

제약조건 (Constraint) 이란?

제약조건이란 테이블에 부적절한 자료가 입력되는 것을 방지하기 위해서 여러 가지 규칙을 적용해 놓는거라 생각하면 된다. 간단하게 테이블안에서 데이터의 성격을 정의하는 것이 바로 제약조건 이다.

- 제약조건은 데이터의 무결성 유지를 위하여 사용자가 지정할 수 있는 성질 이다.
- 모든 제약조건은 데이터 사전(DICTIONARY)에 저장 된다.
- 의미있는 이름을 부여했다면 CONSTRAINT를 쉽게 참조할 수 있다.
- 표준 객체 명명법을 따르는 것이 좋다.
- 제약조건은 테이블을 생성할 당시에 지정할 수도 있고, 테이블 생성 후 구조변경(ALTER)명령어를 통해서도 추가가 가능하다.
- NOT NULL제약조건은 반드시 컬럼 레벨에서만 정의가 가능하다.

NOT NULL 조건

컬럼을 필수 필드화 시킬 때 사용 한다.

```
-- NOT NULL제약조건을 설정하면 ename 컬럼에는 꼭 데이터를 입력해야 한다.
-- emp_nn_ename을 제약조건 명으로 설정 하였다.
SQL> CREATE TABLE emp3 (
      ename VARCHAR2(20) CONSTRAINT emp_nn_ename NOT NULL );

-- 제약조건은 USER_CONSTRAINTS 뷰를 통해서 확인 할 수 있다.
SQL> SELECT CONSTRAINT_NAME
      FROM   USER_CONSTRAINTS
      WHERE  TABLE_NAME = 'EMP3' ;

CONSTRAINT_NAME
-----
emp_nn_ename
```

UNIQUE 조건

데이터의 유일성을 보장(중복되는 데이터가 존재할수 없다) 되고, 자동으로 인덱스가 생성 된다.

```
-- deptno 컬럼에 UNIQUE 제약조건 생성
SQL> ALTER TABLE emp2
      ADD CONSTRAINT emp2_uk_deptno UNIQUE (deptno);

-- 제약 조건의 삭제
SQL> ALTER TABLE emp2
      DROP CONSTRAINT emp2_uk_deptno;
```

CHECK 조건

컬럼의 값을 어떤 특정 범위로 제한할 수 있다.

```
-- comm 컬럼에 1에서 100까지의 값만을 가질수 있는 체크조건 생성
SQL> ALTER TABLE emp2
      ADD CONSTRAINT emp2_ck_comm
      CHECK (comm >= 1 AND comm <= 100);

-- 제약 조건의 삭제
SQL> ALTER TABLE emp2
      DROP CONSTRAINT emp2_ck_comm;

-- 10000,20000,30000,40000,50000의 값만을 가질수 있는 체크조건 생성
SQL> ALTER TABLE emp2
      ADD CONSTRAINT emp2_ck_comm
      CHECK comm IN (10000,20000,30000,40000,50000);
```

DEFAULT(컬럼 기본값) 지정

데이터를 입력 하지 않아도 지정된 값이 기본으로 입력 된다.

```
-- hiredate 컬럼에 값을 입력하지 않아도 오늘 날짜가 입력된다.
SQL> CREATE TABLE emp4 (
      ... (컬럼생략) ...,
      hiredate DATE DEFAULT SYSDATE );
```

PRIMARY KEY 지정

- 기본키는 UNIQUE 와 NOT NULL의 결합과 같다.
- 기본키는 그 데이터 행을 대표하는 컬럼으로서의 역할을 수행하여 다른 테이블에서 외래키들이 참조할 수 있는 키로서의 자격을 가지고 있다. 이를 참조 무결성이라 한다.
- UNIQUE 조건과 마찬가지로 기본키를 정의하면 자동으로 인덱스를 생성하며, 그 이름은 기본 키 제약 조건의 이름과 같다.
- INDEX: 검색 키로서 검색 속도를 향상 시킨다.(UNIQUE,PRIMARY KEY 생성시 자동적으로 생긴다.)

```
-- PRIMARY KEY 생성 예제
SQL> CREATE TABLE emp5 (
      empno NUMBER CONSTRAINT emp5_pk_empno PRIMARY KEY );
```

```
-- ALTER TABLE 명령어로 PRIMARY KEY 생성 예제
ALTER TABLE emp2
ADD CONSTRAINT emp2_pk_empno PRIMARY KEY (empno) ;
```

FOREIGN KEY(외래 키)지정

- 기본키를 참조하는 컬럼 또는 컬럼들의 집합이다.
- 외래키를 가지는 컬럼의 데이터 형은 외래키가 참조하는 기본키의 컬럼과 데이터형과 일치해야 한다. 이를 어기면 참조무결성 제약에 의해 테이블을 생성할수 없다.
- 외래키에 의해 참조되고 있는 기본 키는 삭제 할 수 없다.
- **ON DELETE CASCADE** 연산자와 함께 정의된 외래키의 데이터는 그 기본키가 삭제 될 때 같이 삭제 된다.

```
-- emp 테이블의 deptno 컬럼이 dept 테이블에 deptno 컬럼을 참조하는 외래키 생성
SQL> ALTER TABLE emp2 ADD CONSTRAINT emp2_fk_deptno
      FOREIGN KEY (deptno) REFERENCES dept(deptno);
```

제약 조건의 확인

- **USER_CONS_COLUMNS** : 컬럼에 할당된 제약 조건 조회.
- **USER_CONSTRAINTS** : 유저가 소유한 모든 제약 조건 조회.

```
SQL> SELECT SUBSTR(A.COLUMN_NAME,1,15) COLUMN_NAME,
      DECODE(B.CONSTRAINT_TYPE,
        'P','PRIMARY KEY',
        'U','UNIQUE KEY',
        'C','CHECK OR NOT NULL',
        'R','FOREIGN KEY') CONSTRAINT_TYPE,
      A.CONSTRAINT_NAME
  FROM   USER_CONS_COLUMNS A, USER_CONSTRAINTS B
 WHERE  A.TABLE_NAME = UPPER('&table_name')
        AND A.TABLE_NAME = B.TABLE_NAME
        AND A.CONSTRAINT_NAME = B.CONSTRAINT_NAME
 ORDER BY 1;
```

-- 테이블 명을 입력 하면 됩니다.

table_name의 값을 입력하십시오: emp2

COLUMN_NAME	CONSTRAINT_TYPE	CONSTRAINT_NAME
-----	-----	-----
DEPTNO	CHECK OR NOT NULL	SYS_C001362
	FOREIGN KEY	EMP2_FK_DEPTNO
EMPNO	PRIMARY KEY	EMP2_PK_EMPNO
ENAME	CHECK OR NOT NULL	EMP2_NN_ENAME
MGR	UNIQUE KEY	EMP2_UP_MGR