

테이블 생성

CREATE TABLE table_name 을 이용하여 테이블을 생성할 수 있습니다.

테이블을 생성할 때 명령을 실행하는 USER가 테이블의 소유자가 됩니다.

테이블 생성 Syntax

```
1 CREATE TABLE table_name ( [  
2     column_name1 data_type [ COLLATE collation ] [ column_constraint [ ... ] ]  
3     column_name2 data_type [ COLLATE collation ] [ column_constraint [ ... ] ]  
4     column_name3 data_type [ COLLATE collation ] [ column_constraint [ ... ] ]  
5     ...  
6 ] )
```

테이블에 추가될 컬럼 이름을 column_name에 넣고 컬럼의 데이터 타입을 data_type에 넣습니다. 그리고 column_constraint에 생성한 컬럼의 제약 조건을 넣습니다.

다음은 테이블 생성에 대한 몇 가지 예제입니다.

Create table로 departments 테이블을 생성하고 department_id와 name컬럼을 생성합니다. Department_id의 데이터 타입은 integer이고 이 테이블의 기본키가 됩니다. Name의 데이터 타입은 varchar(50)입니다.

```
1 CREATE TABLE departments(  
2     Department_id integer PRIMARY KEY,  
3     Name varchar(50)  
4 );
```

아래 예제에서는 CREATE TABLE로 films테이블을 생성합니다. 테이블의 컬럼으로는 code, title, did, date_prod, kind, len등이 있습니다.

films 테이블에는 제약 조건으로 code 컬럼은 NULL값을 갖지 않는 NOT NULL조건과 production 테이블에서 date_prod 컬럼이 중복 없이 유일한 UNIQUE조건을 만족해야 합니다.

```
1 CREATE TABLE films (  
2     code char(5) NOT NULL,  
3     title varchar(40),  
4     did integer,  
5     date_prod date,  
6     kind varchar(10),  
7     len interval hour to minute,  
8     CONSTRAINT production UNIQUE(date_prod)  
9 );
```

테이블 변경

테이블을 변경하는 것은 ALTER TABLE table_name 을 이용하여 테이블을 변경할 수 있습니다.

테이블 변경 Syntax

```
1 ALTER TABLE table_name
2 [ADD|DROP|ALTER|SET|RENAME]
3 column_name data_type [ COLLATE collation ] [ column_constraint [ ... ] ]
```

[ADD|DROP|ALTER|SET|RENAME] 는 테이블을 수정할 때 변경할 내용을 선택합니다.

컬럼을 추가할 때는 ADD, 컬럼을 삭제할 때는 DROP, 존재하는 컬럼의 데이터 타입을 변경할 때는 ALTER, 테이블을 다른 TABLESPACE나 SCHEMA로 이동시킬 때는 SET, 테이블, 컬럼, 제약조건의 이름을 변경할 때는 RENAME을 사용합니다.

테이블에 변경할 컬럼 이름을 column_name에 넣고 컬럼의 데이터 타입을 data_type에 넣습니다. 그리고 column_constraint에 변경할 컬럼의 제약 조건을 넣습니다.

다음은 테이블 변경에 대한 몇 가지 예제입니다.

Distributors 테이블에 데이터타입이 varchar(50)인 address 컬럼을 추가합니다.

```
1 ALTER TABLE distributors
2 ADD COLUMN address varchar(50);
```

Distributors 테이블에 address 컬럼의 이름을 city로 변경합니다.

```
1 ALTER TABLE distributors
2 RENAME COLUMN address TO city;
```

Distributors 테이블에서 zipchk 제약조건을 삭제합니다.

```
1 ALTER TABLE distributors
2 DROP CONSTRAINT zipchk;
```

Distributors 테이블의 tablespace를 fasttablespace tablespace로 이동합니다.

```
1 ALTER TABLE distributors
2 SET TABLESPACE fasttablespace;
```

Distributors 테이블에서 address컬럼의 데이터 타입을 varchar(80)으로 변경하고 name컬럼의 데이터 타입을 varchar(100)으로 변경합니다.

```
1 ALTER TABLE distributors
2 ALTER COLUMN address TYPE varchar(80),
3 ALTER COLUMN name TYPE varchar(100);
```

테이블 삭제

테이블을 삭제하는 것은 DROP TABLE table_name 을 이용하여 테이블을 변경할 수 있습니다.

테이블 삭제 Syntax

```
1 | DROP TABLE [ IF EXISTS ] table_name [, ...] [ CASCADE | RESTRICT ]
```

테이블을 삭제할 때는 테이블을 소유하고 있는 USER나 SUPERUSER만이 테이블을 삭제할 수 있습니다. 테이블전체를 삭제하는 것이 아니라 테이블 내의 행들만 삭제할 때는 DELETE나 TRUNCATE를 사용합니다.

먼저 삭제할 테이블이름을 table_name에 넣습니다. 제약조건으로 CASCADE를 사용할 경우 이 테이블과 연관된 view나 개체들도 함께 삭제합니다. RESTRICT를 사용할 경우 다른 개체가 종속되어 있을 경우 테이블삭제를 제한합니다.

다음은 Films 테이블과 distributors 테이블을 삭제 예제입니다.

```
1 | DROP TABLE films, distributors;
```