**Le lien :** <https://www.youtube.com/watch?v=ICQq2ZqhHok>

**Définition en General**

DATA DEFINTION LANGUAGE(DDL) : CREATE, DROP, ALTER

DATA MANIPULATION LANGUAGE(DML) :INSERT,UPDATE,DELETE,SELECT

Note : L'objet [Trigger](https://msdn.microsoft.com/library/microsoft.sqlserver.management.smo.trigger(v=SQL.130).aspx) représente des déclencheurs traditionnels du langage de manipulation de données (DML). Dans SQL Server 2008 et les versions ultérieures, les déclencheurs DDL sont également pris en charge. Les déclencheurs DDL sont représentés par l'objet [DatabaseDdlTrigger](https://msdn.microsoft.com/library/microsoft.sqlserver.management.smo.databaseddltrigger(v=SQL.130).aspx) et l'objet [ServerDdlTrigger](https://msdn.microsoft.com/library/microsoft.sqlserver.management.smo.serverddltrigger(v=SQL.130).aspx).

Coure Triggers

**Définition** :

* C’est un programme qui se déclenche automatiquement suite à un évènement.
* Triggers sont utiliser avec les DML (INSERT, UPDATE, DELETE)
* INSTEAD OF : Au lieu INSERT, UPDATE, DELETE Faire {INSTRUCTION}
* FOR : Avec … Faire {INSTRUCTION }
* AFTER : Apre … Faire { INSTRUCTION }
* Exemple : AFTER Insert Faire {Cette insertion est bien Ajouter}
* ROLLBACK : Annulation de l’évènement
* Exemple : AFTER Insert Faire {Cette insertion ne va pas être ajouter ROLLBACK ;}
* On peut utiliser ROLLBACK avec FOR et AFTER .

Lorsque le trigger ce déclenche il va créer deux tables virtuelles qui sont Inserted Table et Deleted Table.

|  |  |  |
| --- | --- | --- |
| **Evènement** | **Inserted Table** | **Deleted Table** |
| **Insert** | Values( ) | Null |
| **Delete** | Null | Deleted Value() |
| **Update** | Nouveaux Values() | Ancien Values() |

Exemple Pratique :

Création d’un tableau Customer dans les colonnes sont Id et Name :

**Création de la Table SQL Commande :**

CREATE TABLE [dbo].[Customer](

[Id] [int] PRIMARY KEY,

[Name] [nvarchar](255) NULL)

**Insertion dans la Table SQL Commande :**

INSERT INTO [dbo].[Customer] ([Id] ,[Name])

VALUES (1,’Customer’), (2,’Customer1’), (3,’Customer2’) ;

1. **Création d’un trigger Sql Commande avec INSTEAD OF :**

Create Trigger Affichage\_Trigger on [TriggerTest].[dbo].[Customer]

INSTEAD OF insert, delete

As Begin

SELECT [Id] ,[Name] FROM [dbo].[Customer]

END

**Insertion apres la creation du Trigger :**

INSERT INTO [dbo].[Customer] ([Id],[Name])

VALUES (4,'Customer3')

**Résultat :**

|  |  |
| --- | --- |
| **Id** | **Name** |
| **1** | **Customer** |
| **2** | **Customer1** |
| **3** | **Customer2** |

**Conclusion :**

**Pas D’insertion dans la table car le Trigger «**Affichage\_Trigger » **est activée.**

**Note : Pour effacer un trigger on utilise la commande :**

**Drop Trigger [Nom\_du\_trigger]**

1. **Création d’un trigger Sql Commande avec FOR :**

Create Trigger Affichage\_Trigger on [TriggerTest].[dbo].[Customer]

For insert, delete

As Begin

SELECT [Id] ,[Name] FROM [dbo].[Customer]

END

**Insertion apres la creation du Trigger :**

INSERT INTO [dbo].[Customer] ([Id],[Name])

VALUES (4,'Customer3')

**Résultat :**

|  |  |
| --- | --- |
| **Id** | **Name** |
| **1** | **Customer** |
| **2** | **Customer1** |
| **3** | **Customer2** |
| **4** | **Customer3** |

**Conclusion :**

**l’insertion dans la table est effectuer avec succès avec un affichages enregistrement car le Trigger «**Affichage\_Trigger » **est activée.**

**4.Création d’un trigger Sql Commande avec AFTER :**

Create Trigger Affichage\_Trigger on [TriggerTest].[dbo].[Customer]

AFTER insert, delete

As Begin

SELECT [Id] ,[Name] FROM [dbo].[Customer]

END

**Insertion après la création du Trigger :**

INSERT INTO [dbo].[Customer] ([Id],[Name])

VALUES (5,'Customer4')

**Résultat :**

|  |  |
| --- | --- |
| **Id** | **Name** |
| **1** | **Customer** |
| **2** | **Customer1** |
| **3** | **Customer2** |
| **4** | **Customer3** |
| **5** | **Customer4** |

**Conclusion :**

**l’insertion dans la table est effectuer avec succès avec un affichages enregistrement car le Trigger «**Affichage\_Trigger » **est activée.**

**Deuxième exercice : Trigger En Delete**

**On a ajouter une Table Commande :**

CREATE TABLE [dbo].[Commande](

[Id] [int] Primary Key ,

[Libelle] [nvarchar](255) NULL,

IDCustomer [int] REFERENCES Customer(Id) )

**Insertion table Commande :**

**Id Libelle IDCustomer**

**1 Cmd 1 3**

**2 Cmd 2 3**

**3 Cmd 3 3**

**4 Cmd 4 1**

**Note :**

delete Customer where id=3

**Dans ce cas on peut pas supprimer le client dans l’id est 3 .**

**Car il est reffèrencè dans d’autre table .**

**On var créer un Trigger  Instead of** **:**

Create Trigger Delete\_Trigger on [TriggerTest].[dbo].[Customer]

Instead of delete

As Begin

Delete Commande where IDCustomer=(select Id from deleted)

Delete Customer where id=(select Id from deleted )

END

**On va executer la requette de Delete avec sql :**

delete Customer where Id=3 ;

Resultat : avec le trigger Instead of on a put effacer le customer dans l’Id est egale a 3 .

Note : on peut pas utiliser un trigger avec For et After dans cette situation.

**Deuxième exercice : Trigger En Insert .**

create Trigger [dbo].[Verifier\_Age\_Client\_Trigger] on [TriggerTest].[dbo].[Customer]

Instead of insert

As Begin

if(select Age from inserted)<20

Begin

Print 'age client est infèrieure a 20';

Rollback ;

END

if(select Age from inserted)>20

Begin

insert into Customer (Id,Name,Age) Values((select Id from inserted),(select Name from inserted), (Select Age From inserted) )

||

insert into Customer (select \* from inserted)

END

END