Chapter 8

테이블 생성(CREATE TABLE)

테이블 구조 변경(ALTER TABLE)

테이블 삭제(DROP TABLE)

테이블 데이터 비우기(TRUNCATE)

오라클 데이터베이스 객체 이름을 지정하는 표준 규칙에 따라서 데이터베이스 테이블과 열의 이름을 정합니다. 다음은 이름 만드는 규칙입니다.

- 테이블 이름과 열 이름은 문자로 시작해야 하며 1-30 문자 길이를 가질 수 있습니다.
- 이름은 오직 A-Z, a-z, 0-9, _ (underscore), $ 그리고 # (유효한 문자이지만 권장되지는 않습니다.) 문자만을 포함해야 합니다.
- 이름은 오라클 서버 사용자에 의해 소유된 다른 객체의 이름과 중복되어서는 안 됩니다.
- 이름은 오라클 예약어이어서는 안 됩니다.

이름을 만들 때에는 테이블이나 다른 데이터베이스 객체에 대한 서술적인 이름을 사용합니다. 다른 테이블에도 일관되게 똑같은 엔티티 이름을 지정합니다. 예를 들면, EMPLOYEES 테이블과 DEPARTMENTS 테이블에서 부서 번호 열은 DEPARTMENT_ID라고 명명합니다.

이름은 대소문자를 구분하지 않습니다. 예를 들면, EMPLOYEES는 employees와 똑같이 취급 합니다.

```
CREATE  TABLE [schema.]table_name (
      column  data_type [DEFAULT expr],
      ...
);
```

구문형식에서:

- *schema* : 소유자의 이름과 똑같습니다.
- *table_name* : 테이블의 이름입니다.
- DEFAULT *expr*: INSERT 문장에서 값을 생략할 경우에 입력될 디폴트 값을 명시합니다.
- *column*: 열의 이름입니다.
- *data_type* : 열의 데이터타입과 길이입니다.

테이블 생성!

다음 구문은 DEPTNO, DNAME 그리고 LOC 이라는 세 개의 열을 생성합니다.

```
SQL> CREATE TABLE dept (
  2      deptno  NUMBER(2),
  3      dname   VARCHAR2(14),
  4      loc     VARCHAR2(13) );

Table DEPT이(가) 생성되었습니다.
```

데이터 형	설명
VARCHAR2(size)	가변 길이 문자 데이터입니다. (최대 크기는 명시해야 하며, 최소 크기는 1, 최대 크기는 4000 바이트입니다.)
CHAR(size)	size Byte 길이의 고정 길이 문자 데이터입니다. (디폴트이며 최소 크기는 1, 최대 크기는 2000바이트입니다.)
NUMBER(p, s)	전체 p자리와 소수점 이하 s자리를 가지는 숫자(소수점은 자리수에서 제외); 정밀도는 십진 자리수의 최대 개수이며, 스케일은 소수점 오른쪽의 자리수입니다. (정밀도는 1에서 38까지의 범위이며 스케일은 -84에서 127까지의 범위일 수 있습니다.)
DATE	January 1, 4712 B.C. 와 December 31, 9999 A.D.사이의 날짜와 시간 값입니다.
LONG	2 기가바이트까지의 가변 길이 문자 데이터입니다.
CLOB	4 기가바이트까지의 단일 바이트 문자 데이터입니다.
RAW(size)	size 길이의 원시 이진 데이터입니다. 최대 크기는 2000 입니다. (최대 크기는 명시해야 합니다.)
LONG RAW	2 기가바이트까지의 가변 길이 원시 이진 데이터입니다.
BLOB	4 기가바이트까지의 이진 데이터입니다.
BFILE	4 기가바이트까지의 외부 파일에 저장된 이진 데이터입니다.

사본 데이터를 생성할 때 한다고 했었죠?

CTAS(Create Table As Select) 구문은 현재 있는 테이블과 같은 구조를 갖는 테이블을 생성할 수 있도록 합니다. CTAS 구문을 이용한 테이블 복제 시 NOT NULL 제약조건을 제외한 다른 제약조건을 복사하지 않습니다.

```
CREATE TABLE table AS SELECT statement
```

구문에서...

- AS SELECT *statement* : 테이블이 만들어질 서브쿼리를 입력합니다. SELECT 문장의 WHERE 절이 항상 FALSE일 경우 데이터는 포함하지 않고 구조만 갖는 테이블을 생성합니다. 서브쿼리이지만 괄호(( ))를 포함하지 않습니다. SELECT 문장에서 조회하는 열에 수식이 사용될 경우 열별칭을 사용해야 합니다. 열 별칭은 생성되는 테이블의 열 이름으로 사용됩니다.

다음 구문은 SELECT한 결과대로 테이블의 구조를 생성하고 데이터도 저장합니다.

```
SQL> CREATE TABLE emp2 AS SELECT * FROM employees;

Table EMP2이(가) 생성되었습니다.

SQL> SELECT COUNT(*) FROM emp2;

  COUNT(*)
----------
       107
```

Chapter 8

테이블 생성(CREATE TABLE)
테이블 구조 변경(ALTER TABLE)
테이블 삭제(DROP TABLE)
테이블 데이터 비우기(TRUNCATE)

ALTER TABLE 문장에 ADD 절을 사용하여 테이블에 열을 추가할 수 있습니다. 열을 추가하거나 수정할 수 있지만, 열이 어디에 나타날지를 명시할 수 없습니다. 새로운 열은 마지막 열이 됩니다. 새로운 열 추가 시 제약사항과 함께 추가할 수는 없습니다.

```
ALTER TABLE table_name
ADD         (column  data_type  [DEFALT  expr]
           [, column  data_type] … );
```

ALTER TABLE 문장에 MODIFY 절을 사용하여 테이블에 있는 기존의 열을 변경할 수 있습니다.

```
ALTER TABLE table_name
MODIFY      (column  data_type  [DEFALT  expr]
           [, column  data_type] … );
```

열 추가

- ADD 절을 사용하여 열을 추가합니다.
- 새로운 열이 마지막 열이 됩니다.

열을 추가하거나 수정할 수 있지만, 열이 어디에 나타날지를 명시할 수 없습니다. 새로운 열은 마지막 열이 됩니다. 새로운 열 추가 시 제약사항과 함께 추가할 수는 없습니다.

```
ALTER TABLE table_name
ADD         (colume datatype);
```

열 추가

다음 구문은 ADD 절을 이용하여 새로운 열을 추가합니다.

```
SQL> ALTER TABLE emp_dept50
  2  ADD (job VARCHAR2(10));

Table EMP_DEPT50이(가) 변경되었습니다.
```

다음 구문은 새로운 열이 추가된 것을 확인합니다.

```
SQL> SELECT * FROM emp_dept50;
```

	EMPLOYEE_ID	FIRST_NAME	ANN_SAL	HIRE_DATE	JOB
1	120	Matthew	96000	04/07/18	(null)
2	121	Adam	98400	05/04/10	(null)
3	122	Payam	94800	03/05/01	(null)
4	123	Shanta	78000	05/10/10	(null)

열을 추가할 때 테이블이 이미 어떤 행을 포함하고 있다면, 새로운 열은 이미 존재하는 모든 행에 대해서 Null로 초기화합니다.

열 이름 변경

- RENAME COLUMN 절을 사용하여 테이블의 열 이름을 변경할 수 있습니다.

컬럼 이름의 변경은 오라클 9i(Release 2)이후 가능해 졌으며 ALTER TABLE 문에서 RENAME COLUMN 절을 사용합니다.

```
ALTER TABLE table_name
RENAME COLUMN old_name TO new_name;
```

열 이름 변경

다음 구문은 RENAME COLUMN 절을 이용하여 열 이름을 변경합니다.

```
SQL> ALTER TABLE emp_dept50
  2  RENAME COLUMN job TO job_id;

Table EMP_DEPT50이(가) 변경되었습니다.
```

다음 구문은 테이블의 구조를 확인합니다.

```
SQL> DESC emp_dept50;

이름          널         유형
----------- -------- ------------
EMPLOYEE_ID           NUMBER(6)
FIRST_NAME            VARCHAR2(20)
ANN_SAL               NUMBER
HIRE_DATE   NOT NULL DATE
JOB_ID                VARCHAR2(10)
```

열 수정

- MODIFY 절을 사용하여 열을 수정합니다.
- 기존의 데이터를 손상되게 크기를 조정할 수는 없습니다.

```
ALTER TABLE   table
MODIFY        (column datatype);;
```

열 수정

열을 수정할 때 데이터 타입의 크기는 가능한 범위내에서 얼마든지 늘릴 수 있습니다. 그러나 열의 크기를 축소할 경우 이미 저장되어 있는 데이터를 손상시키도록 축소할 수 없습니다.

다음 구문은 first_name 열의 크기를 변경합니다.

```
SQL> ALTER TABLE emp_dept50
  2  MODIFY (first_name VARCHAR2(30));

Table EMP_DEPT50이(가) 변경되었습니다.
```

열의 크기를 줄일 수 있습니다.

```
SQL> ALTER TABLE emp_dept50
  2  MODIFY (first_name VARCHAR2(10));

Table EMP_DEPT50이(가) 변경되었습니다.
```

열 삭제

- ALTER TABLE 문을 DROP COLUMN 절과 함께 사용하여 테이블에서 열을 삭제할 수 있습니다.
- 이 기능은 Oracle8i부터 사용할 수 있습니다.

```
ALTER TABLE table
DROP COLUMN column;
```

열 삭제는...

- 삭제를 하려는 열은 데이터를 포함하거나 포함하지 않을 수 있습니다.
- 한 번에 하나의 열만 삭제할 수 있습니다.
- 테이블 변경 후 테이블에 열이 하나 이상 있어야 합니다.
- 삭제된 열은 복구할 수 없습니다.

열 삭제

다음 구문은 JOB_ID 열을 삭제합니다.

```
SQL> ALTER TABLE emp_dept50
  2  DROP COLUMN job_id;

Table EMP_DEPT50이(가) 변경되었습니다.
```

테이블의 구조를 확인해 보면 JOB_ID 열이 삭제된 것을 확인할 수 있습니다.

```
SQL> DESC emp_dept50;

이름          널         유형
----------- -------- ------------
EMPLOYEE_ID            NUMBER(6)
FIRST_NAME             VARCHAR2(10)
ANN_SAL                NUMBER
HIRE_DATE    NOT NULL DATE
```

Chapter 8

테이블 생성(CREATE TABLE)
테이블 구조 변경(ALTER TABLE)
테이블 삭제(DROP TABLE)
테이블 데이터 비우기(TRUNCATE)

테이블 삭제

- 테이블의 모든 데이터와 구조가 삭제됩니다.
- 어떤 결정되지 않은 트랜잭션이 커밋됩니다.
- 모든 인덱스가 삭제됩니다.
- 이 문장은 롤백할 수 없습니다.

DROP TABLE 문장은 오라클 테이블의 정의를 삭제합니다. 테이블을 삭제할 때 데이터베이스는 테이블에 있는 모든 데이터와 그와 연관된 모든 인덱스를 상실합니다. CASCADE CONSTRAINTS 절은 참조 제약조건이 있는 열을 포함하더라도 테이블을 삭제합니다.

```
DROP TABLE table_name [CASCADE CONSTRAINTS]
```

테이블 삭제

일단 실행된 DROP TABLE 문장은 복구할 수 없습니다. 오라클 서버는 DROP TABLE 문장을 실행할 때 삭제에 대한 질문을 하지 않습니다. 여러분이 해당 테이블을 소유하거나 high-level 권한을 가지고 있다면, 테이블은 즉시 삭제될 것입니다. 모든 DDL 문장이 커밋될 것이므로 트랜잭션을 영구적으로 만듭니다.

다음 구문은 테이블을 삭제합니다.

```
SQL> DROP TABLE employees_dept50;

Table EMPLOYEES_DEPT50이(가) 삭제되었습니다.
```

이후 DESC 문장으로 테이블의 구조를 보려할 때 해당 명령은 실패합니다.

```
SQL> DESC employees_dept50;

ERROR:
------------------------------------
오류: EMPLOYEES_DEPT50 객체가 존재하지 않습니다.
```

Chapter 8

테이블 생성(CREATE TABLE)
테이블 구조 변경(ALTER TABLE)
테이블 삭제(DROP TABLE)
테이블 데이터 비우기(TRUNCATE)

위험하기 때문에 많이 사용되지 않습니다

- TRUNCATE TABLE 문장으로 테이블의 모든 행을 삭제합니다.
- 해당 테이블에 사용된 기억공간을 해제 합니다.
- TRUNCATE를 사용하여 삭제한 행을 롤백 할 수 없습니다.
- TRUNCATE 대안적으로, DELETE 문장을 사용하여 행을 삭제합니다.

DDL 문장은 TRUNCATE TABLE 문장입니다. TRUNCATE TABLE 문장을 사용할 때 삭제한 행을 롤백 할 수 없습니다.

```
TRUNCATE TABLE table_name;
```

여러분은 테이블의 소유자이거나 테이블을 삭제할 수 있는 DELETE ANY TABLE 시스템 권한을 가져야 합니다. DELETE 문장은 테이블의 모든 행을 삭제할 수 있지만, 저장 공간을 해제할 수는 없습니다. 그러나 TRUNCATE 문장은 테이블의 모든 행을 제거하고 저장 공간을 해제합니다.

다음 구문은 테이블 데이터 전체를 삭제합니다.

```
SQL> TRUNCATE TABLE emp2;

Table EMP2이(가) 잘렸습니다.
```

TRUNCATE 구문은 데이터를 삭제하지만 테이블이 구조는 남아 있습니다. 다음 구문을 실행했을 때 아무것도 출력되지는 않지만 에러는 발생하지 않습니다.

```
SQL> SELECT * FROM emp2;
```