请对bank数据库做以下查询，请勿使用“with”语句。

（1）找出余额最多的账户信息，包括账户编号、可用余额、开户日期和最后活跃日期。输出属性包括ACCOUNT\_ID, AVAIL\_BALANCE, OPEN\_DATE, LAST\_ACTIVITY\_DATE。

SELECT ACCOUNT\_ID, AVAIL\_BALANCE, OPEN\_DATE, LAST\_ACTIVITY\_DATE

from account

ORDER BY AVAIL\_BALANCE

DESC LIMIT 1;

SELECT ACCOUNT\_ID, AVAIL\_BALANCE, OPEN\_DATE, LAST\_ACTIVITY\_DATE

from account

where AVAIL\_BALANCE = (

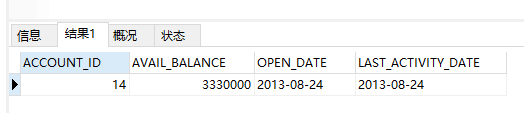
SELECT max(AVAIL\_BALANCE)

FROM account as a

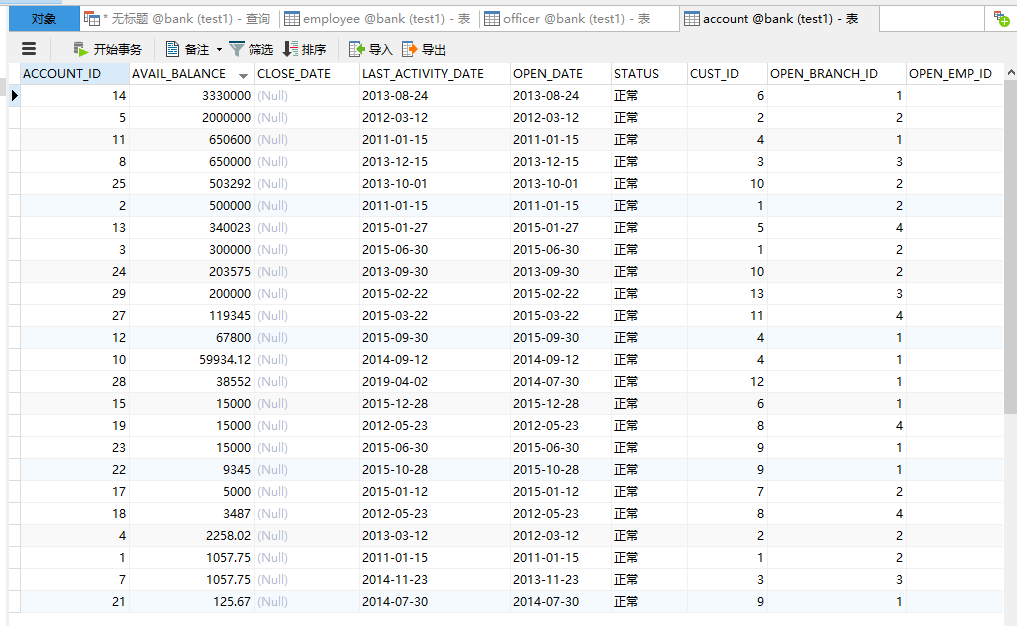
where ACCOUNT\_ID = a.ACCOUNT\_ID

);

结果如下：



将account排序后，符合上述结果



（2）找出所属部门与其上级领导所属部门不同的员工姓名，输出一个字段，属性命名为name（用CONCAT函数）。输出属性包括name。

SELECT concat(e1.LAST\_NAME ,'',e1.FIRST\_NAME) as full\_name

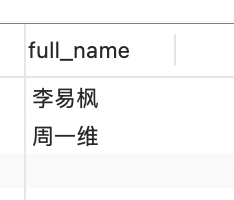
FROM employee as e1, employee as e2

where e1.SUPERIOR\_EMP\_ID = e2.EMP\_ID

AND

e1.DEPT\_ID != e2.DEPT\_ID;

结果如下:



（3）找出平均余额最多的支行名称（注：可能存在并列最多的情况）。输出属性包括NANE。

SELECT (sum(a.AVAIL\_BALANCE) / count(a.OPEN\_BRANCH\_ID)) as res, b.NAME

FROM account as a, branch as b

where a.OPEN\_BRANCH\_ID = b.BRANCH\_ID

GROUP BY b.`NAME`

ORDER BY res

desc LIMIT 1;

结果如下：



（4）找出身份证号以“3”开头的个人的（对私账户）和社会信用代码以“1”开头的客户（对公账户），将此属性命名为code；随后将其按照字符串从小到大排序，筛选出前3个。输出属性包括code。

（5）找出至少拥有两个账户的个人客户（individual表）的姓名（命名为name）和年龄（命名为age）。（注：可使用FROM\_DAYS、TO\_DAYS和NOW函数计算年龄）。输出属性包括name和age。

SELECT FLOOR(DATEDIFF(CURRENT\_DATE,i.BIRTH\_DATE)/365) as age,

concat(i.LAST\_NAME ,'', i.FIRST\_NAME)as full\_name

FROM individual as i, account as a

where i.CUST\_ID = a.CUST\_ID

GROUP BY i.CUST\_ID

HAVING count(a.ACCOUNT\_ID)>= 2;结果如下：

（6）找出工龄大于5年，且办理的执行交易数大于3次的员工信息，按其入职时间从先到后顺序输出。（注：可使用FROM\_DAYS、TO\_DAYS和NOW函数计算工龄）。输出属性包括EMP\_ID。

（7）查询至少购买了编号为“3”的客户所购买的所有产品的客户编号。输出属性包括CUST\_ID。

（8）查询购买了编号为“3”的客户购买的产品完全相同的客户编号。输出属性包括CUST\_ID。

（9）请对2015年的交易历史进行报表汇总，具体查询输出要求为：首先对交易月份（命名为month）和交易类型编码进行分组，接着对交易月份进行分组，最后输出2015年销售总额。输出属性包括month，TXN\_TYPE\_CD，sum（销售总额）。

（10）请对2015年的交易历史进行报表汇总，使用union集合操作实现cube汇总查询。输出属性包括month，TXN\_TYPE\_CD，sum（销售总额）。