

القادحات في قواعد المعطيات

Triggers in Databases

1- استخدام القادحات لفرض قيود العمل Using triggers to force constraints

القادح بالتعريف هو نوع خاص من الإجراءات المخزنة التي تنفذ تلقائياً عندما تتغير البيانات الموجودة في جدول محدد. إذ يتم استدعاء القادح استجابة إلى أحد تعليمات Insert, Update, Delete.

تتم معاملة القادح والتعليمة التي حفزته كمنافلة واحدة وهي منافلة يمكن التراجع عنها من داخل القادح وذلك في حال حصول خطأ.

توفر مخدمات قواعد البيانات طريقتين أساسيتين لفرض قيود العمل وهما القيود Constraints والقادحات Triggers.

2- مقارنة القادحات بالقيود

الفائدة الأساسية للقادحات تكمن في احتوائها على تعليمات معالجة معقدة، وبالتالي فهي أوسع من القيود في معالجتها.

تكون القادحات مفيدة في الحالات التي لا تحقق فيها القيود الحاجات الوظيفية لتطبيق ما:

- يمكن للقادحات أن تسلسل التعديلات إلى الجداول المرتبطة، ولكن هذه السلسلة يمكن فرضها أيضاً عبر قيود التكامل المرجعي واستخدام التسلسل للقيود المرجعية يكون عادة أكثر فعالية.
- يمكن للقادحات أن تفرض القيود المعقدة التي يصعب فرضها باستخدام قيد الاختبار Check Constraint. لا يمكن لقيد الاختبار أن يشير إلى أعمدة موجودة في جداول أخرى، أما القادحات فهي قادرة على ذلك.
- يمكن للقادح أن يختبر حالة الجدول قبل وبعد التعديل، وأن يقوم بعمليات معينة بناء على ذلك.
- يمكن إضافة عدة قادحات لنفس نمط التغيير على الجدول.

3- إنشاء القادحات

قواعد بناء القادحات

- مالك الجدول هو المستخدم الافتراضي الذي يحق له إنشاء القادحات المقابلة. ولا يمكن لمالك الجدول أن يمنح هذا الحق لغيره.
- يمكن إنشاء القادحات في قاعدة البيانات الحالية فقط، رغم أن القادح يمكنه استخدام جداول موجودة في قواعد أخرى.
- لا يمكن إنشاء قادحات على جداول مؤقتة أو خاصة بالنظام رغم أنه يمكن استخدام الجداول المؤقتة ضمن القادح.
- لا يمكن تعريف قادحات الاستبدال للحذف والتعديل على الجداول التي تحوي مفاتيح أجنبية معرفة بأفعال مرتبطة مثل التسلسل CASCADE أو إسناد قيمة NULL أو قيمة افتراضية DEFAULT.
- لا تؤدي تعليمة بتر الجدول Truncate Table (وهي تعليمة ماثلة لتعليمة الحذف DELETE بدون شرط فلترة أي حذف جميع السجلات من الجدول) إلى تحفيز القادح البديل عن الحذف.
- يمكن فقط لمالك الجدول، وأعضاء المجموعة sysadmin، وأعضاء المجموعات الخاصة بقاعدة البيانات db_owner، db_ddladmin أن ينفذوا تعليمة Create Trigger. ولا يمكن نقل هذا الحق إلى بقية المستخدمين.

4- تصميم القادحات

هناك نمطان من القادحات:

- القادحات الاستبدالية Instead Of Triggers: وهي قادحات تنفذ عوضاً عن التغيير الأصلي. أي إذا كان هذا القادح استبدالاً لعملية الإضافة Insert فلا تنفذ عملية الإضافة وإنما ينفذ القادح عوضاً عنه. بمعنى آخر لو أردنا أن تتم عملية الإضافة الأصلية فإن مسؤولية الإضافة تقع على عاتق القادح (أي يجب أن نكتب كود الإضافة في القادح).

يمكن تعريف القادحات الاستبدالية على المناظير متعددة الجداول، الأمر الذي يسمح بتوسيع أنماط التحديث الممكنة عبر هذه المناظير.

- **القادحات اللاحقة After Triggers:** وهي قادحات تنفذ بعد نجاح الفعل الموافق لها (Insert, Update, Delete)، في حال فشلت عملية (الإضافة أو الحذف أو التعديل) فلن يتم استدعاء القادح اللاحق المرتبط.

يتم التحقق من القيود بعد القادحات من النمط Instead Of وقبل القادحات من النمط After. إذا سبب القادح Instead Of اختراقاً لأحد القيود فإن المناقشة الموافقة له تلغى، ولا يتم تنفيذ القادح After الموافق إن وجد.

انتبه:

يمكن لأي جدول أو منظار أن يحوي **قادحاً استبدالياً وحيداً**، بينما يمكنه أن يحوي **عدة قادحات لاحقة** خاصة بكل فعل ممكن.

5- استخدام جداول السجلات المضافة والمحذوفة Inserted and Deleted Tables

يقوم مخدم البيانات بإنشاء وإدارة الجدولين Inserted, Deleted. هذان الجدولان هما جدولان مؤقتان مخزنان في الذاكرة ويحويان القيم السابقة Deleted واللاحقة Inserted الناتجة عن العملية.

يخزن جدول المحذوفات **Deleted** نسخة عن السجلات القديمة المتأثرة بكل من عمليتي الحذف والتعديل.

يخزن جدول الإضافات **Inserted** نسخة عن السجلات الجديدة الناتجة عن كل من عمليتي الإضافة والتعديل.

لأنه يعتبر أن عملية التعديل هي حذف للقيم القديمة وإدراج للقيم الجديدة.

أثناء **عمليتي الحذف والتعديل** يقوم المخدم بحذف السجلات المتأثرة بالعملية من الجدول الموافق وإضافتها إلى الجدول Deleted.

أثناء عمليتي **الإضافة والتعديل** يقوم المخدم بتخزين القيم المحدثة أو الجديدة للسجلات في الجدول Inserted. أي أن هذا الجدول يحوي نسخاً من السجلات الجديدة المراد إضافتها إلى الجدول المرتبط.

بالتالي: جدول المحذوفات لا يحوي أي سجلات للقادحات البديلة عن الإضافة، كما أن جدول المضافات لا يحوي أي سجلات للقادحات البديلة عن الحذف.

بنية الجداول في الأمثلة:

```
CREATE TABLE Employee
(
    Emp_ID int identity,
    Emp_Name varchar(55),
    Emp_Sal decimal (10,2)
)

Insert into Employee values ('hasan',1000);
Insert into Employee values ('wesam',1200);
Insert into Employee values ('samer',1100);
Insert into Employee values ('lamis',1300);
Insert into Employee values ('fahed',1400);

create table Employee_log
(
    Emp_ID int,
    Emp_Name varchar(55),
    Emp_Sal decimal(10,2),
```

```
Audit_Action varchar(100) ,
Audit_Timestamp datetime
)
```

6- تصميم القادحات الاستبدالية:

ملاحظة جانبية: استخدام القادحات العودية Recursive Triggers

القادح العودي هو القادح الذي يقوم باستدعاء نفسه من ضمن تعريف القادح عند تنفيذ العملية الموافقة للقادح على نفس الجدول.

هناك نوعان من العودية:

- **العودية المباشرة:** وتحدث عندما ننفذ نفس التعليمة التي فعلت القادح من ضمن القادح مباشرة. مثلاً إذا عرفنا قادحاً بديلاً عن التحديث TrigU على الجدول T. فعندما نجري عملية تحديث لسجلات في T فإن القادح TrigU ينفذ. إذا قمنا ضمن تعريف القادح TrigU بتحديث الجدول T مجدداً فإن القادح TrigU ينفذ بالعودية المباشرة.
 - **العودية غير المباشرة:** وتحدث عندما يحفز القادح المعنى قادحاً آخر معرف على جدول ثان، الذي يقوم بدوره بتنفيذ التعليمة التي فعلت القادح الأول مما يؤدي إلى إعادة استدعاء القادح بالعودية غير المباشرة. مثلاً نعرف بديل إضافة Trig1 على الجدول T1 الذي يستدعي ضمنه قادح الإضافة Trig2 المعرف على الجدول T2. إذا قام القادح Trig2 بعملية إضافة على الجدول T1 فإن القادح Trig1 ينفذ بالعودية غير المباشرة.
- بشكل عام القادحات ليست عودية ما لم نعط قيمة لخيار قاعدة البيانات RECURSIVE TRIGGER القيمة ON. إذا وضعنا قيمة هذا الخيار OFF فإنه يتم منع العودية المباشرة فقط.

لمنع العودية غير المباشرة يجب أن نسمح إلى خيار مخدّم البيانات nested triggers القيمة 0.

الفائدة الأساسية للقادحات الاستبدالية هي لدعم المناظير القابلة للتحديث.

إذا تضمن منظر ما على أكثر من جدول، عندها يجب استخدام القادح الاستبدالي لدعم عمليات الإضافة والحذف والتعديل عبر هذا المنظر للجدول المرتبطة به.

6.1. القادح الاستبدالي مع عملية الإضافة: Instead of Insert Trigger

يقوم القادح البديل عن الاستبدال بالتنفيذ عوضاً عن فعل الإضافة العادي المنفذ من قبل المخدم. هذا القادح مفيد عادة لتمكين الإضافة عبر منظر إلى مجموعة الجداول المضمنة فيه (أو بعضها على الأقل).

في مثالنا هنا نقوم بمناقشة الموظف المضاف بشرط أن لا يكون راتبه أقل 1000

```
CREATE TRIGGER trigger_instead_insert ON Employee
INSTEAD OF Insert
AS
declare @emp_id int, @emp_name varchar(55),
        @emp_sal decimal(10,2), @log_action varchar(100);

select @emp_id=i.Emp_ID from inserted i;
select @emp_name=i.Emp_Name from inserted i;
select @emp_sal=i.Emp_Sal from inserted i;
SET @log_action='Inserted Record -- Instead Of Insert
Trigger.';
BEGIN
    if (@emp_sal <= 1000)
    begin
        print 'Cannot Insert where salary < 1000 '
        ROLLBACK;
    end
    else
    begin
        Insert into
```

```

Employee (Emp_Name,Emp_Sal)
values (@emp_name,@emp_sal) ;
Insert into
Employee_log(Emp_ID,Emp_Name,Emp_Sal,Audit_Action,Audit_Timestamp)
values (@@identity,@emp_name,@emp_sal
,@log_action,getdate() ) ;
PRINT 'Record Inserted Successfully .'
END
END

----- ثم جرب الأمثلة التالية -----
insert into Employee values ('wedad',1300)
insert into Employee values ('Ahmad',900)

```

6.2. القادح الاستبدالي مع عملية التعديل:

يجب عدم تنفيذ عملية التعديل على الموظف في حال كان الراتب الجديد أقل من 1000

```

CREATE TRIGGER trgInsteadOfUpdate ON dbo.Employee
INSTEAD OF Update
AS
declare @emp_id int, @emp_name varchar(55),
        @emp_sal decimal(10,2), @log_action varchar(100);
select @emp_id=i.Emp_ID from inserted i;
select @emp_name=i.Emp_Name from inserted i;
select @emp_sal=i.Emp_Sal from inserted i;
SET @log_action='Updated Record -- Instead Of Update';
BEGIN
if (@emp_sal <= 1000)
begin
RAISERROR('Cannot update where salary < 1000',16,1);
ROLLBACK;
end
else
begin
update Employee set emp_name = @emp_name
, emp_sal = @emp_sal
where emp_id = @emp_id;
insert into Employee_log(Emp_ID,Emp_Name,Emp_Sal
,Audit_Action,Audit_Timestamp)
values (@emp_id,@emp_name,@emp_sal,
@log_action,getdate());
PRINT 'Record Updated successfully';
END
END

----- ثم جرب الأمثلة التالية -----

update Employee set Emp_Sal = '1400' where emp_id = 6
update Employee set Emp_Sal = '900' where emp_id = 6

```

لعرض رسائل الخطأ يمكن استخدام أي من التعليمتين:

```

RAISERROR('Cannot Insert where salary < 1000',16,1);
print 'Cannot Insert where salary < 1000'

```

6.3. القادح الاستبدالي مع عملية الحذف:
يتم حذف الموظف بشرط أن يكون راتبه أقل من 1200

```
CREATE TRIGGER trgAfterDelete ON dbo.Employee
INSTEAD OF DELETE
AS
declare @empid int, @empname varchar(55), @empsal
decimal(10,2), @audit_action varchar(100);
select @empid=d.Emp_ID FROM deleted d;
select @empname=d.Emp_Name from deleted d;
select @empsal=d.Emp_Sal from deleted d;
Begin
if (@empsal>1200)
begin
RAISERROR('Cannot delete where salary > 1200',16,1);
ROLLBACK;
end
else
begin
delete from Employee where Emp_ID=@empid;

insert into Employee_log (Emp_ID,Emp_Name,Emp_Sal,
                        Audit_Action,Audit_Timestamp)
values (@empid,@empname,@empsal
        , 'Deleted -- Instead Of Delete Trigger.',getdate() );
PRINT 'Record Deleted -- Instead Of Delete Trigger.'
end
END

----- ثم جرب الأمثلة التالية -----
DELETE FROM Employee where emp_id = 1
DELETE FROM Employee where emp_id = 3
```

7- تصميم القادحات اللاحقة:
a. القادح اللاحق بعد عملية الإضافة After Insert Trigger

```
CREATE TRIGGER trgAfterInsert on Employee
FOR INSERT
AS declare @empid int, @empname varchar(55),
@empsal decimal(10,2), @audit_action varchar(100);
select @empid=i.Emp_ID from inserted i;
select @empname=i.Emp_Name from inserted i;
select @empsal=i.Emp_Sal from inserted i;
set @audit_action='Inserted Record -- After Insert Trigger.';

insert into Employee_log (Emp_ID,Emp_Name,Emp_Sal,
                        Audit_Action,Audit_Timestamp)
values (@empid,@empname,@empsal,
        @audit_action,getdate());
PRINT 'AFTER INSERT trigger fired.'
```

-----ثم جرب الأمثلة التالية-----
insert into Employee (Emp_Name, Emp_Sal) **values** ('sami', 1000);

ملاحظة: حتى لو قام القادح الاستبدالي بإلغاء عملية الاضافة واستبدالها بأي تعليمة أخرى، يستمر تنفيذ القادح اللاحق لعملية الاضافة بعد تعليمة الاضافة، وكان عملية الاضافة قد تمت بنجاح.

b. القادح اللاحق بعد عملية التعديل After Update Trigger

```
CREATE TRIGGER trgAfterUpdate ON dbo.Employee
FOR UPDATE
AS
declare @empid int, @empname varchar(55),
        @empsal decimal(10,2), @audit_action varchar(100);
select @empid=i.Emp_ID from inserted i;
select @empname=i.Emp_Name from inserted i;
select @empsal=i.Emp_Sal from inserted i;

if update(Emp_Name)
    set @audit_action='Update on Emp_name ';
else
    if update (Emp_Sal)
        set @audit_action='Update on Salary..';
    else
        if update(Emp_Name) AND update (Emp_Sal)
            set @audit_action='Update on Salary and NAmE';

insert into
    Employee_log (Emp_ID, Emp_Name, Emp_Sal,
                  Audit_Action, Audit_Timestamp)
    values (@empid, @empname, @empsal,
            @audit_action, getdate());
PRINT 'AFTER UPDATE trigger fired.'
```

-----ثم جرب الأمثلة التالية-----
update Employee **set** Emp_Name='Pawan' **Where** Emp_ID =6;

التابع update في السطر: **if update** (Emp_Name) يستخدم للتحقق من أن الحقل Emp_Name قد تم تعديل قيمته (أي هو يستخدم لمعرفة الحقل الذي تغيرت قيمته من الحقل الذي لم تتغير قيمته)

c. القادح اللاحق بعد عملية الحذف After Delete Trigger

```
CREATE TRIGGER trgAfterDelete ON dbo.Employee
FOR DELETE
AS
declare @empid int, @empname varchar(55),
        @empsal decimal(10,2), @audit_action varchar(100);
select @empid=d.Emp_ID FROM deleted d;
select @empname=d.Emp_Name from deleted d;
select @empsal=d.Emp_Sal from deleted d;
select @audit_action='Deleted -- After Delete Trigger.';
```

```

insert into Employee_log
    (Emp_ID, Emp_Name, Emp_Sal, Audit_Action, Audit_Timestamp)
values (@empid, @empname, @empsal, @audit_action, getdate());
PRINT 'AFTER DELETE TRIGGER fired.'

```

----- ثم جرب الأمثلة التالية -----

```

DELETE FROM Employee where emp_id = 5

```

8- تعديل القادحات

لتعديل القادح يمكننا أن نحذفه ومن ثم نعيد تعريفه كما يمكننا تعديل القادح بتعليمة واحدة.

إذا تغير اسم أي غرض مستخدم في القادح فيجب عندها تعديل القادح بما يتوافق، لذلك قبل تعديل اسم أي غرض يجب عرض قائمة الأغراض المرتبطة به لتحديد وجود أي قادحات متأثرة بهذا التغيير.

كما يمكن تعديل أي قادح يملكه المستخدم من قبل المستخدم نفسه (حكما في قاعدة البيانات الحالية) كما يمكن لمدير النظام أن يغير اسم أي قادح لأي مستخدم.

9- حذف القادحات

عند حذف قادح ما فإن الجدول أو المنظار الموافق لا يتأثر. أما إذا حذفنا جدولا أو منظارا فإن جميع القادحات المرتبطة ستحذف تلقائيا. يحق لمالك القادح فقط إضافة إلى مدير النظام إمكانية حذف القادح.

نستخدم التعليمة DROP TRIGGER كما يبين المثال التالي:

```

DROP TRIGGER InsteadTrigger

```

----- انتهت المحاضرة -----