

数据库实验七 综合实验-水果商店进阶二

原创

我超爱Debug 于 2021-05-05 15:03:17 发布 2046 收藏 1

分类专栏: [数据库](#) 文章标签: [数据库](#) [触发器](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/qq_43741419/article/details/116425425

版权



[数据库](#) 专栏收录该内容

10 篇文章 1 订阅

订阅专栏

文章目录

一、实验目的与要求:

二、实验内容:

1.在客户表customers中添加“VIP”字段,默认值0,根据已购买的累计金额自动更新等级,如累计100~199,一星VIP折扣9.9,200~499二星VIP折扣9.7,500~999三星VIP折扣9.5,1000~1999,四星VIP折扣9,后8.5折。

2.查询指定客户的“总优惠金额”,即累计每次购买时为客户优惠金额。

三、实验小结

一、实验目的与要求:

综合运用SQL语言相关知识如变量、游标、函数、触发器等解决实际问题。

二、实验内容:

设计并完成以下实验,要求附上源码(非截图),测试效果截图

1.在客户表customers中添加“VIP”字段,默认值0,根据已购买的累计金额自动更新等级,如累计100~199,一星VIP折扣9.9,200~499二星VIP折扣9.7,500~999三星VIP折扣9.5,1000~1999,四星VIP折扣9,后8.5折。

① 根据已购买情况修改客户表的VIP信息

源码:

```

先把vip字段添加，默认值为0
ALTER table customers
add VIP INT(11) not NULL DEFAULT(0);
再设计游标，更新等级
create PROCEDURE C_starts()
BEGIN
    DECLARE cc_id INT(11);
    DECLARE csum_price decimal(10,2);
    DECLARE done TINYINT DEFAULT 0;
    DECLARE C_starts cursor for select c_id,sum(original_price) FROM orders group by c_id;
    DECLARE CONTINUE HANDLER FOR NOT found set done=1;
open C_starts;
read_cur:LOOP
    FETCH C_starts INTO cc_id,csum_price;
    IF done then LEAVE read_cur;
    END IF;
    UPDATE customers_copy set VIP=
CASE
    WHEN csum_price BETWEEN 0 AND 100 THEN 0
    WHEN csum_price BETWEEN 100 AND 199 THEN 1
    WHEN csum_price BETWEEN 200 AND 499 THEN 2
    WHEN csum_price BETWEEN 500 AND 999 THEN 3
    WHEN csum_price BETWEEN 1000 AND 1999 THEN 4
    WHEN csum_price>1999 THEN 5
END
WHERE c_id=cc_id;
END loop read_cur;
CLOSE C_starts;
END

```

运行测试结果截图:

| c_id | c_name | c_address | c_city | c_zip | c_contact | c_email | VIP |
|-------|----------|---------------------|---------|--------|-----------|-------------------|-----|
| 10000 | 林秀秀 | 泉州 | 泉州 | 362000 | xiuxiulin | (Null) | 2 |
| 10001 | RedHook | 200 Street | Tianjin | 300000 | LiMing | LMing@163.com | 3 |
| 10002 | Stars | 333 Fromage Lane | Dalian | 116000 | Zhangbo | Jerry@hotmail.com | 2 |
| 10003 | Netbhood | 1 Sunny Place | QingDao | 266000 | LuoCong | (Null) | 0 |
| 10004 | JOTO | 829 Riverside Drive | Haikou | 570000 | YangShan | sam@hotmail.com | 2 |

| o_num | o_date | c_id | original_price | discount | pay |
|-------|---------------------|-------|----------------|----------|-------|
| 30001 | 2018-09-01 00:00:00 | 10001 | 560 | 1 | 560 |
| 30002 | 2018-09-12 00:00:00 | 10003 | 12.8 | 1 | 12.8 |
| 30003 | 2018-09-30 00:00:00 | 10004 | 320 | 1 | 320 |
| 30004 | 2018-10-03 00:00:00 | 10002 | 460 | 1 | 460 |
| 30005 | 2018-10-08 00:00:00 | 10001 | 218 | 1 | 218 |
| 30006 | 2020-04-15 10:30:07 | 10000 | 45 | 1 | 45 |
| 30007 | 2020-04-15 10:30:44 | 10000 | 23.3 | 1 | 23.3 |
| 30008 | 2020-04-15 10:30:52 | 10000 | 35.8 | 1 | 35.8 |
| 30009 | 2020-04-15 10:31:12 | 10000 | 156.1 | 1 | 156.1 |
| 30010 | 2020-04-15 10:31:30 | 10000 | 45.2 | 1 | 45.2 |

② 在订单表orders中插入新订单时自动计算并插入当前客户的折扣信息，注不修改原订单的折扣源码:

```

CREATE
TRIGGER `new_discount` BEFORE INSERT ON `orders`
FOR EACH ROW BEGIN
  DECLARE Oc_vip INT;
  SET Oc_vip=(SELECT vip FROM customers WHERE c_id=new.c_id);
  SET new.discount=(
  CASE
    WHEN Oc_vip=0 THEN 1
    WHEN Oc_vip=1 THEN 0.99
    WHEN Oc_vip=2 THEN 0.97
    WHEN Oc_vip=3 THEN 0.95
    WHEN Oc_vip=4 THEN 0.9
    WHEN Oc_vip=5 THEN 0.85
  END);
END;

```

运行测试结果截图:

现在orders中插入了新的o_num30011和30012

后来又在orderitems中插入数据

| o_num | o_date | c_id | original_price | discount | pay |
|-------|---------------------|-------|----------------|----------|-------|
| 30001 | 2018-09-01 00:00:00 | 10001 | 560 | 1 | 560 |
| 30002 | 2018-09-12 00:00:00 | 10003 | 12.8 | 1 | 12.8 |
| 30003 | 2018-09-30 00:00:00 | 10004 | 320 | 1 | 320 |
| 30004 | 2018-10-03 00:00:00 | 10002 | 460 | 1 | 460 |
| 30005 | 2018-10-08 00:00:00 | 10001 | 218 | 1 | 218 |
| 30006 | 2020-04-15 10:30:07 | 10000 | 45 | 1 | 45 |
| 30007 | 2020-04-15 10:30:44 | 10000 | 23.3 | 1 | 23.3 |
| 30008 | 2020-04-15 10:30:52 | 10000 | 35.8 | 1 | 35.8 |
| 30009 | 2020-04-15 10:31:12 | 10000 | 156.1 | 1 | 156.1 |
| 30010 | 2020-04-15 10:31:30 | 10000 | 45.2 | 1 | 45.2 |
| 30011 | 2020-05-27 11:09:00 | 10001 | 520 | 0.95 | 494 |
| 30012 | 2020-05-27 11:09:25 | 10000 | 520 | 0.97 | 504.4 |

https://blog.csdn.net/qq_43741419

| o_num | o_item | f_id | quantity | item_price |
|-------|--------|------|----------|------------|
| 30001 | 1 | a1 | 20 | 5.2 |
| 30001 | 2 | b2 | 20 | 6.4 |
| 30001 | 3 | bs1 | 20 | 11.2 |
| 30001 | 4 | a1 | 20 | 5.2 |
| 30002 | 1 | b3 | 2 | 6.4 |
| 30003 | 1 | c0 | 100 | 3.2 |
| 30004 | 1 | o2 | 50 | 9.2 |
| 30005 | 1 | c0 | 5 | 3.2 |
| 30005 | 2 | b1 | 10 | 10.2 |
| 30005 | 3 | a2 | 10 | 2.2 |
| 30005 | 4 | m1 | 5 | 15.6 |
| 30006 | 1 | lxx1 | 10 | 4.5 |
| 30007 | 2 | lxx2 | 10 | 2.33 |
| 30008 | 3 | lxx3 | 10 | 3.58 |
| 30009 | 2 | a1 | 20 | 5.2 |
| 30009 | 4 | lxx4 | 10 | 5.21 |
| 30011 | 1 | a1 | 100 | 5.2 |
| 30012 | 1 | a1 | 100 | 5.2 |

https://blog.csdn.net/qq_43741419

2. 查询指定客户的“总优惠金额”，即累计每次购买时为客户优惠金额。

源码：

```
SET GLOBAL log_bin_trust_function_creators=1;
CREATE FUNCTION All_discount(cid INT)
RETURNS DECIMAL(8,2)
RETURN (
  SELECT SUM(original_price-pay)
  FROM
  orders
  where c_id=cid
  GROUP BY c_id
)
```

运行测试结果截图：

| 定义 | 高级 | 注释 | 信息 | 结果 |
|-----------------------|----|----|----|----|
| `All_discount`(10001) | | | | |
| | | | | 26 |

| 定义 | 高级 | 注释 | 信息 | 结果 |
|-----------------------|----|----|----|------|
| `All_discount`(10000) | | | | |
| | | | | 15.6 |

三、实验小结

1. 实验中遇到的问题及解决过程

在“在订单表orders中插入新订单时自动计算并插入当前客户的折扣信息，注不修改原订单的折扣”最后测试的时候刚开始就直接在orders里面插入。后来觉得为了数据的统一性还是在orderitems中插入，然后就也可以用到上次实验的触发器。

2. 实验中产生的错误及原因分析

暂无

3. 实验体会和收获。

我想收回我上次实验中说的对触发器和游标有了深入的认识这句话，因为在这次实验中，其实有些方法中的细节刷新了我对触发器和游标的认识，我觉得吧，自己最多能说句，能用触发器和游标来解决问题，但是不断地学习总有不断地学习知识点的刷新，加油吧。