

SCHEMA 구조

SCHEMA는 OBJECT들의 논리적인 집합을 말합니다. SCHEMA는 TABLE, VIEW, SEQUENCE, SYNONYM, DOMIAN, FUNCTION등의 OBJECT들로 구성되어 있습니다.

SCHEMA를 사용하는 이유는 논리적 집합체를 만들어서 관리의 편의성을 높이고 여러 USER들 간의 간섭 없이 접속할 수 있게 합니다.

SCHEMA 조회

현재 존재하고 있는 SCHEMA를 조회하기 위해서는 pgAdmin의 Object Browser에서 검색하거나 psql에서 \dn 명령어로 조회할 수 있습니다.

```
1 | \dn
```

CREATE SCHEMA

```
1 | 1. CREATE SCHEMA schema_name [ AUTHORIZATION user_name ] [ schema_element [ ..
2 |
3 | 2. CREATE SCHEMA AUTHORIZATION user_name [3. schema_element [ ... ] ]
4 |
5 | 4. CREATE SCHEMA IF NOT EXISTS schema_name [ AUTHORIZATION user_name ]
6 |
7 | 5. CREATE SCHEMA IF NOT EXISTS AUTHORIZATION user_name
```

1. CREATE SCHEMA 'schemaname'을 입력하여 SCHEMA를 생성합니다. 이름을 입력하지 않을 경우 USER의 이름이 SCHEMA이름으로 사용됩니다. (단 Pg_로 시작하는 이름은 스키마로 불가능 합니다.)

2. AUTHORIZATION 'username' 스키마를 소유한 USER의 이름을 입력합니다. 이를 생략할 경우 접속되어 있던 USER가 default값으로 저장되고 SUPERUSER만이 다른 USER가 소유한 SCHEMA를 만들 수 있습니다.

3. schema_element [...]을 입력하여 SCHEMA 내에서 객체를 정의하는 SQL문을 작성합니다. CREATE TABLE, CREATE VIEW, CREATE INDEX, CREATE SEQUENCE, CREATE TRIGGER, GRANT등이 포함될 수 있습니다.

4. CREATE SCHEMA IF NOT EXISTS 'schema_name' 을 입력하면 특정 이름이 SCHEMA에 없을 경우 그 SCHEMA를 생성합니다.

5. CREATE SCHEMA IF NOT EXISTS AUTHORIZATION 'user_name' 은 USER가 소유한 SCHEMA가 존재하지 않을 때 SCHEMA를 생성합니다.

SCHEMA 생성 예제

```
1 | CREATE SCHEMA TEST_S;
```

[그림] SCHEMA 생성 결과 화면

```
File Edit View Search Terminal Help

postgres=# CREATE SCHEMA TEST_S;
CREATE SCHEMA
postgres=# \dn
List of schemas
Name | Owner
-----+-----
public | postgres
test_s | postgres
(2 rows)

postgres=#
```

USER를 지정해준 SCHEMA 생성 예제

```
1 | CREATE SCHEMA TEST_S2 AUTHORIZATION TEST1;
```

[그림] SCHEMA 생성 결과 화면

```
daou@
File Edit View Search Terminal Help

postgres=# CREATE SCHEMA TEST_S2 AUTHORIZATION TEST1;
CREATE SCHEMA
postgres=# \dn
List of schemas
Name | Owner
-----+-----
public | postgres
test_s | postgres
test_s2 | test1
(3 rows)

postgres=#
```

TABLE과 VIEW를 추가한 SCHEMA 생성 예제

```
1 | CREATE SCHEMA TEST_S3
2 | CREATE TABLE comtecopseq(table_name varchar(20));
```

[그림] SCHEMA 생성 결과 pgAdmin3 화면

