

문장 레벨 트리거와 행 레벨 트리거

FOR EACH ROW 옵션 절의 사용 유무에 따라 결정

FOR EACH ROW 옵션 절을 사용시 : 행 레벨 트리거

문장 트리거

트리거링 사건에 대해 딱 한번만 실행

컬럼의 각 데이터 행을 제어할 수 없다

컬럼의 데이터 값에 관계없이 컬럼에 변화가 일어남을 감지하여 실행되는 트리거

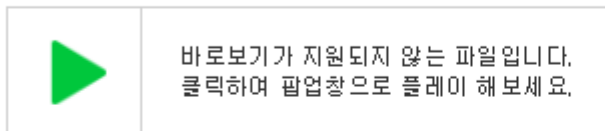


행 트리거

컬럼의 각 데이터 행이 변경될 때마다 실행

실제 그 데이터 행의 값을 제어할 수 있는 트리거

실제 값을 수정, 변경 또는



저장하기 위해 사용.

행 트리거에서 실제 데이터를 제어하기 위해 사용하는 문자

:old와 :new



SQL 문에 따른 사용 방법

- INSERT 문의 경우

입력할 데이터의 값이 ":new, 컬럼 이름"에 저장.

- UPDATE 문의 경우

변경 전의 데이터는 ":old, 컬럼 이름"에 저장되고, 새로운 데이터 값은 ":new, 컬럼 이름"에 저장.

- DELETE 문의 경우

삭제되는 컬럼 값이 ":old, 컬럼 이름"에 저장.

트리거링 시점

SALARY 테이블

EMP_NO	EMP_NAME	SAL	BONUS	
1001	SAM	1500000	500000	→ BEFORE 문장 트리거
1002	KIM	2500000	450000	→ BEFORE 행 트리거
1003	PARK	2800000	720000	→ AFTER 행 트리거
				→ AFTER 문장 트리거

실행결과

```
SQL> CREATE OR REPLACE TRIGGER ins_audit
2 BEFORE insert ON CLONE_EMP
3 BEGIN
4     IF(TO_CHAR(sysdate,'DY') IN ('FRI','SAT','SUN')) THEN
5         RAISE_APPLICATION_ERROR(-20001,'금요일부터 일요일까지는
6                                     입력작업을 할 수 없습니다.');
```

트리거가 생성되었습니다.

```
SQL> INSERT INTO CLONE_EMP VALUES(0001,'황희',45000);
```

1 개의 행이 만들어졌습니다.

```
SQL> SELECT * FROM CLONE_EMP;
```

EMPNO	ENAME	SAL
1	황희	45000
7566	JONES	2975
7698	BLAKE	2850
7839	KING	5000
7788	SCOTT	3000
7902	FORD	3000

6 개의 행이 선택되었습니다.

```
SQL> SELECT user_id, TO_CHAR(log_date,'YY/MM/DD HH24:MI:SS'),
2         behavior,old_values,new_values
3 FROM audit_user;
```

USER_ID	TO_CHAR(LOG_DATE, BEHAVIOR
SCOTT	08/05/07 23:44:20 INSERT