Unit VI – Database Storage Strategies

Chapter 11 –

SQL Triggers

Database Triggers

- A Trigger defines an action the database should take when some database related event occurs. A database trigger is a stored subprogram associated with a table. Triggers are executed when a specific data manipulation command are performed on specific tables.
- A trigger is an **event-condition-action (ECA)** rule.
  - ✓ When event occurs, test condition; if condition is satisfied, execute action

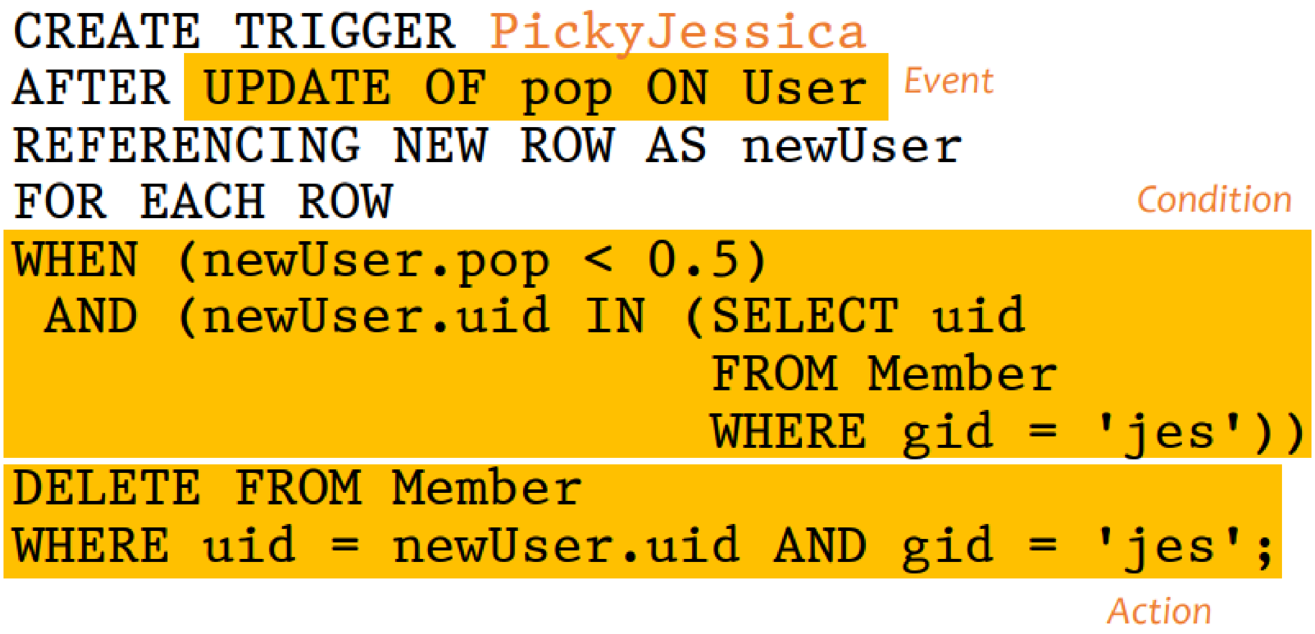
- Include the following:
  - ✓ event (e.g., an update/insert operation)
  - ✓ Condition
  - ✓ action (to be taken when the condition is satisfied)
  - The actions can be specified for:
    - ➢ informing the user the violation of the specified constraint
    - ➢ executing a specific store procedure or
    - ➢ triggering other update operations

- The action is to be executed automatically if the condition is satisfied when the event occurs

Example 1:

- Event: some user’s popularity is updated
- Condition: the user is a member of “Jessica’s Circle,” and pop drops below 0.5
- Action: kick that user out of Jessica’s Circle

Trigger example 1



```
CREATE TRIGGER PickyJessica
AFTER UPDATE OF pop ON User
REFERENCING NEW ROW AS newUser
FOR EACH ROW
WHEN (newUser.pop < 0.5)
 AND (newUser.uid IN (SELECT uid
                      FROM Member
                      WHERE gid = 'jes'))
DELETE FROM Member
WHERE uid = newUser.uid AND gid = 'jes';
```

Trigger options

- Possible events include:
  - INSERT ON table
  - DELETE ON table
  - UPDATE [OF column] ON table
- Granularity—trigger can be activated:

Row Level Triggers

- Row Level triggers execute once for each row in a transaction. Row level triggers are create using FOR EACH ROW clause in the create trigger command.

Statement Level Triggers

- Statement Level triggers execute once for each transaction.
- FOR EACH STATEMENT that performs modification
- For example, if you insert 100 rows in a single transaction then statement level trigger will be executed once.

BEFORE and AFTER Triggers

✓Timing—action can be executed:

- AFTER or BEFORE the triggering event

✓ Since triggers occur because of events, they may be set to occur immediately before or after those events.

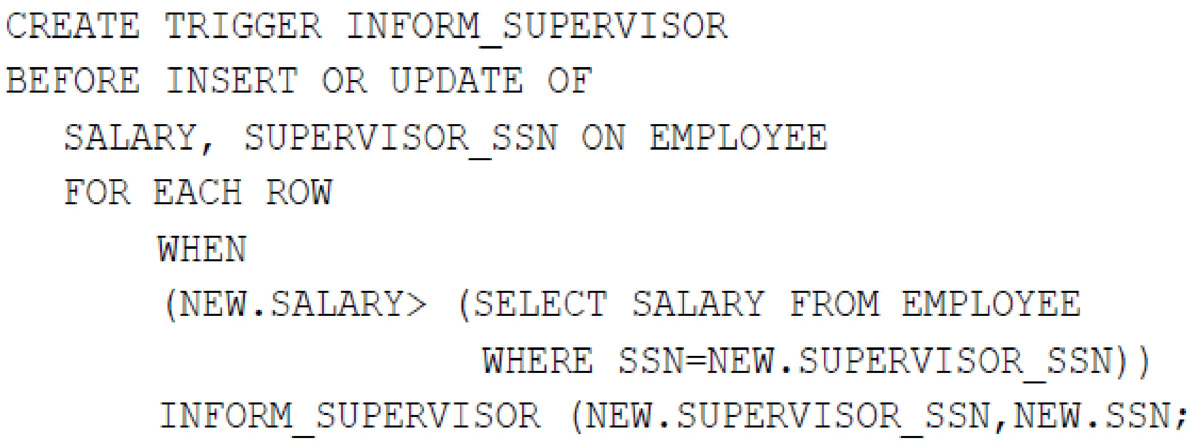
Transition variables

- OLD ROW: the modified row before the triggering event
- NEW ROW: the modified row after the triggering event

- OLD TABLE: a hypothetical read-only table containing all rows to be modified before the triggering event
- NEW TABLE: a hypothetical table containing all modified rows after the triggering event

SQL Triggers: Example 2

- A trigger to compare an employee’s salary to his/her supervisor during insert or update operations:



```
CREATE TRIGGER INFORM_SUPERVISOR
BEFORE INSERT OR UPDATE OF
   SALARY, SUPERVISOR_SSN ON EMPLOYEE
   FOR EACH ROW
       WHEN
       (NEW.SALARY> (SELECT SALARY FROM EMPLOYEE
                     WHERE SSN=NEW.SUPERVISOR_SSN))
       INFORM_SUPERVISOR (NEW.SUPERVISOR_SSN,NEW.SSN;
```