

#Section9

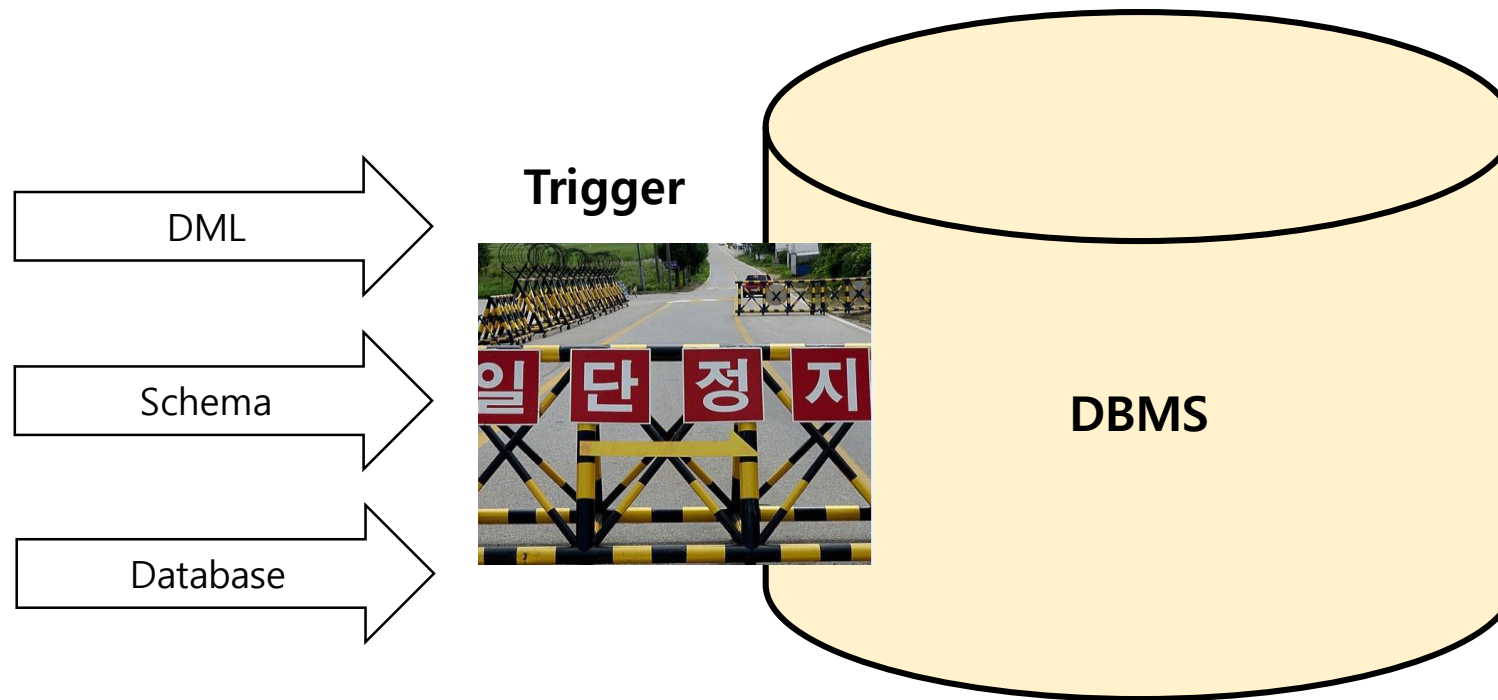
Triggers (DBMS 검문소)

Trigger 개요

A trigger is like a stored procedure that Oracle Database invokes automatically whenever a specified event occurs.

트리거는 지정된 이벤트가 발생할 때마다 오라클 데이터베이스가 호출하는 저장 프로시저와 같다.

- DB 운영자 입장에서 App. DB 개발자로부터 DB를 관리하는 DB 관리의 마지막 보루
- DML 뿐 아니라 DDL Level 까지 통제가 가능



DML Trigger

CREATE OR REPLACE TRIGGER 트리거이름

[필수] BEFORE or AFTER -----> 트리거의 발생 시점, 보통 Before 를 많이 사용

[필수] DELETE OR INSERT OR UPDATE ON 테이블명-----> 3개를 선택해서 사용 (INSERT/UPDATE/DELETE)

[선택] REFERENCING NEW AS NEW OLD AS OLD -----> NEW , OLD 에 대한 alias 사용 가능,

[선택] FOR EACH ROW [statement] -----> 생략될경우 statement 실행

-----> statement : 해당 이벤트에 한번만 실행됨

[선택] WHEN (new.column = '~~') -----> 트리거의 조건을 생성

DECLARE --- PL/SQL 구문

n number;

BEGIN

-- NEW : 새로 추가되거나 변경된 후의 값에 트리거가 적용된다.(INSERT : 입력할 값, UPDATE : 수정할 값)

-- OLD : 변경 전의 값에 트리거가 적용된다. (UPDATE : 수정 전 값, DELETE : 삭제할 값)

IF INSERTING THEN ~~ ; -----> insert 이벤트 실행

ELSIF UPDATING THEN ~~ : -----> update 이벤트 실행

ELSIF DELETING THEN ~~ : -----> delete 이벤트 실행

END IF;

END;

Compound DML Trigger

A compound DML trigger created on a table or editioning view can fire at multiple timing points.

복합 트리거는 여러 타이밍 지점에서 실행 가능

[Sample]

```
Create or Replace TRIGGER TG_CPD_CST_INFO
FOR DELETE OR INSERT OR UPDATE ON CST_INFO----> 3개를 선택해서 사용 (INSERT/UPDATE/DELETE)
COMPOUND TRIGGER
  -- 변수 선언
  n number;
  v_birth cst_info.birth%type;

  AFTER EACH ROW IS
  BEGIN

    IF INSERTING THEN DBMS_OUTPUT.put_line('TG_DEPARTMENTS_BEFORE - INSERT');
    ELSIF UPDATING THEN DBMS_OUTPUT.put_line('TG_DEPARTMENTS_BEFORE - UPDATE');
    END IF;

  End AFTER EACH ROW;

  AFTER STATEMENT IS
  BEGIN

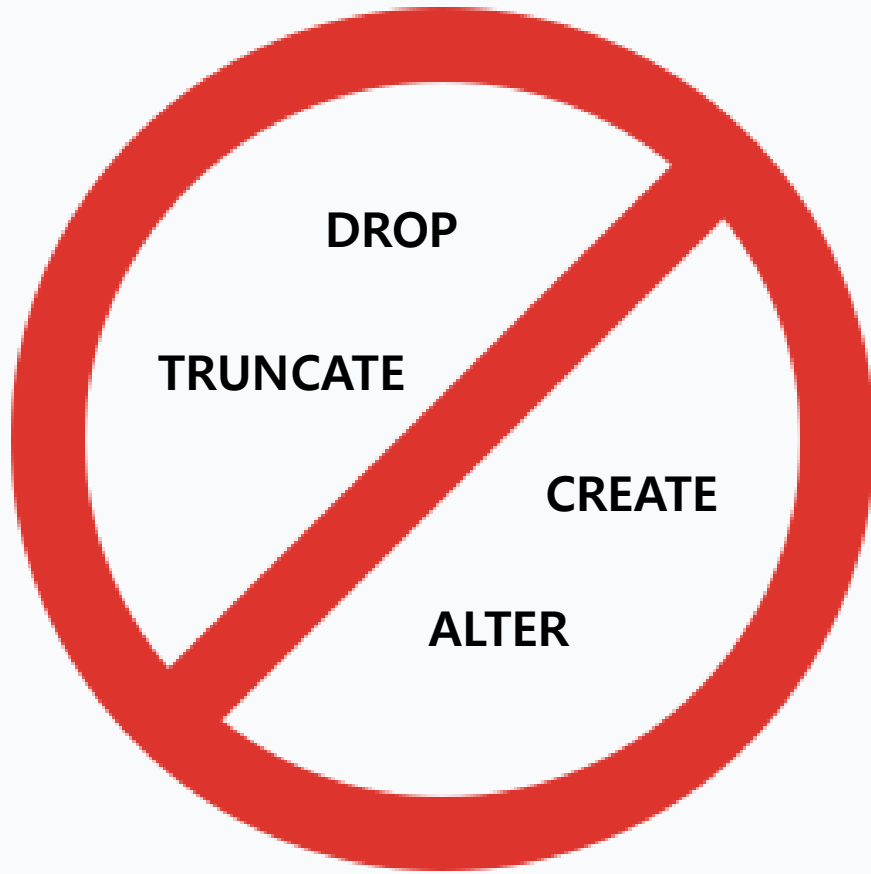
    select count(*)
    into n
    from cst_info;

    DBMS_OUTPUT.put_line('n -> '||n);
    IF DELETING THEN Raise_Application_Error(-20001, 'Dont delete this table'); End IF;

  END AFTER STATEMENT;

END TG_CPD_CST_INFO;
```

DDL Trigger



[Sample]

```
CREATE OR REPLACE TRIGGER TG_DDL
```

```
BEFORE
```

```
DROP OR TRUNCATE ON USER_TMP.SCHEMA
```

```
BEGIN
```

```
    dbms_output.put_line('ora_dict_obj_name >' || ora_dict_obj_name);
```

```
    dbms_output.put_line('ora_dict_obj_type >' || ora_dict_obj_type);
```

```
    IF ora_dict_obj_type in ('TABLE') Then
```

```
        Raise_Application_Error(-20001, 'Warning !! Warning !! Warning !!
```

```
        This command cannot be executed. !!');
```

```
    End If;
```

```
END;
```