

Leçon 5 : Les Triggers

AYIKPA KACOUTCHY JEAN : Enseignant -
Chercheur



Table des matières

I - 1- Généralité des triggers	3
II - Application 1 :	4
III - 2- Traitement des Triggers	5
IV - Application 2 :	7
Solutions des exercices	8

1- Généralité des triggers



■ Définition : 1.1- Définition

- *Un trigger est un programme qui se déclenche automatiquement suite à un événement, à la différence d'une procédure stockée, on ne peut pas appeler un trigger explicitement.*
- *En base de données, l'événement est une instruction du langage de manipulation de données qui modifie la base (INSERT, DELETE, UPDATE).*
- *Ces triggers font partie du schéma de la base, leur code compilé est conservé (comme pour les programmes stockés).*
- *Les triggers (ou déclencheurs) sont des objets de la base de données. Attachés à une table, ils vont déclencher l'exécution d'une instruction, ou d'un bloc d'instructions, lorsqu'une, ou plusieurs lignes sont insérées, supprimées ou modifiées dans la table à laquelle ils sont attachés.*

1.2- Intérêt des triggers

- Les triggers peuvent servir à vérifier des contraintes que l'on ne peut pas définir de façon déclarative
- Ils peuvent aussi gérer de la redondance d'information.
- Ils peuvent aussi servir à collecter des informations sur les mises à jour de la base.
- etc

Application 1 :



Exercice

[solution n°1 p.8]

Un Trigger permet :

- ☐ la modification automatique des données
- ☐ le déclenchement automatiquement d'un événement
- ☐ l'enregistrement automatique des données

2- Traitement des Triggers

III

2.1- Création d'un trigger

Syntaxe :

DELIMITER |

CREATE TRIGGER nom_trigger moment_trigger evenement_déclencheur

ON nom_table FOR EACH ROW

BEGIN

corps_trigger;

END|

CREATE TRIGGER : il s'agit de la commande à exécuter pour créer un Trigger.

moment_trigger : Lorsqu'un trigger est déclenché, ses instructions peuvent être exécutées à deux moments différents : soit juste avant que l'événement déclencheur n'ait lieu (*BEFORE*), soit juste après (*AFTER*).

evenement_déclencheur : Trois événements différents peuvent déclencher l'exécution des instructions d'un trigger en l'occurrence *INSERT*, *UPDATE*, *DELETE*.

ON nom_table : c'est là que l'on définit à quelle table le trigger est attaché.

FOR EACH ROW : signifie littéralement pour chaque ligne de la table rattachée au trigger.

corps_trigger : c'est le contenu du trigger. Comme pour les procédures stockées, il peut s'agir soit d'une seule instruction, soit d'un bloc d'instructions.

DELIMITER | : Un délimiteur est tout simplement (par défaut), le caractère ;. C'est-à-dire le caractère qui permet de délimiter les instructions, mais pour une procédure avec un bloc d'instructions il faut changer le délimiteur de la fin de procédure pour éviter une erreur. Dans notre cas nous utilisons le délimiteur .

Exemple :

Soit la table suivant :

Client(idcli,nomcli,prencli,sexecli,datenaiscli)

Créer un trigger sur la table client qui intervient après un enregistrement.

```
| DELIMITER |
```

```

2 CREATE TRIGGER Apres_insert_client AFTER INSERT
3 ON Client FOR EACH ROW
4 BEGIN
5 corps_trigger;
6 END |

```

2.2- OLD et NEW

OLD : désigne la ligne avant modification

NEW : désigne la ligne après modification.

Exemple :

Soit la table suivant :

Client(idcli,nomcli,prencli,nomjeunefille,sexecli,datenaiscli)

Le nom de jeune fille d'une client doit être NULL si le client à un sexe m

```

1 DELIMITER |
2 CREATE TRIGGER Avant_insert_client BEFORE INSERT
3 ON Client FOR EACH ROW
4 BEGIN
5 IF NEW.sexecli = 'm' THEN /**Vérifie si le sexe est m
6 SET NEW.nomjeunefille := NULL; /***** affecte NULL à nomjeunefille
7 END IF; /*** Fin de la structure IF
8 END |
9

```

2.4- Suppression d'une procédure stockée

Pour supprimer un trigger, on utilise DROP.

DROP TRIGGER nom_trigger;

Exemple :

Soit la table suivant :

Client(idcli,nomcli,prencli,sexecli,datenaiscli)

Supprimer la procédure Apres_insert_client

```

1 DROP TRIGGER Apres_insert_client;

```

Application 2 :

IV

Exercice

[solution n°2 p.8]

Énoncé :

Soit la table suivante :

LIVRE(NUMLIV,NOMAUT ,TITRELIV,GENRELIV,PRIXLIV)

- Créer un trigger before_update_livre qui permet d'augmenter de 10% le prix du livre si l'auteur est BERTIN.

NB : Respecter l'ordre de création des propriétés

Solution :

```

[ ]
[ ] before_update_livre [ ]
[ ] LIVRE [ ]
[ ]
IF [ ] THEN
[ ] = [ ] + ([ ] * 0.1);
END IF;
[ ]

```

Solutions des exercices



> Solution n°1

Exercice p. 4

Un Trigger permet :

- ☐ la modification automatique des données
- ☒ le déclenchement automatiquement d'un événement
- ☐ l'enregistrement automatique des données

> Solution n°2

Exercice p. 7

Énoncé :

Soit la table suivante :

LIVRE(NUMLIV,NOMAUT ,TITRELIV,GENRELIV,PRIXLIV)

- Créer un trigger before_update_livre qui permet d'augmenter de 10% le prix du livre si l'auteur est BERTIN.

NB : Respecter l'ordre de création des propriétés

Solution :

```
DELIMITER |  
  
CREATE TRIGGER before_update_livre BEFORE UPDATE  
ON LIVRE FOR EACH ROW  
BEGIN  
IF NEW.NOMAUT= ' BERTIN ' THEN  
SET NEW.PRIXLIV= OLD.PRIXLIV + (OLD.PRIXLIV * 0.1);  
END IF;  
END |
```