

河南工业大学实验报告

课 程 数据库系统原理及应用 实验名称 实验六 触发器实验

院 系 信息科学与工程学院 专业班级 物联网 1603

姓 名 郭治洪 学 号 201616070320

指导老师 许贺洋 日 期 2018. 10

一. 实验目的

1. 理解触发器的作用和工作机制；
2. 掌握某一具体 DBMS 的 SQL 编程语言，在前面创建的数据库基础上，定义 BEFORE 触发器和 AFTER 触发器，并验证设计的触发器是否起作用。

二. 实验内容

1. 触发器的设计；
2. 触发器正确性验证。

三. 实验要求、过程及结果

对实验二建立的 Student 表、Course 表、Teacher 表、Dept 表和 SC 表，定义如下触发器：

1. 设计一个触发器，实现如下功能：在 Student 中，当删除某一同学 S#时，该同学的所有选课也都要删除；
2. 假设 Student 表中某一学生要变更其主码 S#的值，如使其原来的 98030101 变更为 99030131，此时 sc 表中该同学已选课记录的 S#也需自动随其改变。设计一个触发器完成上述功能；
3. 设计一个触发器：当进行 Teacher 表更新元组时，使其工资只能升不能降；

4. 建立 CourSum(S#, Sname, SumCourse)表,其中属性 SumCourse 统计该同学已学习课程的门数,初始值为 0,以后每选修一门都要对其增 1。设计一个触发器自动完成上述功能;
5. 设计一个触发器,实现以下功能: Dept(D#, Dname, Dean)表中 Dean 对应的教师,一定是该系教师 Teacher(T#, Tname, D#, Salary)中工资最高的教师。

A 备份数据库

因为后续触发器对数据删除修改操作会直接影响第二次执行结果,因此为了安全起见应该备份数据库!!!

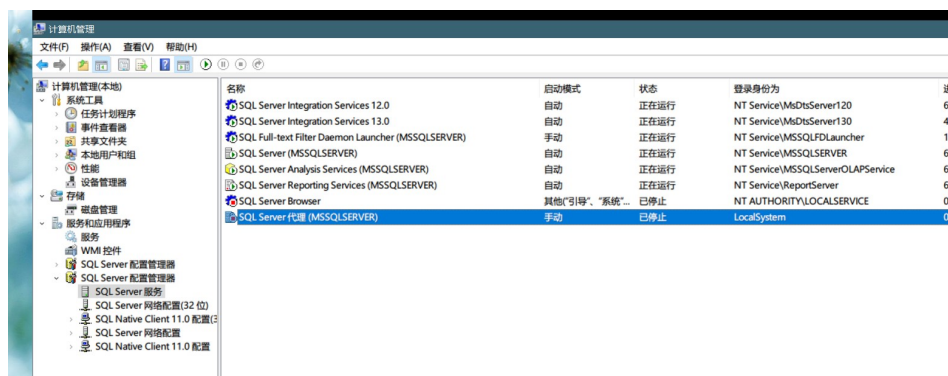
涉及增删改实验等操作数据一定要小心!!!

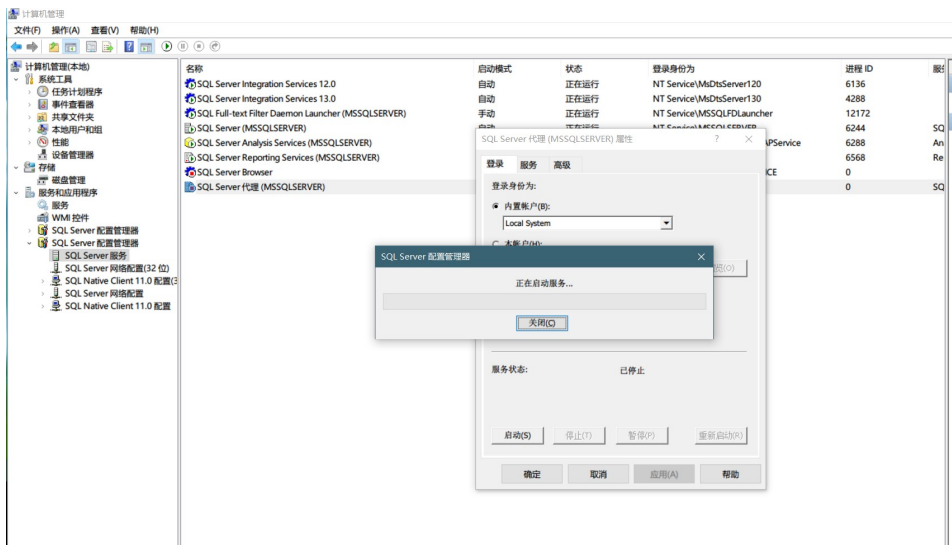
涉及增删改实验等操作数据一定要小心!!!

涉及增删改实验等操作数据一定要小心!!!

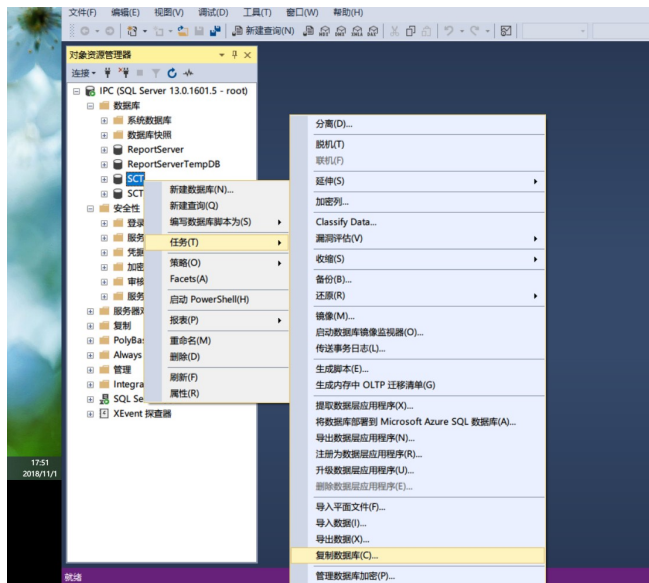
打开计算机管理-服务和应用程序-SQL SERVER 配置管理器
-SQL SERVER 服务找到 SQL SERVER 代理

登录身份选择内置账户 Local System, 手动启动服务。

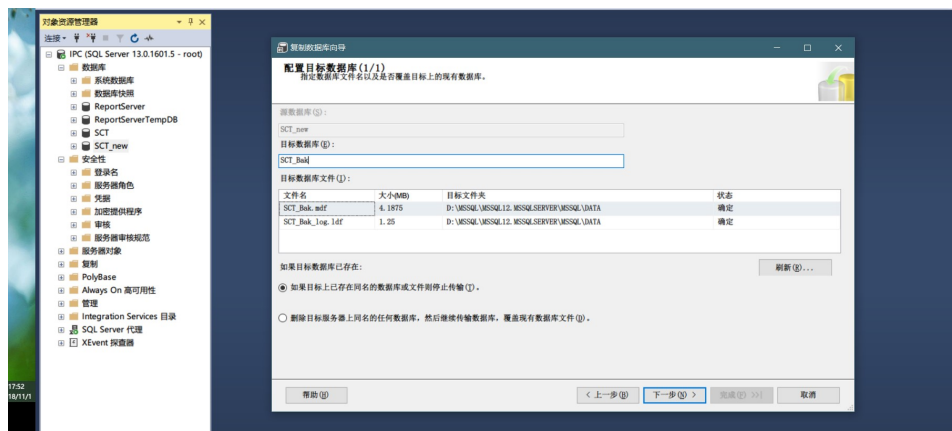




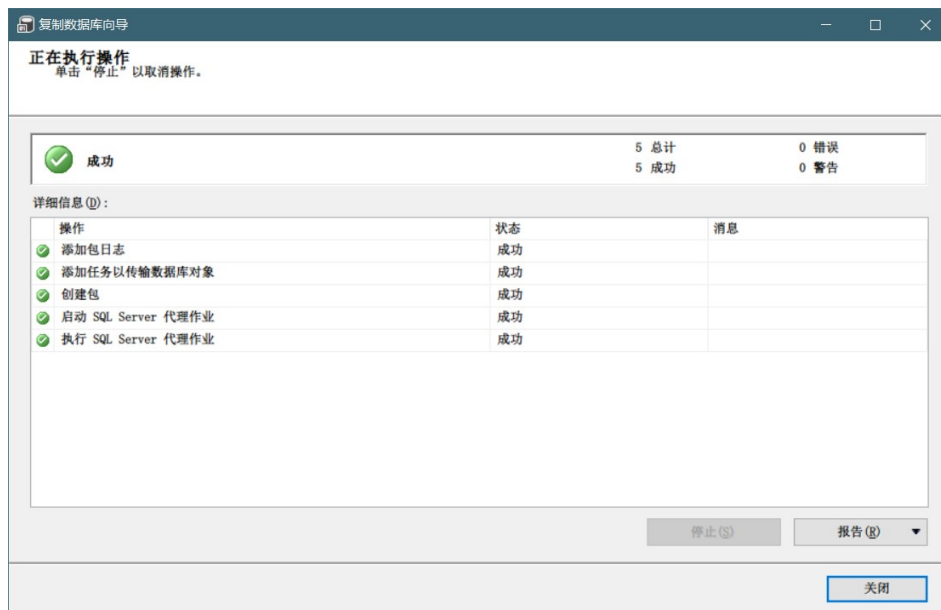
打开 SSMS，对备份数据库选择复制数据库



一切按照默认



等待备份成功:



B 实验源代码

```
USE SCT_new1; --使用复制之后的数据库防止数据变动
```

--参考:

<http://www.cnblogs.com/liushuijinger/archive/2012/06/10/2543941.html>

--1. 设计一个触发器，实现如下功能：在Student中，当删除某一同学S#时，该同学的所有选课也都要删除；

```
PRINT '创建触发器1'
```

```
GO
```

```
CREATE TRIGGER TRIGGER_S#_DEL
```

```
ON Student
```

```
INSTEAD OF DELETE --必须在删除之前执行触发器，否则将会Student删信息提示外键错误而无法删除！！
```

--因为外键的关系，因此我们需要在删除学生表之前删除课程表

--INSTEAD OF 是用触发器本身的SQL语句（触发器AS块中内容）代替需要执行的语句去执行

--INSTEAD OF只有一个，FOR/AFTER触发器可以有多个

--Inserted表：对于插入记录操作来说，插入表里存放的是要插入的数据；对于更新记录操作来说，插入表里存放的是要更新的记录。

--Deleted表：对于更新记录操作来说，删除表里存放的是被更新记录；对于删除记录操作来说，删除表里存入的是被删除的旧记录

```
/*
```

对于此题触发器，如果要通过学号删除某同学信息，应该先删除他的选课信息然后删除他的信息。

不然存在外键情况下是直接删除学生信息的！！

触发器执行过程：

有删除动作，触发器是触发删除动作执行。

然后系统会忽略外键关系把要删除信息记录到Deleted虚表，我们要通过这个虚表来删除所需要删除的数据。

只有这样删除动作才能执行，才不会提示外键冲突!!!

*/

AS

--触发器中代替删除动作执行的语句块，只要出现删除动作，触发器就会立刻生成虚表和执行下部分。

--删除动作将不会起效果，因为使用触发器该语句块替代，但是会将该动作作为虚表给触发器使用，搞清这些概念!!!

--参考:

<http://www.cnblogs.com/liushuijinger/archive/2012/06/10/2543941.html>

BEGIN

DELETE FROM SC WHERE S# IN (SELECT S# FROM deleted) --删除SC表中数据

DELETE FROM Student WHERE S# IN (SELECT S# FROM deleted) --删除

Student表中数据

END

GO

--测试触发器1

GO

PRINT '触发器1测试开始'

PRINT '源数据: '

PRINT 'Student表: '

SELECT * FROM Student

PRINT 'SC表: '

SELECT * FROM SC

PRINT '修改数据: '

DELETE FROM Student WHERE S#='98030102'

PRINT 'Student表: '

SELECT * FROM Student

PRINT 'SC表: '

SELECT * FROM SC

GO

--2. 假设Student表中某一学生要变更其主码S#的值，如使其原来的98030101变更为99030131，此时sc表中该同学已选课记录的S#也需自动随其改变。设计一个触发器完成上述功能;

PRINT '创建触发器2'

GO

CREATE TRIGGER TRIGGER_S#_UPDATE

ON Student

INSTEAD OF UPDATE --必须在更新之前执行触发器，否则可能将会在Student修改信息提示外键错误

AS

BEGIN

UPDATE SC SET SC.S#=I.S# FROM inserted AS I,deleted AS D WHERE

SC.S#=D.S#

UPDATE Student SET Student.S#=I.S# FROM inserted AS I,deleted AS D

```

WHERE Student.S#=D.S#
END
GO
PRINT '触发器2测试开始'
PRINT '源数据: '
PRINT 'Student表: '
SELECT * FROM Student
PRINT 'SC表: '
SELECT * FROM SC
PRINT '修改数据: '
UPDATE Student SET Student.S#='99030131' WHERE Student.S#='98030101'
PRINT 'Student表: '
SELECT * FROM Student
PRINT 'SC表: '
SELECT * FROM SC
GO
--4. 建立CourSum(S#, Sname, SumCourse)表,其中属性SumCourse统计该同学已学习课程的门数,初始值为0,以后每选修一门都要对其增1。设计一个触发器自动完成上述功能;
--触发器触发时会自动统计相关信息插入数据到CourSum表
PRINT '创建触发器4'
GO
CREATE TRIGGER TRIGGER_SUM_COURSE
ON SC
AFTER INSERT,UPDATE,DELETE
AS
BEGIN
    --直接删除老表
    IF EXISTS (SELECT * FROM dbo.SysObjects WHERE ID =
object_id(N'[CourSum]') AND OBJECTPROPERTY(ID, 'IsTable') = 1)
        DROP TABLE CourSum
    --生成新表
    SELECT DISTINCT Student.S#,Student.Sname,COUNT(SC.C#) AS
[SumCourse]
    INTO CourSum
    FROM Student,SC
    WHERE SC.S#=Student.S#
    GROUP BY Student.S#,Student.Sname
END
GO
PRINT '触发器4测试开始'
PRINT '源数据: '
PRINT 'SC表: '
SELECT * FROM SC
PRINT '修改数据: '

```

```

DELETE FROM SC WHERE SC.S#='98040203'
INSERT INTO SC VALUES('98040202','004',66);
PRINT 'SC表: '
SELECT * FROM SC
PRINT 'CourSum表: '
SELECT * FROM CourSum
GO

--3. 设计一个触发器: 当进行Teacher表更新元组时, 使其工资只能升不能降;
--5. 设计一个触发器, 实现以下功能: Dept(D#, Dname, Dean)表中Dean对应的教师,
一定是该系教师Teacher(T#, Tname, D#, Salary)中工资最高的教师。
-- 根据汪洋同学的翻译, 这个意思是给老师加薪不得高于该系系主任的工资
--INSTEAD OF只有一个, FOR/AFTER触发器可以有多个
PRINT '创建触发器3, 5'
GO
CREATE TRIGGER TRIGGER_SALARY_UPDATE
ON Teacher
INSTEAD OF UPDATE --这是在修改之前就执行触发器
AS
BEGIN
    DECLARE @diff REAL
    --进行修改差额比对
    SELECT @diff=I.Salary-D.Salary FROM deleted AS D,inserted AS I
    IF(@diff<0)
        RAISERROR('只允许上调工资',16,1)
    ELSE
        --得到需要修改的工资和工号以及系号
        DECLARE @NewSalary REAL
        DECLARE @NewT# CHAR(3)
        DECLARE @NewD# CHAR(2)
        SELECT @NewSalary=I.Salary,@NewT#=I.T#,@NewD#=I.D# FROM
inserted AS I
        DECLARE @DeptHeadSalary REAL
        DECLARE @DeptHeadT# CHAR(3)
        --得到该系主任的工资和工号
        SELECT @DeptHeadSalary=Teacher.Salary,@DeptHeadT#=Teacher.T#
        FROM Dept,Teacher WHERE Teacher.D#=Dept.D# AND
Teacher.Tname=Dept.Dean AND Dept.D#=@NewD#
        --该系主任不存在, 无法修改
        IF(@DeptHeadT# IS NULL)
            RAISERROR('上调工资之前请指定系主任',16,1)
        --1.系主任存在, 且调工资的是其他老师, 其他老师工资小于他的工资, 可以修改
        --2.或者修改系主任工资
        ELSE IF((@DeptHeadT# IS NOT NULL AND @DeptHeadT#!=@NewT# AND
@DeptHeadSalary>@NewSalary)

```

```

        OR(@DeptHeadT# IS NOT NULL AND @DeptHeadT#=@NewT#))
    UPDATE Teacher SET Teacher.Salary=I.Salary FROM
Teacher,inserted AS I WHERE I.T#=Teacher.T#
    ELSE IF(@DeptHeadT# IS NOT NULL AND @DeptHeadT#!=@NewT# AND
@DeptHeadSalary<@NewSalary)
        RAISERROR('上调工资不得高于最高工资',16,1)
END
GO
PRINT '触发器3，5测试开始'
PRINT '源数据: '
SELECT * FROM Teacher
PRINT '修改数据: '
PRINT '未指定系主任测试'
UPDATE Teacher SET Teacher.Salary=1800.00 WHERE T#='001'
SELECT * FROM Teacher
INSERT INTO Teacher VALUES('007','李五','03',2000.00);
INSERT INTO Teacher VALUES('008','李三','01',2000.00);
INSERT INTO Teacher VALUES('009','李四','02',2000.00);
INSERT INTO Teacher VALUES('010','李六','04',2000.00);
SELECT * FROM Teacher
PRINT '指定系主任测试其他老师'
PRINT '以下结果应该失败'
UPDATE Teacher SET Teacher.Salary=2800.00 WHERE T#='002'
SELECT * FROM Teacher
PRINT '以下结果应该成功'
UPDATE Teacher SET Teacher.Salary=1800.00 WHERE T#='002'
SELECT * FROM Teacher
PRINT '系主任测试'
PRINT '以下结果应该成功'
UPDATE Teacher SET Teacher.Salary=2800.00 WHERE T#='007'
SELECT * FROM Teacher
PRINT '以下结果应该失败'
UPDATE Teacher SET Teacher.Salary=2000.00 WHERE T#='007'
SELECT * FROM Teacher
GO
--删除触发器
PRINT '删除所有触发器'
GO
DROP TRIGGER TRIGGER_S#_DEL
DROP TRIGGER TRIGGER_S#_UPDATE
DROP TRIGGER TRIGGER_SALARY_UPDATE
DROP TRIGGER TRIGGER_SUM_COURSE
GO

```


C 实验结果

创建触发器 1

触发器 1 测试开始

源数据:

Student 表:

S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030102	张四	男	20	03	980301
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
98040203	王五	女	19	04	980402

(6 行受影响)

SC 表:

S#	C#	Score
98030101	001	92
98030101	002	85
98030101	003	88
98040202	002	90
98040202	003	80
98040202	001	55
98040203	003	56
98030102	001	54
98030102	002	85
98030102	003	48

(10 行受影响)

修改数据:

(3 行受影响)

(1 行受影响)

(1 行受影响)

Student 表:

S#	Sname	Ssex	Sage	D#	Sclass
----	-------	------	------	----	--------

98030101	张三	男	20	03	980301
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
98040203	王五	女	19	04	980402

(5 行受影响)

SC 表:

S#	C#	Score
98030101	001	92
98030101	002	85
98030101	003	88
98040202	002	90
98040202	003	80
98040202	001	55
98040203	003	56

(7 行受影响)

创建触发器 2

触发器 2 测试开始

源数据:

Student 表:

S#	Sname	Ssex	Sage	D#	Sclass
98030101	张三	男	20	03	980301
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
98040203	王五	女	19	04	980402

(5 行受影响)

SC 表:

S#	C#	Score
98030101	001	92
98030101	002	85
98030101	003	88
98040202	002	90
98040202	003	80
98040202	001	55

98040203 003 56

(7 行受影响)

修改数据:

(3 行受影响)

(1 行受影响)

(1 行受影响)

Student 表:

S#	Sname	Ssex	Sage	D#	Sclass
98030103	张五	男	19	03	980301
98040201	王三	男	20	04	980402
98040202	王四	男	21	04	980402
98040203	王五	女	19	04	980402
99030131	张三	男	20	03	980301

(5 行受影响)

SC 表:

S#	C#	Score
99030131	001	92
99030131	002	85
99030131	003	88
98040202	002	90
98040202	003	80
98040202	001	55
98040203	003	56

(7 行受影响)

创建触发器 4

触发器 4 测试开始

源数据:

SC 表:

S#	C#	Score
99030131	001	92
99030131	002	85
99030131	003	88

98040202 002 90
98040202 003 80
98040202 001 55
98040203 003 56

(7 行受影响)

修改数据：

(2 行受影响)

(1 行受影响)

(2 行受影响)

(1 行受影响)

SC 表：

S#	C#	Score
99030131	001	92
99030131	002	85
99030131	003	88
98040202	002	90
98040202	003	80
98040202	001	55
98040202	004	66

(7 行受影响)

CourSum 表：

S#	Sname	SumCourse
98040202	王四	4
99030131	张三	3

(2 行受影响)

创建触发器 3, 5

触发器 3, 5 测试开始

源数据：

T#	Tname	D#	Salary
001	赵三	01	1200
002	赵四	03	1400

003	赵五	03	1000
004	赵六	04	1100

(4 行受影响)

修改数据:

未指定系主任测试

消息 50000, 级别 16, 状态 1, 过程 TRIGGER_SALARY_UPDATE, 行 24 [批起始行 141]

上调工资之前请指定系主任

(1 行受影响)

T#	Tname	D#	Salary
001	赵三	01	1200
002	赵四	03	1400
003	赵五	03	1000
004	赵六	04	1100

(4 行受影响)

(1 行受影响)

(1 行受影响)

(1 行受影响)

(1 行受影响)

T#	Tname	D#	Salary
001	赵三	01	1200
002	赵四	03	1400
003	赵五	03	1000
004	赵六	04	1100
007	李五	03	2000
008	李三	01	2000
009	李四	02	2000
010	李六	04	2000

(8 行受影响)

指定系主任测试其他老师

以下结果应该失败

消息 50000, 级别 16, 状态 1, 过程 TRIGGER_SALARY_UPDATE, 行 31 [批起始行 141]
上调工资不得高于最高工资

(1 行受影响)

T#	Tname	D#	Salary
001	赵三	01	1200
002	赵四	03	1400
003	赵五	03	1000
004	赵六	04	1100
007	李五	03	2000
008	李三	01	2000
009	李四	02	2000
010	李六	04	2000

(8 行受影响)

以下结果应该成功

(1 行受影响)

(1 行受影响)

T#	Tname	D#	Salary
001	赵三	01	1200
002	赵四	03	1800
003	赵五	03	1000
004	赵六	04	1100
007	李五	03	2000
008	李三	01	2000
009	李四	02	2000
010	李六	04	2000

(8 行受影响)

系主任测试

以下结果应该成功

(1 行受影响)

(1 行受影响)

T#	Tname	D#	Salary
----	-------	----	--------

001	赵三	01	1200
002	赵四	03	1800
003	赵五	03	1000
004	赵六	04	1100
007	李五	03	2800
008	李三	01	2000
009	李四	02	2000
010	李六	04	2000

(8 行受影响)

以下结果应该失败

消息 50000, 级别 16, 状态 1, 过程 TRIGGER_SALARY_UPDATE, 行 10 [批起始行 141]

只允许上调工资

(1 行受影响)

(1 行受影响)

T#	Tname	D#	Salary
001	赵三	01	1200
002	赵四	03	1800
003	赵五	03	1000
004	赵六	04	1100
007	李五	03	2000
008	李三	01	2000
009	李四	02	2000
010	李六	04	2000

(8 行受影响)

删除所有触发器

四. 实验中的问题及心得

这玩意坑真的很多, 而且书上和老师教的根本没法用, 因此我们都自学了很多。还有感谢汪洋同学帮助, 我逐渐理清思路并且让我的程序更加完善。另外感谢老师的耐心等待, 给我时间让我 Debug, 我觉得我应该多加练习少写 Debug。继续加油!