数据库大作业实验报告

第五个分任务：

需要实现的功能：创建一个触发器，能根据每个班的学生变动情况自动增减班级表和系表的人数字段的值。

分析与设计：

1.触发器是一种特殊的存储过程，是一个在修改指定表中的数据执行的存储过程；触发器主要是通过事件进行触发而被执行的；触发器是建立在表或视图之上的。

2.由题意可知，我们需要实现的触发器是通过某个班学生的“增加”或“删除”这个事件来自动执行的；并且我们需要建立在表“student”之上。

3.综上，我们的设计思路为，定义一个触发器，它是建立在学生表之上的，根据”学生表”的学生信息的”插入(删除)”操作自动的对”班级表”和”系表”中的总人数“增1(减1)”。

具体实现：

CREATE TRIGGER mytrigger /\*定义一个触发器，触发器的名字为mytrigger\*/

on student /\*触发器mytrigger是建立在学生表student上的\*/

for insert,delete /\*当对student表进行插入删除操作时自动执行触发器\*/

AS /\*执行触发器时，执行下述操作\*/

update depart /\*插入时更新系表中系的总人数\*/

set depart.dsum=depart.dsum+1 /\*将我们增加的学生所在的系的总人数加1\*/

where depart.dnum=(

select inserted.dnum

from inserted

)

update class /\*插入时更新班级表的总人数\*/

set class.csum=class.csum+1/\*将我们增加的学生所在的班的总人数加1\*/

where class.cnum=(

select inserted.cnum

from inserted

)

update depart /\*删除时更新系表总人数\*/

set depart.dsum=depart.dsum-1/\*将我们增加的学生所在的系的总人数加1\*/

where depart.dnum=(

select deleted.dnum

from deleted

)

update class /\*删除时更新班表总人数\*/

set class.csum=class.csum-1/\*将我们增加的学生所在的班的总人数加1\*/

where class.cnum=(

select deleted.cnum

from deleted

)

问题及解决方法：

问题现象：在了解了触发器的基本原理之后，首先在触发器中添加了对学生表“增加”信息的相应操作，这部分执行成功之后，在加入“删除”操作后并没有得到相应结果。

解决过程：首先发现了删除时的操作语句写错，更改后仍然没有成功；再次结合触发器的原理，发现在定义自动执行触发器的操作时，忘记添加“for delete”；修改后可以显示预想结果。

检测触发器功能是否实现的过程：

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

截图存放处

第六个分任务：

需要实现的功能：创建一个存储过程，给定一个班的旧班号和新班号，把所有相关表中的此班的旧班号改为新班号，并使用输出参数返回此班的人数。

分析与设计：

1.存储过程是一种被存储在数据库内、允许用户声明变量、可由应用程序调用、根据条件执行的具有很强编程功能的数据库对象。包含两个过程：创建存储过程，执行存储过程。

2.由题意可知，我们需要在存储过程中声明三个变量：“新班号”、“旧班号”、“此班人数”，其中“此班人数”为输出参数；执行的语句为修改“班级表”和“学生表”中的班级号。

3.综上，我们创建的触发器为：定义三个变量，修改两个表中的班级号；执行触发器的过程为：用户自定义“旧班号”和“新班号”，存储过程修改后返回该班人数。

具体实现如下：

/\*假如在创建表时定义了UPDATE CASCADE,则按照下述方法定义存储过程\*/

CREATE PROCEDURE change\_cnum /\*创建一个存储过程，名字为change\_cnum\*/

@oldcnum CHAR(20),@newcnum CHAR(20),@cnumber INT OUTPUT

/\*定义变量旧班号：oldcnum；新班号：newcnum；输出班级人数：cnumber\*/

AS /\*存储过程执行下述操作\*/

UPDATE class /\*级联修改班表中的班级号\*/

SET class.cnum=@newcnum,@cnumber=csum

WHERE class.cnum=@oldcnum

DECLARE @cnumber INT /\*执行存储过程，声明返回值\*/

EXEC change\_cnum 'C101','CTEST',@cnumber OUTPUT /\*将班号C101改为CTEST\*/

PRINT STR(@cnumber) /\*输出班级人数\*/

/\*假如在创建表时没有定义UPDATE CASCADE,则按照下述方法定义存储过程\*/

CREATE PROCEDURE change\_cnum

@oldcnum CHAR(20),@newcnum CHAR(20),@cnumber INT OUTPUT

AS

UPDATE class /\*修改班表中的班级号\*/

SET class.cnum=@newcnum,@cnumber=csum

WHERE class.cnum=@oldcnum

UPDATE student /\*修改学生表中的班级号\*/

SET student.cnum=@newcnum

WHERE [student.cnum=@oldcnum](mailto:student.cnum=@oldcnum)

问题及解决方法：

1.问题现象：在定义存储过程后，执行时总在我们自定义变量的地方出现错误

解决方法：所有自定义的变量在程序中出现的时候都需要在其前边加上‘@‘符号，问题解决

2.问题现象：当我们使用已经定义“UPDATE CASCADE”的创建过程，在执行存储过程之后，打开所涉及的表“班表”和“系表”，并没有在原来的数量上增加或减少，刷新之后没有变化，但是我们明明在创表的时候对外码定义了“UPDATE CASCADE”

解决方法：当我们把已经打开的“学生表”、“班表”和“系表”关闭之后再打开，即能看到修改后的结果。

检测触发器功能是否实现的过程：

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

截图存放处

第七个分任务：

需要实现的功能：

编写脚本，使用游标完成如下功能：确定系表中人数字段的值与实际学生数是否相符。如果不相符，把人数字段的值改为实际数，并打印出此系的系号、系名、原人数、实际人数。

设计与分析：

1.游标提供了一种灵活操作从表中检索出来的数据集的方法，能把集合的操作转换为对单个记录的处理。就本质而言，游标实际上是一种能从包括多条数据记录的结果集中每次提取一条记录的机制。

2.由题意可知，我们需要将系表中的“系号”和“系人数”作为一个集合选出来，然后使用游标，一次对一条记录的系人数做以判断；判断是否等于“学生表”中系号为当前游标所指系号的总人数；然后做相应处理。

3.综上，我们定义的游标为：声明游标(从depart表中选择一个集合)；打开游标(将游标指向所选集合的第一个元组)；读取游标数据(将记录中的游标数据读入自定义变量)；用游标修改和删除当前数据(如果人数不相等，则修改)；关闭游标；释放游标。

具体实现：

DECLARE mycursor CURSOR SCROLL/\*定义游标，名为mycursor\*/

FOR SELECT dsum,dnum FROM depart/\*从系表中选择出系人数和系号\*/

OPEN mycursor/\*打开游标\*/

DECLARE @middle INT,@departnum CHAR(20),@temp INT

/\*定义变量，middle:用来记录游标的人数字段；departnum:用来记录游标的系号\*/

/\*temp:一个自定义中间变量\*/

FETCH NEXT FROM mycursor INTO @middle,@departnum

/\*读取游标数据\*/

WHILE(@@fetch\_status=0)

BEGIN

BEGIN

(SELECT @temp=COUNT(\*)/\*统计学生表中属于当前游标所指系号的学生人数\*/

FROM student

WHERE dnum=@departnum

)

IF (@middle!=@temp)/\*如果不相等，执行下述操作\*/

UPDATE depart/\*将系表中的人数字段更新\*/

SET dsum=@temp

WHERE Current of mycursor

END

FETCH NEXT FROM mycursor INTO @middle,@departnum

/\*将游标指向下一个记录并读取数据\*/

END

CLOSE mycursor/\*关闭游标\*/

DEALLOCATE mycursor/\*释放游标\*/

问题及解决方法：

1.问题现象：在语句编译过程中提到BEGIN出现错误

解决方法：在实际了解了“WHILE”循环中如何使用BEGIN...END以及“IF”中使用BEGIN....END的使用方法，将BEGIN和END的位置做以调整即可

2.问题现象：我们需要判断系表中的人数是否和实际学生表中该系人数相等时，需要SELECT出学生表中的该系人数；当两个人数值不相等时，还需要利用实际人数对系表中的人数进行修改。但是SELECT语句较为复杂，所以我们修改这个语句。

解决方法：在脚本前边定义一个中间变量@temp，来记录这个我们频发使用的值；但是要注意的是，脚本中出现temp的地方都要以@temp的形式出现。

检测触发器功能是否实现的过程：

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

截图存放处