

数据库实验lab8

曹广杰

15352015 数据科学与计算机

授课教师：刘玉葆

Content

数据库实验lab8

Content

实验目的

句法汇总

实验内容

触发器建立与insert操作的限制

对Update操作建立触发器

对更新操作作出限制

建立视图与触发器insert函数

实验目的

1. 通过实验加深对数据完整性的理解
2. 学会创建和使用触发器

句法汇总

1. 触发器建立
2. 对insert、update以及delete的触发器设置
3. 建立视图
4. 重写insert的触发器代替insert，实现多个关联表格的插入操作

实验内容

以下的代码依然是使用事务的外层：

```
1 set xact_abort on
2 begin transaction tmp
3 -- TODO
4 commit transaction tmp
```

触发器建立与insert操作的限制

建立一个在worker表上的触发器T4，要求插入记录的sage值必须比表中已记录的最大sage值大。

```

1 go
2 create trigger T4 on worker
3 for insert
4 as
5     if(
6         (select sage
7          from inserted)
8         <=
9         all(select sage
10            from inserted)
11     )
12     begin
13         print 'It is too young'
14         Rollback transaction
15     End

```

句法总结：

1. 创建触发器：

```

1 go
2 create trigger T4 on worker
3 for insert/delete
4 as -- 具体的条件以及触发步骤

```

不能对update创建触发器，

2. 对insert触发器的实现：

inserted作为插入信息的集合，可以当做table使用；在添加限定条件之后，如果触发了限定条件，则使用回滚操作。

3. 回滚操作： Rollback transaction

演示违反触发器的操作

上文中使用的限定条件是插入年龄信息不能小于任何一个元组的年龄，所以此处我们插入一个更小的年龄作为示范：

```

1 set xact_abort on
2 begin transaction tmp
3
4 insert into worker
5 values('0002', 'Wanggang', 'm', 13, 'magic')
6
7 select *
8 from worker
9
10 commit transaction tmp

```

该年龄小于数据库中的最大年龄，于是插入失败，数据库报错：

```
1 It is too young
2 消息 3902, 级别 16, 状态 1, 过程 T4, 第 15 行
3 COMMIT TRANSACTION 请求没有对应的 BEGIN TRANSACTION。
4 语句已终止。
```

第一句话是笔者设置的报错信息，其余的报错信息主要来自违反了触发器。

对Update操作建立触发器

建立一个在worker表上的触发器T5，要求当更新一个记录的时候，表中记录的sage值要比老记录的sage值大，因为一般工资级别只能升不能降。

```
1 go
2 create trigger T5 on worker
3 for update
4 as
5     if(
6         (select sage
7          from inserted)
8         <
9         (select sage
10          from deleted)
11     )
12     begin
13         print 'Why much younger!?'
14         Rollback transaction
15     End
```

句法分析：update的触发器使用不允许存在updated的集合，也就是说，update的操作实际上是先delete再insert，因此，在使用的时候，需要分别作为参照集进行限定。

添加违反该触发器的语句

```
1 set xact_abort on
2 begin transaction tmp
3
4 update worker
5 set sage = 7
6 where number = '0001'
7
8 select *
9 from worker
10
11 commit transaction tmp
```

由于我们之前限定，年龄不能负增长，因此添加7的任务应该后失败，不出意料的：

```
1 Why much younger!?'
2 消息 3902, 级别 16, 状态 1, 过程 T5, 第 15 行
3 COMMIT TRANSACTION 请求没有对应的 BEGIN TRANSACTION。
4 语句已终止。
```

对更新操作作出限制

为worker表建立触发器T6，禁止修改编号为00001的记录。

```
1 go
2 create trigger T6 on worker
3 for update
4 as
5     if(select number
6        from inserted) = '0001'
7     begin
8         print 'Cannot change it, Permission denied.'
9         Rollback transaction
10    End
```

只是需要限定更新的元组不能是编号为'0001'的，并没有什么新的语法。

建立视图与触发器insert函数

建立基于students和stu_card两个表的视图，创建INSTEADOF触发器使该视图支持更新操作，并演示更新操作。

先是建立基于students和stu_card的视图：

```
1 create view StuStucard as
2 select st.sid, st.sname, st.email, st.grade,
3        sc.card_id, sc.stu_id, sc.remained_money
4 from students st, stu_card sc
5 where sc.stu_id = st.sid
```

接下来是重载insert函数：

```

1  create trigger trg_ins_ssc on StuStucard
2  instead of insert
3  as
4  begin
5      set nocount on
6      if(not exists
7          (select students.sid
8           from students, inserted
9           where students.sid = inserted.stu_id)
10     )
11     begin
12         insert into students
13             select sid, sname, email, grade
14             from inserted
15         insert into stu_card
16             select card_id, stu_id, remained_money
17             from inserted
18     End
19     else
20         print 'Reputated tuple.Insert fail'
21 end

```

添加了触发器之后，只要有插入操作就会执行触发器，而不再是原来的insert操作。用当前的触发器代替语法insert实现对于关联的多个table的插入操作。