



SQL,PL/SQL(ORACLE)

4-2. 제약조건

제약 조건 CONSTRAINT

(1)개요

테이블의 해당 컬럼에 올바른 데이터를 받기위해 바르지 않은 입력/변경/삭제 가 발생되지 않도록 테이블 생성(CREATE)시 또는 변경(ALTER)시 설정하는 규칙을 말한다.

예) 회원가입시 회원의 아이디는 동일하거나 없을 수 없다.
초등학교 성적관리에서 7학년은 필요없다.



(2) 종류

1) PRIMARY KEY (식별키, 기본키) PK

하나의 테이블에 오직 하나만 존재하며 자동으로 인덱스가 부여되는,
하나의 ROW 데이터를 대표하는 대표 컬럼 .

2) FOREIGN KEY (참조키 / 외래키) FK

부모테이블의 PK(PRIMARY KEY)를 참조하는 컬럼

3) UNIQUE KEY (유일키)

PK가 아니더라도 컬럼의 모든 값이 유일하게해야 하는 경우
에 사용하는 제약조건이다.
즉, 중복데이터를 허용하지 않는 컬럼

cf) PK와 차이점

첫째, NULL을 입력할 수 있다.

둘째, 하나의 테이블에 여러개의 유일키 생성 가능

4) CHECK

조건에 맞는 데이터만 입력되도록 조건을 부여한 제약조건
조건으로는 데이터의 값의 범위, 특정패턴숫자,
문자값 설정이 가능한 제약 조건

5) NOT NULL

NULL 이 입력되어서는 안되는 컬럼에 부여하는 제약조건으로
COLUMN-LEVEL 으로만 부여할 수 있는 제약조건

cf) PK는 NOT NULL 조건을 가지고 있음



예) EMP 테이블 참고

```
CREATE TABLE EMP_TAB(  
  EMPNO NUMBER(4),  
  ENAME VARCHAR2(10),  
  JOB_ID VARCHAR2(9),  
  MGR NUMBER(4) CONSTRAINT EMP_SELF_KEY REFERENCES EMP(EMPNO),  
  HIRE_DATE DATE,  
  SAL NUMBER(7),  
  COMM NUMBER(1),  
  DEPTNO NUMBER(2) NOT NULL,  
  CONSTRAINT EMP_DEPTNO_FK FOREIGN KEY(DEPTNO) REFERENCES DEPT(DEPTNO),  
  CONSTRAINT EMP_EMPNO_PK PRIMARY KEY(EMPNO)  
);
```

오류나는 DDL이다. 수정해서 완성하시오.

예) test_TAB 테이블 생성

```
CREATE TABLE test_TAB(  
  EMPNO NUMBER(4),  
  CONSTRAINT test_TAB_PK PRIMARY KEY(EMPNO)  
);
```

제약조건 생성확인

```
select constraint_name from user_constraints;
```

```
ALTER table test_TAB ADD CONSTRAINT test_TAB_PK PRIMARY KEY(EMPNO);  
ALTER table test_TAB DROP CONSTRAINT test_TAB_PK ;
```

정답

create table dept as select * from departments where 1=2;

-- 복사한 dept 테이블 컬럼명 변경 deptno 및 기본키 할당.

```
CREATE TABLE EMP_TAB(  
  EMPNO NUMBER(4),  
  ENAME VARCHAR2(10),  
  JOB_ID VARCHAR2(9),  
  MGR NUMBER(4) CONSTRAINT EMP_SELF_KEY  
    REFERENCES EMP_TAB(EMPNO),  
  HIRE_DATE DATE,  
  SAL NUMBER(7),  
  COMM NUMBER(1),  
  DEPTNO NUMBER(2) NOT NULL,  
  CONSTRAINT EMP_DEPTNO_FK FOREIGN KEY(DEPTNO)  
    REFERENCES DEPT(DEPTNO),  
  CONSTRAINT EMP_EMPNO_PK PRIMARY KEY(EMPNO)  
);
```



1. 이름없는 제약조건을 갖는 테이블

```
CREATE TABLE test_dept(  
  deptno NUMBER PRIMARY KEY,  
  dname VARCHAR2(20) DEFAULT '개발부',  
  loc CHAR(1) CHECK(loc IN('1', '2')));
```

```
CREATE TABLE test_emp(  
  no NUMBER(4) PRIMARY KEY,  
  name VARCHAR2(20) NOT NULL,  
  loc VARCHAR2(4) CHECK(loc IN('서울', '부산')),  
  jumin CHAR(13) UNIQUE,  
  deptno NUMBER REFERENCES test_dept(deptno) );
```

```
INSERT INTO test_dept (deptno, loc) VALUES(10, '2');  
INSERT INTO test_dept VALUES(20, '기획부', '1');  
INSERT INTO test_dept VALUES(30, '영업부', '2');  
INSERT INTO test_dept VALUES(40, '관리부', '1');
```

제약조건 알아보기

```
SQL> select constraint_name from user_constraints;
```



2.이름있는 제약조건을 갖는 테이블1

```
CREATE TABLE test2_dept(  
  deptno NUMBER CONSTRAINT test2_pk_dept_deptno PRIMARY KEY,  
  dname VARCHAR2(15) DEFAULT '개발부',  
  loc CHAR(1) CONSTRAINT test2_ck_dept_loc CHECK(loc IN('1', '2')));
```

```
CREATE TABLE test2_emp(  
  no NUMBER(4) CONSTRAINT test2_pk_emp_no PRIMARY KEY,  
  name VARCHAR2(10) CONSTRAINT test2_nn_emp_name NOT NULL,  
  loc VARCHAR2(4) CONSTRAINT test2_ck_emp_loc CHECK(loc IN('서울', '부산')),  
  jumin CHAR(13) CONSTRAINT test2_uq_emp_jumin UNIQUE,  
  deptno NUMBER(2) CONSTRAINT test2_fk_emp_deptno REFERENCES test2_dept(deptno));
```

```
INSERT INTO test2_dept (deptno, loc) VALUES(10, '2');  
INSERT INTO test2_dept VALUES(20, '기획부 ', '1');  
INSERT INTO test2_dept VALUES(30, '영업부 ', '2');  
INSERT INTO test2_dept VALUES(40, '관리부 ', '1');
```

제약조건 알아보기

```
SQL> select constraint_name from user_constraints;
```



3.이름있는 제약조건을 갖는 테이블2

```
CREATE TABLE test3_dept(  
  deptno NUMBER(2),  
  dname VARCHAR2(15) default '개발부',  
  loc CHAR(1),  
  CONSTRAINT test3_dept_deptno_pk PRIMARY KEY (deptno),  
  CONSTRAINT test3_dept_loc_ck CHECK(loc IN('1', '2'))  
);
```

```
CREATE TABLE test3_emp(  
  no NUMBER(4),  
  name VARCHAR2(10) CONSTRAINT test3_emp_name_nn NOT NULL,  
  loc VARCHAR2(4),  
  jumin CHAR(13),  
  deptno NUMBER(2),  
  CONSTRAINT test3_emp_no_pk PRIMARY KEY (no),  
  CONSTRAINT test3_emp_loc_ck CHECK (loc IN('서울', '부산')),  
  CONSTRAINT test3_emp_jumin_uq UNIQUE (jumin),  
  CONSTRAINT test3_emp_deptno_fk FOREIGN KEY (deptno) REFERENCES  
    test3_dept(deptno)  
);
```

제약조건 알아보기

```
SQL> select constraint_name from user_constraints;
```



- 제약조건 이름 검색하기

```
select CONSTRAINT_NAME from USER_CONSTRAINTS;
```

	CONSTRAINT_NAME	STATUS
69	TEST2 CK DEPT LOC	ENABLED
70	TEST2 PK DEPT DEPTNO	ENABLED
71	TEST3 DEPT LOC CK	ENABLED
72	TEST3 DEPT DEPTNO PK	ENABLED
73	TEST3 EMP NAME NN	ENABLED
74	TEST3 EMP LOC CK	ENABLED
75	TEST3 EMP NO PK	ENABLED
76	TEST3 EMP JUMIN UQ	ENABLED

- 제약조건을 수정 할 수는 없고 **삭제만 가능하다.**

```
ALTER TABLE test3_dept
DROP CONSTRAINT test3_dept_deptno_pk CASCADE;
```

- 제약 조건 **추가하기**

```
ALTER TABLE test3_dept
ADD CONSTRAINT test3_dept_deptno_pk PRIMARY KEY(deptno);
```

- 제약조건 < **활성화 상태 검색** >

```
select CONSTRAINT_NAME , STATUS from USER_CONSTRAINTS;
```

- 제약조건 **비활성화 시키기**

```
ALTER TABLE test3_emp DISABLE CONSTRAINT test3_emp_name_nn;
select CONSTRAINT_NAME , STATUS from USER_CONSTRAINTS;
```

```
ALTER TABLE test3_emp ENABLE CONSTRAINT test3_emp_name_nn;
select CONSTRAINT_NAME , STATUS from USER_CONSTRAINTS;
```



게시판에 해당하는 부모테이블을 만들고
답글에 해당하는 자식테이블을 만들어서 부모 글번호를 참조하도록
하시오.

글번호(정수) WNUM 기본키
작성자(문자) WRITER NULL허용안함
글제목(문자) TITLE NULL허용안함
글내용(문자) CON NULL허용안함
작성날짜(날자) WDATE NULL허용안함 기본값 시스템현재날자
조회수(정수) VCOUNT 기본값 0

```
CREATE TABLE TEST_BOARD(  
WNUM NUMBER(4),  
WRITER VARCHAR2(10) CONSTRAINT test_b_writer_nn NOT NULL,  
TITLE VARCHAR2(4) CONSTRAINT test_b_title_nn NOT NULL,  
CON VARCHAR2(4) CONSTRAINT test_b_con_nn NOT NULL,  
WDATE CHAR(13) default SYSDATE CONSTRAINT test_b_wdate_nn  
NOT NULL,  
VCOUNT NUMBER(2),  
CONSTRAINT test_b_wnum_pk PRIMARY KEY (WNUM)  
);
```



게시판에 해당하는 부모테이블을 만들고
답글에 해당하는 자식테이블을 만들어서 부모 글번호를 참조하도록
하시오.

```
CREATE TABLE TEST_BOARD_COMM(  
  COMM_ID NUMBER(4),  
  WRITER VARCHAR2(10) CONSTRAINT test_b_com_writer_nn NOT NULL,  
  TITLE VARCHAR2(4) CONSTRAINT test_b_com_title_nn NOT NULL,  
  CON VARCHAR2(4) CONSTRAINT test_b_com_con_nn NOT NULL,  
  WDATE CHAR(13) default SYSDATE CONSTRAINT test_b_com_wdate_nn NOT  
  NULL,  
  wnum NUMBER(4) ,  
  VCOUNT NUMBER(2),  
  CONSTRAINT test_b_com_id_wnum_pk PRIMARY KEY (COMM_ID) ,  
  CONSTRAINT test_b_com_wnum_fk FOREIGN KEY (wnum)  
  REFERENCES TEST_BOARD(wnum)  
);
```

```
select constraint_name from user_constraints;
```

