테이블 컬럼의 관리

테이블의 컬럼은 ADD, MODIFY, DROP연산자를 통해서 관리 할 수 있습니다.

ADD 연산자

테이블에 새로운 컬럼을 추가 할 때 사용 한다.

```
-- VARCHAR2의 데이터 형을 가지는 addr 컬럼을 emp 테이블에 추가
SQL> ALTER TABLE emp ADD (addr VARCHAR2(50));
```

MODIFY 연산자

테이블의 컬럼을 수정 하거나 NOT NULL컬럼으로 변경 할 수 있으며, 컬럼이 이미 테이터를 가지고 있을 경우 다른 데이터형으로 변경이 불가능 하다.

```
-- ename 컬럼을 VARCHAR2 50자리로 수정한 예제.
SQL> ALTER TABLE emp MODIFY (ename VARCHAR2(50));
SQL> ALTER TABLE emp MODIFY (ename VARCHAR2(50) NOT NULL);
```

DROP 연산자

테이블 컬럼을 삭제 하거나, 테이블의 제약 조건을 삭제 할 때 사용 한다.

```
-- 컬럼의 삭제 문법
SQL> ALTER TABLE table_name DROP COLUMN column_name

-- 제약 조건의 삭제 예제
SQL> ALTER TABLE emp DROP PRIMARY KEY ;

-- CASCADE 연산자와 함께 사용하면 외래키에 의해 참조되는 기본키도 삭제 할 수 있다.
SQL> ALTER TABLE emp DROP CONSTRAINT emp_pk_empno CASCADE;
```

기존 테이블의 복사

- 기존 테이블을 부분, 또는 완전히 복사할 때에 서브쿼리를 가진 CREATE TABLE 명령어를 사용해서 쉽게 복사 할 수 있다.

- 하지만 제약 조건, 트리거, 그리고 테이블 권한은 새로운 테이블로 복사되지 않는다.

- 제약조건은 NOT NULL 제약 조건만 복사 된다.


```
-- 한번 실습해 보세요.
SQL> CREATE TABLE emp2
     AS SELECT * FROM emp;
```

테이블의 테이블스페이스 변경

Oracle 8i부터는 ALTER TABLE ~ MOVE TABLESPACE 명령어로 쉽게 테이블의 테이블스페이스를 변경할 수 있다.


```
-- 한번 실습해 보세요. (test라는 테이블스페이스가 있어야 겠죠)
SQL> ALTER TABLE emp
     MOVE TABLESPACE test;
```

테이블의 TRUNCATE

- 테이블을 Truncate하면 테이블의 모든 행이 삭제되고 사용된 공간이 해제 된다.
- TRUNCATE TABLE은 DDL명령이므로 롤백 데이터가 생성되지 않는다.
- DELETE명령으로 데이터를 지우면 롤백명령어로 복구 할 수 있지만, TRUNCATE로 데이터를 삭제하면 롤백을 할 수가 없다.
- 행당 인덱스도 같이 잘려 나간다.
- 외래키가 참조중인 테이블은 TRUNCATE할 수 없다.
- TRUNCATE 명령을 사용하면 삭제 트리거가 실행되지 않는다.

테이블의 삭제 (DROP TABLE)

```
-- emp 테이블 삭제
SQL> DROP TABLE emp;

-- CASCADE CONSTRAINT는 외래키에 의해 참조되는 기본키를 포함한 테이블일 경우
-- 기본키를 참조하던  외래 키 조건도 같이 삭제 한다.
SQL> DROP TABLE emp CASCADE CONSTRAINT;
```