



实 验 伍建霖 学 20337251 日 期: 2022.12.18
人: 号:

联系方式: _____

一. 实验目的

二. 实验内容和要求

三. 实验重点和难点

实验难点：自定义函数的参数传递方法。

四. 实验工具

MySQL、SQL Server、Navicat

五. 实验过程

■ 在 MySQL 环境下，执行以下操作


(1) 无参数的自定义函数


①定义一个自定义函数，更新所有订单的(含税折扣价)总价，并返回所有订单的总价之和。


/*该自定义函数与实验6.1中Proc_CalTotalPrice() 存储过程类似， 区别在于该自定义函数具有一个REAL类型的返回值。*/

```
set global log_bin_trust_function_creators=TRUE;
CREATE FUNCTION FUN_CalTotalPrice() RETURNS REAL
BEGIN
    DECLARE res REAL;
    UPDATE Orders
    SET totalprice=/*更新所有订单的含税折扣价总价*/
        (SELECT SUM(extendedprice*(1-discount) *(1+tax) )
        FROM Lineitem
        WHERE Orders.orderkey=Lineitem.orderkey) ;
    SELECT SUM(totalprice) INTO res/*计算所有订单的含税折扣价总价之和*/
    FROM Orders;
    RETURN res;/*返回总价之和*/
END;
```

验证截图如下：

▼  函数

 FUN_CalTotalPrice

 Proc_CalTotalPrice

②执行自定义函数 FUN_CalTotalPrice() 。

```
SELECT FUN_CalTotalPrice() ;
/*执行自定义函数，其返回值以结果集的方式返回和显示。*/
```

验证截图如下：

信息	摘要	结果 1	剖析	状态
FUN_CalTotalPrice()				
		114818773878.2785		

(2) 有参数的自定义函数

①定义一个自定义函数，更新并返回给定订单的总价。


```
CREATE FUNCTION FUN_CalTotalPrice4Order(p_okey INTEGER) RETURNS REAL
BEGIN
    DECLARE res REAL;
    UPDATE Orders
    SET totalprice=/*更新给定编号的订单的含税折扣价总价*/
        (SELECT SUM(extendedprice*(1-discount) *(1+tax) )
        FROM Lineitem
        WHERE Orders.orderkey=Lineitem.orderkey)
    WHERE orderkey=p_okey ;
```


```


SELECT totalprice INTO res/*查找给定订单的总价*/
FROM Orders
WHERE orderkey=p_okey;
RETURN res;/*返回给定订单的总价*/
END;


```


验证截图如下：

▼  函数


 FUN_CalTotalPrice

 FUN_CalTotalPrice4Order

 Proc_CalTotalPrice

 Proc_CalTotalPrice4Customer

 Proc_CalTotalPrice4Customer2

 TestIndex

② 执行自定义函数FUN_CalTotalPrice4Order()

```

/*更新并返回5365号订单的总价*/
select FUN_CalTotalPrice4Order(5365) ;

```

验证截图如下：

信息	摘要	结果 1	剖析	状态
		FUN_CalTotalPrice4Order(
		283522.350515		

(3) 有局部变量的自定义函数

① 定义一个自定义函数，计算并返回某个顾客的所有订单的总价。

```

CREATE FUNCTION FUN_CalTotalPrice4Customer(p_custname CHAR(25) ) RETURNS REAL
BEGIN
    DECLARE L_custkey INTEGER;/*局部变量L_custkey*/
    DECLARE res REAL;
    SELECT custkey INTO L_custkey/*查找给定客户名的客户编号*/
    FROM Customer
    WHERE name=trim(p_custname) ;
    SELECT CONCAT('custkey is ', L_custkey) into @user_prompt;
    /*提示客户编号信息*/
    /*更新指定客户编号的所有订单的含税折扣价总价*/
    UPDATE Orders
    SET totalprice=
        (SELECT SUM(extendedprice*(1-discount) *(1+tax) )
        FROM Lineitem
        WHERE Orders.orderkey=Lineitem.orderkey)
    WHERE Orders.custkey=L_custkey ;

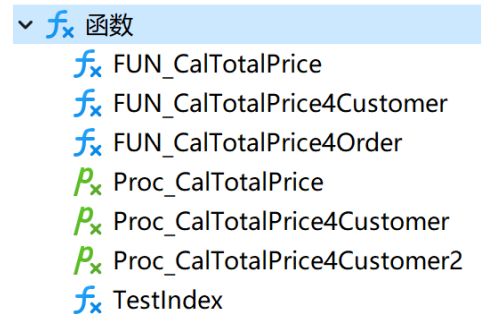
```

```

/*计算指定客户编号的所有订单的含税折扣价总价之和*/
SELECT SUM(totalprice) INTO res
FROM Orders
WHERE custkey=L_custkey;
RETURN res; /*返回总价之和*/
END;

```

验证截图如下：



②执行自定义函数FUN_CalTotalPrice4Customer()。

```
SELECT FUN_CalTotalPrice4Customer('符少静'), @user_prompt;
```

验证截图如下：

信息	摘要	结果 1	剖析	状态
FUN_CalTotalPrice4Custor@user_prompt				
▶ 16296283.2819 custkey is 56065				

(4) 有输出参数的自定义函数

①定义一个自定义函数, 计算并返回某个顾客的所有订单的总价。

```

/*该函数定义一个输入参数p_custname, 一个输出参数p_totalprice, 还有一个返回值类型REAL, 通过输出参数的定义, 该函数可以返回两个或者两个以上的值。该函数与FUN_CalTotalPrice4Customer()基本类似, 区别只在于该函数多了一个输出参数。*/
CREATE FUNCTION FUN_CalTotalPrice4Customer2(p_custname CHAR(25), out p_totalprice REAL) RETURNS REAL
BEGIN
    DECLARE L_custkey INTEGER;
    DECLARE res REAL;
    SELECT custkey INTO L_custkey
    FROM Customer
    WHERE name=trim(p_custname) ;
    SELECT CONCAT('custkey is ', L_custkey) into @user_prompt;
    UPDATE Orders
    SET totalprice=
        (SELECT SUM(extendedprice*(1-discount) *(1+tax) )
        FROM Lineitem

```

```

        WHERE Orders.orderkey=Lineitem.orderkey)
WHERE custkey=L_custkey;
SELECT SUM(totalprice) IN T0p_totalprice
FROM Orders
WHERE custkey=L_custkey;
set Res = p_totalprice;
RETURN res;
END;
```

验证截图如下：

信息

1064 - You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'out p_totalprice REAL) RETURNS REAL
BEGIN

根据出错信息，可知：在 MySQL 中，自定义函数不能拥有输出参数，这是因为自定义函数自身就是输出参数。

②执行自定义函数FUN_CalTotalPrice4Customer2()。

注：在MySQL中，以下语句失败。

```
SELECT FUN_CalTotalPrice4Customer2('符少静', @totalprice) ;
```

信息


1305 - FUNCTION tpch3.FUN_CalTotalPrice4Customer2 does not exist







(5) 删除自定义函数

删除自定义函数 FUN_CalTotalPrice4Order。

```
DROP FUNCTION FUN_CalTotalPrice4Order;
```

验证截图如下：

▼  函数

-  FUN_CalTotalPrice
-  FUN_CalTotalPrice4Customer
-  Proc_CalTotalPrice
-  Proc_CalTotalPrice4Customer
-  Proc_CalTotalPrice4Customer2
-  TestIndex

六. 与实验结果相关的文件

无

七. 思考题

(1) 试分析自定义函数与存储过程的区别与联系。

答:

1. 存储过程，功能强大，可以执行包括修改表等一系列数据库操作，也可以创建为 SQL Server 启动时自动运行的存储过程。

自定义函数，用户定义函数不能用于执行一组修改全局数据库状态的操作。

2. 存储过程，可以使用非确定函数。

自定义函数，不允许在用户定义函数主体中内置非确定函数。

3. 存储过程，可返回记录集。

自定义函数，可以返回表变量。

4. 存储过程，其返回值不能被直接引用。

自定义函数，其返回值可以被直接引用。

5. 存储过程，用 EXECUTE 语句执行。

自定义函数，在查询语句中调用

(2) 如何使得自定义函数可以返回多个值?如何利用?

答: 返回 table

```
Select * from func();
```

八. 实验总结

存储过程、用户自定义函数可以通过 CALL 和 SELECT 语句调用。需要说明的是:

①存储过程、用户自定义函数如果带有 OUT 或 IN OUT 参数， 则参数对应位置在调用时必须使用 NULL 或其他常量占位。运行所得是一个结果集， 结果集由一条或多条 RECORD 组成， 每条 RECORD 中字段的顺序是 OUT 或 IN OUT 参数对应的字段在前，最后返回 RETURN 语句对应的字段。

②SELECT 调用， 就是执行普通的 SELECT 语句。对于存储过程，不能和其他任何常量、函数、存储过程等一并构成表达式使用，只能单独作为一个表达式出现在 SELECT 语句中。对于用户自定义函数， 如果没有 OUT 或 IN OUT 参数，可以和其他常量、变量、对象名如字段名等组合成表达式使用。带有 OUT 或 IN OUT 参数的函数不可以参与表达式的计算。