

CONSTRAINT

[의미] 데이터의 무결성(INTEGRITY)을 보장하기 위해서 사용되는 제약사항

[종류] ① 선언적 무결성 제약사항(Declarative Integrity Constraint)

② TRIGGER

③ APPLICATION LOGIC

선언적 무결성 제약사항

[종류]

① PRIMARY KEY : 대표성, 고유성, 존재성 → NOT NULL, 1개만 정의, Unique Index

② UNIQUE KEY : 고유성, N개 정의, NULL 허용, Unique Index 생성

NULL이 여러개 인 경우 어떻게 되나 ?

③ CHECK : 값의 유효성 제약, Boolean 연산 처리

④ NOT NULL : 필수 입력 사항

⑤ FOREIGN KEY : 테이블간의 관계 정의, 기본키를 다른 테이블에 저장 → 연결고리
, 내용에 의한 참조

[LEVEL]

① TABLE LEVEL

② COLUMN LEVEL

[생성시기]

① TABLE 생성시 생성

② TABLE 생성후 임의의 시점에 추가

NOT NULL (필수입력사항)

SQL> ED T_CONS.SQL

-- SQL SCRIPT FILE로 작업

-- SQL SCRIPT FILE을 사용하는 이유?

-- 1. 반복작업시 2. 여러 SQL문장을 순차적으로 사용하기 위해서

```
DROP TABLE SCOTT.CUSTOMER;
CREATE TABLE SCOTT.CUSTOMER (
    ID          VARCHAR2(8)    NOT NULL,
    PWD         VARCHAR2(8)    CONSTRAINT CUSTOMER_PWD_NN NOT NULL,
    NAME        VARCHAR2(20),  -- 이름
    SEX         CHAR(1),       -- 성별 [M|F] M:MALE F:FEMALE
    AGE         NUMBER(3)      -- 나이
) TABLESPACE USERS
PCTFREE 5 PCTUSED 60 INITRANS 2 MAXTRANS 20
STORAGE(INITIAL 100K NEXT 100K MINEXTENTS 3 MAXEXTENTS 10 PCTINCREASE 0);
```

[요구] ㉠ Block 공간 활용 Parameter PCTFREE, PCTUSED, INITRANS, MAXTRANS에 대해 설명 하십시오

㉡ TABLESPACE 정의 및 용도 와 STORAGE에 대해 설명 하십시오

SQL> START T_CONS.SQL or @ T_CUSTOMER.SQL

- ① DESC CUSTOMER -- id,pwd column 관찰...
 - ② INSERT INTO CUSTOMER(ID,PWD,NAME,SEX,AGE) VALUES('xman','ok','kang','M',21);
 - ③ INSERT INTO CUSTOMER(ID,PWD,NAME,SEX,AGE) VALUES('XMAN','no','kim','T',-20); -- ID 중복?, 성별? 나이?
-- implicit null insert
 - ④ INSERT INTO CUSTOMER(ID,NAME,AGE) VALUES('zman','son',99); -- PWD가 입력되지 않은 상황
-- explicit null insert
 - ⑤ INSERT INTO CUSTOMER(ID,PWD,NAME,AGE) VALUES('rman',NULL,'jjang',24); -- PWD ?
 - ⑥ INSERT INTO CUSTOMER(ID,PWD,NAME,AGE) VALUES('','pwd','jjang',24); -- ID ?
 - ⑦ UPDATE CUSTOMER SET AGE= -1, NAME = NULL; -- 사건발생...
 - ⑧ UPDATE CUSTOMER SET PWD = NULL WHERE ID = 'XMAN'; -- ID가 XMAN인 ROW만 수정
 - ⑨ UPDATE CUSTOMER SET PWD = NULL; -- UPDATE시 WHERE절이 없는 경우? 전체수정
 - ⑩ SELECT * FROM CUSTOMER;
- DATA DICTIONARY에서 CONSTRAINT 정보 조회
- ⑪ SELECT TABLE_NAME,CONSTRAINT_NAME,CONSTRAINT_TYPE,SEARCH_CONDITION
FROM USER_CONSTRAINTS WHERE TABLE_NAME = 'CUSTOMER';
 - ⑫ SELECT TABLE_NAME,CONSTRAINT_NAME,POSITION,COLUMN_NAME FROM USER_CONS_COLUMNS
WHERE TABLE_NAME = 'CUSTOMER' ORDER BY CONSTRAINT_NAME,POSITION;
- [요구] ㉓ DATA DICTIONARY에 대해서 설명하고 DBA_ , ALL_ , USER_ 에 대해서 설명 하십시오

CHECK: BOOLEAN CHECK

SQL> ED T_CONS2.SQL -- SQL SCRIPT FILE로 작업

```

DROP TABLE SCOTT.CUSTOMER2;
CREATE TABLE CUSTOMER2(
    ID VARCHAR2(8) NOT NULL,
    PWD VARCHAR2(8) CONSTRAINT CUSTOMER2_PWD_NN NOT NULL,
    NAME VARCHAR2(20),
    SEX CHAR(1) CONSTRAINT CUSTOMER2_SEX_CK CHECK (SEX IN ('M','F')),
    AGE NUMBER(3) CHECK ( AGE > 0 AND AGE < 100)
);

```

- ⑬ INSERT INTO CUSTOMER2(ID,PWD,NAME,SEX, AGE) VALUES('xman','ok','kang','M',21);
- ⑭ INSERT INTO CUSTOMER2(ID,PWD,NAME,SEX,AGE) VALUES('xman','ok','jjang','M',20); -- ID 중복

- ① INSERT INTO CUSTOMER2(ID,PWD,NAME,SEX,AGE) VALUES('XMAN','no','kim','M',-20); -- ID 중복?,나이?
- ② INSERT INTO CUSTOMER2(ID,PWD,NAME,AGE) VALUES('asura','ok','joo',99); -- 성별?
- ③ INSERT INTO CUSTOMER2(ID,PWD,NAME,SEX,AGE) VALUES('harisu','ok','susu','T',33); -- 성별?
- ④ INSERT INTO CUSTOMER2(ID,PWD,NAME,SEX,AGE) VALUES('shinsun','ok','도사','M',999);
- ⑤ UPDATE CUSTOMER SET AGE = AGE + 1; --??

ID	AGE
-----	-----
xman	21
xman	20
asura	99

// STATEMENT LEVEL ROLLBACK, TRANSACTION LEVEL ROLLBACK

UNIQUE: 컬럼 또는 컬럼조합의 고유한 값을 보장

SQL> ED T_CONS3.SQL

```

DROP TABLE CUSTOMER3;
CREATE TABLE CUSTOMER3(
    ID      VARCHAR2(8)    NOT NULL CONSTRAINT CUSTOMER3_ID_UK UNIQUE,
    PWD     VARCHAR2(8)    NOT NULL,
    NAME    VARCHAR2(20),
    SEX     CHAR(1)        DEFAULT 'M' CONSTRAINT CUSTOMER_SEX_CK CHECK (SEX IN ('M','F')),
    MOBILE  VARCHAR2(14)   UNIQUE,          -- 핸드폰 번호
    AGE     NUMBER(3)      DEFAULT 18
);

```

// 동일한 이름을 가진 제약사항이 생성될수 있는가?

- ⑥ INSERT INTO CUSTOMER3(ID,PWD,NAME,MOBILE, AGE) VALUES('xman','ok','kang','011-3333',21); -- 성별?
- ⑦ INSERT INTO CUSTOMER3(ID,PWD,NAME, MOBILE,AGE) VALUES('yman','yes','lee','011-3333',28); --핸폰?
- ⑧ INSERT INTO CUSTOMER3(ID,PWD,NAME, MOBILE,AGE) VALUES('XMAN','yes','kim','011-3334',33); --ID중복
- ⑨ INSERT INTO CUSTOMER3(ID,PWD,NAME, MOBILE,AGE) VALUES('xman','yes','lee','011-3335',-21);--ID중복
- ⑩ INSERT INTO CUSTOMER3(ID,PWD,NAME, MOBILE) VALUES('무명인','yes',NULL, NULL); -- NULL?

// 테이블 생성후 제약사항 신규 추가

- ⑪ ALTER TABLE CUSTOMER3 ADD CONSTRAINT CUSTOMER_NAME_SEX_UK UNIQUE(NAME,SEX); --조합,2개
- ⑫ ALTER TABLE CUSTOMER3 MODIFY(NAME NOT NULL); -- NOT NULL
- ⑬ INSERT INTO CUSTOMER3(ID,PWD,NAME, SEX) VALUES('rman','yes','ksh','M');
- ⑭ INSERT INTO CUSTOMER3(ID,PWD,NAME, SEX) VALUES('Rman','yes','ksh','F'); -- 이름 중복 허용?
- ⑮ INSERT INTO CUSTOMER3(ID,PWD,NAME, SEX) VALUES('RmaN','yes','ksh','M'); -- 조합의 중복 ??

SELECT * FROM CUSTOMER3;

// CONSTRAINT(제약사항) 확인

① SELECT INDEX_NAME, INDEX_TYPE, UNIQUENESS FROM USER_INDEXES
WHERE TABLE_NAME = 'CUSTOMER3';

// INDEX 생성 여부 확인

② SELECT INDEX_NAME, COLUMN_POSITION, COLUMN_NAME FROM USER_IND_COLUMNS
WHERE TABLE_NAME = 'CUSTOMER3' ORDER BY INDEX_NAME, COLUMN_POSITION;

[요구~캔]

③ INDEX에 대해서 설명하고 UNIQUE 제약사항이 걸린 컬럼에 UNIQUE INDEX 가 생성되는 이유를 설명 하십시오

PRIMARY KEY : ROW(RECORD)의 유일성을 보장하는 식별자

SQL> ED T_CONS4.SQL

```
DROP TABLE CUSTOMER4;
CREATE TABLE CUSTOMER4(
    ID VARCHAR2(8) CONSTRAINT CUSTOMER_ID_PK PRIMARY KEY,
    PWD VARCHAR2(8) NOT NULL,
    NAME VARCHAR2(20),
    SEX CHAR(1) DEFAULT 'M' CONSTRAINT CUSTOMER_SEX_CK CHECK (SEX IN ('M','F')),
    MOBILE VARCHAR2(14) CONSTRAINT CUSTOMER_MOBILE_UK UNIQUE,
    AGE NUMBER(3) DEFAULT 18
);
```

// Default의 의미: 입력시 컬럼의 값이 지정되지 않는 경우 NULL 값대신 저장하는 값

입력시 컬럼에 값이 지정되지 않으면 기본적으로 NULL이 들어 간다.

③ INSERT INTO CUSTOMER4(ID,PWD,NAME,MOBILE) VALUES('zman','ok','한국','011');

④ INSERT INTO CUSTOMER4(ID,PWD,NAME) VALUES('xman','ok','king');

⑤ INSERT INTO CUSTOMER4(ID,PWD,NAME) VALUES('xman','power','zzang'); -- ID중복??

⑥ INSERT INTO CUSTOMER4(ID,PWD,NAME) VALUES('Xman','korea','dbzzang'); -- ID중복??

⑦ INSERT INTO CUSTOMER4(ID,PWD,NAME) VALUES(lower('xMan'),'ok','zzang'); -- values절에서 함수 사용

⑧ INSERT INTO CUSTOMER4(PWD,NAME) VALUES('ok','kim'); -- Implicit null ??

⑨ UPDATE CUSTOMER SET ID = NULL; -- NOT NULL

⑩ UPDATE CUSTOMER SET ID = 'XMAN'; -- UNIQUENESS

⑪ SELECT INDEX_NAME, INDEX_TYPE, UNIQUENESS FROM USER_INDEXES
WHERE TABLE_NAME = 'CUSTOMER3';

⑫ SELECT INDEX_NAME, COLUMN_POSITION, COLUMN_NAME FROM USER_IND_COLUMNS

WHERE TABLE_NAME = 'CUSTOMER3' ORDER BY INDEX_NAME,COLUMN_POSITION;

[요구]

- ㉔ 컬럼 조합으로 구성되는 PK를 가지는 테이블을 설계(자신만의)한후 SQL Script를 작성 하십시오

FOREIGN KEY : 테이블간(테이블내)의 참조 무결성(REFERENTIAL INTEGRITY)을 보장

SQL> ED T_CONS5.SQL

```
DROP TABLE 부서;

CREATE TABLE 부서( 부서ID      VARCHAR2(2)  CONSTRAINT DEPARTMENT_부서ID_PK PRIMARY KEY,
                    부서명      VARCHAR2(10)                                -- 부서명
);

DROP TABLE EMPLOYEE;
CREATE TABLE EMPLOYEE( EMPID      VARCHAR2(8),                                -- 사원 고유 ID
                        부서ID      VARCHAR2(2),                                -- 사원 근무 부서 ID
                        CONSTRAINT EMPLOYEE_부서_부서ID_FK FOREIGN KEY(부서ID)
                        REFERENCES 부서(부서ID)                                -- TABLE LEVEL 제약사항
);

DESC 부서;
DESC EMPLOYEE;
```

// 테이블 생성시 발생하는 에러를 수정 하십시오

// 사원 정보 입력

① INSERT INTO EMPLOYEE VALUES('XMAN','10');

-- 부모가 없는 자식 데이터

// 부서 정보 입력

② INSERT INTO 부서(부서ID,부서명) VALUES('10','관리실');

-- 부서ID VARCHAR2

③ INSERT INTO 부서(부서ID,부서명) VALUES('20','전산실');

// 사원 정보 입력

④ INSERT INTO EMPLOYEE VALUES('XMAN','10');

⑤ INSERT INTO EMPLOYEE VALUES('XMAN','XX');

-- 존재하지 않는 유형 부서에서 근무 ?

⑥ DELETE FROM 부서 WHERE 부서ID = '10';

-- 10번부서에서 근무하는 사원들은?

⑦ DROP TABLE 부서;

⑧ DROP TABLE EMPLOYEE;

[요구]

⑥ 부서테이블을 삭제하는 2가지 방법의 예제를 SQL Script로 작성 하십시오

SQL> ED T_CONS5_2.SQL

```

DROP TABLE DEPARTMENT;
DROP TABLE EMPLOYEE;

CREATE TABLE DEPARTMENT (DEPTNO VARCHAR2(2) CONSTRAINT DEPARTMENT_DEPTNO_PK PRIMARY KEY,
                           DNAME VARCHAR2(10) CONSTRAINT DEPARTMENT_DNAME_NN NOT NULL
);

CREATE TABLE EMPLOYEE (사번 VARCHAR2(8) PRIMARY KEY,
                        이름 VARCHAR2(10),
                        DEPTNO VARCHAR2(2) NOT NULL,
                        CONSTRAINT EMPLOYEE_DEPARTMENT_DEPTNO_FK FOREIGN KEY(DEPTNO)
                        REFERENCES DEPARTMENT(DEPTNO)
);

```

[요구] 테이블 생성시 발생하는 에러를 수정하고 식별자의 최대 길이를 조사해서 설명 하십시오

① `SELECT TABLE_NAME, CONSTRAINT_NAME, CONSTRAINT_TYPE`
`FROM DBA_CONSTRAINTS WHERE TABLE_NAME IN ('DEPARTMENT', 'EMPLOYEE');`

② 부서 데이터가 존재 하지 않는 상황에서 사원 정보를 입력시 참조 무결성 에러 발생
 SQL 오류: ORA-02291: 무결성 제약조건(SCOTT.사원_부서_부서번호_FK)이 위배되었습니다
 - 부모 키가 없습니다

`INSERT INTO 사원(사번,이름,부서번호) VALUES('XMAN', 'TUNER', '10');`

③ 부서 정보 입력

`INSERT INTO DEPARTMENT(DEPTNO,DNAME) VALUES('10','관리팀');`
`INSERT INTO DEPARTMENT(DEPTNO,DNAME) VALUES('20','전산팀');`
`INSERT INTO DEPARTMENT(DEPTNO,DNAME) VALUES('30','영업팀');`

④ 10번,20번 DEPARTMENT에 근무하는 사원

`INSERT INTO EMPLOYEE(사번,이름,DEPTNO) VALUES('XMAN','TUNER',10);`
`INSERT INTO EMPLOYEE(사번,이름,DEPTNO) VALUES('YMAN','DBA',20);`

⑤ 30번 DEPARTMENT(존재하지 않는 DEPARTMENT) 에 근무하는 EMPLOYEE 입력시 참조 무결성 발생
 SQL 오류: ORA-02291: 무결성 제약조건(SCOTT.EMPLOYEE_DEPARTMENT_DEPTNO_FK)이 위배되었습니다
 - 부모 키가 없습니다

`INSERT INTO EMPLOYEE(사번,이름,DEPTNO) VALUES('ZMAN','DEVELOPER',30);`

⑥ 근무자가 없는 30번DEPARTMENT 폐지

```
DELETE FROM DEPARTMENT WHERE DEPTNO = 30;
```

⑦ 근무자가 있는 10번DEPARTMENT 폐지시 참조 무결성으로 에러가 발생한다.

SQL 오류: ORA-02292: 무결성 제약조건(SCOTT.EMPLOYEE_DEPARTMENT_DEPTNO_FK)이 위배되었습니다

- 자식 레코드가 발견되었습니다

```
DELETE FROM DEPARTMENT WHERE DEPTNO = 10;
```

⑧ 근무자가 있는 10번DEPARTMENT 폐지 방법은?

해당 DEPARTMENT 근무자를 퇴사(DELETE)처리 하거나 다른 DEPARTMENT로 이관(UPDATE)하고 폐지한다.

```
UPDATE EMPLOYEE SET DEPTNO=20 WHERE DEPTNO=10;
```

```
DELETE FROM DEPARTMENT WHERE DEPTNO = 10;
```