

3. 테이블의 생성과 수정 그리고 삭제(2)[테이블 제약조건]

3.2 테이블의 제약조건

제약조건 (Constraint)

제약조건이란 **테이블에 부적절한 자료가 입력되는 것을 방지**하기 위해서 여러 가지 규칙을 적용해 놓는거라 생각하면 됩니다. 간단하게 테이블안에서 데이터의 성격을 정의하는 것이 바로 제약조건 입니다.

- 데이터의 무결성 유지를 위하여 사용자가 지정할 수 있는 성질 입니다.
- 모든 CONSTRAINT는 데이터 사전(DICTIONARY)에 저장 됩니다.
- 의미있는 이름을 부여했다면 CONSTRAINT를 쉽게 참조할 수 있습니다.
- 표준 객체 명명법을 따르는 것이 좋습니다.
- 제약조건은 테이블을 생성할 당시에 지정할 수도 있고,
테이블 생성 후 구조변경(ALTER) 명령어를 통해서도 추가가 가능합니다.
- NOT NULL 제약조건은 반드시 컬럼 레벨에서만 정의가 가능합니다.

****NOT NULL 조건** : 컬럼을 필수 필드화 시킬 때 사용합니다.

```
SQL> CREATE TABLE emp(  
    ename VARCHAR2(20) NOT NULL );
```

이런식으로 하면 ename 컬럼에는 꼭 데이터를 입력해야만 합니다.

여기서 emp_nn_ename은 (테이블이름_제약조건이름_컬럼이름) 형식으로 CONSTRAINT NAME을 정의 합니다.

CONSTRAINT NAME은 USER_CONSTRAINTS 뷰(VIEW)를 통해서 확인할 수 있습니다.

***** 제약조건 확인하기 *****

```
SQL> SELECT * FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS  
WHERE TABLE_NAME='MEMBER';
```

	CONSTRAINT_CATALOG	CONSTRAINT_SCHEMA	CONSTRAINT_NAME	TABLE_SCHEMA	TABLE_NAME	CONSTRAINT_TYPE	ENFORCED
▶	def	mydb	PRIMARY	mydb	member	PRIMARY KEY	YES

**** UNIQUE 조건 ****

데이터의 유일성을 보장(중복되는 데이터가 존재할 수 없습니다.)

자동으로 index가 생성 됩니다.

```
SQL> ALTER TABLE DEPT ADD CONSTRAINT DEPT_DEPTNO UNIQUE(DEPTNO);
```

테이블이 변경 되었습니다.

이런식으로 하면 deptno 컬럼에 중복된 데이터가 들어갈 수 없습니다.

-- 제약조건의 삭제

```
SQL> ALTER TABLE emp  
      DROP CONSTRAINT DEPT_DEPTNO;
```

테이블이 변경 되었습니다.

**** CHECK 조건 ****

컬럼의 값을 어떤 특정 범위로 제한 할 수 있습니다.

```
SQL> ALTER TABLE emp  
      ADD CONSTRAINT emp_ck_sal  
      CHECK (SAL >= 50 AND SAL<= 10000 );
```

테이블이 변경 되었습니다.

comm컬럼은 체크조건에서 제한을 하고 있으므로 1에서 100까지의 값만을 가질 수 있습니다.

또 체크 조건에서는 IN 연산자를 사용할 수 있습니다.

```
SQL> ALTER TABLE emp  
      ADD CONSTRAINT emp_ck_comm  
      CHECK ( comm IN (10000,20000,30000,40000,50000));
```

테이블이 변경 되었습니다.

comm 컬럼은 10000,20000,30000,40000,50000의 값만을 가질 수 있습니다.

**** DEFAULT 조건 ****

컬럼 기본값 지정 : 데이터 입력시에 입력을 하지 않아도 지정된 값이 입력될수 있습니다.

```
SQL> CREATE TABLE emp(  
    hiredate DATE DEFAULT NOW() );
```

이런식으로 하면 hiredate 컬럼에 INSERT를 하지 않아도 오늘 날짜가 들어갑니다.

**** FOREIGN KEY 지정 ****

기본키를 참조하는 컬럼 또는 컬럼들의 집합입니다.

※ 외래키를 가지는 컬럼의 데이터 형은 외래키가 참조하는 기본키의 컬럼과 데이터형이 일치해야 합니다. 이를 어기면 참조무결성 제약에의해 테이블을 생성할 수 없습니다.

※ 외래키에 의해 참조되고 있는 기본키는 삭제할 수 없다.

※ ON DELETE CASCADE 연산자와 함께 정의된 외래키의 데이터는 그 기본키가 삭제 될 때 같이 삭제 됩니다.

```
SQL> ALTER TABLE emp ADD CONSTRAINT emp_fk_deptno  
    FOREIGN KEY (deptno) REFERENCES dept(deptno) ON DELETE CASCADE;
```

테이블이 변경되었습니다.

ON DELETE CASCADE

: 다른 테이블의 기존 행에 있는 외래 키에서 참조하는 키가 포함된 행을 삭제하려고 하면 해당 외래 키가 포함되어 있는 모든 행도 삭제되도록 지정

**** 외래키 삭제하기**

```
SQL> ALTER TABLE [테이블명] DROP FOREIGN KEY [제약조건이름];
```