# 实验七 触发器

## 一、实验目的

（1）理解触发器的用途、类型和工作原理

（2）掌握利用T-SQL语句创建和维护触发器的方法

（3）掌握利用企业管理器创建、维护触发器的方法

## 二、实验内容

### 创建after触发器并测试

#### 第一题

创建一个在插入时触发的触发器sc\_insert,当向sc表插入数据时，须确保插入的学号已在Student表中存在，并且还须确保插入的课程号在Course表中存在；若不存在，则给出相应的提示信息，并取消插入操作，提示信息要求指明插入信息是学号不满足条件还是课程号不满足条件。（注：先取消相关外键）

##### 第一题SQL语句

IF EXISTS(SELECT \*

FROM sysobjects

WHERE name = 'sc\_insert' AND type = 'TR')

BEGIN

DROP TRIGGER sc\_insert

PRINT '已删除'

END

ELSE

BEGIN

PRINT '可创建'

END

GO

CREATE TRIGGER sc\_insert

ON SC

AFTER INSERT

AS

DECLARE @sno VARCHAR(20), @cno VARCHAR(20)

SELECT @sno = Sno, @cno = Cno

FROM INSERTED

IF NOT EXISTS(SELECT \*

FROM Student

WHERE Sno = @sno)

BEGIN

ROLLBACK TRAN

RAISERROR('在Student表中不存在', 16, 10)

END

IF NOT EXISTS(SELECT \*

FROM Course

WHERE Cno = @cno)

BEGIN

ROLLBACK TRAN

RAISERROR ('在Course表中不存在', 16, 10)

END

GO

##### 执行结果

##### 验证

1. 学号不满足条件

INSERT

INTO SC(Sno, Cno, Grade)

VALUES ('20110050', '001', 50);

1. 课程号不满足条件

INSERT

INTO SC(Sno, Cno, Grade)

VALUES ('20110001', '050', 50);

1. 满足条件

INSERT

INTO SC(Sno, Cno, Grade)

VALUES ('20110003', '004', 50)

#### 第二题

为Course表创建一个触发器Course\_del，当删除了Course表中的一条课程信息时，同时将表sc表中相应的学生选课记录删除掉。

##### 第二题SQL代码

IF EXISTS(SELECT \*

FROM sysobjects

WHERE name = 'Course\_del' AND type = 'TR')

DROP TRIGGER Course\_del

GO

CREATE TRIGGER Course\_del

ON Course

AFTER DELETE

AS

BEGIN TRANSACTION

DECLARE @cno VARCHAR(20)

SELECT @cno = Cno

FROM DELETED

WHILE EXISTS(SELECT \*

FROM SC

WHERE Cno = @cno)

BEGIN

DELETE

FROM SC

WHERE Cno = @cno

END

COMMIT TRANSACTION

GO

##### 执行结果

##### 验证

DELETE

FROM Course

WHERE Cno = '009'

DELETE

FROM SC

WHERE Cno = '009'

SELECT \*

FROM SC

WHERE Cno = '009'

#### 第三题

在Course表中添加一个平均成绩avg\_Grade字段（记录每门课程的平均成绩），创建一个触发器Grade\_modify，当SC表中的某学生的成绩发生变化时，则Course表中的平均成绩也能及时相应的发生改变。

##### 第三题SQL

IF EXISTS (SELECT \*

FROM sysobjects

WHERE name = 'Grade\_modify' AND type = 'TR')

BEGIN

DROP TRIGGER Grade\_modify;

PRINT '已删除！'

END

PRINT '可以正常建立触发器！'

GO

CREATE TRIGGER Grade\_modify

ON SC

after INSERT, DELETE, UPDATE

AS

BEGIN TRANSACTION

IF UPDATE(Grade)

BEGIN

UPDATE Course

SET avg\_grade = (SELECT SUM(Grade)/COUNT(\*)

FROM SC

WHERE Course.Cno = SC.Cno)

WHERE Cno IN (SELECT Cno FROM INSERTED)

END

ELSE

BEGIN

UPDATE Course

SET avg\_grade = (SELECT SUM(Grade)/COUNT(\*)

FROM SC

WHERE Course.Cno = SC.Cno)

WHERE Cno IN (SELECT Cno FROM DELETED)

END

COMMIT TRANSACTION

GO

##### 执行结果

##### 验证

SELECT \*

FROM Course;

INSERT

INTO SC(Sno, Cno, Grade)

VALUES ('20110111', '001', 100)

INSERT

INTO SC(Sno, Cno, Grade)

VALUES ('20110111', '002', 100)

INSERT

INTO SC(Sno, Cno, Grade)

VALUES ('20110111', '003', 100)

SELECT \*

FROM Course;

### 创建instead of 触发器并测试

#### 第一题

创建一视图Student\_view,包含学号、姓名、课程号、课程名、成绩等属性，在Student\_view上创建一个触发器Grade\_moidfy，当对Student\_view中的学生的成绩进行修改时，实际修改的是sc中的相应记录。

##### 第一题SQL语句

IF EXISTS (SELECT \*

FROM sysobjects

WHERE name = 'student\_view' AND type = 'v')

BEGIN

DROP VIEW student\_view;

PRINT '已删除！'

END

PRINT '可以正常建立视图！'

GO

CREATE VIEW student\_view

AS

SELECT Student.Sno, Student.Sname, Course.Cno, Course.Cname, SC.Grade

FROM SC JOIN Student ON SC.Sno = Student.Sno

JOIN Course ON SC.Cno = Course.Cno

WHERE SC.Sno = Student.Sno AND Course.Cno = SC.Cno

GO

IF EXISTS (SELECT \*

FROM sysobjects

WHERE name = 'grade\_modify' AND type = 'TR')

BEGIN

DROP TRIGGER grade\_modify;

PRINT '已删除！'

END

PRINT '可以正常建立触发器！'

GO

CREATE TRIGGER grade\_modify

ON student\_view

INSTEAD OF UPDATE

AS

BEGIN TRANSACTION

DECLARE @grade int, @cno VARCHAR(20), @sno VARCHAR(20)

SELECT @grade = Grade, @sno = Sno, @cno = Cno

FROM INSERTED

UPDATE student\_view

SET Grade = @grade

WHERE Sno = @sno AND Cno = @cno

COMMIT TRANSACTION

GO

#### 第二题

在SC表中插入一个getcredit字段（记录某学生，所选课程所获学分的情况），创建一个触发器ins\_credit，当更改（注：含插入时）SC表中的学生成绩时，如果新成绩大于等于60分，则该生可获得这门课的学分，且该学分须与Course表中的值一致；如果新成绩小于60分，则该生未能获得学分，修改值为0。

##### 第二题SQL语句

IF EXISTS (SELECT \*

FROM sysobjects

WHERE name = 'ins\_credit' AND type = 'TR')

BEGIN

DROP TRIGGER ins\_credit;

PRINT '已删除！'

END

PRINT '可以正常建立触发器！'

GO

CREATE TRIGGER ins\_credit

ON SC

INSTEAD OF UPDATE, DELETE, INSERT

AS

BEGIN TRANSACTION

DECLARE @cno VARCHAR(20), @sno VARCHAR(20), @grade TINYINT,

@cno\_old VARCHAR(20), @sno\_old VARCHAR(20)

IF UPDATE(Grade)

BEGIN

IF NOT EXISTS (SELECT \* FROM DELETED)

BEGIN

SELECT @cno = Cno, @sno = Sno, @grade = Grade

FROM INSERTED

INSERT

INTO SC(Sno, Cno, Grade)

SELECT Sno, Cno, Grade FROM INSERTED

UPDATE Student

SET getcredit = (SELECT COUNT(\*)

FROM SC

WHERE Grade >= 60 AND Sno = @sno)

WHERE Sno = @sno

END

ELSE

BEGIN

SELECT @cno = Cno, @sno = Sno, @grade = Grade

FROM INSERTED

UPDATE Student

SET getcredit = (SELECT COUNT(\*)

FROM SC

WHERE Grade >= 60 AND Sno = @sno)

WHERE Sno = @sno

END

END

ELSE

BEGIN

SELECT @cno = Cno, @sno = Sno, @grade = Grade

FROM DELETED

DELETE

FROM SC

WHERE Sno = @sno AND Cno = @cno

UPDATE Student

SET getcredit = (SELECT COUNT(\*)

FROM SC

WHERE Grade >= 60 AND Sno = @sno)

WHERE Sno = @sno

END

COMMIT TRANSACTION

GO

##### 运行结果

### 使用T-SQL语句管理和维护

#### 第一题

用系统存储过程sp\_helptrigger查看触发器Grade\_modify的相关信息

##### SQL语句

sp\_helptrigger SC

GO

##### 运行结果

#### 第二题

使用系统存储过程sp\_helptext查看触发器Grade\_modify中的定义内容。

##### SQL语句

sp\_helptext grade\_modify

GO

##### 运行结果

#### 第三题

使用select语句查看触发器Grade\_modify的定义内容。

##### SQL语句

SELECT text

FROM syscomments

WHERE text LIKE '%grade\_modify%';

GO

##### 运行结果

#### 第四题

用系统存储过程sp\_depends查看触发器Grade\_modify的相关性。

##### SQL语句

sp\_depends grade\_modify

GO

##### 运行结果

#### 第五题

将sc\_insert触发器改为instead of 触发器，实现的功能不变。

##### SQL语句

IF EXISTS (SELECT \*

FROM sysobjects

WHERE name = 'ins\_credit' AND type = 'TR')

DROP TRIGGER ins\_credit

GO

ALTER TRIGGER sc\_insert

ON SC

INSTEAD OF INSERT

AS

DECLARE @sno VARCHAR(20), @cno VARCHAR(20), @grade tinyint

SELECT @sno = Sno, @cno = Cno, @grade = Grade

FROM INSERTED

IF NOT EXISTS(SELECT \*

FROM Student

WHERE Sno = @sno)

BEGIN

ROLLBACK TRAN

RAISERROR('学号在Student表中不存在!', 16, 10)

END

IF NOT EXISTS(SELECT \*

FROM Course

WHERE Cno = @cno)

BEGIN

ROLLBACK TRAN

RAISERROR('课程号在Course中不存在!', 16, 10)

END

INSERT

INTO SC(Sno, Cno, Grade)

VALUES (@sno, @cno, @grade)

GO

##### 运行结果

#### 第六题

将触发器sc\_insert删除。

##### SQL语句

DROP TRIGGER SC\_insert

##### 运行结果

### 使用SQL Server Management Studio管理存储过程

#### 第一题

在SQL Server Management Studio中重新创建刚删除的触发器sc\_insert

##### 解答

点击新建触发器

然后再输入即可

#### 第二题

查看触发器sc\_insert 的内容。

##### 解答

双击触发器即可

#### 第三题

删除触发器sc\_insert

##### 解答

点击删除触发器即可