《数据库概论》实验二 高级SQL 实验报告

姓名 傅小龙 学号 191220029 联系方式 1830970417(QQ)

实验环境

操作系统: Windows 10

Mysql Workbench 8.0 CE

mysql-8.0.19-winx64

实验过程

任务1(1)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

drop procedure if exists gen\_pruduct\_sell\_list;

delimiter //

create procedure gen\_pruduct\_sell\_list(in pName Varchar(40))

begin

select C.customerName, O.orderNo, OD.quantity, OD.price

from Product P, Customer C, ordermaster O, orderdetail OD

where P.productNo = OD.productNo and C.customerNo = O.customerNo and O.orderNo = OD.orderNo and P.productName = pName

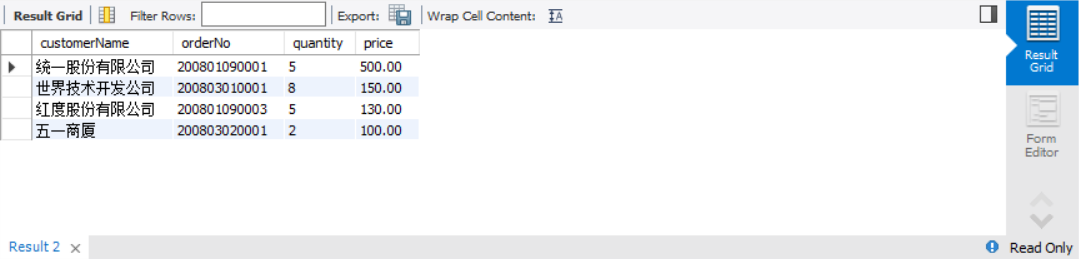
order by OD.price desc;

end;

//

call gen\_pruduct\_sell\_list('32M DRAM');

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



任务1(2)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

drop procedure if exists earlierHired\_employee;

delimiter //

create procedure earlierHired\_employee(in eNumber Char(8))

begin

select E.employeeNo, E.employeeName, E.gender, E.hireDate, E.department

from employee E

where E.hireDate < (

select E.hireDate

from employee E

where E.employeeNo = eNumber

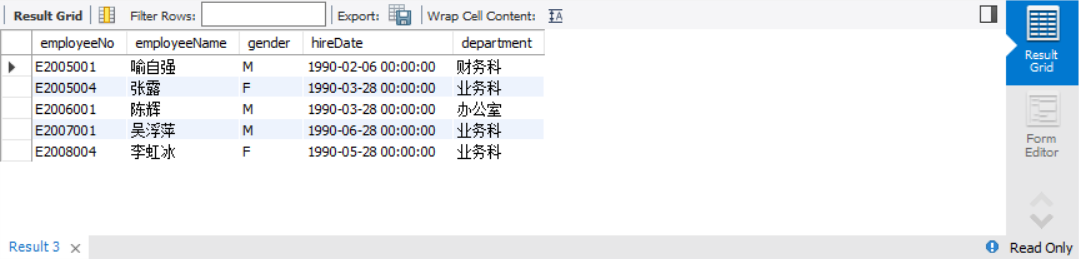
);

end

//

call earlierHired\_employee('E2008005');

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



任务2(1)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

drop function if exists avg\_order\_price;

delimiter //

create function avg\_order\_price(pName Varchar(40))

returns float

DETERMINISTIC

begin

declare avg\_price float;

select avg(OD.price) into avg\_price

from orderdetail OD, Product P

where OD.productNo = P.productNo and P.productName = pName

group by P.productName;

return avg\_price;

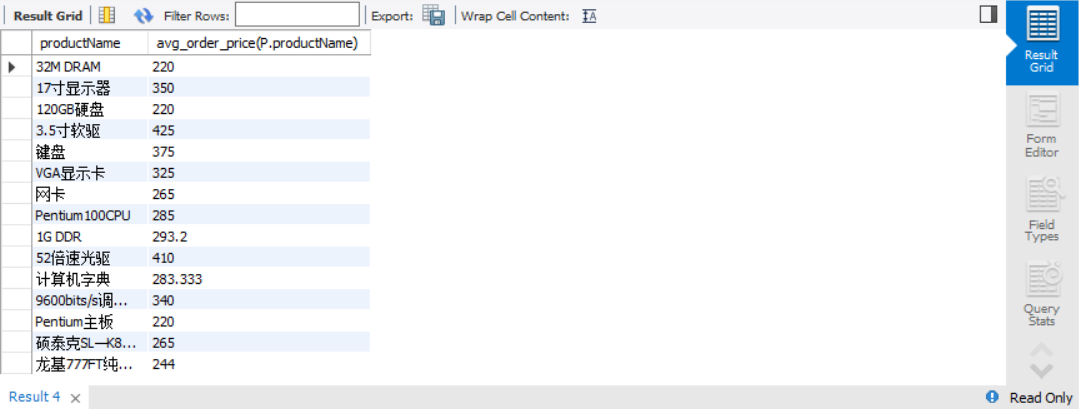
end;

//

select P.productName, avg\_order\_price(P.productName)

from Product P;

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



任务2(2)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

drop function if exists sum\_product\_sell;

delimiter //

create function sum\_product\_sell(pNo char(9))

returns integer

DETERMINISTIC

begin

declare sum\_sell integer;

select sum(OD.quantity) into sum\_sell

from orderdetail OD

where OD.productNo = pNo

group by OD.productNo;

return sum\_sell;

end;

//

select P.productNo, P.productName, sum\_product\_sell(P.productNo)

from Product P

where sum\_product\_sell(P.productNo) > 4;

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



任务3(1)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

drop trigger if exists pPrice\_insert;

delimiter //

create trigger pPrice\_insert before insert on product

for each row

begin

if new.productPrice > 1000 then

set new.productPrice = 1000;

end if;

end;

//

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

采用如下sql语句进行测试：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

insert into product

(productNo, productName, productClass, productPrice)

VALUES

('P20090001', 'test1', 'test', 100),

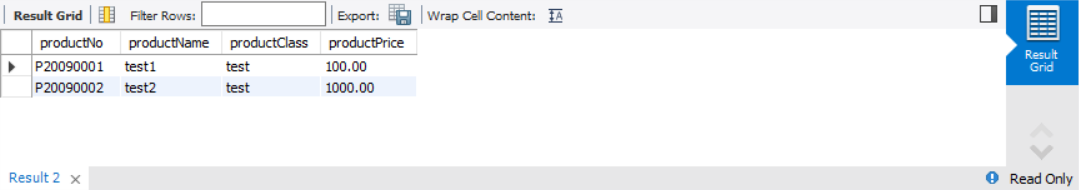
('P20090002', 'test2', 'test', 12000);

select \* from product where productNo like 'P2009%';

delete from product where productNo like 'P2009%';

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

可以看到编号为'P20090002'的产品价格被修改为了1000.



任务3(2)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

drop trigger if exists raise\_salary;

delimiter //

create trigger raise\_salary before insert on ordermaster

for each row

begin

declare old\_salary float;

set old\_salary = (

select salary

from employee

where employeeNo = new.employeeNo

);

if (

select hireDate

from employee

where employeeNo = new.employeeNo

) < date('1992-01-01') then

update employee set salary = old\_salary \* 1.08 where employeeNo = new.employeeNo;

else

update employee set salary = old\_salary \* 1.05 where employeeNo = new.employeeNo;

end if;

end

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

采用如下sql语句进行测试：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

insert into employee

(employeeNo, employeeName, gender, birthday, address, telephone, hireDate, department, headShip, salary)

VALUES

('E2021001', 'test1', 'M', '1968-01-06', '上海市福州路135号', NULL, '1992-02-28', '业务科', '职员', 1000),

('E2021002', 'test2', 'M', '1968-01-06', '上海市福州路135号', NULL, '1991-02-28', '业务科', '职员', 1000),

('E2021003', 'test2', 'M', '1968-01-06', '上海市福州路135号', NULL, '1991-02-28', '业务科', '职员', 1000);

insert into ordermaster

(orderNo, customerNo, employeeNo, orderDate, orderSum, invoiceNo)

VALUES

('P20210001', 'C20050001', 'E2021001', '2008-01-09', 0, 'I202100001'),

('P20210002', 'C20050001', 'E2021002', '2008-01-09', 0, 'I202100001');

select employeeNo, hireDate, salary

from employee

where employeeNo like 'E2021%';

delete from ordermaster where orderNo like '2021%';

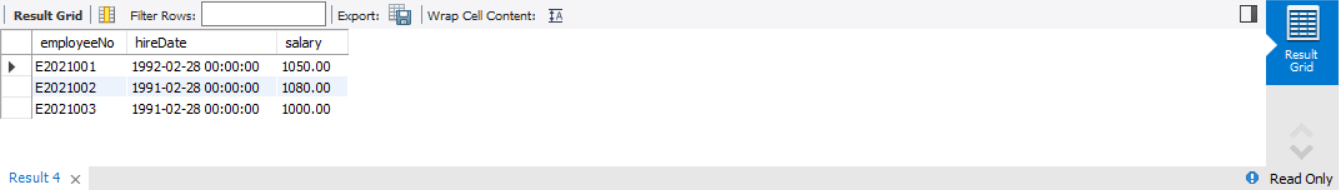
SET FOREIGN\_KEY\_CHECKS = 0;

delete from employee where employeeNo like 'E2021%';

SET FOREIGN\_KEY\_CHECKS = 1;

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

可以看到1992年前入职的的'E2021002'员工在完成一个订单后工资增长了8%, 1992年后入职的'E2021001'员工在完成一个订单后工资增长了5%, 而没有完成订单的'E2021003'员工工资没有增长.



任务4(1)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

任务3(2)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

任务3(2)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

任务3(2)：

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

-- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

实验中遇到的困难及解决办法

[详细说明你认为本次实验中比较困难的地方，也可以对实验设计提出建议]

参考文献及致谢

[如果你参考了任何书籍、网页，或与他人进行了讨论，请在此注明]