

一 . 实验目的:

创建示例数据库 AdventureWorks 中表 Sales.SalesOrderDetail 的 Update 触发器。

二 . 实验内容:

- 1) .使用修改记录中的 ProductID 和 ModifiedDate 从 Production.ProductListPriceHistory 表中查找相应的公开报价 ListPrice。即 ProductID 相同，修改记录中的 ModifiedDate 在 Production.ProductListPriceHistory 表中的 StartDate 与 EndDate 之间。
- 2) .使用 RAISEERROR 抛出错误信息。
- 3) .修改信息记录表 Production.ProductUpdateLog 的内容：记录编号、订单编号、订单明细编号、产品编号、产品的公开报价、修改前产品的单价、修改后产品的单价、修改者的登录名。
- 4) .给出触发器的源代码和简要的说明（可以在代码中使用注释进行说明）。
- 5) .设计触发器测试方案并给出测试的命令和结果，必要时可对测试结果进行分析。

三 . 实验过程:

SQLQuery1.sql | -...-PA01GNG\tc (63)) * X

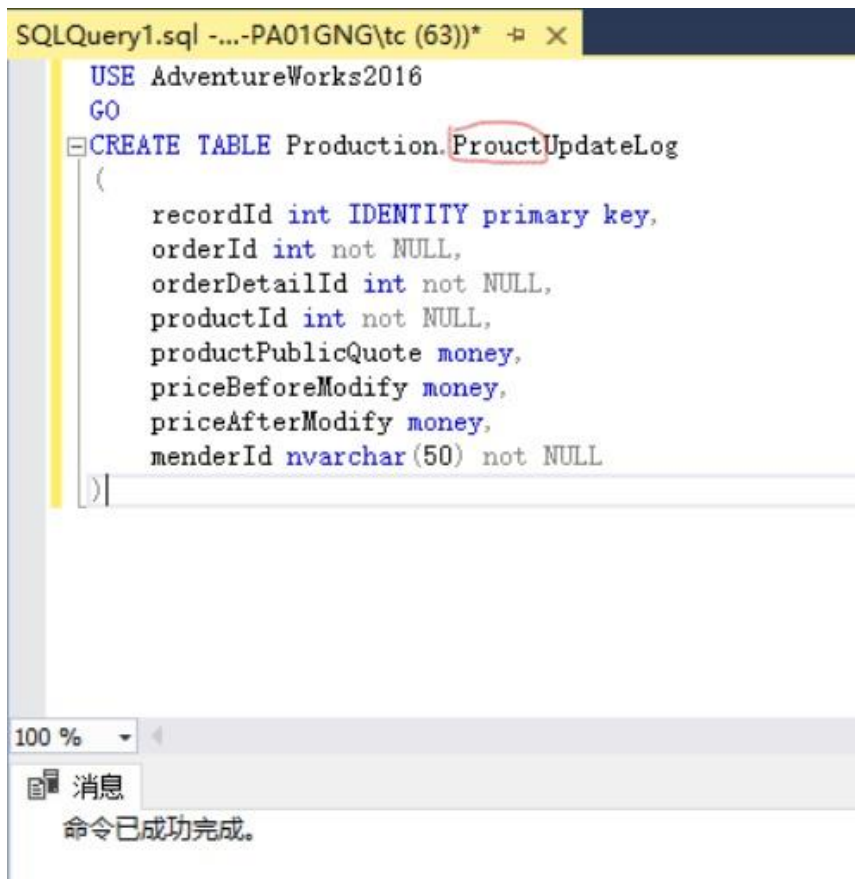
```
USE AdventureWorks2016
GO
SELECT * FROM Sales.SalesOrderDetail
GO
```

100 %

结果 消息

	SalesOrderID	SalesOrderDetailID	CarrierTrackingNumber	OrderQty	ProductID	SpecialOfferID	UnitPrice	UnitPr
1	43659	1	4911~403C~98	1	776	1	2024.994	0.00
2	43659	2	4911~403C~98	3	777	1	2024.994	0.00
3	43659	3	4911~403C~98	1	778	1	2024.994	0.00
4	43659	4	4911~403C~98	1	771	1	2039.994	0.00
5	43659	5	4911~403C~98	1	772	1	2039.994	0.00
6	43659	6	4911~403C~98	2	773	1	2039.994	0.00
7	43659	7	4911~403C~98	1	774	1	2039.994	0.00
8	43659	8	4911~403C~98	3	714	1	28.8404	0.00
9	43659	9	4911~403C~98	1	716	1	28.8404	0.00
10	43659	10	4911~403C~98	6	709	1	5.70	0.00
11	43659	11	4911~403C~98	2	712	1	5.1865	0.00

首先我们查询了表 Sales.SalesOrderDetail 的所有内容，以便后边的操作



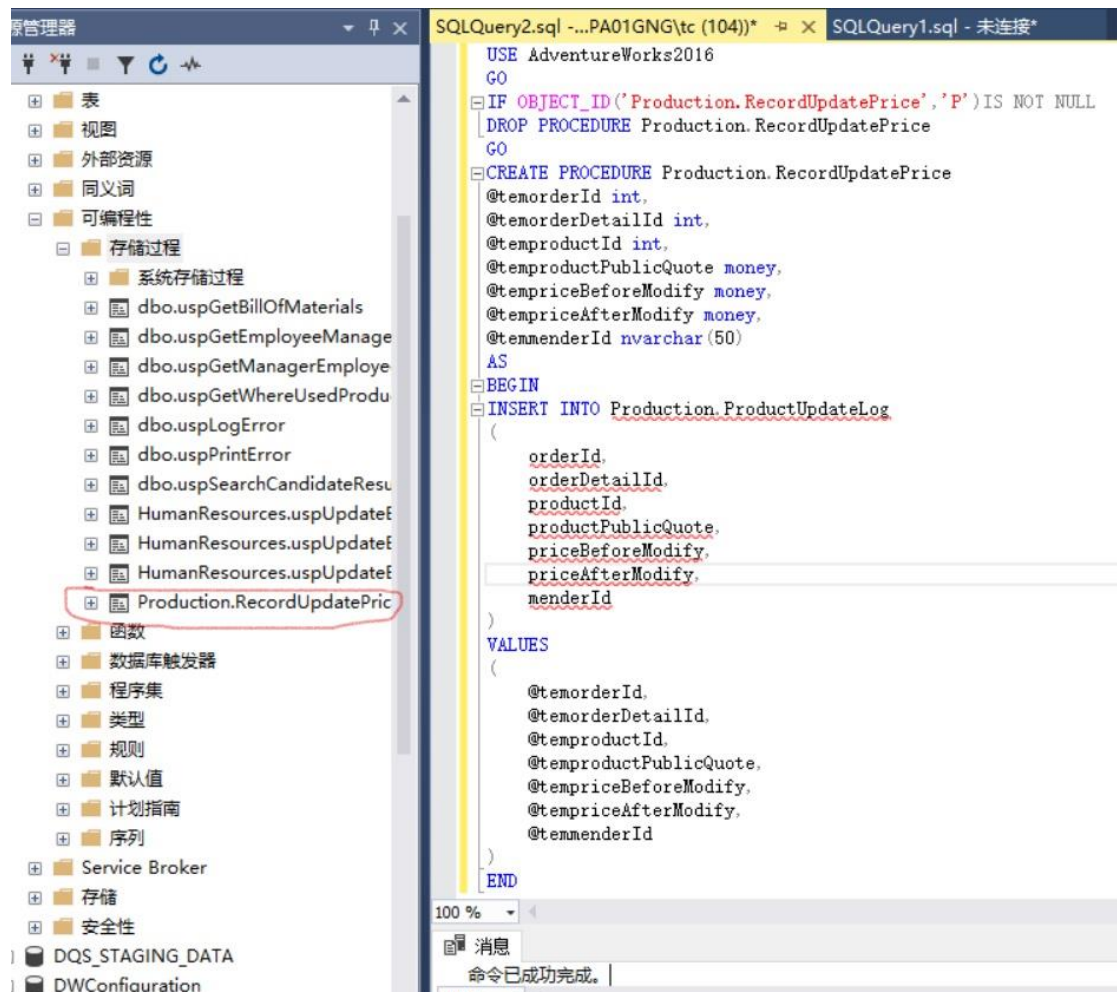
```
SQLQuery1.sql -...-PA01GNG\tc (63))*
USE AdventureWorks2016
GO
CREATE TABLE Production.ProuctUpdateLog
(
    recordId int IDENTITY primary key,
    orderId int not NULL,
    orderDetailId int not NULL,
    productId int not NULL,
    productPublicQuote money,
    priceBeforeModify money,
    priceAfterModify money,
    menderId nvarchar(50) not NULL
)
```

100 %

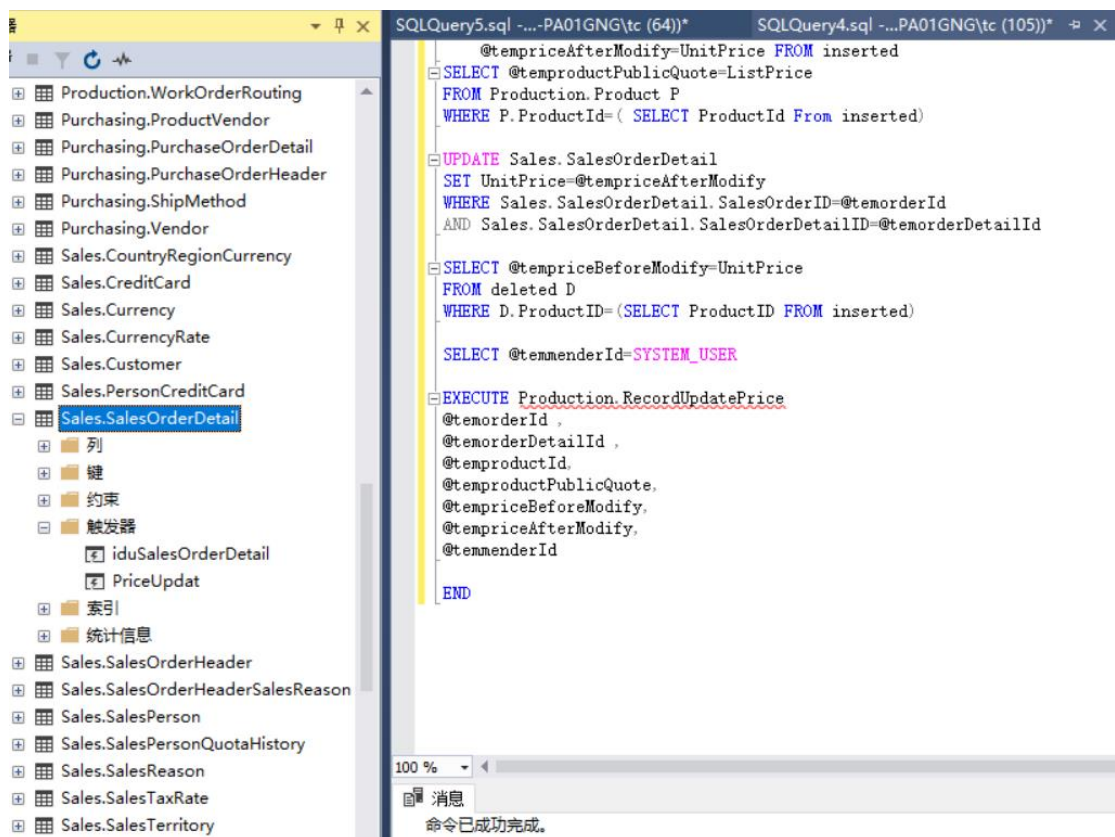
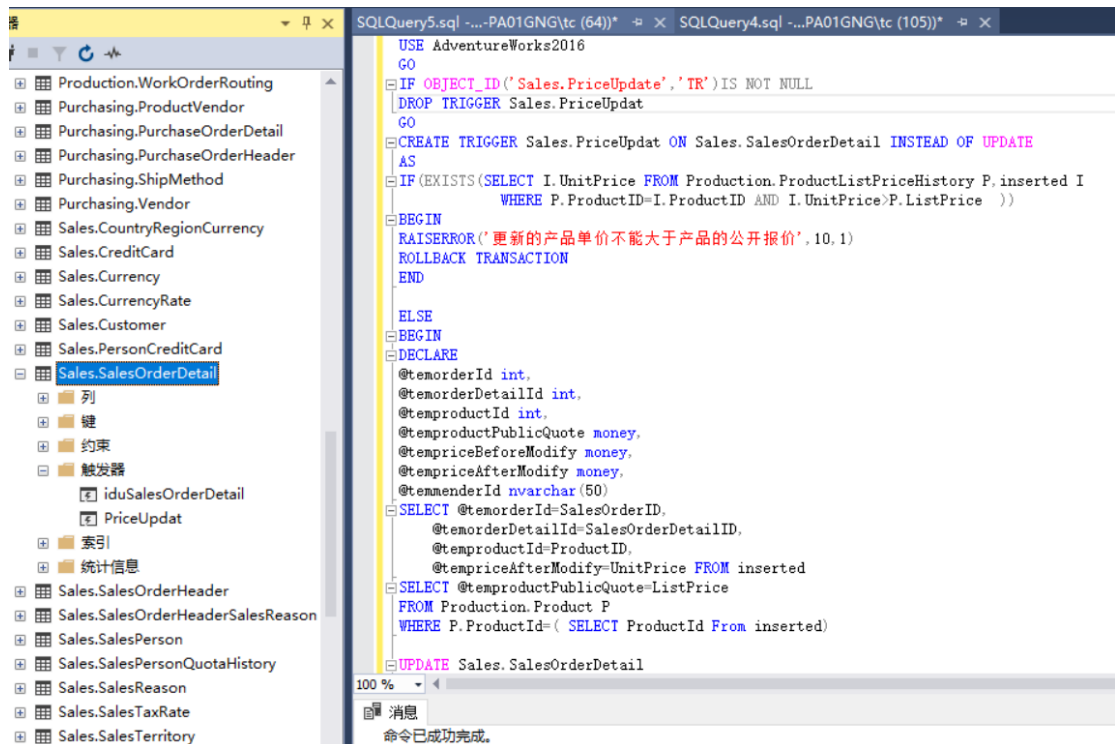
消息

命令已成功完成。

根据实验要求创建了表 Sales.SalesOrderDetail 的日志更新表 Production.ProductUpdateLog,创建时打错了表名(上图红线圈出的部分),后改为 Product,该表内容包括记录编号(recordId)、订单编号(orderId)、订单明细编号(orderDetailId)、产品编号(productId)、产品的公开报价(productPublicQuote)、修改前产品的单价(priceBeforeModify)、修改后产品的单价(priceAfterModify)、修改者的登录名(menderId),其中 recordId 作为主键且自动增长。



先判断是否有重名存储过程，若有便删除该存储过程，创建一个存储过程 Production.RecordUpdatePrice，该过程的参数为订单编号、订单明细编号、产品编号、产品的公开报价、修改前产品的单价、修改后产品的单价、修改者的登录名相对应的七个变量，变量名依次和上图中声明的变量相对应。创建成功后更新会在左侧的 AdventureWork2016/可编程性/存储过程中出现刚创建的存储过程（上图红线已圈出）。



判断是否有重名触发器，若有则删除，创建 instead of 触发器 Sales.PriceUpdat。判断修改的产品单价是否大于产品公开报价，若大于则进行回滚，调用存储过程 Production.RecordUpdatePrice。在调用该存储过程之前先申明七个变量用于存储参数（订单编号、订单明细编号、产品编号、产品的公开报价、修改前产品的单价、修改后产品的单价、修改者的登录名）。其中订单编号、订单明细编号、产品编号和修改后单

价取自 inserted 表, 产品公开报价取自 Production.ProductListPriceHistory, 修改前的产品价格取自 deleted 表, 修改者登录名通过系统内置函数 SYSTEM_USER 获得。

SQLQuery5.sql -...-PA01GNG\tc (64))* SQLQuery4.sql -...PA01GNG\tc (105))*
SELECT* from Sales.SalesOrderDetail

	SalesOrderID	SalesOrderDetailID	CarrierTrackingNumber	OrderQty	ProductID	SpecialOfferID	UnitPrice	UnitPrice
1	43659	1	4911~403C~98	1	776	1	2024.994	0.00
2	43659	2	4911~403C~98	3	777	1	2024.994	0.00
3	43659	3	4911~403C~98	1	778	1	2024.994	0.00
4	43659	4	4911~403C~98	1	771	1	2039.994	0.00
5	43659	5	4911~403C~98	1	772	1	2039.994	0.00
6	43659	6	4911~403C~98	2	773	1	2039.994	0.00
7	43659	7	4911~403C~98	1	774	1	2039.994	0.00
8	43659	8	4911~403C~98	3	714	1	28.8404	0.00
9	43659	9	4911~403C~98	1	716	1	28.8404	0.00
10	43659	10	4911~403C~98	6	709	1	5.70	0.00
11	43659	11	4911~403C~98	2	712	1	5.1865	0.00
12	43659	12	4911~403C~98	4	711	1	20.1865	0.00
13	43660	13	6431~4D57~63	1	762	1	419.4589	0.00
14	43660	14	6431~4D57~63	1	758	1	874.794	0.00
15	43661	15	4E0A~4F89~AE	1	745	1	809.76	0.00
16	43661	16	4E0A~4F89~AE	1	743	1	714.7043	0.00
17	43661	17	4E0A~4F89~AE	2	747	1	714.7043	0.00
18	43661	18	4E0A~4F89~AE	4	712	1	5.1865	0.00

SQLQuery5.sql -...-PA01GNG\tc (64))* SQLQuery4.sql -...PA01GNG\tc (105))*
select* from Production.ProductListPriceHistory

	ProductID	StartDate	EndDate	ListPrice	ModifiedDate
1	707	2011-05-31 00:00:00.000	2012-05-29 00:00:00.000	33.6442	2012-05-29 00:00:00.000
2	707	2012-05-30 00:00:00.000	2013-05-29 00:00:00.000	33.6442	2013-05-29 00:00:00.000
3	707	2013-05-30 00:00:00.000	NULL	34.99	2013-05-09 00:00:00.000
4	708	2011-05-31 00:00:00.000	2012-05-29 00:00:00.000	33.6442	2012-05-29 00:00:00.000
5	708	2012-05-30 00:00:00.000	2013-05-29 00:00:00.000	33.6442	2013-05-29 00:00:00.000
6	708	2013-05-30 00:00:00.000	NULL	34.99	2013-05-09 00:00:00.000
7	709	2011-05-31 00:00:00.000	2012-05-29 00:00:00.000	9.50	2012-05-29 00:00:00.000
8	710	2011-05-31 00:00:00.000	2012-05-29 00:00:00.000	9.50	2012-05-29 00:00:00.000
9	711	2011-05-31 00:00:00.000	2012-05-29 00:00:00.000	33.6442	2012-05-29 00:00:00.000
10	711	2012-05-30 00:00:00.000	2013-05-29 00:00:00.000	33.6442	2013-05-29 00:00:00.000
11	711	2013-05-30 00:00:00.000	NULL	34.99	2013-05-09 00:00:00.000
12	712	2011-05-31 00:00:00.000	2012-05-29 00:00:00.000	8.6442	2012-05-29 00:00:00.000
13	712	2012-05-30 00:00:00.000	2013-05-29 00:00:00.000	8.6442	2013-05-29 00:00:00.000
14	712	2013-05-30 00:00:00.000	NULL	8.99	2013-05-09 00:00:00.000
15	713	2011-05-31 00:00:00.000	2012-05-29 00:00:00.000	48.0673	2012-05-29 00:00:00.000
16	713	2012-05-30 00:00:00.000	2013-05-29 00:00:00.000	48.0673	2013-05-29 00:00:00.000
17	713	2013-05-30 00:00:00.000	NULL	49.99	2013-05-09 00:00:00.000
18	714	2011-05-31 00:00:00.000	2012-05-29 00:00:00.000	48.0673	2012-05-29 00:00:00.000

查看表 Sales.SalesOrderDetail 和表 Production.ProductListPriceHistory 的明细状况

```
SQLQuery5.sql -...-PA01GNG\tc (64))* X SQLQuery4.sql -...-PA01GNG\tc (105))*  
UPDATE Sales.SalesOrderDetail  
SET UnitPrice=3000  
WHERE Sales.SalesOrderDetail.SalesOrderDetailID=1
```

100 %

消息

(1 行受影响)

(1 行受影响)

(1 行受影响)

将订单明细编号为 1 的行的 UnitPrice 更改为 3000，结果执行成功

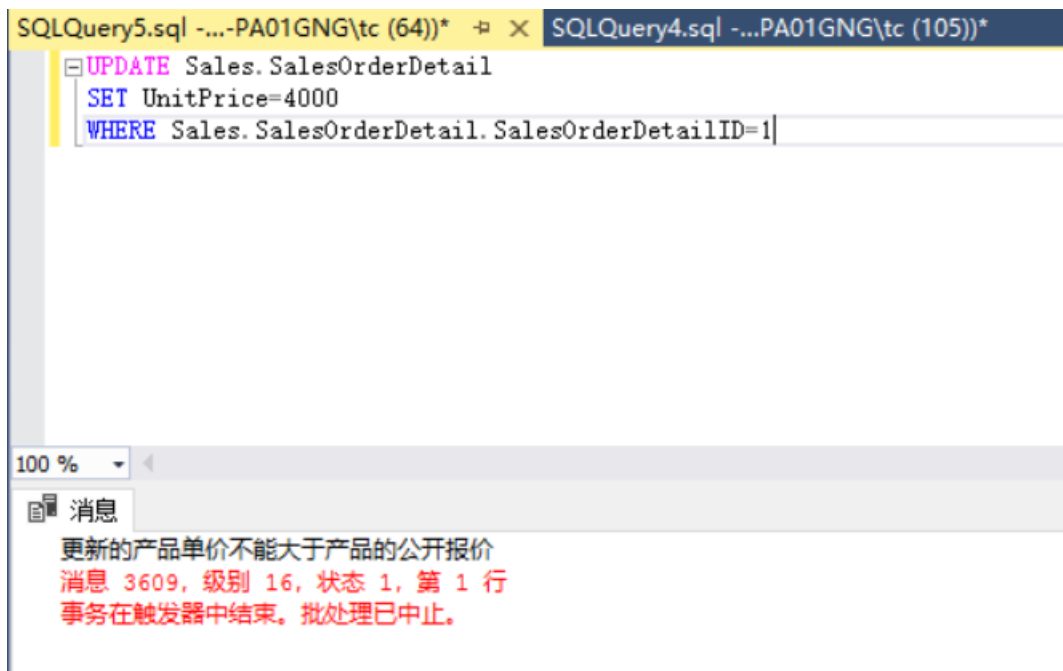
```
SQLQuery5.sql -...-PA01GNG\tc (64))* X SQLQuery4.sql -...-PA01GNG\tc (105))*  
SELECT * FROM Sales.SalesOrderDetail
```

100 %

结果 消息

	SalesOrderID	SalesOrderDetailID	CarrierTrackingNumber	OrderQty	ProductID	SpecialOfferID	UnitPrice	UnitPrice
1	43659	1	4911-403C-98	1	776	1	3000.00	0.00
2	43659	2	4911-403C-98	3	777	1	2024.994	0.00
3	43659	3	4911-403C-98	1	778	1	2024.994	0.00
4	43659	4	4911-403C-98	1	771	1	2039.994	0.00
5	43659	5	4911-403C-98	1	772	1	2039.994	0.00
6	43659	6	4911-403C-98	2	773	1	2039.994	0.00
7	43659	7	4911-403C-98	1	774	1	2039.994	0.00
8	43659	8	4911-403C-98	3	714	1	28.8404	0.00
9	43659	9	4911-403C-98	1	716	1	28.8404	0.00
10	43659	10	4911-403C-98	6	709	1	5.70	0.00
11	43659	11	4911-403C-98	2	712	1	5.1865	0.00
12	43659	12	4911-403C-98	4	711	1	20.1865	0.00
13	43660	13	6431-4D57-83	1	762	1	419.4589	0.00
14	43660	14	6431-4D57-83	1	758	1	874.794	0.00
15	43661	15	4EOA-4F89-AE	1	745	1	809.76	0.00
16	43661	16	4EOA-4F89-AE	1	743	1	714.7043	0.00
17	43661	17	4EOA-4F89-AE	2	747	1	714.7043	0.00
18	43661	18	4EOA-4F89-AE	4	712	1	5.1865	0.00

从上图中我们可以看出订单明细编号为 1 的行的 UnitPrice 已经更改为 3000



将订单明细编号为 1 的行的 UnitPrice 更改为 4000，结果执行失败，因为我们可以从前面对表 Production.ProductListPriceHistory 的查询结果知道 ProductID 为 1 的行对应的 ListPrice 为 3364 小于 4000，故而更新失败

The screenshot shows a SQL query window with the following text:

```
select* from Production.ProductUpdateLog
```

Below the query, a results pane displays the following data:

	recordId	orderId	orderDetailId	productId	productPublicQuote	priceBeforeModify	priceAfterModify	menderId
1	1	43659	1	776	3374.99	2024.994	3000.00	MicrosoftAccount
2	2	43659	1	776	3374.99	3000.00	2024.00	MicrosoftAccount
3	3	43659	1	776	3374.99	2024.00	3000.00	MicrosoftAccount
4	4	43659	1	776	3374.99	3000.00	2024.00	MicrosoftAccount
5	5	43659	1	776	3374.99	2024.00	3000.00	MicrosoftAccount
6	6	43659	1	776	3374.99	3000.00	3300.00	MicrosoftAccount
7	7	43659	1	776	3374.99	3300.00	3000.00	MicrosoftAccount

通过查询表 Production.ProductUpdateLog 可以看见之前所有成功的更新操作