

Oracle DBMS

<DML & DDL의 CRUD>

CRUD	DML- Data	DDL- Table
Create	INSERT	CREATE
Read	SELECT	DESC
Update	UPDATE	ALTER, RENAME
Delete	DELETE	TRUNCATE, DROP

<FOREIGN KEY>

FK(FOREIGN KEY) - 두 키와의 상관관계를 지정

PK(PRIMARY KEY)

EMP TABLE의 DEPTNO와 연결되어서 테이블이 만들어짐.

<TABLE 생성 - FK와 PK의 상관관계>

```
CREATE TABLE DEPT_FK(  
    DEPTNO NUMBER(2) CONSTRAINT DEPTFK_DEPTNO_PK PRIMARY KEY,  
    DNAME VARCHAR2(14),  
    LOC VARCHAR2(13)  
);
```

PK를 참조하여 테이블을 만들어낸다.

PK - NOT NULL, UNIQUE

```
CREATE TABLE EMP_FK(  
    EMPNO NUMBER(4) CONSTRAINT EMPFK_EMPNO_PK PRIMARY KEY,  
    ENAME VARCHAR2(10),  
    JOB VARCHAR2(9),  
    MGR NUMBER(4),  
    HIREDATE DATE,  
    SAL NUMBER(7,2),  
    COMM NUMBER(7,2),  
    DEPTNO NUMBER(2) CONSTRAINT EMPFK_DEPTNO_FK REFERENCES DEPT_FK  
(DEPTNO)  
);
```

EMP에서는 PK, FK 모두 가진다.

REFERENCES - 참고하다

FK는 참조 테이블의 컬럼을 참조한다.

<데이터 삽입>

```
INSERT INTO EMP_FK
```

```
VALUES(9999, 'TEST_NNAME', 'TEST_JOB', NULL, TO_DATE('2001/01/01', 'YYYY/MM/DD'),
      3000, NULL, 10);
```

데이터가 없기 때문에 에러 발생

ORA-02291: integrity constraint (SCOTT.EMPFK_DEPTNO_FK) violated - parent key not found

```
INSERT INTO DEPT_FK
VALUES(10, 'TEST_DNAME', 'TEST_LOC');
```

```
SELECT * FROM DEPT_FK;
```

정상적으로 입력이 된다.

그 이후 에러난 데이터를 다시 입력하면 정상적으로 삽입이 된다.

FK - FK로 TABLE에서 VALUE를 참조하기 때문에 FK는 잘 설정해야함.

```
DELETE FROM DEPT_FK
```

```
WHERE DEPTNO = 10;
```

오류 보고 -

ORA-02292: integrity constraint (SCOTT.EMPFK_DEPTNO_FK) violated - child record found

<CHECK - 데이터 형태와 범위 정하기>

CHECK를 사용하여 제약조건을 테이블 컬럼내에 넣어놓을 수 있다.

<테이블 생성시 CHECK 사용하여 제약조건 설정>

```
CREATE TABLE TABLE_CHECK(
  LOGIN_ID VARCHAR2(20) CONSTRAINT TBLCK_LOGINID_PK PRIMARY KEY,
  LOGIN_PWD   VARCHAR2(20)   CONSTRAINT      TBLCK_LOGINPW_CK      CHECK
  (LENGTH(LOGIN_PWD) > 3),
  TEL VARCHAR2(20)
);
```

```
DESC TABLE_CHECK;
```

<제약조건과 테이블>

```
INSERT INTO TABLE_CHECK
VALUES ('TEST_ID', '123', '010-1234-5678');
```

제약조건 에러 걸림

ORA-02290: check constraint (SCOTT.TBLCK_LOGINPW_CK) violated

```
INSERT INTO TABLE_CHECK
VALUES ('TEST_ID', '1234', '010-1234-5678');
```

```
SELECT * FROM TABLE_CHECK;
```

제약조건에 맞게 테이블 생성하여야 함.

```
SELECT OWNER, CONSTRAINT_NAME, CONSTRAINT_TYPE, TABLE_NAME  
FROM USER_CONSTRAINTS  
WHERE TABLE_NAME LIKE 'TABLE_CHECK';
```

체크로 생성한 제약조건도 C로 나타난다.

<DEFAULT 기본값>

<DEFAULT 제약 조건 설정>

```
CREATE TABLE TABLE_DEFAULT(  
    LOGIN_ID VARCHAR2(20) CONSTRAINT TBLCK2_LOGINID_PK PRIMARY KEY,  
    LOGIN_PWD VARCHAR2(20) DEFAULT '1234',  
    TEL VARCHAR2(20)  
);
```

```
DESC TABLE_DEFAULT;
```

<DEFAULT에서의 널값 입력과 입력하지 않을 때 차이>

```
INSERT INTO TABLE_DEFAULT VALUES ('TEST_ID', NULL, '010-1234-5678');  
INSERT INTO TABLE_DEFAULT (LOGIN_ID, TEL) VALUES ('TEST_ID2', '010-1234-5678');  
SELECT * FROM TABLE_DEFAULT;
```

NULL입력을 하면 NULL값 입력하지 않으면 디폴트로 설정한 값(1234)이 들어가게 된다.

<사용자, 권한, 롤관리 - DBA(DataBase Administrator)의 역할>

SYSTEM은 만든 것이 아님

SCOTT만 만든 계정이다.

<계정 생성>

- SCOTT에서 생성

```
CREATE USER ORCLSTUDY  
IDENTIFIED BY ORACLE;
```

에러가 뜬다.

ORA-01031: insufficient privileges

01031. 00000 - "insufficient privileges"

*Cause: An attempt was made to perform a database operation without the necessary privileges.

*Action: Ask your database administrator or designated security administrator to grant you the necessary privileges

권한이 없어 생성이 되지 않음.
시스템(DBA)에서 하는 것이다.
사용자는 사용자를 만들 수가 없음.

사용자는 생성되었지만 접속을 해야 함.

GRANT로 권한을 부여해야 한다.

<사용자에게 권한 부여>

```
GRANT CREATE SESSION TO ORCLSTUDY;
```

<사용자 정보 조회>

```
SELECT * FROM ALL_USERS  
WHERE USERNAME = 'ORCLSTUDY';  
SELECT * FROM DBA_USERS  
WHERE USERNAME = 'ORCLSTUDY';  
SELECT * FROM DBA_OBJECTS  
WHERE OWNER = 'ORCLSTUDY';
```

OWNER대신 USERNAME을 사용하게 되면 아래 오류가 나타난다.

ORA-00904: "USERNAME": invalid identifier

계정접속해야 생성한 것이 나타남.

+눌러서 NAME(구분하기 위해서 사용자 이름과 동일하게 작성) / 사용자 이름 / 비번 써주고 입력

<계정에서의 CRUD>

계정 생성 (Create) : CREATE USER 문을 사용하여 새로운 계정을 생성할 수 있습니다. CREATE USER 문은 사용자 이름, 비밀번호, 권한 등을 지정하여 계정을 생성합니다.

계정 조회 (Read) : DBA_USERS 또는 ALL_USERS 뷰를 사용하여 데이터베이스에 등록된 모든 계정을 조회할 수 있습니다. 또한, 각 계정에 부여된 권한을 확인할 수 있습니다.

계정 수정 (Update) : ALTER USER 문을 사용하여 계정의 비밀번호, 권한 등을 수정할 수 있습니다. 이때, SYSDBA 권한을 가진 계정으로 로그인하여 ALTER USER 문을 실행해야 합니다.

계정 삭제 (Delete) : DROP USER 문을 사용하여 계정을 삭제할 수 있습니다. 이때, SYSDBA 권한을 가진 계정으로 로그인하여 DROP USER 문을 실행해야 합니다. 삭제된 계정에 속한 객체는 CASCADE 옵션을 사용하여 함께 삭제할 수 있습니다.

<사용자 패스워드 변경 - DBA>

```
ALTER USER ORCLSTUDY  
IDENTIFIED BY ORCL;
```

<사용자 삭제 - DBA>

```
DROP USER ORCLSTUDY;  
ORA-01940: cannot drop a user that is currently connected
```

01940. 00000 - "cannot drop a user that is currently connected"

*Cause: Attempt was made to drop a user that is currently logged in.

*Action: Make sure user is logged off, then repeat command.

접속중일 때 에러가 뜸

<모두 삭제 - 사용자 및 오브젝트까지 삭제>

DROP USER ORCLSTUDY CASCADE;

<권한 관리>

CREATE USER ORCLSTUDY

IDENTIFIED BY ORACLE;

<권한 부여>

GRANT RESOURCE, CREATE SESSION, CREATE TABLE TO ORCLSTUDY;

<권한 부여 취소>

REVOKE RESOURCE, CREATE TABLE FROM ORCLSTUDY;

<권한 부여 취소 후>

CREATE TABLE TEMP2(

COL1 VARCHAR(20));

ORA-01031: insufficient privileges

01031. 00000 - "insufficient privileges"

*Cause: An attempt was made to perform a database operation without the necessary privileges.

*Action: Ask your database administrator or designated security administrator to grant you the necessary privileges

권한 부여 취소해서 되지 않음.

<SCOTT에서 다른 사용자(ORCLSTUDY)에게 권한 부여>

CREATE TABLE TEMP(

COL1 VARCHAR(20),

COL2 VARCHAR(20)

);

GRANT SELECT ON TEMP TO ORCLSTUDY;

GRANT INSERT ON TEMP TO ORCLSTUDY;

또는

GRANT SELECT, INSERT ON TEMP

TO ORCLSTUDY;

다른 계정의 권한을 받아서 데이터 접근이 가능해진다.

<다른사용자의 테이블 입력 - ORCLSTUDY에서>

SELECT * FROM SCOTT.TEMP;

```
INSERT INTO SCOTT.TEMP VALUES('TEXT', 'FROM ORCLSTUDY');  
SELECT * FROM SCOTT.TEMP;
```

<권한 취소>

REVOKE 사용

```
REVOKE SELECT, INSERT ON TEMP FROM ORCLSTUDY;
```

권한이 없어 조회 실패하게 됨.

사용자가 사용자에게 DB를 볼 수 있는 것을 가능하게 할 수 있고 제한 할 수 있다.

<롤 관리>

ROLES

<연결> - CREATE SESSION

RESOURCE

TABLE, SEQUENCE를 비롯한 여러 Object를 생성할 수 있는 기본 시스템이 리소스

<사용자 정의 롤>

임의로 롤 권한을 포함하여 부여 할 수 있음

```
CREATE ROLE ROLESTUDY; - 사용자 롤을 만드는데 이름 : ROLESTUDY
```

```
GRANT CONNECT, RESOURCE, CREATE VIEW, CREATE SYNONYM
```

```
TO ROLESTUDY;
```

<ORCLSTUDY에 롤 권한 부여>

```
GRANT ROLESTUDY TO ORCLSTUDY;
```

<부여된 롤 권한 확인>

```
SELECT * FROM USER_SYS_PRIVS;
```

```
SELECT * FROM USER_ROLE_PRIVS;
```

<롤 권한 취소>

REVOKE

<롤 삭제하기>

DROP