

TRIGGER

I trigger, come le asserzioni, sono innescati automaticamente a seguito di operazioni di aggiornamento. A differenza delle asserzioni, però, consentono di specificare le azioni da compiere sulla BD in corrispondenza di determinati eventi (**BD active**).

Un trigger (o regola) si basa sul modello ECA (event-condition, action):

- Evento: operazione di aggiornamento che attiva la regola
- Condizione: determina se l'azione debba essere eseguita; è facoltativa
- Azione: sequenza di istruzioni SQL da eseguire quando l'evento si verifica e la condizione (se esiste) è verificata

I trigger vengono usati essenzialmente per mantenere l'integrità dei dati, calcolare dati derivati o gestire eccezioni, ecc.

Schema Logico

- ALUNNO(**CF**, nome, cognome, data_nascita, sostegno)
- INSEGNANTE(**CF**, nome, cognome, data_nascita, di_ruolo, materia)
- CLASSE(**id**, anno, sezione, id_scuola*)
- AULA(**id**, mq, #max:stud, nome_fasciaEtà*, piano, id_plesso*)
- PLESSO(**id**, indirizzo, #piani, ascensore)
- FASCIAETÀ(**nome**, min, max)
- SCUOLA(**id**, nome, telefono, tipo, tempoPieno)
- INSEGNA(**id ins***, **id classe***, **AS**, ore)
- ASSEGNAMENTO(**id classe***, **id aula***, **AS**, numerosa)
- STORICOISCRIZIONE(**id alunno***, **id classe***, **AS**, data)

1. All'inserimento di una nuova tupla in "Storiciscrizione", se l'alunno iscritto è il 25° della propria classe, si imposti l'attributo *numerosa* della tabella "Assegnamento" a True per la tupla corrispondente alla classe **X** per l'anno scolastico **AS** se l'iscrizione inserita riguarda la classe **X** anno scolastico **AS**.

Operazione: Insert on Storiciscrizioni

Tupla inserita: (a1, c2, as1, d2) - NEW.id_classe=c2, NEW.AS=as1

CREATE TRIGGER CheckNumerosa **AFTER INSERT ON** Storiciscrizione
FOR EACH ROW

```
IF (25==(SELECT count(*)
        FROM Storiciscrizione AS S
        WHERE S.id_classe=NEW.id_classe and S.AS=NEW.AS))
THEN
    UPDATE Assegnamento
    SET numerosa=True
    WHERE id_classe=NEW.id_classe and AS=NEW.AS;
```

2. Alla rimozione di una tupla di "Storiciscrizione", se l'alunno iscritto era il 25° della propria classe, si imposti l'attributo *numerosa* della tabella "Assegnamento" a False per la tupla

corrispondente alla classe **X** per l'anno scolastico **AS** se l'iscrizione eliminata riguarda la classe **X** anno scolastico **AS**.

Operazione: Delete * From Storiciscrizioni where <cond>

Tupla rimossa: (a1, c2, as1, d2) - OLD.id_classe=c2, OLD.AS=as1

CREATE TRIGGER CheckNonNumerosa **AFTER DELETE ON** Storiciscrizione
FOR EACH ROW

IF (24==(SELECT count(*)

FROM Storiciscrizione **AS** S

WHERE S.id_classe=OLD.id_classe and S.AS=OLD.AS))

THEN

UPDATE Assegnamento

SET numerosa=FALSE

WHERE id_classe=OLD.id_classe and AS=OLD.AS;

3. All'aggiornamento di una tupla della tabella "Aula", se i nuovi *mq* sono inferiori ai vecchi, verificare che il nuovo #max:stud non sia superiore al vecchio

Vecchia tupla: (a1, 25, 15,) – OLD.mq=25, OLD.#max:stud=15

Nuova tupla: (a1, 20, 18,) – NEW.mq=20, NEW.#max:stud=18

CREATE TRIGGER CheckMq **BEFORE UPDATE ON** Aula
FOR EACH ROW

IF (NEW.mq<OLD.mq and NEW.#max:stud>OLD.#max:stud)

THEN RAISE ERROR;

4. Impedire l'iscrizione di un alunno ad una specifica classe se l'iscrizione comporta lo sfioramento del numero massimo di studenti ospitabili dall'aula associata alla classe stessa

Operazione: Insert on Storiciscrizioni

Tupla inserita: (a1, c2, as1, d2) - NEW.id_classe=c2, NEW.AS=as1

CREATE TRIGGER CheckCapienza **BEFORE INSERT ON** Storiciscrizione
FOR EACH ROW

IF ((SELECT #max:stud

FROM Assegnamento as X, Aula as Y

WHERE X.id_classe=NEW.id_classe and X.AS=NEW.AS

and X.id_aula=Y.id))==(SELECT count(*)

FROM Storiciscrizione **AS** S

WHERE S.AS=NEW.AS and

S.id_classe=NEW.id_classe))

THEN RAISE ERROR;

5. assegnare ad ogni nuovo alunno sostegno=false, a meno che non sia diversamente specificato

Operazione: Insert on Alunno

Tupla inserita: (a1, n2, c1, d2, NULL) - NEW.sostegno=NULL

```
CREATE TRIGGER Sostegno BEFORE INSERT ON Alunno  
FOR EACH ROW  
IF ( NEW.sostegno==NULL)  
THEN sostegno = false;
```

NOTA: L'aggiornamento della tupla (a1, n2, c1, d2, NULL) in (a1, n2, c1, d2, false) viene effettuato prima che essa sia inserita nel DB.

Alternativamente, si può adottare la seguente soluzione (meno efficiente)

```
CREATE TRIGGER Sostegno AFTER INSERT ON Alunno  
FOR EACH ROW  
IF ( NEW.sostegno==NULL)  
THEN  
    UPDATE Alunno  
    SET sostegno= false  
    WHERE CF=NEW.CF
```