

▶ Chapter 06: PostgreSQL 사용자 관리

SQL 활용 프로그래밍

SQL Application Programming





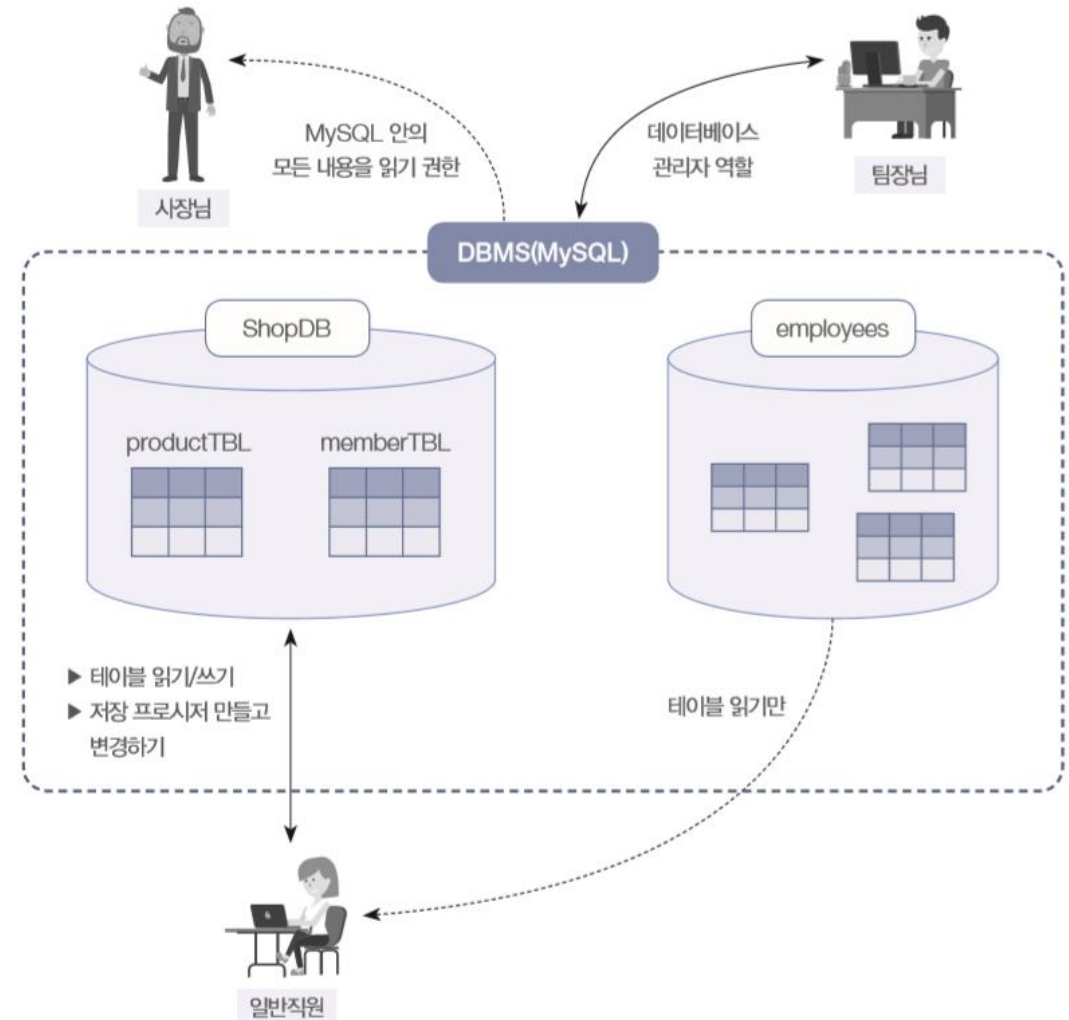
CHAPTER 06 PostgreSQL 사용자 관리

PostgreSQL의 사용자 관리에 대해 알아본다.

SECTION 01 사용자 관리하기

DB 사용자 관리의 필요성

- 실무에서의 DB 관리
 - 데이터베이스를 혼자 사용하는 것이 아닌 다양한 사용자나 응용프로그램에서 접속해 사용함
 - 모든 사용자에게 모든 데이터베이스 권한을 준다면 고의나 실수로 인해 데이터가 유출, 유실되는 사고가 생길 수 있음
 - 이를 방지하기 위해 별도의 역할을 만들고, 역할마다 모든 권한이 아닌 적당한 권한을 부여해 관리



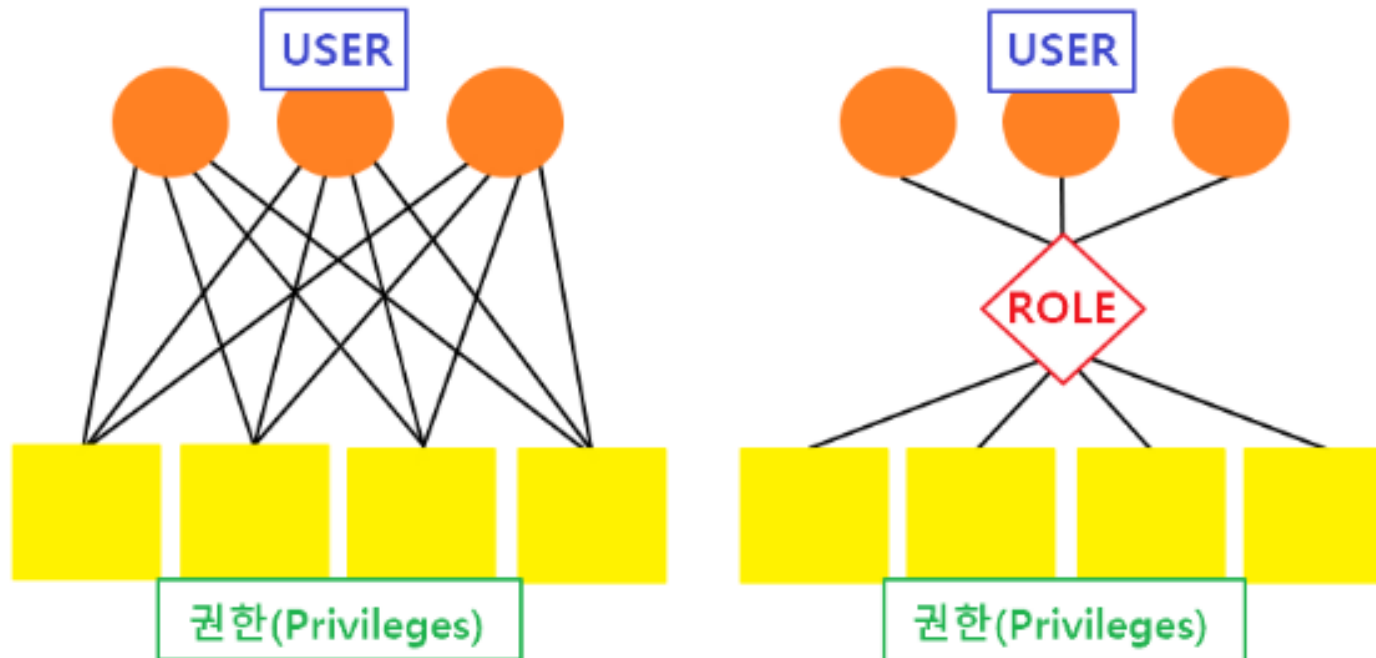
[그림 5-57] 사용자에게 부여된 다양한 권한

SECTION 01 사용자 관리하기

DB 사용자 관리의 필요성

실무에서의 DB 관리

- 사용자 (user): DB를 사용하는 계정.
따로 생성하지 않으면 기본 사용자인 postgres가 있음
- 역할 (role): 데이터베이스 권한들을 모아서 관리할 수 있도록 도와 줌.
한 명의 user를 role으로 간주하거나 여러 명의 user를 묶어 role으로 간주할 수 있음



SECTION 01 사용자 관리하기

사용자 생성

- **CREATE USER** 사용자 이름 **WITH** 옵션 **PASSWORD** 암호;

사용자 제거

- **DROP USER** 사용자 이름;

사용자 조회

- **SELECT * FROM PG_USER;**

OPTION	DEFAULT	OPTION
SUPERUSER	NOSUPERUSER	해당 USER에게 SUPERUSER권한을 줌
CREATEDB	NOCREATEDB	DATABASE를 생성하는 권한을 정의
CREATEUSER	NOCREATEUSER	스스로 새로운 유저를 생성하는 권한을 정의
INHERIT	NOINHERIT	DATABASE의 권한을 다른 구성원들에게 상속하는 역할
LOGIN	NOLOGIN	USER가 LOGIN하는 권한을 부여
CONNECTION LIMIT connlimit	-1	로그인 시 동시연결을 지원하는 기능
[ENCRYPTED] PASSWORD 'password'	[UNCRYPTED '']	'password'를 입력하고 인증이 필요 없는 경우 옵션 생략이 가능

SECTION 01 사용자 관리하기

SUPERUSER 권한을 가지고 암호가 1234인 user1과,
DB생성 권한을 가지고 암호가 0000인 user2 생성 해보기

- `create user user1 with SUPERUSER password '1234';`
`create user user2 with CREATEDB password '0000';`

사용자 조회해보기

- `select * from pg_user;`

	username name	usesysid oid	usecreatedb boolean	usesuper boolean	userepl boolean	usebypassrls boolean	passwd text	valuntil timestamp with time zone	useconfig text[]
1	postgres	10	true	true	true	true	*****	[null]	[null]
2	user1	16651	false	true	false	false	*****	[null]	[null]
3	user2	16652	true	false	false	false	*****	[null]	[null]

SECTION 01 사용자 관리하기

역할 생성

- **CREATE ROLE** 역할 이름 **WITH** 옵션;

역할 제거

- **DROP ROLE**

역할 조회

- **SELECT * FROM PG_ROLES;**

대부분의 권한을 가짐

DB를 생성할 수 있는 권한

역할을 생성할 수 있는 권한

```
CREATE ROLE name [ [ WITH ] option [ ... ] ]
```

option:

```
SUPERUSER | NOSUPERUSER  
| CREATEDB | NOCREATEDB  
| CREATEROLE | NOCREATEROLE  
| INHERIT | NOINHERIT  
| LOGIN | NOLOGIN  
| REPLICATION | NOREPLICATION  
| BYPASSRLS | NOBYPASSRLS  
| CONNECTION LIMIT connlimit  
| [ ENCRYPTED ] PASSWORD 'password'  
| VALID UNTIL 'timestamp'  
| IN ROLE role_name [, ...]  
| IN GROUP role_name [, ...]  
| ROLE role_name [, ...]  
| ADMIN role_name [, ...]  
| USER role_name [, ...]  
| SYSID uid
```

SECTION 01 사용자 관리하기

- SUPERUSER 권한을 가지는 director 역할과, DB 생성 권한을 가지는 staff 역할 만들어보기
 - `create role director with superuser;`
`create role staff with createdb;`
- 역할 조회해보기
 - `select * from pg_roles;`

	rolname name	rolsuper boolean	rolinherit boolean	rolcreatorole boolean	rolcreatedb boolean	rolcanlogin boolean	rolreplication boolean	rolconnlimit integer
12	postgres	true	true	true	true	true	true	
13	director	true	true	false	false	false	false	
14	staff	false	true	false	true	false	false	
15	user1	true	true	false	false	true	false	
16	user2	false	true	false	true	true	false	

SECTION 01 사용자 관리하기

권한 추가

- **GRANT** 권한 **ON** 대상 **TO** 사용자 이름/역할 이름;
 - 권한: SELECT, INSERT, DELETE 등.
ALL PRIVILEGES를 넣으면 모든 권한이 부여 됨
 - 대상: 테이블, 스키마 등

권한 제거

- **REVOKE** 권한 **ON** 대상 **FROM** 사용자 이름/역할 이름;
 - 권한: SELECT, INSERT, DELETE 등
ALL PRIVILEGES를 넣으면 모든 권한이 제거 됨
 - 대상: 테이블, 스키마 등

SECTION 01 사용자 관리하기

- Staff 역할에 select, insert 권한 추가해보기
 - `grant select, insert on employees to staff;`
- Director 역할에 모든 권한 추가해보기
 - `grant all privileges on employees to director;`
- Staff 역할에서 모든 권한 제거해보기
 - `revoke all privileges on employees from staff;`
- Staff 역할 제거해보기
 - `drop role staff;`