

## 트리거의 개념

- 트리거는 프로시저와 유사하지만 실행되는 원리가 다르다
- 프로시저는 작성 후 EXECUTE 명령어를 통해서 직접 실행하지만 트리거는 어떤 이벤트(주로 DML)가 발생했을 때 '내부적'으로 실행되는 저장된 프로시저이다.

## 트리거의 개념

- 트리거의 특징 : 프로시저의 형태이지만 단독으로 사용되기 보다 DML 구문에 붙여서 사용이 된다.
- 트리거의 종류
  - AFTER 트리거 : DML등의 작업이 일어났을 때 작동하는 트리거, 해당 작업 후에 작동한다.(테이블에만 작동)
  - BEFORE 트리거 : DML등의 작업이 일어났을 때 작동하는 트리거, 해당 작업 직전에 작동하다.

## 트리거의 개념

- 트리거를 사용하기에 앞서 실습용 테이블을 구성해봅니다.

```
CREATE TABLE dept_organ  
AS  
SELECT * FROM department WHERE 0=1;  
  
DROP TABLE dept_copy;  
  
CREATE TABLE dept_copy  
AS  
SELECT * FROM department WHERE 0=1;
```

## 트리거의 개념

- INSERT에 적용할 트리거를 생성하기

```
CREATE OR REPLACE TRIGGER trigger_sample1
  AFTER INSERT
  ON dept_ordin
  for each row
BEGIN
  IF inserting then
    dbms_output.put_line('Insert Trigger 발생');
    insert into dept_copy
      values (:new.dno, :new.dname, :new.loc);
  END IF;
END;
```

각 행마다 적용

## 트리거의 개념

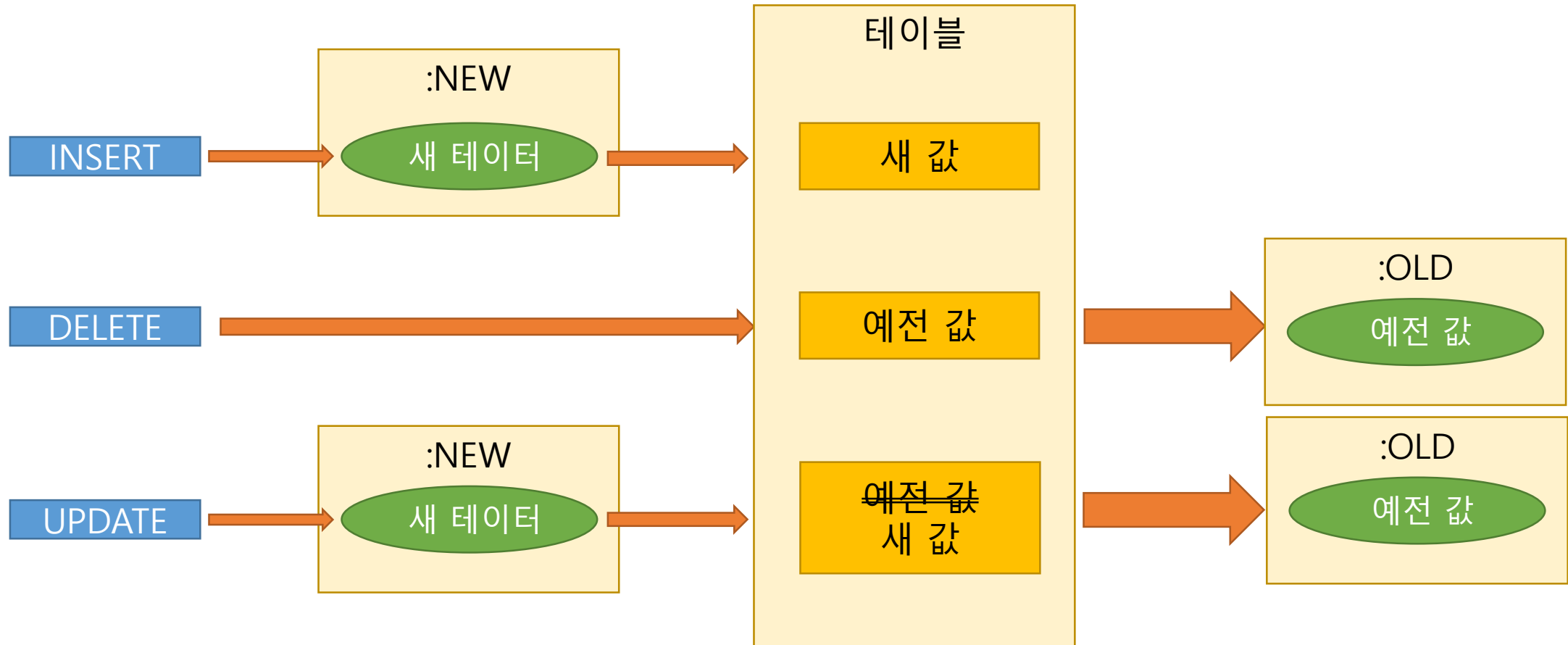
- INSERT를 통해서 데이터를 추가해 봅니다.

```
INSERT INTO dept_origin  
VALUES (10, 'ACCOUNTING', 'NEW YORK');
```

- origin테이블과 copy테이블 둘 다 데이터가 입력된 것을 확인할 수 있다

## 트리거의 개념

- 트리거에서 생성하는 임시 테이블 :NEW, :OLD



## 트리거의 개념

- UPDATE와 DELETE 트리거를 만들어 보자
  - 백업 테이블에 컬럼을 추가한다.

```
ALTER TABLE dept_copy  
ADD modType NCHAR(2);
```

- 트리거를 만든다.

```
CREATE OR REPLACE TRIGGER trigger_sample2  
AFTER UPDATE OR DELETE  
ON dept_organ  
for each row  
DECLARE  
    v_modType NCHAR(2);  
BEGIN  
    IF updating then  
        dbms_output.put_line('Updating Trigger 발생');  
        v_modType := '수정';  
    ELSIF deleting then  
        dbms_output.put_line('Deleting Trigger 발생');  
        v_modType := '삭제';  
    END IF;  
    INSERT INTO dept_copy  
    VALUES (:old.dno, :old.dname, :old.loc, v_modType);  
END;
```

## 트리거의 개념

- UPDATE를 통해 데이터를 변경하고 copy테이블을 확인해 봅니다.

```
UPDATE dept_ordin SET dno=30, dname='SALES', loc='CHICAGO'  
WHERE dno=10;
```

```
SELECT * FROM dept_ordin;  
SELECT * FROM dept_copy;
```

- DELETE를 통해 데이터를 삭제하고 copy 테이블을 확인해 봅니다.

```
DELETE FROM dept_ordin  
WHERE dno=30;
```

```
SELECT * FROM dept_ordin;  
SELECT * FROM dept_copy;
```