

# 数据库概论实验二:高级SQL 实验报告

田永铭 221900180

[tianym2022@smail.nju.edu.cn](mailto:tianym2022@smail.nju.edu.cn)

## 一、实验环境

操作系统版本: Windows 11

SQL软件版本: MySQL Workbench 8.0, MySQL Router 8.0, MySQL Server 8.0, MySQL Shell 8.0

python版本: python 3.8.1

## 二、实验过程

### 1.在SQL中实验

和第一次实验一样，建立好数据库链接，打开写代码的界面，导入数据，即可开始实验。

#### Q1.1

**注意：**由于有些代码一行太长了，所以我增加了换行，若要实际运行代码，请注意。同时，附件也有代码文件，能直接运行。

**这次的实验有几个注意点：**

编写语句需要用delimiter重新定义代码块的结束语，否则会出现一个存储函数或者存储过程还没结束系统就以为有分号就结束的问题。

同时，为了更加方便地检验代码，在每个代码前面加上drop，防止被之前自己写的错误的代码干扰。然后利用call或者SQL语句来进行检验。

以下分别展示执行代码和执行结果：

```
delimiter <>
drop procedure if exists q1_1 <>
create procedure q1_1(in product_name_to_search_for varchar(40))
begin
    select om.customerNo, c.customerName, om.orderNo,
           od.quantity, od.quantity*od.price as total_price
    from OrderMaster om, OrderDetail od, Customer c, Product p
    where p.productName = product_name_to_search_for and om.orderNo = od.orderNo
           and om.customerNo = c.customerNo and od.productNo = p.productNo
    order by total_price desc;

end
<>
delimiter ;

call q1_1("32M DRAM");
```

customerNo	customerName	orderNo	quantity	total_price
C20050001	统一股份有限公司	200801090001	5	2500.00
C20070002	世界技术开发公司	200803010001	8	1200.00
C20080001	红度股份有限公司	200801090003	5	650.00
C20050004	五一商厦	200803020001	2	200.00

## Q1.2

```
delimiter <>
drop procedure if exists q1_2 <>
create procedure q1_2(in employee_no char(8))
begin
    select e.employeeNo, e.employeeName, e.gender,
           e.hireDate, e.department
    from Employee e
    where e.hireDate < (
        select e2.hireDate
        from Employee e2
        where e2.employeeNo = employee_no
              and e2.department = e.department
    );
end
<>
delimiter ;

call q1_2("E2008005");
```

Result Grid   Filter Rows:   Export:   Wrap Cell Content:					
	employeeNo	employeeName	gender	hireDate	department
▶	E2005001	喻自强	M	1990-02-06 00:00:00	财务科

## Q2.1

```

delimiter <>
drop function if exists q2_1 <>
create function q2_1(product_name varchar(40))
    returns double deterministic
begin
    declare average double ;
    select sum(od.price*od.quantity)/sum(od.quantity) into average
        from OrderDetail od, Product p
        where p.productName = product_name and p.productNo = od.productNo;
    return average;
end
<>
delimiter ;

select distinct p.productName as productName, q2_1(p.productName) as average
from Product p;

```

	productName	average
▶	32M DRAM	227.5
	17寸显示器	425
	120GB硬盘	215
	3.5寸软驱	350
	键盘	375
	VGA显示卡	337.5
	网卡	258.571428571
	Pentium100CPU	290
	1G DDR	315.466666666
	52倍速光驱	362.5
	计算机字典	286.363636363
	9600bits/s调...	342
	Pentium主板	245.714285714
	硕泰克SL—K8...	265.714285714
	龙基777FT纯...	215.333333333

Q2.2

```
delimiter <>
drop function if exists q2_2 <>
create function q2_2(product_no char(9))
    returns integer deterministic
begin
    declare total integer ;
    select sum(od.quantity) into total
        from OrderDetail od
        where od.productNo = product_no;
    return total;
end
<>
delimiter ;

select p.productNo, p.productName,
        q2_2(p.productNo) as quantity
from Product p
where q2_2(p.productNo) > 4;
```

	productNo	productName	quantity
▶	P20050001	32M DRAM	20
	P20050003	120GB硬盘	8
	P20050004	3.5寸软驱	7
	P20050005	键盘	6
	P20060002	网卡	7
	P20060003	Pentium 100CPU	12
	P20070001	1G DDR	15
	P20070002	52倍速光驱	12
	P20070003	计算机字典	11
	P20070004	9600bits/s调制解调	5
	P20080001	Pentium主板	7
	P20080002	硕泰克SL—K8AN-R...	7

Q3.1

**注意**，在第三问中，我们需要将触发器写到orderdb中对应语句的前面设置，然后一起全部运行，再运行检验代码进行检验。

```
delimiter <>
drop trigger if exists q3_1 <>
create trigger q3_1
    before insert on Product
    for each row
begin
    if new.productPrice > 1000
        then set new.productPrice = 1000;
    end if ;
end
<>
delimiter ;

select p.productNo as productNo,p.productName as productName,
p.productPrice as productPrice
from Product p;
```

	productNo	productName	productPrice
▶	P20050001	32M DRAM	80.70
	P20050002	17寸显示器	700.00
	P20050003	120GB硬盘	300.00
	P20050004	3.5寸软驱	35.00
	P20050005	键盘	100.60
	P20060001	VGA显示卡	1000.00
	P20060002	网卡	66.00
	P20060003	Pentium100CPU	200.00
	P20070001	1G DDR	256.00
	P20070002	52倍速光驱	200.00
	P20070003	计算机字典	100.00
	P20070004	9600bits/s调...	320.00
	P20080001	Pentium主板	890.00
	P20080002	硕泰克SL—K8...	1000.00
	P20080003	龙基777FT纯...	900.00

Q3.2

```
delimiter <>
drop trigger if exists q3_2 <>
create trigger q3_2
    after insert on OrderMaster
    for each row
begin
    declare hire_year int;
    select YEAR(hireDate) into hire_year from Employee where employeeNo = new.employeeNo;
    if hire_year >= '1992'
        then
            update Employee set salary = salary * 1.05 where employeeNo = new.employeeNo;
        else
            update Employee set salary = salary * 1.08 where employeeNo = new.employeeNo;
    end if;
end
<>
delimiter ;

select e.employeeNo,e.employeeName,e.salary
from Employee e;
```



	employeeNo	employeeName	salary
▶	E2005001	喻自强	5800.00
	E2005002	E2005001	3023.31
	E2005003	张小娟	2866.50
	E2005004	张露	4100.00
	E2005005	张小东	1800.00
	E2006001	陈辉	4000.00
	E2006002	韩梅	2600.00
	E2006003	刘风	2500.00
	E2007001	吴浮萍	2500.00
	E2007002	高代鹏	2000.00
	E2008001	陈诗杰	3456.00
	E2008002	张良	2976.75
	E2008003	黄梅莹	3348.00
	E2008004	李虹冰	3672.00
	E2008005	张小梅	5000.00

## 2.利用高级语言在python中实验

首先需要学习如何在python中链接到orderdb数据库，上网查阅资料后我学会了这个操作，相关参考文献见最后部分。框架如下：

```
# 引入库
import pymysql

# 链接数据库
db = pymysql.connect(host='localhost',user='root',password='86161163',port=3306,db='orderdb')

# 尝试执行操作
try:
    # 获取游标
    cursor = db.cursor()
    sql = '''
        # 填写对应语句
        ...

    cursor.execute(sql)
    # 获取结果并打印
    result = cursor.fetchall()
    for row in result:
        print(row)

# 错误处理
except Exception:print("error!")

# 关闭游标和数据库
cursor.close()
db.close()
```

此后只展示核心代码块，略去框架。代码和运行结果依次如下：

### Q4.1

```
sql = '''
    select employeeNo, employeeName, salary
    from Employee
    order by salary DESC
    limit 20
    ...
```

```
( 'E2005001', '喻自强', Decimal('5800.00'))
( 'E2008005', '张小梅', Decimal('5000.00'))
( 'E2005004', '张露', Decimal('4100.00'))
( 'E2006001', '陈辉', Decimal('4000.00'))
( 'E2008004', '李虹冰', Decimal('3400.00'))
( 'E2008001', '陈诗杰', Decimal('3200.00'))
( 'E2008003', '黄梅莹', Decimal('3100.00'))
( 'E2008002', '张良', Decimal('2700.00'))
( 'E2005003', '张小娟', Decimal('2600.00'))
( 'E2006002', '韩梅', Decimal('2600.00'))
( 'E2006003', '刘风', Decimal('2500.00'))
( 'E2007001', '吴浮萍', Decimal('2500.00'))
( 'E2005002', '张小梅', Decimal('2400.00'))
( 'E2007002', '高代鹏', Decimal('2000.00'))
( 'E2005005', '张小东', Decimal('1800.00'))
```

**注意**,经检验,这个数据库一共就没有20个人,所以该结果正常,就是排名前20的薪水对应的员工数据。

## Q4.2

```
sql = '''
    insert Customer values('C20080002','泰康股份有限公司','010-5422685', '天津市','220501');
    ...

sql='''
    select * from Customer;
    ...
```

```
( 'C20050001', '统一股份有限公司', '022-3566021', '天津市', '220012')
( 'C20050002', '兴隆股份有限公司', '022-3562452', '天津市', '220301')
( 'C20050003', '上海生物研究室', '010-2121000', '北京市', '108001')
( 'C20050004', '五一商厦', '021-4532187', '上海市', '210100')
( 'C20060001', '大地商城', '010-1165152', '北京市', '100803')
( 'C20060002', '联合股份有限公司', '021-4568451', '上海市', '210100')
( 'C20070001', '南昌市电脑研制中心', '0791-4412152', '南昌市', '330046')
( 'C20070002', '世界技术开发公司', '021-4564512', '上海市', '210230')
( 'C20070003', '万事达股份有限公司', '022-4533141', '天津市', '220400')
( 'C20080001', '红度股份有限公司', '010-5421585', '北京市', '100800')
( 'C20080002', '泰康股份有限公司', '010-5422685', '天津市', '220501')
```



### Q4.3

```
sql = '''
    delete
    from Employee
    where salary > 5000;
    ...

sql='''
    select e.employeeNo,e.employeeName,e.salary
    from Employee e;
    ...
```

```
('E2005002', '张小梅', Decimal('2400.00'))
('E2005003', '张小娟', Decimal('2600.00'))
('E2005004', '张露', Decimal('4100.00'))
('E2005005', '张小东', Decimal('1800.00'))
('E2006001', '陈辉', Decimal('4000.00'))
('E2006002', '韩梅', Decimal('2600.00'))
('E2006003', '刘风', Decimal('2500.00'))
('E2007001', '吴浮萍', Decimal('2500.00'))
('E2007002', '高代鹏', Decimal('2000.00'))
('E2008001', '陈诗杰', Decimal('3200.00'))
('E2008002', '张良', Decimal('2700.00'))
('E2008003', '黄梅莹', Decimal('3100.00'))
('E2008004', '李虹冰', Decimal('3400.00'))
('E2008005', '张小梅', Decimal('5000.00'))
```

共14个

### Q4.4

```
sql = '''
    update
    Product p
    set p.productPrice = 0.5*p.productPrice
    where p.productPrice > 1000;
    ...

sql='''
    select p.productNo, p.productName, p.productPrice
    from Product p;
    ...
```

```
( 'P20050001', '32M DRAM', Decimal('80.70'))
( 'P20050002', '17寸显示器', Decimal('700.00'))
( 'P20050003', '120GB硬盘', Decimal('300.00'))
( 'P20050004', '3.5寸软驱', Decimal('35.00'))
( 'P20050005', '键盘', Decimal('100.60'))
( 'P20060001', 'VGA显示卡', Decimal('600.30'))
( 'P20060002', '网卡', Decimal('66.00'))
( 'P20060003', 'Pentium100CPU', Decimal('200.00'))
( 'P20070001', '1G DDR', Decimal('256.00'))
( 'P20070002', '52倍速光驱', Decimal('200.00'))
( 'P20070003', '计算机字典', Decimal('100.00'))
( 'P20070004', '9600bits/s调制解调', Decimal('320.00'))
( 'P20080001', 'Pentium主板', Decimal('890.00'))
( 'P20080002', '硕泰克SL-K8AN-RL主板', Decimal('550.00'))
( 'P20080003', '龙基777FT纯平显示器', Decimal('900.00'))
```

## Q5.1

```
sql = '''
    update
    Employee e
    set e.salary = 200 + e.salary
    where e.department = '业务科'
    ...

sql='''
    select e.employeeNo, e.employeeName, e.salary
    from Employee e;
    ...
```

```
( 'E2005001', '喻自强', Decimal('5800.00'))
( 'E2005002', '张小梅', Decimal('2600.00'))
( 'E2005003', '张小娟', Decimal('2800.00'))
( 'E2005004', '张露', Decimal('4300.00'))
( 'E2005005', '张小东', Decimal('2000.00'))
( 'E2006001', '陈辉', Decimal('4000.00'))
( 'E2006002', '韩梅', Decimal('2800.00'))
( 'E2006003', '刘凤', Decimal('2700.00'))
( 'E2007001', '吴浮萍', Decimal('2700.00'))
( 'E2007002', '高代鹏', Decimal('2000.00'))
( 'E2008001', '陈诗杰', Decimal('3200.00'))
( 'E2008002', '张良', Decimal('2900.00'))
( 'E2008003', '黄梅莹', Decimal('3300.00'))
( 'E2008004', '李虹冰', Decimal('3600.00'))
( 'E2008005', '张小梅', Decimal('5000.00'))
```

Q5.2

**注意**，我不太理解为什么作业突然在这里强调一个利用游标的要求，因为之前每个问题都利用了游标。我上网查了以下游标的使用，发现也可以一行一行地获取数据，我猜测可能是需要利用这个特点将数据格式化的更好，于是代码和结果如下：

```
sql='''
    select c.customerName, c.address, c.telephone
    from Customer c;
'''

cursor.execute(sql)

def format_row(row):
    return "{:<12}\t{:<12}\t{:<12}".format(*row)

# 输出表头
print("名称\t\t\t\t\t", "地址\t\t\t\t\t", "电话\t\t\t\t\t")

result = cursor.fetchone()
while result:
    # 输出数据行
    print(format_row(result))
    result = cursor.fetchone()
```

名称	地址	电话
统一股份有限公司	天津市	022-3566021
兴隆股份有限公司	天津市	022-3562452
上海生物研究室	北京市	010-2121000
五一商厦	上海市	021-4532187
大地商城	北京市	010-1165152
联合股份有限公司	上海市	021-4568451
南昌市电脑研制中心	南昌市	0791-4412152
世界技术开发公司	上海市	021-4564512
万事达股份有限公司	天津市	022-4533141
红度股份有限公司	北京市	010-5421585

### 三、实验中遇到的困难及解决办法

#### (1) 困难1：这次作业需要自主学习的内容比较多

这次作业需要自主学习的东西比较多，比如说存储函数和存储过程、触发器、高级语言链接SQL，这些都需要我查阅资料认真学习。不过，只要花了时间，相对也比较容易就能够学会。

#### (2) 困难2：对于题意的理解

这是一个相对比较大的项目作业，建立在一个比较完整的场景设置之中，理解起来需要一定时间。我选择结合E-R图片，自己也画一画、理一理关系，最终理解了作业的整体框架。

#### (3) 困难3：结束符问题

这在一开始我就已经进行过讨论，这是我遇到的第一个问题，也很好解决，通过delimiter来换结束符就行。

#### (4) 困难4：函数需要声明确定性

通过查看报错，我得知函数return需要加上deterministic来保证每次结果都一致，否则运行不了。

### 四、参考文献及致谢

本人所有作业独立完成，所有不会的地方通过自行查找资料完成，参考资料如下：

[Python编程连接SQL数据库](#)

[Mysql数据库 13.SQL语言 触发器](#)

老师提供的PPT（存储过程和存储函数）