KH정보교육원

Oracle SQL 개요

Oracle - SQL 개요

KH정보교육원

SQL이란?

1. Structured Query Language의 약자 (구조화된 질의 언어)
2. 관계형 데이터베이스의 데이터를 조회하거나 조작하기 위해 사용하는 표준 검색 언어
3. 원하는 데이터를 찾는 방법이나 절차를 기술하는 것이 아닌 조건을 기술하여 작성함
4. DBMS에 따라 사용되는 SQL 문법이 다름

- 기본 SQL 문법 종류

1. 데이터 정의어 : DDL (Data Definition Language)
2. 데이터 조작어 : DML (Data Manipulation Language)
3. 데이터 제어어 : DCL (Data Control Language)
4. 트랜젝션 제어어 : TCL (Transaction Control Language)

Oracle - SQL 개요

KH정보교육원

DDL(Data Definition Language) 이란?

1. 데이터베이스의 구조를 정의하거나 변경, 삭제 하기 위해 사용하는 언어

2. 주로 DB 관리자 또는 설계자가 사용함

3. CREATE(개체 생성), ALTER(개체 수정), DROP(개체 삭제), TRUNCATE(개체 초기화)
   ex) DB를 생성/수정/삭제 하거나 TABLE을 생성/수정/삭제 등

Oracle - SQL 개요

KH정보교육원

DML(Data Manipulation Language)이란?

1. Data를 조작하기 위해 사용하는 언어
2. Data의 삽입, 수정, 삭제, 조회 등의 동작을 제어함
3. Data를 이용하려는 사용자와 시스템간의 인터페이스를 직접적으로 제공하는 언어
4. 가장 많이 사용됨
5. INSERT(데이터 삽입), UPDATE(데이터 수정), DELETE (데이터 삭제)

- DQL (Data Query Language)

1. 데이터를 검색(추출)하기 위해 사용되는 언어
2. SELECT (데이터 검색)

Oracle - SQL 개요

KH정보교육원

DCL / TCL 이란?

DCL (Data Control Language)	1. 사용자의 권한이나, 관리자 설정 등을 처리 2. GRANT(유저 권한 생성), REVOKE(유저 권한 삭제)
TCL (Transaction Control Language)	1. 트랜 잭션 관리 처리 언어 2. COMMIT(트랜잭션 종료처리후 저장), ROLLBACK(트랜잭션 취소), SAVEPOINT(임시저장)

KH정보교육원

DDL(CREATE)

DDL(Data Definition Languge)

KH정보교육원

DDL이란 무엇인가?

데이터 정의 언어이다. 객체(Object)를 만들고(CREATE), 수정하고(ALTER), 삭제(DROP)하는 구문을 말한다.

오라클 객체의 종류

테이블(TABLE), 뷰(VIEW), 시퀀스(SEQUENCE), 인덱스(INDEX), 패키지(PACKAGE), 프로시저(PROCEDUAL), 함수(FUNCTION), 트리거(TRIGGER), 동의어(SYNONYM), 사용자(USER)가 있다.

DDL(Data Definition Languge)

KH정보교육원

CREATE - 사용자만들기

[표현식]

CREATE USER 계정명 IDENTIFIED BY 계정비밀번호;

```
CREATE USER KH IDENTIFIED BY KH; -- 계정 생성(with 비밀번호)
GRANT  CONNECT TO KH; -- 연결 권한 부여  => DCL
GRANT RESOURCE TO KH; -- 데이터 조작 권한 부여 => DCL, SYS로!!
```

DDL(Data Definition Languge)

KH정보교육원

CREATE - 테이블만들기

[표현식]

CREATE TABLE 테이블명 (컬럼명 자료형(크기), 컬럼명 자료형(크기), …);

```
CREATE TABLE MEMBER (
    MEMBER_ID VARCHAR2(20),
    MEMBER_PWD VARCHAR2(20),
    MEMBER_NAME VARCHAR2(20)
);
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	MEMBER_ID	VARCHAR2(20 BYTE)	Yes	(null)	1	(null)
2	MEMBER_PWD	VARCHAR2(20 BYTE)	Yes	(null)	2	(null)
3	MEMBER_NAME	VARCHAR2(30 BYTE)	Yes	(null)	3	(null)

Oracle - Table

KH정보교육원

1. 데이터를 담고 있는 개체
2. Table은 기본적으로 행과 열을 이용하여 데이터를 표현함

MEMBER Table

MEMBER_ID	MEMBER_PWD	MEMBER_NAME
user11	pass11	일용자
user22	pass22	이용자
user33	pass33	삼용자

① Relation | ② Column(Domain) | ③ Row (Tuple) | ④ Field

Oracle - Table

KH정보교육원

1. 데이터를 담고 있는 개체
2. Table은 기본적으로 행과 열을 이용하여 데이터를 표현함

Student Table

Grade	Name	Age	Gender	phoneNumber
1	김철구	20	남	01012341122
2	최거폭	32	남	01078754885
2	이벤쯔	22	남	01033554040
1	김꽃님	21	여	01050509999
3	박윰댕	24	여	01099582356
4	주창현	24	남	01022441148

① Relation | ② Column(Domain) | ③ Row (Tuple) | ④ Field

Oracle - 데이터 타입

KH정보교육원

오라클에서 데이터를 표현하기 위한 데이터 타입

구분	데이터 타입	하위 데이터 타입	설명
숫자	NUMBER		숫자
문자	CHARACTER	CHAR	고정길이 문자 (최대 2000Byte/최소 1Byte)
		VARCHAR2	가변길이 문자 (최대 4000 Byte/ 최소 1Byte)
		NCHAR	CHAR와 동일, 유니코드 문자
		NVARCHAR2	VARCHAR와 동일, 유니코드 문자
		LONG	가변길이 문자(최대 2GByte)
날짜	DATE		날짜
	TIMESTAMP		연도, 월, 일, 시, 분, 초, 밀리초 까지 입력가능
데이터	LOB	CLOB	가변길이 문자(최대 4GByte)
		BLOB	Binary Data

Oracle - 데이터 타입

KH정보교육원

NUMBER([P, S])

P : 표현할 수 있는 전체 숫자 자리수(1~38)

S : 소수점 이하 자리수 (-84 ~ 127)

1234.678의 값을 다음의 데이터 타입으로 저장한다면?

NUMBER(7,3)	->	1234.678
NUMBER(7)	->	1234
NUMBER	->	1234.678
NUMBER(7, 1)	->	1234.6
NUMBER(5, -2)	->	1200

Oracle - 데이터 타입

KH정보교육원

CHAR(SIZE [byte | char])

SIZE : 포함될 문자(열) 크기

지정한 크기보다 작은 문자(열)이 입력되고 남는 공간은 공백으로 채움

데이터는 ''를 사용하여 표기하고, 대/소문자를 구분함

만약 'ORACLE'이라는 데이터를 CHAR 타입으로 저장한다면? (영어 1글자 1Byte)

CHAR(6) -> ORACLE

CHAR(9) -> ORACLE(공백 3칸)

CHAR(3) -> 에러

만약 '김치' 라는 데이터를 CHAR 타입으로 저장한다면? (한글 1글자는 3Byte, expres버전)

CHAR(6) -> 김치

CHAR(9) -> 김치(공백3Byte,한글 1글자)

CHAR(3) -> 에러

Oracle - 데이터 타입

KH정보교육원

VARCHAR(SIZE [byte | char])

SIZE : 포함될 문자(열) 크기

지정한 크기보다 작은 문자(열)이 입력되고 남는 공간은 공백으로 채움

데이터는 ''를 사용하여 표기하고, 대/소문자를 구분함

만약 'ORACLE'이라는 데이터를 CHAR 타입으로 저장한다면?

VARCHAR(6) -> ORACLE

VARCHAR(9) -> ORACLE(공백 0칸)

VARCHAR(3) -> 에러

만약 '김치' 라는 데이터를 CHAR 타입으로 저장한다면?

VARCHAR(6) -> 김치

VARCHAR(9) -> 김치(공백0Byte)

VARCHAR(3) -> 에러

Oracle - 데이터 타입

KH정보교육원

DATE

일자(세기/년/월/일) 및 시간(시/분/초) 정보를 관리

기본적으로 화면에 년/월/일 정보만 표기 됨

날짜의 연산 및 비교가 가능함

연산	결과 타입	설명
날짜+숫자	DATE	작성한 숫자 만큼 며칠 후를 의미
날짜-숫자	DATE	작성한 숫자 만큼 며칠 전을 의미
날짜-날짜	NUMBER	두 날짜의 차이(일수)를 의미
날짜+숫자/24	DATE	날짜+시간의 의미

DDL(Data Definition Languge)

KH정보교육원

오라클의 데이터형

데이터형	설명
CHAR(크기)	고정길이 문자 데이터
VARCHAR2(크기)	가변길이 문자 데이터(최대 2,000 Byte)
NUMBER	숫자 데이터(최대 40자리)
NUMBER(길이)	숫자 데이터로, 길이 지정 가능하다 (최대 38자리)
DATE	날짜 데이터(BC 4712년 1월 1일 ~ AD 4712년 12월 31일)
LONG	가변 길이 문자형 데이터(최대 2GB)
LOB	2GB까지의 가변길이 바이너리 데이터 저장 가능 (이미지, 실행파일 등을 저장할 수 있음)
ROWID	DB에 저장되지 않는 행을 식별할 수 있는 고유 값
BFILE	대용량의 바이너리 데이터 저장 가능(최대 4GB)
TIMESTAMP	DATE형의 확장된 형태이다.
INTERVAL YEAR TO MONTH	년과 월을 이용하여 기간을 저장한다.
INTERVAL DAY TO SECONT	일, 시, 분, 초를 이용하여 기간을 저장한다.

DDL(Data Definition Languge)

KH정보교육원

컬럼 주석

[표현식]

COMMENT ON CULUMN 테이블명.컬럼명 IS '주석내용';

```
COMMENT ON COLUMN MEMBER.MEMBER_ID IS '회원아이디';
COMMENT ON COLUMN MEMBER.MEMBER_PWD IS '비밀번호';
COMMENT ON COLUMN MEMBER.MEMBER_NAME IS '회원이름';
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	MEMBER_ID	VARCHAR2(20 BYTE)	Yes	(null)	1	회원아이디
2	MEMBER_PWD	VARCHAR2(20 BYTE)	Yes	(null)	2	비밀번호
3	MEMBER_NAME	VARCHAR2(30 BYTE)	Yes	(null)	3	회원이름

KH정보교육원

DDL(ALTER, DROP)

DDL(Data Definition Language)

KH정보교육원

DDL - ALTER

테이블에 정의된 내용을 수정할 때 사용하는 데이터 정의어이다.

컬럼의 추가/삭제, 제약조건의 추가/삭제, 컬럼의 자료형 변경, DEFAULT값 변경,

테이블명/컬럼명/제약조건의 이름 변경 등을 할 수 있다.

컬럼 추가

```
ALTER TABLE DEPT_COPY
ADD (LNAME VARCHAR2(20));
```

SQL | 인출된 모든 행: 9(0.016초)

	DEPT_ID	DEPT_TITLE	LOCATION_ID
1	D1	인사관리부	L1
2	D2	회계관리부	L1
3	D3	마케팅부	L1
4	D4	국내영업부	L1
5	D5	해외영업1부	L2
6	D6	해외영업2부	L3
7	D7	해외영업3부	L4
8	D8	기술지원부	L5
9	D9	전략기획팀	L1

SQL | 인출된 모든 행: 9(0초)

	DEPT_ID	DEPT_TITLE	LOCATION_ID	CNAME
1	D1	인사관리부	L1	(null)
2	D2	회계관리부	L1	(null)
3	D3	마케팅부	L1	(null)
4	D4	국내영업부	L1	(null)
5	D5	해외영업1부	L2	(null)
6	D6	해외영업2부	L3	(null)
7	D7	해외영업3부	L4	(null)
8	D8	기술지원부	L5	(null)
9	D9	전략기획팀	L1	(null)

DDL(Data Definition Language)

KH정보교육원

DDL - ALTER

테이블에 정의된 내용을 수정할 때 사용하는 데이터 정의어이다.

컬럼의 추가/삭제, 제약조건의 추가/삭제, 컬럼의 자료형 변경, DEFAULT값 변경,

테이블명/컬럼명/제약조건의 이름 변경 등을 할 수 있다.

컬럼 추가

```
ALTER TABLE DEPT_COPY
ADD (CNAME VARCHAR2(40) DEFAULT '한국');
```

SQL | 인출된 모든 행: 9(0초)

	DEPT_ID	DEPT_TITLE	LOCATION_ID	CNAME
1	D1	인사관리부	L1	(null)
2	D2	회계관리부	L1	(null)
3	D3	마케팅부	L1	(null)
4	D4	국내영업부	L1	(null)
5	D5	해외영업1부	L2	(null)
6	D6	해외영업2부	L3	(null)
7	D7	해외영업3부	L4	(null)
8	D8	기술지원부	L5	(null)
9	D9	전략기획팀	L1	(null)

SQL | 인출된 모든 행: 9(0초)

	DEPT_ID	DEPT_TITLE	LOCATION_ID	CNAME	LNAME
1	D1	인사관리부	L1	(null)	한국
2	D2	회계관리부	L1	(null)	한국
3	D3	마케팅부	L1	(null)	한국
4	D4	국내영업부	L1	(null)	한국
5	D5	해외영업1부	L2	(null)	한국
6	D6	해외영업2부	L3	(null)	한국
7	D7	해외영업3부	L4	(null)	한국
8	D8	기술지원부	L5	(null)	한국
9	D9	전략기획팀	L1	(null)	한국

DDL(Data Definition Language)

KH정보교육원

DDL - ALTER

제약조건 추가

```
ALTER TABLE DEPT_COPY
ADD CONSTRAINT DCOPY_DID_PK PRIMARY KEY(DEPT_ID),
ADD CONSTRAINT DCOPY_DTITLE_UNQ UNIQUE (DEPT_TITLE),
MODIFY LNAME CONSTRAINT DCOPY_LNAME_NN NOT NULL;
```

```
SELECT UC.CONSTRAINT_NAME,
       UC.CONSTRAINT_TYPE,
       UC.TABLE_NAME,
       UCC.COLUMN_NAME,
       UC.SEARCH_CONDITION
FROM USER_CONSTRAINTS UC
JOIN USER_CONS_COLUMNS UCC ON (UC.CONSTRAINT_NAME = UCC.CONSTRAINT_NAME)
WHERE UC.TABLE_NAME = 'DEPT_COPY';
```

SQL | 인출된 모든 행: 5(0.125초)

	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	COLUMN_NAME	SEARCH_CONDITION
1	SYS_C007230	C	DEPT_COPY	DEPT_ID	"DEPT_ID" IS NOT NULL
2	SYS_C007231	C	DEPT_COPY	LOCATION_ID	"LOCATION_ID" IS NOT NULL
3	DCOPY_LNAME_NN	C	DEPT_COPY	LNAME	"LNAME" IS NOT NULL
4	DCOPY_DID_PK	P	DEPT_COPY	DEPT_ID	(null)
5	DCOPY_DTITLE_UNQ	U	DEPT_COPY	DEPT_TITLE	(null)

DDL(Data Definition Language)

KH정보교육원

DDL - ALTER

컬럼 수정

```
ALTER TABLE DEPT_COPY
MODIFY DEPT_ID CHAR(3)
MODIFY DEPT_TITLE VARCHAR(30)
MODIFY LOCATION_ID VARCHAR2(2)
MODIFY CNAME CHAR(20)
MODIFY LNAME DEFAULT '미국';
```

	COLUMN_N...	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DEPT_ID	CHAR(2 BYTE)	No	(null)	1	(null)
2	DEPT_TITLE	VARCHAR2(35 BYTE)	Yes	(null)	2	(null)
3	LOCATION_ID	CHAR(2 BYTE)	No	(null)	3	(null)
4	CNAME	VARCHAR2(20 BYTE)	Yes	(null)	4	(null)
5	LNAME	VARCHAR2(20 BYTE)	No	'한국'	5	(null)

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DEPT_ID	CHAR(3 BYTE)	No	(null)	1	(null)
2	DEPT_TITLE	VARCHAR2(30 BYTE)	Yes	(null)	2	(null)
3	LOCATION_ID	VARCHAR2(2 BYTE)	No	(null)	3	(null)
4	CNAME	CHAR(20 BYTE)	Yes	(null)	4	(null)
5	LNAME	VARCHAR2(20 BYTE)	No	'미국'	5	(null)

DDL(Data Definition Language)

KH정보교육원

DDL - ALTER

컬럼 삭제

```
ALTER TABLE DEPT_COPY
DROP COLUMN DEPT_ID;
```

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DEPT_ID	CHAR(3 BYTE)	No	(null)	1	(null)
2	DEPT_TITLE	VARCHAR2(30 BYTE)	Yes	(null)	2	(null)
3	LOCATION_ID	VARCHAR2(2 BYTE)	No	(null)	3	(null)
4	CNAME	CHAR(20 BYTE)	Yes	(null)	4	(null)
5	LNAME	VARCHAR2(20 BYTE)	No	'미국'	5	(null)

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DEPT_TITLE	VARCHAR2(30 BYTE)	Yes	(null)	1	(null)
2	LOCATION_ID	VARCHAR2(2 BYTE)	No	(null)	2	(null)
3	CNAME	CHAR(20 BYTE)	Yes	(null)	3	(null)
4	LNAME	VARCHAR2(20 BYTE)	No	'미국'	4	(null)

DDL(Data Definition Language)

KH정보교육원

DDL - ALTER

컬럼 삭제

```
CREATE TABLE TB1 (
    PK NUMBER PRIMARY KEY,
    FK NUMBER REFERENCES TB1,
    COL1 NUMBER,
    CHECK (PK > 0 AND COL1 > 0)
);
```

```
ALTER TABLE TB1
DROP COLUMN PK;
```

```
오류 보고 -
SQL 오류: ORA-12992: cannot drop parent key column
12992. 00000 -  "cannot drop parent key column"
*Cause:    An attempt was made to drop a parent key column.
*Action:   Drop all constraints referencing the parent key column, or
           specify CASCADE CONSTRAINTS in statement.
```

** 컬럼 삭제시 참조하고 있는 컬럼이 있다면 컬럼 삭제를 못한다.

```
ALTER TABLE TB1
DROP COLUMN PK CASCADE CONSTRAINT;
```

```
Table TB1이(가) 변경되었습니다.
```

DDL(Data Definition Language)

KH정보교육원

DDL - ALTER

제약조건 삭제

```
ALTER TABLE DEPT_COPY
DROP CONSTRAINT DCOPY_DID_PK
DROP CONSTRAINT DCOPY_DTITLE_UNQ
MODIFY LNAME NULL;
```

SQL | 인출된 모든 행: 5(0.125초)

	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	COLUMN_NAME	SEARCH_CONDITION
1	SYS_C007230	C	DEPT_COPY	DEPT_ID	"DEPT_ID" IS NOT NULL
2	SYS_C007231	C	DEPT_COPY	LOCATION_ID	"LOCATION_ID" IS NOT NULL
3	DCOPY_LNAME_NN	C	DEPT_COPY	LNAME	"LNAME" IS NOT NULL
4	DCOPY_DID_PK	P	DEPT_COPY	DEPT_ID	(null)
5	DCOPY_DTITLE_UNQ	U	DEPT_COPY	DEPT_TITLE	(null)

	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	COLUMN_NAME	SEARCH_CONDITION
1	SYS_C007231	C	DEPT_COPY	LOCATION_ID	"LOCATION_ID" IS NOT NULL

DDL(Data Definition Language)

KH정보교육원

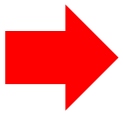
DDL - ALTER

컬럼 이름변경

```
ALTER TABLE DEPT_COPY
RENAME COLUMN DEPT_ID TO DEPT_CODE;
```

SQL | 인출된 모든 행: 9(0초)

	DEPT_TITLE	LOCATION_ID	CNAME	LNAME
1	인사관리부	L1	(null)	한국
2	회계관리부	L1	(null)	한국
3	마케팅부	L1	(null)	한국
4	국내영업부	L1	(null)	한국
5	해외영업1부	L2	(null)	한국
6	해외영업2부	L3	(null)	한국
7	해외영업3부	L4	(null)	한국
8	기술지원부	L5	(null)	한국
9	전략기획팀	L1	(null)	한국



SQL | 인출된 모든 행: 9(0.016초)

	DEPT_NAME	LOCATION_ID	CNAME	LNAME
1	인사관리부	L1	(null)	한국
2	회계관리부	L1	(null)	한국
3	마케팅부	L1	(null)	한국
4	국내영업부	L1	(null)	한국
5	해외영업1부	L2	(null)	한국
6	해외영업2부	L3	(null)	한국
7	해외영업3부	L4	(null)	한국
8	기술지원부	L5	(null)	한국
9	전략기획팀	L1	(null)	한국

DDL(Data Definition Language)

KH정보교육원

DDL - ALTER

제약조건 이름 변경

```
ALTER TABLE USER_FOREIGNKEY
RENAME CONSTRAINT SYS_C007211 TO
UF_UP_NN;
```

```
ALTER TABLE USER_FOREIGNKEY
RENAME CONSTRAINT SYS_C007212 TO
UF_UN_PK;
```

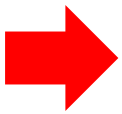
```
ALTER TABLE USER_FOREIGNKEY
RENAME CONSTRAINT SYS_C007213 TO
UF_UI_UQ;
```

```
ALTER TABLE USER_FOREIGNKEY
RENAME CONSTRAINT SYS_C007214 TO
UF_GC_FK;
```

```
SELECT UC.CONSTRAINT_NAME 이름,
       UC.CONSTRAINT_TYPE 유형,
       UCC.COLUMN_NAME 컬럼명,
       UC.R_CONSTRAINT_NAME 참조,
       UC.DELETE_RULE 삭제규칙
FROM USER_CONSTRAINTS UC
JOIN USER_CONS_COLUMNS UCC ON (UC.CONSTRAINT_NAME = UCC.CONSTRAINT_NAME)
WHERE UC.TABLE_NAME = 'USER_FOREIGNKEY';
```

SQL | 인출된 모든 행: 4(0.109초)

	이름	유형	컬럼명	참조	삭제규칙
1	SYS_C007211	C	USER_PWD	(null)	(null)
2	SYS_C007212	P	USER_NO	(null)	(null)
3	SYS_C007213	U	USER_ID	(null)	(null)
4	SYS_C007214	R	GRADE_CODE	SYS_C007210	CASCADE



SQL | 인출된 모든 행: 4(0초)

	이름	유형	컬럼명	참조	삭제규칙
1	UF_UP_NN	C	USER_PWD	(null)	(null)
2	UF_UN_PK	P	USER_NO	(null)	(null)
3	UF_UI_UQ	U	USER_ID	(null)	(null)
4	UF_GC_FK	R	GRADE_CODE	SYS_C007210	CASCADE

DDL(Data Definition Language)

KH정보교육원

DDL - ALTER

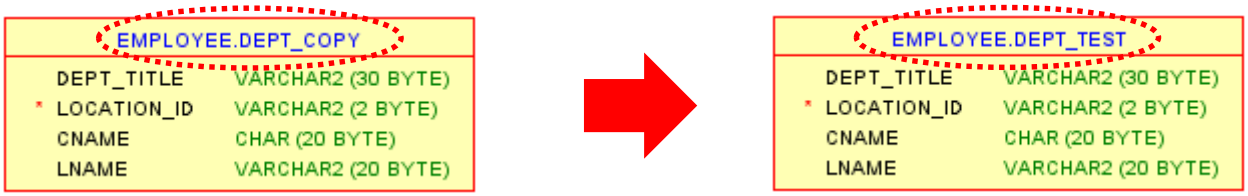
테이블 이름 변경

```
ALTER TABLE DEPT_COPY RENAME TO DEPT_TEST;
```

혹은

```
RENAME DEPT_COPY TO DEPT_TEST;
```

```
테이블 이름이 변경되었습니다.
```



DDL(Data Definition Language)

KH정보교육원

DDL - DROP

테이블 삭제

```
DROP TABLE DEPT_TEST CASCADE CONSTRAINT;
```

```
Table DEPT_TEST이(가) 삭제되었습니다.
```

KH정보교육원

DCL(GRANT, REVOKE)

DCL(Data Control Language)

KH정보교육원

DCL이란?

DB에 대한 보안, 무결성, 복구 등 DBMS를 제어하기 위한 언어이다.

GRANT(권한할당), REVOKE(권한해제), COMMIT(실행), ROLLBACK(복구)

COMMIT, ROLLBACK 은 트랜젝션에 관련된 언어로 TCL로 구분하기도 함

DCL - GRANT

사용자 또는 ROLE에 대하여 권한을 부여 가능

GRANT [System_Privilege|role] TO [user|role|PUBLIC] WITH ADMIN OPTION

- System_privilege : 부여할 시스템 권한의 이름
- role : 부여할 데이터베이스 역할의 이름
- user,role : 부여할 사용자 이름과 다른 데이터 베이스 역할 이름
- PUBLIC : 시스템 권한, 또는 데이터베이스 역할을 모든 사용자에게 부여할 수 있음
- WITH ADMIN OPTION :
  권한을 부여받은 사용자도 부여 받은 권한을 다른 사용자 또는 역할로 부여 가능

DCL(Data Control Language)

KH정보교육원

DCL – 접속 권한 부여하기

sqlplus 접속

1. 신규 사용자 생성 (system 계정으로 생성하기)

```
create user test01 identified by test01;
```

2. 접속 시도

```
conn test01/test01;
```

=> 접속 실패됨 (접속 권한이 없음)

3. system 계정으로 접속후 권한 부여

```
grant connect to test01;
```

4. 접속 재시도

```
conn test01/test01;
```

=> 접속 성공

=> show user;

DCL(Data Control Language)

KH정보교육원

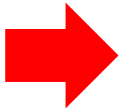
DCL – 데이터 보기 권한 부여하기

```
CREATE TABLE COFFEE
(
 PRODUCT_NAME VARCHAR2(20) PRIMARY KEY,
 PRICE NUMBER NOT NULL,
 COMPANY VARCHAR2(20) NOT NULL
);

INSERT INTO COFFEE VALUES('맥심커피',30000,'MAXIM');
INSERT INTO COFFEE VALUES('카누커피',50000,'MAXIM');
INSERT INTO COFFEE VALUES('네스카페커피',40000,'nescafe');

COMMIT; -- COMMIT을 하지 않으면 다른 계정에 적용이 되어 있지 않아서 안보임
```

```
GRANT SELECT ON kh.COFFEE TO test01;
```



```
SELECT * FROM kh.COFFEE;
```

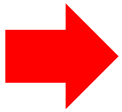
성공!

DCL(Data Control Language)

KH정보교육원

DCL – 데이터 입력 권한 부여하기

```
GRANT INSERT ON kh.COFFEE TO test01;
```



```
INSERT INTO kh.COFFEE VALUES('프렌치카페', 20000, '남양유업');
```

성공!

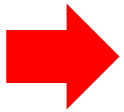
DCL(Data Control Language)

KH정보교육원

DCL - REVOKE

사용자 또는 ROLE에 대하여 권한을 회수 가능

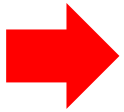
```
REVOKE INSERT ON kh.COFFEE FROM test01;
```



```
INSERT INTO kh.COFFEE VALUES('맥스웰', 25000, '동서식품');
```

실패

```
REVOKE SELECT ON kh.COFFEE FROM test01;
```



```
SELECT * FROM kh.COFFEE;
```

실패

KH정보교육원

DDL 제약조건

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS)

테이블 작성시 각 컬럼에 대한 기록에 대해 제약조건을 설정할 수 있다.

데이터 무결성 보장이 주 목적이다. 입력 데이터에 문제가 없는지 검사와 데이터의 수정/삭제 가능 여부 검사 등을 위해 사용한다.

제약 조건	설명
NOT NULL	데이터에 NULL을 허용하지 않는다.
UNIQUE	중복된 값을 허용하지 않는다.
PRIMARY KEY	NULL을 허용하지 않고, 중복 값을 허용하지 않는다. 컬럼의 고유 식별자로 사용하기 위함이다.
FOREIGN KEY	참조되는 테이블의 컬럼의 값이 존재하면 허용한다.
CHECK	저장 가능한 데이터 값의 범위나 조건을 지정하여 설정한 값만 허용한다.

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - NOT NULL

해당 컬럼에 반드시 값이 기록되어야 하는 경우, 특정 컬럼에 값을 저장하거나 수정할 때 NULL 값을 허용하지 않도록 컬럼 레벨에서 제한한다.

```
CREATE TABLE USER_NOCONS(
    USER_NO NUMBER,
    USER_ID VARCHAR2(20),
    USER_PWD VARCHAR2(30),
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50)
);
```

```
Table USER_NOCONS이(가) 생성되었습니다.
```

```
INSERT INTO USER_NOCONS
VALUES(1, 'user01', 'pass01', '홍길동', '남',
        '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_NOCONS
VALUES(2, NULL, NULL, NULL, NULL,
        '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

	USER_NO	USER_ID	USER_PWD	USER_NAME	GENDER	PHONE	EMAIL
1	1	USER01	PASS01	홍길동	남	010-1234-5678	hong123@kh.or.kr
2	2	(null)	(null)	(null)	(null)	010-1234-5678	hong123@kh.or.kr

** 컬럼에 아무 제약조건을 설정하지 않을 경우, NULL값이 문제 없이 삽입됨

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - NOT NULL

```
CREATE TABLE USER_NOTNULL(
    USER_NO NUMBER NOT NULL,
    USER_ID VARCHAR2(20) NOT NULL,
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50)
);
```

```
Table USER_NOTNULL이(가) 생성되었습니다.
```

```
INSERT INTO USER_NOTNULL
VALUES(1, 'user01', 'pass01', '홍길동', '남',
        '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_NOTNULL
VALUES(2, NULL, NULL, NULL, NULL,
        '010-1234-5678', 'hong123@kh.or.kr');
```

```
오류 보고 -
SQL 오류: ORA-01400: cannot insert NULL into ("EMPLOYEE"."USER_NOTNULL"."USER_ID")
01400. 00000 -  "cannot insert NULL into (%s)"
*Cause:    An attempt was made to insert NULL into previously listed objects.
*Action:   These objects cannot accept NULL values.
```

	USER_NO	USER_ID	USER_PWD	USER_NAME	GENDER	PHONE	EMAIL
1	1	user01	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr

** NOT NULL 제약조건이 설정된 컬럼에 NULL값이 입력되면, 행 자체를 삽입하지 않는다.

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - UNIQUE

컬럼에 입력 값에 대해 중복을 제한하는 제약조건이다. 컬럼레벨과 테이블레벨에 설정 가능하다.

```
CREATE TABLE USER_NOCONS(
    USER_NO NUMBER,
    USER_ID VARCHAR2(20),
    USER_PWD VARCHAR2(30),
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50)
);
```

```
INSERT INTO USER_NOCONS
VALUES(1, 'user01', 'pass01', NULL, NULL,
        '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
Table USER_NOCONS이(가) 생성되었습니다.
```

	USER_NO	USER_ID	USER_PWD	USER_NAME	GENDER	PHONE	EMAIL
1	1	user01	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr
2	2	(null)	(null)	(null)	(null)	010-1234-5678	hong123@kh.or.kr
3	1	user01	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr

** UNIQUE 제약조건이 없는 컬럼은 중복된 값도 저장을 허용한다.

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - UNIQUE

```
CREATE TABLE USER_UNIQUE(
    USER_NO NUMBER,
    USER_ID VARCHAR2(20) UNIQUE,
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50)
);
```

```
Table USER_UNIQUE이(가) 생성되었습니다.
```

```
INSERT INTO USER_UNIQUE
VALUES(1, 'user01', 'pass01', '홍길동', '남',
       '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_UNIQUE
VALUES(1, 'user01', 'pass01', NULL, NULL,
       '010-1234-5678', 'hong123@kh.or.kr');
```

```
오류 보고 -
SQL 오류: ORA-00001: unique constraint (EMPLOYEE.SYS_C007182) violated
00001. 00000 -  "unique constraint (%s.%s) violated"
*Cause:    An UPDATE or INSERT statement attempted to insert a duplicate key.
           For Trusted Oracle configured in DBMS MAC mode, you may see
           this message if a duplicate entry exists at a different level.
*Action:   Either remove the unique restriction or do not insert the key.
```

	TABLE_NAME	COLUMN_NAME	CONSTRAINT_TYPE
1	USER_UNIQUE	USER_ID	U

```
SELECT UCC.TABLE_NAME, UCC.COLUMN_NAME, UC.CONSTRAINT_TYPE
FROM USER_CONSTRAINT UC, USER_CONS_COLUMNS UCC
WHERE UCC.CONSTRAINT_NAME = UC.CONSTRAINT_NAME
AND UCC.CONSTRAINT_NAME = 'SYS_C007182';
```

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - UNIQUE

```
CREATE TABLE USER_UNIQUE2(
    USER_NO NUMBER,
    USER_ID VARCHAR2(20),
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50),
    UNIQUE (USER_ID)  -- 테이블레벨
);
```

```
Table USER_UNIQUE2이(가) 생성되었습니다.
```

```
INSERT INTO USER_UNIQUE2
VALUES(1, 'user01', 'pass01', '홍길동', '남',
        '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_UNIQUE2
VALUES(1, 'user01', 'pass01', NULL, NULL,
        '010-1234-5678', 'hong123@kh.or.kr');
```

```
SQL 오류: ORA-00001: unique constraint (EMPLOYEE.SYS_C007184) violated
00001. 00000 -  "unique constraint (%s.%s) violated"
*Cause:    An UPDATE or INSERT statement attempted to insert a duplicate key.
           For Trusted Oracle configured in DBMS MAC mode, you may see
           this message if a duplicate entry exists at a different level.
*Action:   Either remove the unique restriction or do not insert the key.
```

	USER_NO	USER_ID	USER_PWD	USER_NAME	GENDER	PHONE	EMAIL
1	1	user01	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr

** 중복값이 있는 경우 UNIQUE 제약조건에 의해 행이 삽입되지 않는다.

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - UNIQUE

```
INSERT INTO USER_UNIQUE2
VALUES(1, 'user01', 'pass01', '홍길동', '남',
       '010-1234-5678', 'hong123@kh.or.kr');
```

1 행 이(가) 삽입되었습니다.

```
INSERT INTO USER_UNIQUE2
VALUES(1, NULL, NULL, '홍길동', '남',
       '010-1234-5678', 'hong123@kh.or.kr');
```

1 행 이(가) 삽입되었습니다.

```
INSERT INTO USER_UNIQUE2
VALUES(1, NULL, NULL, '홍길동', '남',
       '010-1234-5678', 'hong123@kh.or.kr');
```

1 행 이(가) 삽입되었습니다.

	USER_NO	USER_ID	USER_PWD	USER_NAME	GENDER	PHONE	EMAIL
1	1	user01	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr
2	1	(null)	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr
3	1	(null)	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr

** UNIQUE 제약조건은 NULL값 중복 저장이 가능하다.

** 해결방법은 테이블 생성시 컬럼레벨에 NOT NULL을 함께 지정하면 된다.

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - UNIQUE

```
CREATE TABLE USER_UNIQUE3(
    USER_NO NUMBER,
    USER_ID VARCHAR2(20),
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50),
    UNIQUE (USER_NO, USER_ID)
);
```

```
Table USER_UNIQUE3이(가) 생성되었습니다.
```

	TABLE_NAME	COLUMN_NAME	CONSTRAINT_NAME	CONSTRAINT_TYPE
1	USER_UNIQUE3	USER_NO	SYS_C007186	U
2	USER_UNIQUE3	USER_ID	SYS_C007186	U

** 두 개의 컬럼을 묶어서
하나의 UNIQUE 제약조건을 설정한다.

```
INSERT INTO USER_UNIQUE3
VALUES(1, 'user01', 'pass01', '홍길동', '남',
       '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_UNIQUE3
VALUES(2, 'user01', 'pass01', NULL, NULL,
       '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_UNIQUE3
VALUES(2, 'user02', 'pass02', NULL, NULL,
       '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_UNIQUE3
VALUES(1, 'user01', 'pass01', NULL, NULL,
       '010-1234-5678', 'hong123@kh.or.kr');
```

```
SQL 오류: ORA-00001: unique constraint (EMPLOYEE.SYS_C007186) violated
00001. 00000 -  "unique constraint (%s.%s) violated"
*Cause:    An UPDATE or INSERT statement attempted to insert a duplicate key.
           For Trusted Oracle configured in DBMS MAC mode, you may see
           this message if a duplicate entry exists at a different level.
*Action:   Either remove the unique restriction or do not insert the key.
```

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - PRIMARY KEY

테이블에서 한 행의 정보를 구분하기 위한 고유 식별자(Identifier) 역할을 한다.

NOT NULL과 UNIQUE의 의미를 둘 다 가지고 있으며, 한 테이블당 한 개만 설정 가능

```
CREATE TABLE USER_PRIMARYKEY(
    USER_NO NUMBER PRIMARY KEY,
    USER_ID VARCHAR2(20) UNIQUE,
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50)
);
```

```
Table USER_PRIMARYKEY이(가) 생성되었습니다.
```

```
INSERT INTO USER_PRIMARYKEY
VALUES(1, 'user01', 'pass01', '홍길동', '남',
        '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_PRIMARYKEY
VALUES(1, 'user02', 'pass02', '이순신', '남',
        '010-5678-9012', 'lee123@kh.or.kr');
```

```
SQL 오류: ORA-00001: unique constraint (EMPLOYEE.SYS_C007188) violated
00001. 00000 -  "unique constraint (%s.%s) violated"
*Cause:    An UPDATE or INSERT statement attempted to insert a duplicate key.
           For Trusted Oracle configured in DBMS MAC mode, you may see
           this message if a duplicate entry exists at a different level.
*Action:   Either remove the unique restriction or do not insert the key.
```

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - PRIMARY KEY

테이블에서 한 행의 정보를 구분하기 위한 고유 식별자(Identifier) 역할을 한다.

NOT NULL과 UNIQUE의 의미를 둘 다 가지고 있으며, 한 테이블당 한 개만 설정 가능

```
CREATE TABLE USER_PRIMARYKEY(
    USER_NO NUMBER,
    USER_ID VARCHAR2(20) UNIQUE,
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50),
    PRIMARY KEY (USER_NO)
);
```

```
Table USER_PRIMARYKEY이(가) 생성되었습니다.
```

```
INSERT INTO USER_PRIMARYKEY
VALUES(NULL, 'user03', 'pass03', '유관순', '여',
       '010-9999-3131', 'yoo123@kh.or.kr');
```

```
오류 보고 -
SQL 오류: ORA-01400: cannot insert NULL into ("EMPLOYEE"."USER_PRIMARYKEY"."USER_NO")
01400. 00000 -  "cannot insert NULL into (%s)"
*Cause:    An attempt was made to insert NULL into previously listed objects.
*Action:   These objects cannot accept NULL values.
```

	TABLE_NAME	COLUMN_NAME	CONSTRAINT_NAME	CONSTRAINT_TYPE
1	USER_PRIMARYKEY	USER_NO	SYS_C007188	P

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) – PRIMARY KEY

```
CREATE TABLE USER_PRIMARYKEY2(
    USER_NO NUMBER,
    USER_ID VARCHAR2(20),
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50),
    PRIMARY KEY (USER_NO, USER_ID)
);
```

```
Table USER_PRIMARYKEY2이(가) 생성되었습니다.
```

	TABLE_NAME	COLUMN_NAME	CONSTRAINT_NAME	CONSTRAINT_TYPE
1	USER_PRIMARYKEY2	USER_NO	SYS_C007196	P
2	USER_PRIMARYKEY2	USER_ID	SYS_C007196	P

** 두 개의 컬럼을 묶어서
하나의 PRIMARY KEY 제약조건을 설정한다.

```
INSERT INTO USER_PRIMARYKEY2
VALUES(1, 'user01', 'pass01', '홍길동', '남',
       '010-1234-5678', 'hong123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_PRIMARYKEY2
VALUES(1, 'user02', 'pass02', '이순신', '남',
       '010-5678-9012', 'lee123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_PRIMARYKEY2
VALUES(2, 'user01', 'pass01', '유관순', '여',
       '010-9999-3131', 'yoo123@kh.or.kr');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_PRIMARYKEY2
VALUES(1, 'user01', 'pass01', '신사임당', '여',
       '010-9999-9999', 'sin123@kh.or.kr');
```

	USER_NO	USER_ID	USER_PWD	USER_NAME	GENDER	PHONE	EMAIL
1	1	user01	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr
2	1	user02	pass02	이순신	남	010-5678-9012	lee123@kh.or.kr
3	2	user01	pass01	유관순	여	010-9999-3131	yoo123@kh.or.kr

```
오류 보고 -
SQL 오류: ORA-00001: unique constraint (EMPLOYEE.SYS_C007196) violated
00001. 00000 -  "unique constraint (%s.%s) violated"
*Cause:    An UPDATE or INSERT statement attempted to insert a duplicate key.
           For Trusted Oracle configured in DBMS MAC mode, you may see
           this message if a duplicate entry exists at a different level.
*Action:   Either remove the unique restriction or do not insert the key.
```

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) – FOREIGN KEY

참조 무결성을 유지하기 위한 제약조건이다.

즉, 참조된 다른 테이블이 제공하는 값만 사용할 수 있도록 제한을 거는 것이다.

```
CREATE TABLE USER_GRADE(
    GRADE _CODE NUMBER PRIMARY KEY
    GRADE_NAME VARCHAR2(30) NOT NULL
);
```

```
Table USER_GRADE이(가) 생성되었습니다.
```

```
SELECT * FROM USER_GRADE;
```

	GRADE_...	GRADE_NAME
1	10	일반회원
2	20	우수회원
3	30	특별회원

```
INSERT INTO USER_GRADE
VALUES(10, '일반회원');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_GRADE
VALUES(20, '우수회원');
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_GRADE
VALUES(30, '특별회원');
```

```
1 행 이(가) 삽입되었습니다.
```

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - FOREIGN KEY

참조 무결성을 유지하기 위한 제약조건이다.

즉, 참조된 다른 테이블이 제공하는 값만 사용할 수 있도록 제한을 거는 것이다.

```
CREATE TABLE USER_FOREIGNKEY(
    USER_NO NUMBER PRIMARY KEY,
    USER_ID VARCHAR2(20) UNIQUE,
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50),
    GRADE_CODE NUMBER,
    GRADE_CODE NUMBER FOREIGN KEY REFERENCES USER_GRADE (GRADE_CODE)
    혹은
    FOREIGN KEY (GRADE_CODE) REFERENCES USER_GRADE (GRADE_CODE)
);
```

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - FOREIGN KEY

```
INSERT INTO USER_FOREIGNKEY
VALUES(1, 'user01', 'pass01', '홍길동', '남', '010-1234-5678', 'hong123@kh.or.kr', 10);
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_FOREIGNKEY
VALUES(2, 'user02', 'pass02', '이순신', '남', '010-5678-9012', 'lee123@kh.or.kr', 20);
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_FOREIGNKEY
VALUES(3, 'user03', 'pass03', '유관순', '여', '010-9999-3131', 'yoo123@kh.or.kr', 30);
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_FOREIGNKEY
VALUES(4, 'user04', 'pass04', '안중근', '남', '010-2222-1111', 'ahn123@kh.or.kr', NULL);
```

```
1 행 이(가) 삽입되었습니다.
```

```
INSERT INTO USER_FOREIGNKEY
VALUES(5, 'user05', 'pass05', '윤봉길', '남', '010-6666-1234', 'yoon123@kh.or.kr', 50);
```

```
오류 보고 -
SQL 오류: ORA-02291: integrity constraint (EMPLOYEE.SYS_C007202) violated - parent key not found
02291. 00000 - "integrity constraint (%s.%s) violated - parent key not found"
*Cause:    A foreign key value has no matching primary key value.
*Action:   Delete the foreign key or add a matching primary key.
```

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - FOREIGN KEY

참조되는 컬럼과 참조된 컬럼을 통해 테이블간의 관계가 형성된다.

또한, 참조되는 값은 제공되는 값 이외에 NULL을 사용 가능하며, 참조할 테이블의 참조할 컬럼명을 생략할 경우에는 PRIMARY KEY로 설정된 컬럼이 자동으로 참조할 컬럼이 된다.

<USER_GRADE TABLE>

	GRADE_...	GRADE_NAME
1	10	일반회원
2	20	우수회원
3	30	특별회원

<USER_FOREIGNKEY TABLE>

	USER_ID	USER_PWD	USER_NAME	GENDER	PHONE	EMAIL	GRADE_CODE
1	user01	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr	10
2	user02	pass02	이순신	남	010-5678-9012	lee123@kh.or.kr	20
3	user03	pass03	유관순	여	010-9999-3131	yoo123@kh.or.kr	30
4	user04	pass04	안중근	남	010-2222-1111	ahn123@kh.or.kr	(null)

** FOREIGN KEY 제약조건으로 USER_GRADE TABLE의 GRADE_CODE 컬럼을 참조한다.

** USER_GRADE 테이블을 USER_FOREIGNKEY 테이블에서 참조하는 관계이기 때문에,
USER_GRADE 테이블의 데이터 삭제 시 참조무결성에 위배되기 때문에 삭제가 불가능하다.

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - FOREIGN KEY

삭제 옵션

부모 테이블의 데이터 삭제 시 자식 테이블의 데이터를 어떠한 방식으로 처리할지에 대한 내용을 제약조건 설정 시 옵션으로 지정할 수 있다.

기본 삭제 옵션은 ON DELETE RESTRICTED로 지정되어 있다.

```
CREATE TABLE USER_FOREIGNKEY(
    USER_NO NUMBER PRIMARY KEY,
    USER_ID VARCHAR2(20) UNIQUE,
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50),
    GRADE_CODE NUMBER REFERENCES USER_GRADE (GRADE_CODE)
                                        ON DELETE SET NULL
);
```

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - FOREIGN KEY

삭제 옵션

부모 테이블의 데이터 삭제 시 자식 테이블의 데이터를 어떠한 방식으로 처리할지에 대한 내용을 제약조건 설정 시 옵션으로 지정할 수 있다.

기본 삭제 옵션은 ON DELETE RESTRICTED로 지정되어 있다.

```
DELETE FROM USER_GRADE WHERE GRADE_CODE = 10;
```

	GRADE_CODE	GRADE_NAME
1	20	우수회원
2	30	특별회원

USER_ID	USER_PWD	USER_NAME	GENDER	PHONE	EMAIL	GRADE_CODE
user01	pass01	홍길동	남	010-1234-5678	hong123@kh.or.kr	(null)
user02	pass02	이순신	남	010-5678-9012	lee123@kh.or.kr	20
user03	pass03	유관순	여	010-9999-3131	yoo123@kh.or.kr	30
user04	pass04	안중근	남	010-2222-1111	ahn123@kh.or.kr	(null)

** 부모 테이블의 데이터 삭제 시 참조하고 있는 테이블의 컬럼 값이 NULL로 변경된다.

DDL(Data Definition Language)

KH정보교육원

제약 조건(CONSTRAINTS) - FOREIGN KEY

삭제 옵션

```
CREATE TABLE USER_FOREIGNKEY(
    USER_NO NUMBER PRIMARY KEY,
    USER_ID VARCHAR2(20) UNIQUE,
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50),
    GRADE_CODE NUMBER REFERENCES USER_GRADE (GRADE_CODE)
                                        ON DELETE SET CASCADE
);
```

```
DELETE FROM USER_GRADE WHERE GRADE_CODE = 10;
```

	GRADE_CODE	GRADE_NAME
1	20	우수회원
2	30	특별회원

	USER_ID	USER_PWD	USER_NAME	GENDER	PHONE	EMAIL	GRADE_CODE
2	user02	pass02	이순신	남	010-5678-9012	lee123@kh.or.kr	20
3	user03	pass03	유관순	여	010-9999-3131	yoo123@kh.or.kr	30
4	user04	pass04	안중근	남	010-2222-1111	ahn123@kh.or.kr	(null)

** 부모 테이블의 데이터 삭제 시 참조하고 있는 테이블의 컬럼 값이 존재하던 행 전체를 삭제한다.

DDL(Data Definition Language)

KH정보교육원

CHECK문

해당 컬럼에 입력되거나 수정되는 값을 체크하여, 설정된 값 이외의 값이면 에러를 발생시킨다. 비교연산자를 이용하여 조건을 설정하며, 비교값은 리터럴만 사용 가능하고 변하는 값이나 함수 사용이 불가능하다.

```
CREATE TABLE USER_CHECK(
    USER_NO NUMBER PRIMARY KEY,
    USER_ID VARCHAR2(20) UNIQUE,
    USER_PWD VARCHAR2(30) NOT NULL,
    USER_NAME VARCHAR2(30),
    GENDER VARCHAR2(10) CHECK (GENDER IN ('남', '여')),
    PHONE VARCHAR2(30),
    EMAIL VARCHAR2(50),
    GRADE_CODE NUMBER
);
```

```
INSERT INTO USER_CHECK
VALUES(1, 'user01', 'pass01', '홍길동', '남자',
        '010-1234-5678', 'hong123@kh.or.kr');
```

```
오류 보고 -
SQL 오류: ORA-02290: check constraint (EMPLOYEE.SYS_C007225) violated
02290. 00000 -  "check constraint (%s.%s) violated"
*Cause:    The values being inserted do not satisfy the named check

*Action:   do not insert values that violate the constraint.
```

DDL(Data Definition Language)

KH정보교육원

SUBQUERY를 이용한 CREATE TABLE

서브쿼리를 이용해서 SELECT의 조회 결과로 테이블을 생성하는 방법이다. 컬럼명과 데이터타입, 값이 복사되고, 제약조건은 NOT NULL만 복사된다.

```
CREATE TABLE EMPLOYEE_
COPY AS
  SELECT
        EMP_ID,
        EMP_NAME,
        SALARY,
        DEPT_TITLE,
        JOB_NAME
    FROM EMPLOYEE
    LEFT JOIN DEPARTMENT ON(DEPT_CODE = DEPT_ID)
    LEFT JOIN JOB USING(JOB_CODE);
```

SQL | 인출된 모든 행: 23(0.016초)

	EMP_ID	EMP_NAME	SALARY	DEPT_TITLE	JOB_NAME
1	214	방명수	1380000	인사관리부	사원
2	216	차태연	2780000	인사관리부	대리
3	217	전지연	3660000	인사관리부	대리
4	219	임시환	1550000	회계관리부	차장
5	220	이중석	2490000	회계관리부	차장
6	221	유하진	2480000	회계관리부	차장
7	206	박나라	1800000	해외영업1부	사원
.					
.					
.					
21	200	선동일	8000000	총무부	대표
22	218	이오리	2890000	(null)	사원
23	213	하동운	2320000	(null)	대리