

# SQL - DCL

## [1] DCL (데이터 제어어)

- 데이터의 보안, 무결성, 회복, 병행 제어
- 데이터베이스 관리자(DBA)
- 종류

명령어	기능
COMMIT	실제 물리적 디스크로 저장, 정상적으로 완료
ROLLBACK	비정상적으로 종료, 원래의 상태로 복구
GRANT	사용 권한을 부여
REVOKE	사용 권한을 취소

## [2] GRANT / REVOKE

- 데이터베이스 관리자가 데이터베이스 사용자에게 권한을 부여하거나 취소
- GRANT : 권한 부여
- REVOKE : 권한 취소

```
GRANT 사용자등급 TO 사용자_ID_리스트 [IDENTIFIED BY 암호];
REVOKE 사용자등급 FROM 사용자_ID_리스트;
```

예제1 사용자 ID가 "NABI"인 사람에게 데이터베이스 및 테이블을 생성할 수 있는 권한을 부여하는 SQL문을 작성하시오.

```
GRANT RESOURCE TO NABI;
```

- RESOURCE : 데이터베이스 및 테이블 생성 가능자

예제2 사용자 ID가 "STAR"인 사람에게 단순히 데이터베이스 에 있는 정보를 검색할 수 있는 권한을 부여하는 SQL문을 작성하시오.

```
GRANT CONNECT TO STAR;
```

- CONNECT : 단순 사용자

### 테이블 및 속성에 대한 권한 부여 및 취소

```
GRANT 권한_리스트 ON 개체 TO 사용자 [WITH GRANT OPTION];
REVOKE [GRANT OPTION FOR] 권한_리스트 ON 개체 FROM 사용자[CASCADE];
```

- 권한 종류 : ALL, SELECT, INSERT, DELETE, UPDATE 등
- WITH GRANT OPTION : 부여받은 권한을 다른 사용자에게 다시 부여할 수 있는 권한을 부여함
- GRANT OPTION FOR : 다른 사용자에게 권한을 부여할 수 있는 권한을 취소함
- CASCADE : 권한 취소 시 권한을 부여받았던 사용자가 다른 사용자에게 부여한 권한도 연쇄적으로 취소함

**예제3** 사용자 ID가 "NABI"인 사람에게 <고객> 테이블에 대한 모든 권한과 다른 사람에게 권한을 부여할 수 있는 권한까지 부여하는 SQL문을 작성하시오.

```
GRANT ALL ON 고객 TO NABI WITH GRANT OPTION;
```

**예제4** 사용자 ID가 "STAR"인 사람에게 부여한 <고객> 테이블에 대한 권한 중 UPDATE 권한을 다른 사람에게 부여할 수 있는 권한만 취소하는 SQL문을 작성하시오.

```
REVOKE GRANT OPTION FOR UPDATE ON 고객 FROM STAR;
```

### [3] COMMIT

- 트랜잭션 처리가 정상적으로 완료된 후 트랜잭션이 수행한 내용을 데이터베이스에 반영하는 명령어이다.
- Auto Commit : 성공적으로 완료되면 자동으로 COMMIT되고, 실패하면 자동으로 ROLLBACK

### [4] ROLLBACK

- 변경되었으나 아직 COMMIT되지 않은 모든 내용들을 취소하고 데이터베이스를 이전 상태로 되돌리는 명령어
- 일부 변경된 내용만 데이터베이스에 반영되는 비일관성 상태가 될 수 있기 때문에 일부분만 완료된 트랜잭션을 롤백되어야 한다

### [5] SAVEPOINT

- 트랜잭션 내에 ROLLBACK 할 위치인 저장점을 지정하는 명령어
- 저장점을 지정할 때는 이름을 부여
- ROLLBACK 할 때 지정된 저장점까지의 트랜잭션 처리 내용이 모두 취소

<사원>

사원번호	이름	부서
10	김기희	기획부
20	박인사	인사부
30	최재무	재무부

사원번호	이름	부서
40	오영업	영업부

예제1 <사원> 테이블에서 '사원번호'가 40인 사원의 정보를 삭제한 후 COMMIT을 수행하시오.

```
DELETE FROM 사원 WHERE 사원번호 = 40;
COMMIT;
```

해설 DELETE 명령을 수행한 후 COMMIT 명령을 수행했으므로 DELETE 명령으로 삭제된 레코드는 이후 ROLLBACK 명령으로 되돌릴 수 없다.

<사원>

사원번호	이름	부서
10	김기획	기획부
20	박인사	인사부
30	최재무	재무부

예제2 '사원번호'가 30인 사원의 정보를 삭제하시오.

```
DELETE FROM 사원 WHERE 사원번호 = 30;
```

해설 DELETE 명령을 수행한 후 COMMIT 명령을 수행하지 않았으므로 DELETE 명령으로 삭제된 레코드는 이후 ROLLBACK 명령으로 되돌릴 수 있다.

<사원>

사원번호	이름	부서
10	김기획	기획부
20	박인사	인사부

예제3 SAVEPOINT 'S1'을 설정하고 '사원번호'가 20인 사원의 정보를 삭제하시오.

```
SAVEPOINT S1;
DELETE FROM 사원 WHERE 사원번호 = 20;
```

<사원>

사원번호	이름	부서
------	----	----

사원번호	이름	부서
10	김기희	기획부

예제4 SAVEPOINT 'S2'을 설정하고 '사원번호'가 10인 사원의 정보를 삭제하시오.

```
SAVEPOINT S2;  
DELETE FROM 사원 WHERE 사원번호 = 20;
```

<사원>

사원번호	이름	부서
------	----	----

예제5 SAVEPOINT 'S2'까지 ROLLBACK을 수행하시오.

```
ROLLBACK TO S2;
```

해설 ROLLBACK이 적용되는 시점을 'S2'로 지정했기 때문에 예제5의 ROLLBACK에 의해 <사원>테이블의 상태는 예제4의 작업을 수행하기 전으로 되돌려진다.

<사원>

사원번호	이름	부서
10	김기희	기획부

예제6 SAVEPOINT 'S1'까지 ROLLBACK을 수행하시오.

```
ROLLBACK TO S1;
```

해설 ROLLBACK이 적용되는 시점을 'S1'로 지정했기 때문에 예제6의 ROLLBACK에 의해 <사원>테이블의 상태는 예제3의 작업을 수행하기 전으로 되돌려진다.

<사원>

사원번호	이름	부서
10	김기희	기획부
20	박인사	인사부

예제7 SAVEPOINT 없이 ROLLBACK을 수행하시오.

```
ROLLBACK;
```

해설 '직원번호'가 40인 직원의 정보를 삭제한 후 COMMIT을 수행했으므로 예제7의 ROLLBACK이 적용되는 시점은 예제1의 COMMIT 이후 새롭게 작업이 수행되는 예제2의 작업부터이다.

<직원>

직원번호	이름	부서
10	김기혁	기획부
20	박인사	인사부
30	최재무	재무부