Dals	1	8312101128467	83/12/10	055)426-1752	172	64	201	
.....	.....	.....							.....	
10203	윤진욱	Samba7	3	7904021358671	79/04/02	053)487-2698	171	70	102	9905
10107	이광훈	huriky	4	8109131276431	81/10/13	055)736-4981	175	92	101	9903
.....	.....	.....							.....	

입력 예 (1)

30101	홍길동	gdhong	1	8605051827463	86/05/05		182	78	201	
-------	-----	--------	---	---------------	----------	--	-----	----	-----	--

입력성공

입력 예 (2)

	성춘향	chsung	2		86/05/05		182	78	201	
--	-----	--------	---	--	----------	--	-----	----	-----	--

입력실패

- 기본 키 무결성 제약조건은 NULL을 허용하지 않음

입력 예 (3)

10101		Jun123	4	8312101128467					101	9903
-------	--	--------	---	---------------	--	--	--	--	-----	------

입력실패

- 기본 키 무결성 제약조건은 중복을 허용하지 않음, STUDNO 의 값 10101 은 이미 존재함



중복된 값 입력 불가.

```
create table MyUSER02(  
  idx number constraint PKIDX primary key,  
  id varchar2(10) constraint uid unique,  
  name varchar2(20) constraint notname not null,  
  phone varchar2(15),  
  address varchar2(50),  
  score number(6, 2) constraint CHSCORE check(score >= 0 and score <= 100),  
  subject_code number(5),  
  hire_date date default sysdate,  
  marriage char(1) default 'N' constraint CHMARR check (marriage in ('Y','N')));
```

=====무결성 제약 조건 =====

```
insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)  
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');
```

```
insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)  
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');  
-- bbb가 두번실행하게 되면 유일 키값(uid) 위반 >>> 이유 - bbb중복
```

```
insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)  
values(5, 'null', 'null', '02-222-2222', '서울', 120, 100, null, 'Y');  
---체크 제약조건 >>> 이유 - name은 not null이어야 하기 때문
```

```
insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)  
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'Y');  
--체크 제약조건 >>> 이유 : 점수 제약조건 위배
```

```
insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)  
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'y');  
-- 체크제약조건 >>> 이유 : marriage in(Y,N) 대문자로 짤랐기 때문...
```

## 고유키 무결성 제약조건(UNIQUE KEY)

고유 키 무결성 제약조건 존재 : 컬럼의 모든 값이 유일해야 하므로 중복될 수 없음

STUDNO	NAME	USERID	GRADE	IDNUM	BIRTHDATE	TEL	HEIGHT	WEIGHT	DEPTNO	PROFNO
10101	전인하	jun123	4	7904021369824	79/04/02	051)781-2158	176	72	101	9903
20101	이동훈	Dals	1	8312101128467	83/12/10	055)426-1752	172	64	201	
.....	.....	.....							.....	
10203	윤진욱	Samba7	3	7904021358671	79/04/02	053)487-2698	171	70	102	9905
10107	이광훈	huriky	4	8109131276431	81/10/13	055)736-4981	175	92	101	9903
.....	.....	.....							.....	

고유 키 무결성 제약조건이 없음

- GRADE(학년) 컬럼의 값은 중복될 수 있음

입력 예 (1)

30101	홍길동	gdhong	1	8605051827463	86/05/05		182	78	201	
-------	-----	--------	---	---------------	----------	--	-----	----	-----	--

입력성공

입력 예 (2)

30102	성춘향	chsung	2				152	182	78	201
-------	-----	--------	---	--	--	--	-----	-----	----	-----

입력성공

- 고유키 무결성은 NULL 값을 허용하므로 입력 성공

입력 예 (3)

30103	이동훈	jun123	4	8312101128467					101	9903
-------	-----	--------	---	---------------	--	--	--	--	-----	------

입력실패

- USERID 값 'jun123' 과 IDNUM 값 '8312101128467' 값이 이미 존재함

```

create table MyUSER02(
  idx number constraint PKIDX primary key,
  id varchar2(10) constraint uid unique,
  name varchar2(20) constraint notname not null,
  phone varchar2(15),
  address varchar2(50),
  score number(6, 2) constraint CHSCORE check(score >= 0 and score <= 100),
  subject_code number(5),
  hire_date date default sysdate,
  marriage char(1) default 'N' constraint CHMARR check (marriage in ('Y','N')));

```

=====무결성 제약 조건 =====

```

insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');

```

```

insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');

```

-- bbb가 두번실행하게 되면 유일 키값(unid) 위반 >>> 이유 - bbb중복

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', null, '02-222-2222', '서울', 120, 100, null, 'Y');

```

---체크 제약조건 >>> 이유 - name을 not null이어야 하기 때문

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'Y');

```

--체크 제약조건 >>> 이유 : 점수 제약조건 위반

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'y');

```

-- 체크제약조건 >>> 이유 : marriage in(Y,N) 대문자로 잘랐기 때문...



## NOT NULL 무결성 제약조건

STUDNO	NAME	USERID	GRADE	IDNUM	BIRTHDATE	TEL	HEIGHT	WEIGHT	DEPTNO	PROFNO
10101	전인하	jun123	4	7904021369824	79/04/02		176	72	101	9903
20101	이동훈		1	8312101128467		055)426-1752	172	64	201	
.....	.....	.....							.....	
10203	윤진욱	Samba7	3		79/04/02	053)487-2698	171		102	9905
10107	이광훈	huriky		8109131276431	81/10/13	055)736-4981		92	101	9903
.....	.....	.....							.....	

NOT NULL 무결성 제약조건 존재

- STUDNO, NAME 컬럼에는 NULL을 입력할 수 없음

NOT NULL 무결성 제약조건이 없음

- 이 컬럼들의 어느 열에든지 NULL을 입력할 수 있음

-  은 NULL을 의미

입력 예 (1)

30101	홍길동	gdhong	1	8605051827463	86/05/05		182	78	201	
-------	-----	--------	---	---------------	----------	--	-----	----	-----	--

입력성공

입력 예 (2)

		chsung	4	8909071928374					101	9903
--	--	--------	---	---------------	--	--	--	--	-----	------

입력실패

- STUDNO 값 과 NAME 값은 NULL을 허용하지 않음

```

create table MyUSER02(
  idx number constraint PKIDX primary key,
  id varchar2(10) constraint unid unique,
  name varchar2(20) constraint notname not null,
  phone varchar2(15),
  address varchar2(50),
  score number(6, 2) constraint CHSCORE check(score >= 0 and score <= 100),
  subject_code number(5),
  hire_date date default sysdate,
  marriage char(1) default 'N' constraint CHMARR check (marriage in ('Y','N')));

```

=====무결성 제약 조건 =====

```

insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');

```

```

insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');

```

-- bbb가 두번실행하게 되면 유일 키값(unid) 위반 >>> 이유 - bbb중복

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'null', '02-222-2222', '서울', 120, 100, null, 'Y');

```

---체크 제약조건 >>> 이유 - name은 not null이어야 하기 때문

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'Y');

```

--체크 제약조건 >>> 이유 : 점수 제약조건 위반

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'y');

```

-- 체크제약조건 >>> 이유 : marriage in(Y,N) 대문자로 짤랐기 때문...

```

create table MyUSER02(
  idx number constraint PKIDX primary key,
  id varchar2(10) constraint uid unique,
  name varchar2(20) constraint notname not null,
  phone varchar2(15),
  address varchar2(50),
  score number(6, 2) constraint CHSCORE check(score >= 0 and score <= 100),
  subject_code number(5),
  hire_date date default sysdate,
  marriage char(1) default 'N' constraint CHMARR check (marriage in ('Y','N')));

```

=====무결성 제약 조건 =====

```

insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');

```

```

insert into MyUSER02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(3, 'bbb', 'kang', '02-222-2222', '서울', 80, 100, null, 'Y');

```

-- bbb가 두번실행하게 되면 유일 키값(unid) 위반 >>> 이유 - bbb중복

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'null', '02-222-2222', '서울', 120, 100, null, 'Y');

```

---체크 제약조건 >>> 이유 - name은 not null이어야 하기 때문

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'Y');

```

--체크 제약조건 >>> 이유 : 점수 제약조건 위배

```

insert into myuser02(idx, id, name, phone, address, score, subject_code, hire_date, marriage)
values(5, 'null', 'lee', '02-222-2222', '서울', 120, 100, null, 'y');

```

-- 체크제약조건 >>> 이유 : marriage in(Y,N) 대문자로 짤랐기 때문...

# CHECK 무결성 제약조건

CHECK 무결성 제약조건 존재 : 지정한 조건에 부합되는 값만 허용

STUDNO	NAME	USERID	GRADE	IDNUM	BIRTHDATE	TEL	HEIGHT	WEIGHT	DEPTNO	PROFNO
10101	전안하	jun123	4	7904021369824	79/04/02	051)781-2158	176	72	101	9903
20101	이동훈	Dals	1	8312101128467	83/12/10	055)426-1752	172	64	201	
.....	.....	.....							.....	
10203	윤진욱	Samba7	3	7904021358671	79/04/02	053)487-2698	171	70	102	9905
10107	이광훈	huriky	4	8109131276431	81/10/13	055)736-4981	175	92	101	9903
.....	.....	.....							.....	

입력 예 (1)

30101	홍길동	gdhong	1	8605051827463	86/05/05	051)781-2158	182	78	201	
-------	-----	--------	---	---------------	----------	--------------	-----	----	-----	--

입력성공

입력 예 (2)

30102	성춘향	chsung			86/05/05	051)781-2158	78	201		
-------	-----	--------	--	--	----------	--------------	----	-----	--	--

입력성공

- GRADE는 check 무결성 제약조건만 가지므로 NULL은 허용 가능

입력 예 (3)

30103	이동룡	mrlee	6		86/05/05	051)781-2158	78	201		
-------	-----	-------	---	--	----------	--------------	----	-----	--	--

입력실패

```

create table emp02(
  empno number constraint ENO PRIMARY key,
  ename varchar2(30) constraint ENAME not null,
  deptno number,
  constraint FKNO --테이블 조건, 칼럼 조건X
  foreign key(deptno) --TABLE에서 PRIMARY 값을 참조해야함
  references dept02 on delete set null
  -- deptno 키의 조건: FOREIGN KEY

```

CONSTRAINT 제약이름 FOREIGN KEY (자식컬럼명) REFERENCES 부모테이블(부모컬럼명)

1. 참조하는 테이블이 먼저 생성되어 있어야 함.

사원테이블(자식테이블)

EMPNO	ENAME	DEPTNO
111	장훈	10
222	예슬	40
333	성민	30
444	인국	20

References key

참조

부서테이블(부모테이블)

DEPTNO	DNAME
10	개발부
20	영업부
30	관리부
40	총무부

2. 외래키가 참조하는 컬럼은 참조하는 테이블의 기본키(PRIMARYKEY)이어야 함.

Primary key

foreign key(deptno)

Primary key

```
DELETE
FROM DEPT02
WHERE DEPTNO = 40;
select * from DEPT02;
```

DEPTNO	DNAME
----	----
10	개발부
20	영업부
30	관리부

부모 테이블

```
select * from EMP02;
```

EMPNO	ENAME	DEPTNO
----	----	----
111	장훈	10
222	예슬	NULL
333	성민	30
444	인국	20

자식 테이블

제약조건 삭제

**\*ON DELETE SET NULL** : 부모테이블은 삭제되고 자식은 NULL이 된다.  
만일 제약조건 설정시 **relation**만 설정된 경우에는 삭제가 안된다.  
(자식 레코드가 발견되었습니다)라고 출력된다.

데이터 이관 시 글의 내용은  
살리되 그 글을 쓴 회원 및  
해당 참조는 없게 하고  
싶을 때 사용한다.

**\*ON DELETE CASCADE** 를 설정하면 부모, 자식의 레코드가 모두 삭제된다.

임의의 해당 사이트에서 모 회원이 게시글을 작성한 경우, 회원을 먼저 삭제해버리면 삭제 된 회원의 글이  
유령회원의 글이 되므로 게시글을 먼저 삭제하고 그 다음 회원을 삭제해야 하므로 이 경우 사용한다.  
(하지만 실무에서는 실제 회원 내용은 지우지 않고 null값으로 비워 두기도 한다고 한다.)



## 참조 무결성 제약조건 FOREIGN KEY(외래키)

참조 키  $\longrightarrow$   
(referenced key)

부서 테이블

DEPTNO	DNAME	COLLEGE	LOC
101	컴퓨터공학과	100	1호관
102	멀티미디어학과	100	2호관
...	...	...	...
100	정보미디어학부	10	
10	공과대학		

## 학생 테이블

[illegible]

외래 키 (foreign key)

**입력 예 (1)**

30101	홍길동	gdhong	1	8605	민원실	05	055)426-1752	182	78	201
-------	-----	--------	---	------	-----	----	--------------	-----	----	-----

입력성공

입력 예 (2)

30102	성춘향	chsung	2	86	입력종류	182	78		
-------	-----	--------	---	----	------	-----	----	--	--

입력성공

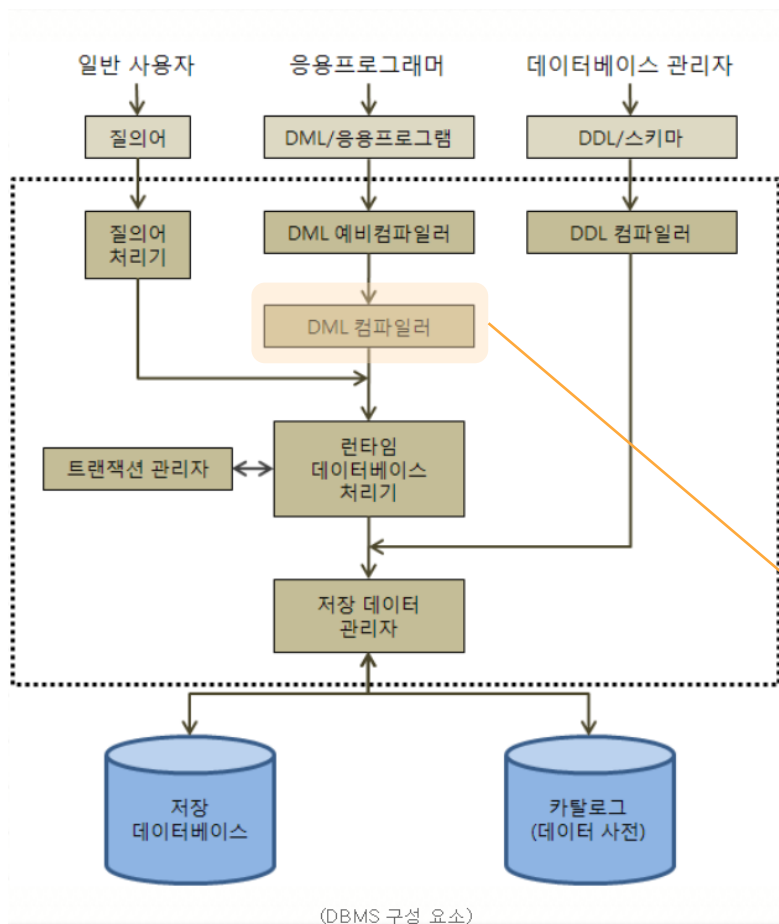
- 참조 무결성 제약조건은 NULL을 허용하므로 입력 성공

입력 예 (3)

30103	아동용	guest	4	731010112777		입력실패	500	9903
-------	-----	-------	---	--------------	--	------	-----	------

입력실패

- DEPTNO의 값 500 은 부서 테이블에 존재하지 않음



## 2. DML (DATA MANIPULATION LANGUAGE)

만들어진 테이블에 관리하기를 원하는 자료들을 입력, 수정, 삭제, 조회하는 명령어

**DBMS:** 사용자와 데이터베이스 사이에 위치하여 데이터베이스를 관리하고 사용자의 요구에 따라 데이터베이스에 대한 연산을 수행해서 정보를 생성해 주는 **S/W**.

**DML**

- INSERT
- UPDATE
- DELETE
- SELECT



--테이블을 다음과 같이 작성해 보자.

테이블명 : dept02

deptno number ==> 기본키, 제약조건 (DNO)

dname varchar2(30) ==> 널 허용안됨, 제약조건명(DNAME)

```
CREATE TABLE DEPT02(  
DEPTNO NUMBER CONSTRAINT DNO PRIMARY KEY,  
DNAME VARCHAR2(30) CONSTRAINT DNAME NOT NULL  
);
```

테이블명 : EMP02

empno number ==>기본키, 제약조건명(ENO)

ename varchar2(30) ==> 널 허용안됨, 제약조건명 (ENAME)

deptno number ==> 외래키, 제약조건명 (FKNO),

대상 데이터를 삭제하고 참조하는 데이터는 NULL로 바꿈

참조 하고 있으므로 참조하는 해당 데이터 한개만 null 되는 것이 맞다.

```
create table emp02(  
empno number constraint ENO PRIMARY key,  
ename varchar2(30) constraint ENAME not null,  
deptno number,  
constraint FKNO  
foreign key(deptno)  
references dept02 on delete set null
```

==>테이블 조건, 칼럼 조건X

==>TABLE에서 PRIMARY KEY값을 참조해야함

==>deptno 키의 조건: FOREIGN KEY

==>대상 데이터를 삭제하고 해당 데이터를 참조하는 데이터는 null로 설정

Status	Parameters	Result1
--------	------------	---------

DEPTNO	DNAME
--------	-------

-----	-----
-------	-------

<

Total 0 records shown

Status	Parameters	Result1
--------	------------	---------

EMPNO	ENAME	DEPTNO
-------	-------	--------

-----	-----	-----
-------	-------	-------

## 2-2 INSERT

<유형1>

INSERT INTO 테이블명 (COLUMN\_LIST)  
VALUES (COLUMN\_LIST에 넣을 VALUE\_LIST);

<유형2>

INSERT INTO 테이블명  
VALUES(전체 COLUMN에 넣을 VALUE\_LIST);

-----DEPT02에 데이터 추가-----  
INSERT INTO DEPT02 VALUES (10, '개발부');  
INSERT INTO DEPT02 VALUES (20, '영업부');  
INSERT INTO DEPT02 VALUES (30, '관리부');  
INSERT INTO DEPT02 VALUES (40, '총무부');

-----EMP02에 데이터 추가-----  
INSERT INTO EMP02 VALUES (111, '장훈', 10);  
INSERT INTO EMP02 VALUES (222, '예슬', 40);  
INSERT INTO EMP02 VALUES (333, '성민', 30);  
INSERT INTO EMP02 VALUES (444, '인국', 20);

-----  
INSERT INTO EMP02 VALUES (555, '인국', 50); ==> 오류 : 50번은 없음  
-----

Status	Parameters	Result1
	DEPTNO DNAME	
	-----	
	10 개발부	
	20 영업부	
	30 관리부	
	40 총무부	
<		
Total 4 records shown		

Status	Parameters	Result1
	EMPNO ENAME DEPTNO	
	-----	
	111 장훈 10	
	222 예슬 40	
	333 성민 30	
	444 인국 20	
<		
Total 4 records shown		

## 2-3 UPDATE

<형식>

UPDATE 테이블명

SET 수정되어야 할 칼럼명 = 수정되기를 원하는 새로운 값;

```
--수정 update / emp02
--emp02테이블에 인국의 부서번호를 10으로 변경하자.
update emp02
set deptno = 10
where ename = '인국';

--emp02테이블에 장훈의 부서번호를 40으로 변경하자
update emp02
set deptno = 40
where ename = '장훈';

--emp02테이블에 def의 부서번호를 20으로 변경하자
update emp02
set deptno = 20
where ename = 'def';

--emp02테이블에 abc의 이름을 으로 king로 변경하자
update emp02
set ename = 'king'
where ename = 'abc';

--emp02 제약조건 확인
SELECT table_name, constraint_name, constraint_type
FROM USER_CONSTRAINTS
WHERE TABLE_NAME = 'emp02';

select * from emp02;
```

SQL Results					
Type	q	Status	Parameters	Result1	
Stat			EMPNO	ENAME	DEPTNO
			-----	-----	-----
✓			555	king	NULL
✓			666	def	20
✓			111	장훈	40
✗			222	예슬	40
✓			333	성민	30
✓			444	인국	10

제약조건 확인

CONSTRAINT\_TYPE : P (PRIMARY KEY)  
U (UNIQUE)  
R (REFERENCE)  
C (CHECK, NOT NULL)

CONSTRAINT\_NAME : SYS\_제약조건타입 + 식별값

제약조건이 있을 시에 삭제

**ON DELETE CASCADE** : 대상데이타를 삭제하고 해당 데이타를 참조하는 데이타도 삭제

**ON DELETE SET NULL** : 대상데이타를 삭제하고 해당 데이타를 참조하는 데이타는 NULL로 변경

**ON DELETE RESTRICTED** : 삭제 대상데이타를 참조하는 데이타가 존재하면 삭제할 수 없다.

수정 RULE

**ON UPDATE CASCADE** : 대상 데이타를 수정하려면 해당 데이타를 참조하는 데이타도 수정.