



جامعة طرابلس كلية تقنية المعلومات



قواعد البيانات المتقدمة Advanced Databases ITSE312

د. عبدالسلام منصور الشريف

a.abdoessalam@uot.edu.ly

المحاضرة العاشرة - القوادح

SQL Triggers

Contents

- ▶ Tables Monitoring
 - ▶ What Are Triggers?
 - ▶ How an INSERT Trigger Works
 - ▶ How a DELETE Trigger Works
 - ▶ How an UPDATE Trigger Works
 - ▶ How an INSTEAD OF Trigger Works
 - ▶ How Nested Triggers Work
 - ▶ Considerations for Recursive Triggers
 - ▶ Practice: Creating Triggers

What are Triggers?

- ▶ Special stored procedures that execute when INSERT, UPDATE, or DELETE statements modify a table
- ▶ Two categories:
 - ▶ AFTER triggers execute after an INSERT, UPDATE, or DELETE statement
 - ▶ INSTEAD OF triggers execute instead of an INSERT, UPDATE, or DELETE statement
- ▶ Trigger and the initiating statement are part of a single transaction


```
CREATE [ OR ALTER ] TRIGGER [schema_name.]trigger_name
ON { table | view }
{AFTER | INSTEAD OF }
{ [ INSERT ] [ , ] [ UPDATE ] [ , ] [ DELETE ] }
AS { sql_statement [ ; ] [ ,...n ] }
```

How does INSERT Trigger work?


- ▶ INSERT statement executed
- ▶ INSERT statement logged
- ▶ AFTER INSERT trigger statements executed

```
CREATE TRIGGER dbo.Reminder
ON [dbo].[Students]
AFTER INSERT, UPDATE
AS RAISERROR ('Notify Student's department', 16, 10);
GO
```

How does DELETE Trigger work?

- DELETE statement executed
 - DELETE statement logged
 - AFTER DELETE trigger statements executed
-
- 

How does UPDATE Trigger work?

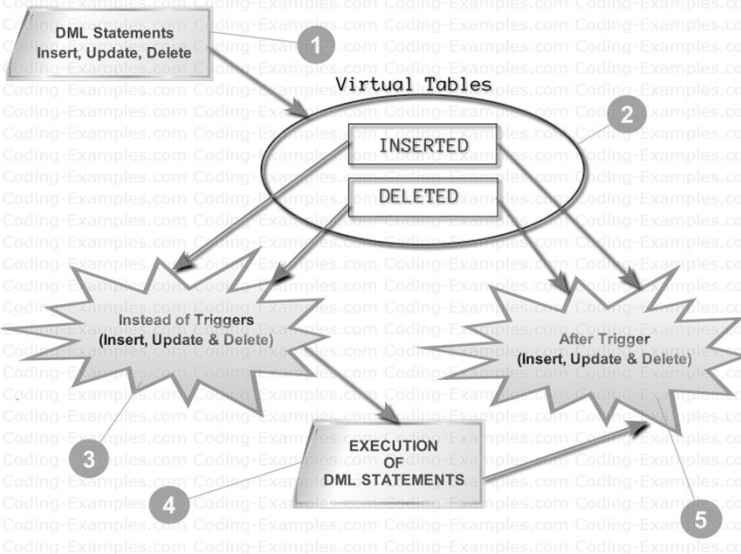
- UPDATE statement executed
 - UPDATE statement logged
 - AFTER UPDATE trigger statements executed
-
- 

How does INSTEAD OF Trigger work?

- ▶ UPDATE, INSERT, or DELETE statement executed
- ▶ Executed statement does not occur
- ▶ INSTEAD OF trigger statements executed

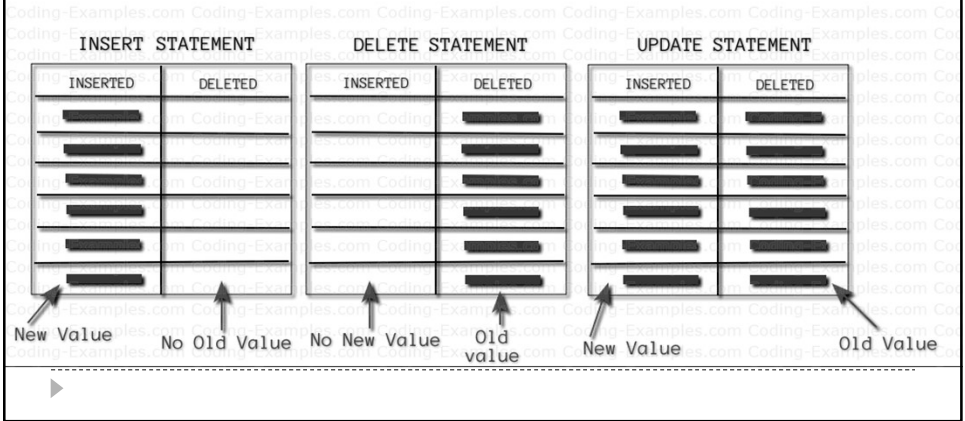


Trigger Execution Order

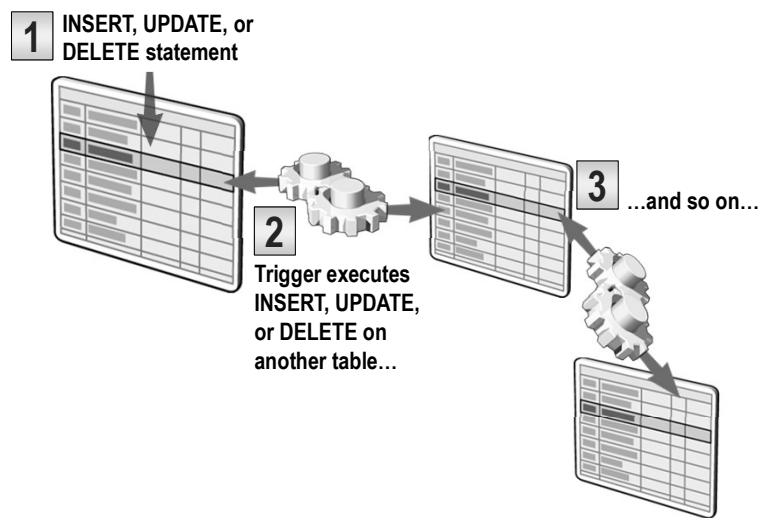


INSERTED & DELETED Virtual Tables

- SQL Server maintains two virtual tables named INSERTED and DELETED. When there is a trigger for the table say ABC, SQL server populates the virtual table while executing the DML statements on that table ABC. Have a look at the below picture.



How Nested Triggers Work



Considerations for Recursive Triggers

- ▶ Disabled by default. To enable:

```
ALTER DATABASE ITDatabase SET RECURSIVE_TRIGGERS ON
```

- ▶ Considerations:

- ▶ Can exceed the 32-level nesting limit without careful design and thorough testing
- ▶ Can be difficult to control the order of table updates
- ▶ Can be replaced with nonrecursive logic



Enabling and Disabling Triggers

- ▶ Triggers are enabled by default when they are created. Disabling a trigger does not drop it.
- ▶ The trigger still exists as an object in the current database. However, the trigger does not fire when any Transact-SQL statements on which it was programmed are executed.
- ▶ Triggers can be re-enabled by using `ENABLE TRIGGER`.

```
[ENABLE | DISABLE] TRIGGER { [ schema_name . ]
trigger_name [ ,...n ] | ALL }
ON object_name [ ; ]
```



Deleting Triggers

► Triggers can be deleted by using:

```
DROP TRIGGER [ IF EXISTS ]  
[schema_name.]trigger_name [ ,...n ] [ ; ]
```

