

Introdução a Banco de dados

Comandos SQL Server – Parte 8

Triggers – SQL Server

Um trigger (“gatilho”) é um objeto programável do banco de dados associado a uma tabela. Trata-se de um procedimento que é invocado automaticamente quando um comando DML é executado na tabela, sendo executado para cada linha afetada. Desta forma, as operações que podem disparar um trigger são:

Triggers – SQL Server

As operações que podem disparar um trigger são:

- ① INSERT
- ① UPDATE
- ① DELETE

Triggers – SQL Server

Modos de disparo de um Trigger:

After – O código presente no trigger é executado após as ações terem sido completadas na tabela especificada.

Instead of - O código presente no trigger é executado no lugar da operação que causou seu disparo.

Triggers – SQL Server

A tabela a seguir compara os dois tipos de Trigger:

| <u>Característica</u> | <u>INSTEAD OF</u> | <u>AFTER</u> |
|-----------------------------|--------------------------------------|---|
| Declaração <u>DML</u> | Simulada, mas não executada | Executada, mas pode ser revertida no <u>trigger</u> (Roll back) |
| <u>Timing</u> | Antes das <u>constraints</u> PK e FK | Após a transação completa, mas antes do <u>commit</u> |
| Nº de eventos por tabela | Um | Múltiplos |
| Aplicável em <u>Views</u> ? | Sim | Não |
| Permite Aninhamento? | Depende das opções do servidor | Depende das opções do servidor |
| É Recursivo? | Não | Depende das opções do Banco de Dados. |

Triggers – SQL Server

Fluxo de Transações

Para desenvolver Triggers, é necessário conhecimento de fluxo geral da transação para evitar conflitos entre os Triggers e Constraints.

As transações se movem através de verificações e códigos na ordem mostrada a seguir:

Triggers – SQL Server

1. Verificação de IDENTITY INSERT
2. Restrição (Constraint) de Nulos (NULL)
3. Checagem de tipos de dados
4. Execução de trigger INSTEAD OF (a execução do DML pára aqui; esse trigger não é recursivo)
5. Restrição de Chave Primária
6. Restrição “Check”
7. Restrição de Chave Estrangeira
8. Execução do DML e atualização do log de transações
9. Execução do trigger AFTER
10. Commit da Transação (Confirmação)

Triggers – SQL Server

Sintaxe do Trigger:

```
CREATE TRIGGER Nome_Trigger  
ON Tabela/View  
[WITH ENCRYPTION]  
AFTER/INSTEAD OF  
[INSERT, UPDATE, DELETE]  
AS  
Código do Trigger
```


Triggers – SQL Server

nome_trigger = Nome do trigger

tabela = Tabela (ou View) onde será executado o trigger.

WITH ENCRYPTION = Criptografa o código do trigger
(opcional)

AFTER = O trigger é disparado quando todas as operações especificadas nas declarações SQL tiverem sido executadas com sucesso

INSTEAD OF = o trigger é disparado no lugar das declarações SQL.

DELETE, UPDATE, INSERT = Definem quais tipos de modificações na tabela especificada ativarão o trigger

Triggers – SQL Server

Criar Trigger AFTER - Exemplo

```
CREATE TRIGGER Teste_trigger_after  
ON Tbl_editoras  
AFTER INSERT  
AS  
PRINT 'Novos dados inseridos na tabela Editoras'
```

Triggers – SQL Server

Criar Trigger AFTER - Exemplo

```
INSERT INTO tbl_editoras  
VALUES ('Editora10')
```

Triggers – SQL Server

```
CREATE TRIGGER trigger_after  
ON Tbl_editoras  
AFTER INSERT  
AS  
INSERT INTO tbl_autores VALUES (25,'José','da Silva')  
INSERT INTO tbl_livro (Nome_Livro, ISBN13, ISBN10, Data_Pub,  
Preco_Livro, ID_Categoria, ID_Autor, ID_Editora)  
VALUES('Livro1','9781118983843', '111898384X', '20150109', 68.35, 1,  
5, 4)
```

Triggers – SQL Server

Criar Trigger AFTER - Exemplo

```
INSERT INTO tbl_editoras  
VALUES ('Editora11')  
SELECT * FROM tbl_autores
```

Após essa inserção o Trigger registrará novos registros na Tbl_autores e Tbl_Livro

Triggers – SQL Server


Criar Trigger INSTEAD OF - Exemplo

```
CREATE TRIGGER Teste_trigger_insteadof  
ON Tbl_autores  
INSTEAD OF INSERT  
AS  
PRINT 'Olá mundo! Não inseri o registro desta vez'
```

Triggers – SQL Server

Criar Trigger INSTEAD OF - Exemplo

**INSERT INTO tbl_autores
VALUES (26, 'Joao', 'Moura')**

 Mensagens

Olá mundo! Não inseri o registro desta vez

(1 linha afetada)

Horário de conclusão: 2022-05-29T11:07:01.4673112-03:00

Triggers – SQL Server

Habilitar e desabilitar Triggers

O administrador do sistema pode desabilitar temporariamente um Trigger se houver necessidade. Para isso, use o comando DDL ALTER TABLE.

ALTER TABLE Nome_tabela

ENABLE/DISABLE TRIGGER Nome_trigger

Triggers – SQL Server

Exemplo desabilitar Triggers

```
ALTER TABLE Tbl_editoras  
DISABLE TRIGGER trigger_after
```

Triggers – SQL Server

Verificar a existência de Triggers

Em uma tabela específica:

EXEC sp_helptrigger @tabname=Nome_tabela

EXEC sp_helptrigger @tabname=tbl_editoras

Triggers – SQL Server

Verificar a existência de Triggers

No banco de dados todo:

USE Nome_banco_de_dados

SELECT *

FROM sys.triggers

WHERE is_disabled = 0

Triggers – SQL Server

Verificar a existência de Triggers

USE Biblioteca

SELECT *

FROM sys.triggers

WHERE IS_disabled = 0 OR IS_disabled = 1

Triggers – SQL Server

Determinando as colunas atualizadas

A função UPDATE() retorna True se uma coluna especificada for afetada por uma transação DML.

Podemos criar um gatinho que executa um código caso uma coluna especificada seja alterada por comando DML usando uma função.

Triggers – SQL Server

```
CREATE TRIGGER Trigger_after_autores
ON Tbl_autores
AFTER INSERT, UPDATE
AS
IF UPDATE(Nome_autor)
    BEGIN
        PRINT 'O nome do autor foi alterado'
    END
ELSE
    BEGIN
        PRINT 'O nome do autor não foi alterado'
    END
```

Triggers – SQL Server

```
UPDATE tbl_autores  
SET Nome_Autor='Joao'  
WHERE ID_Autor=10
```

Mensagens

O nome do autor foi alterado

(1 linha afetada)

Horário de conclusão: 2022-05-29T11:38:17.6126486-03:00

Triggers – SQL Server

```
UPDATE tbl_autores  
SET Sobrenome_Autor='Guimarães'  
WHERE ID_Autor=10
```

Mensagens

O nome do autor não foi alterado

(1 linha afetada)

Horário de conclusão: 2022-05-29T11:52:07.7994176-03:00

Triggers – SQL Server

Aninhamento de Triggers DML

Um Trigger, ao ser disparado, pode executar uma declaração DML que leva ao disparo de outro Trigger.

Para isso, a opção de servidor “Permitir que gatilhos disparem outros gatilhos.

```
EXEC sp_configure 'Nested Triggers', 1  
RECONFIGURE
```

Triggers – SQL Server

Aninhamento de Triggers DML

“Permitir que gatilhos disparem outros gatilhos.



Triggers – SQL Server

Triggers Recursivos

Um Trigger recursivo é um tipo de Trigger AFTER aninhado.

O Trigger recursivo ocorre quando um Trigger executa uma declaração DML que o dispara novamente.

Triggers – SQL Server

Triggers Recursivos

Podemos habilitar ou desabilitar os Triggers recursivos com o comando ALTER DATABASE.

```
ALTER DATABASE Nome_banco_de_dados  
SET RECURSIVE_TRIGGERS ON/OFF
```

```
ALTER DATABASE BIBLIOTECA  
SET RECURSIVE_TRIGGERS ON/OFF
```

Triggers – SQL Server

Triggers Recursivo – Exemplo

Vamos criar uma tabela chamada “Trigger_recurso” com um campo PK, Tipo INT, chamado código, no banco de dados Biblioteca:

```
CREATE TABLE tbl_trigger_recurso(  
Codigo INT PRIMARY KEY)
```

Triggers – SQL Server

Triggers Recursivo – Exemplo

Vamos inserir um registro que irá disparar o trigger a seguir, o qual por sua vez irá inserir outro registro, que o disparará novamente:

```
CREATE TABLE tbl_trigger_recursoivo(  
Codigo INT PRIMARY KEY)
```

Triggers – SQL Server

Triggers Recursivo – Exemplo

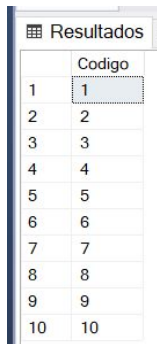
```
CREATE TRIGGER Trigger_rec on tbl_trigger_recurso  
AFTER INSERT  
AS  
DECLARE @cod INT  
SELECT  
@cod = MAX(codigo)  
FROM tbl_trigger_recurso  
IF @cod<10  
BEGIN  
INSERT INTO Tbl_Trigger_recurso SELECT MAX(codigo)+1  
FROM Tbl_trigger_recurso  
END  
ELSE  
BEGIN  
PRINT 'Trigger recursivo finalizado'  
END
```

Triggers – SQL Server

Triggers Recursivo – Exemplo

INSERT INTO tbl_trigger_recursivo

VALUES(1)



| | Codigo |
|----|--------|
| 1 | 1 |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 6 | 6 |
| 7 | 7 |
| 8 | 8 |
| 9 | 9 |
| 10 | 10 |

Após o comando INSERT o Trigger vai inserir mais 9 automática até atingir o código 10.

Triggers – SQL Server

Triggers Recursivo – Exemplo

Vamos inserir um registro que irá disparar o trigger a seguir, o qual por sua vez irá inserir outro registro, que o disparará novamente:

```
CREATE TABLE tbl_trigger_recursoivo(  
Codigo INT PRIMARY KEY)
```

```
CREATE TRIGGER TGR_TESTE
ON TBL_LIVRO
FOR INSERT
AS
BEGIN
    DECLARE
        @VALOR DECIMAL(10,2),
        @COD INT
    SELECT @COD = ID_Livro, @valor = Preco_Livro FROM tbl_livro
    IF @valor>60
        UPDATE TBL_LIVRO SET PRECO_LIVRO = @valor * 0.90
        WHERE ID_Livro = @COD
    ELSE
        UPDATE TBL_LIVRO SET PRECO_LIVRO = @valor * 0.95
        WHERE ID_Livro = @COD
    END
```

```
INSERT INTO tbl_livro (Nome_Livro, ISBN13, ISBN10, Data_Pub, Preco_Livro, ID_Categoria, ID_Autor, ID_Editora)
VALUES('Narnia','9781118983888', '111898777X', '20150109', 100, 1, 5, 4)
```

```
INSERT INTO tbl_livro (Nome_Livro, ISBN13, ISBN10, Data_Pub, Preco_Livro, ID_Categoria, ID_Autor, ID_Editora)
VALUES('Revista do Chaves','9781118983222', '111898333X', '20150109', 10, 1, 5, 4)
```

```
SELECT * FROM tbl_livro
```

129 %

Resultados Mensagens

| | ID_Livro | Nome_Livro | ISBN13 | ISBN10 | ID_Categoria | ID_Autor | ID_Editora | Data_Pub | Preco_livro |
|---|----------|--|---------------|------------|--------------|----------|------------|------------|-------------|
| 1 | 1 | Linux Command Line and Shell Scripting | 9781118983843 | 111898384X | 1 | 5 | 4 | 2015-01-09 | 68.35 |
| 2 | 2 | SSH, o Shell Seguro | 9780596008956 | 0596008953 | 1 | 1 | 2 | 2005-05-17 | 58.30 |
| 3 | 3 | Using Samba | 9780596002565 | 0596002564 | 1 | 2 | 2 | 2003-12-21 | 61.45 |
| 4 | 4 | Fedora and Red Hat Linux | 9780133477436 | 0133477436 | 1 | 3 | 1 | 2014-01-10 | 62.24 |
| 5 | 5 | Windows Server 2012 Inside Out | 9780735666313 | 0735666318 | 1 | 4 | 3 | 2013-01-25 | 66.80 |
| 6 | 6 | Microsoft Exchange Server 2010 | 9780735640610 | 0735640610 | 1 | 4 | 3 | 2010-12-01 | 47.20 |
| 7 | 7 | Practical Electronics for Inventors | 9781259587542 | 1259587541 | 1 | 13 | 5 | 2016-03-24 | 67.80 |
| 8 | 9 | Narnia | 9781118983888 | 111898777X | 1 | 5 | 4 | 2015-01-09 | 90.00 |
| 9 | 10 | Revista do Chaves | 9781118983222 | 111898333X | 1 | 5 | 4 | 2015-01-09 | 9.50 |