

总体评价



王文涛 未通关

学号: U202215357

分班: 计算机科学2201班

截止前完成关卡: 17/19

最新完成关卡: 17/19

完成效率: --

课堂最高完成效率: --

通关时间	计时规则 ?	实训总耗时 ?	评测次数	查重扣分	补交扣分	最终成绩	总评
--	页面停留时长	4时 37分 4秒	46	--	--	32.0/38.0	良好

阶段成绩

关卡	任务名称	开启时间	代码修改 行数	评测次 数	完成时间	实训耗时	是否查看 答案	经验值	关卡得 分	调 分
1	金融应用场景介绍,查询客户主要信息	2024-04-24 15:33	3	1	2024-04-29 18:37	15分 2秒	否	100/100	1.00/1	1.00
2	邮箱为null的客户	2024-04-29 18:37	3	1	2024-04-29 18:41	3分 41秒	否	100/100	1.00/1	1.00
3	既买了保险又买了基金的客户	2024-04-29 18:41	12	3	2024-04-29 19:04	12分 42秒	否	300/300	1.00/1	1.00
4	办理了储蓄卡的客户信息	2024-04-29 19:04	4	1	2024-04-29 19:07	2分 47秒	否	300/300	1.00/1	1.00
5	每份金额在30000 ~ 50000之间的理财产品	2024-04-29 19:07	4	1	2024-04-29 19:14	7分 12秒	否	200/200	1.00/1	1.00
6	产品收益的个数	2024-04-29	4	1	2024-04-29	1时 31分	否	700/700	2.00/2	2.00

6	商品收益的众数	19:14	4	1	20:46	42秒	否	700/700	3.00/3	3.00
7	未购买任何理财产品的武汉居民	2024-04-29 20:46	9	3	2024-05-06 15:25	9分 53秒	否	300/300	1.00/1	1.00
8	持有两张信用卡的用户	2024-05-06 15:25	10	1	2024-05-06 15:31	6分 46秒	否	300/300	1.00/1	1.00
9	购买了货币型基金的客户信息	2024-05-06 15:32	8	4	2024-05-06 15:43	11分 33秒	否	500/500	2.00/2	2.00
10	投资总收益前三名的客户	2024-05-06 15:43	9	2	2024-05-06 15:58	14分 46秒	否	700/700	2.00/2	2.00
11	黄姓客户持卡数量	2024-05-06 15:58	5	4	2024-05-06 19:13	20分 6秒	否	500/500	2.00/2	2.00
12	客户理财、保险与基金投资总额	2024-05-06 19:14	19	4	2024-05-06 19:35	18分 46秒	否	900/900	3.00/3	3.00
13	客户总资产	2024-05-06 19:36	33	2	2024-05-06 19:42	5分 51秒	否	1000/1000	3.00/3	3.00
14	第N高问题	2024-05-06 19:42	7	3	2024-05-06 19:47	5分 27秒	否	800/800	2.00/2	2.00
15	基金收益两种方式排名	2024-05-06 19:47	33	8	2024-05-06 20:16	28分 30秒	否	1000/1000	3.00/3	3.00
16	持有完全相同基金组合的客户	2024-05-06 20:16	26	5	2024-05-06 20:45	17分 50秒	否	1000/1000	3.00/3	3.00
17	购买基金的高峰期	2024-05-06 20:45	--	0	--	--	否	0/1000	0.00/3	0.00

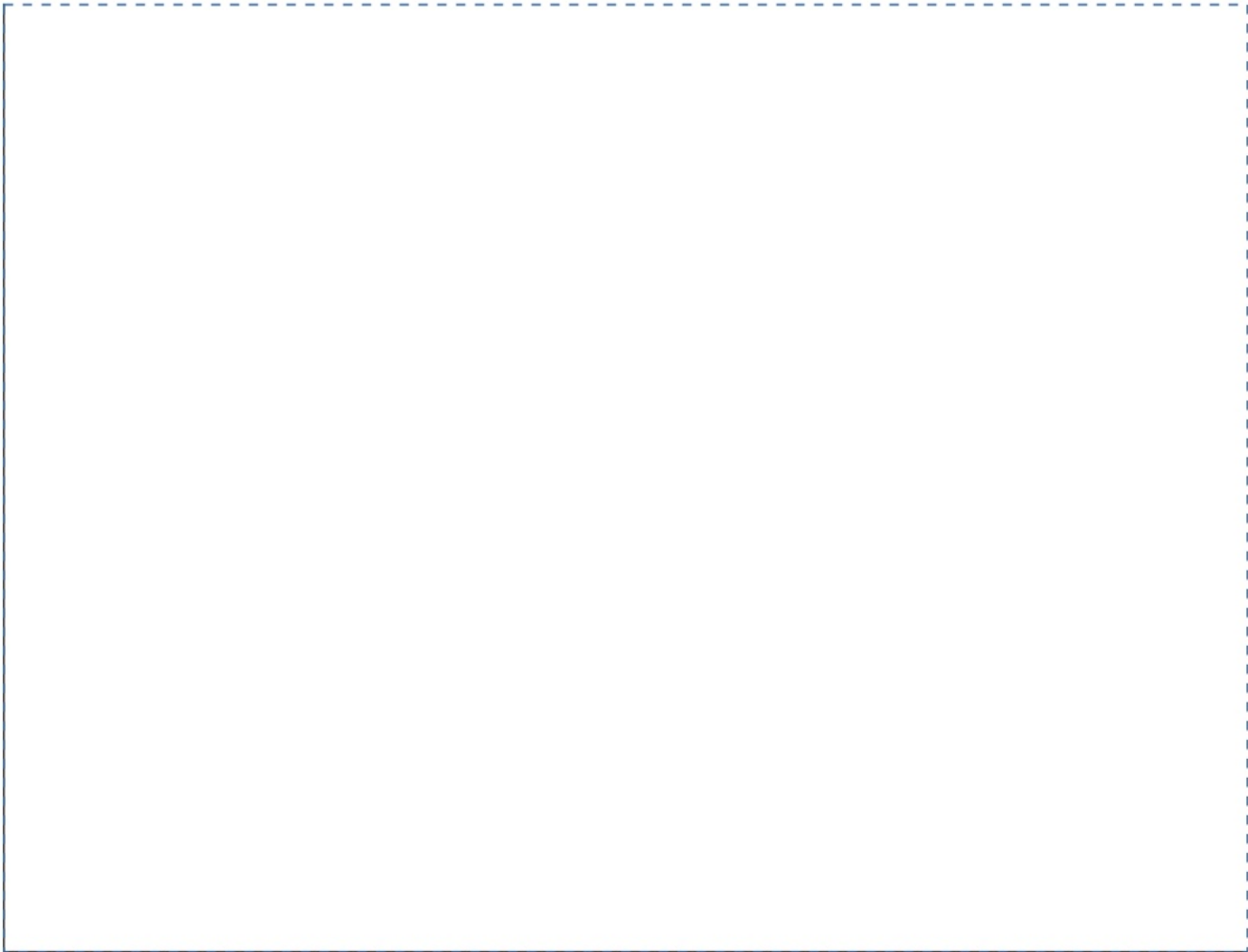
18	至少有一张信用卡余额超过5000元的客户 信用卡总余额	2024-05-10 15:03	13	2	2024-05-10 15:07	4分 30秒	否	500/500	2.00/2	2.00
19	以日历表格式显示每日基金购买总金额	2024-05-10 15:07	1	0	--	--	否	0/900	0.00/3	0.00

图形统计

学习效率

学习效率=log(实训总得分/实训总耗时)

学习效率分布区间



学生位置

能力值

姓名 王文涛

学号 U202215357

学习效率

当前排名

1个小圆点代表1个学生的学习效率

红色水滴表示当前学生的学习效率值

小圆点越高，学习效率越高

橙色横向箭头：中位值，表示整个课堂学生表现的平均值

姓名

王文涛

能力值 (实训获得经验值/实训评测次数)

姓名	王义涛
学号	U202215357
能力	

1个小圆点代表1个学生的能力值
红色水滴表示当前学生的能力值
小圆圈越大，评测次数越少



实训详情



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 3

src/test1/test1.sql

▼

```
1  -- 1) 查询所有客户的名称、手机号和邮箱信息。查询结果按照客户编号排序。
2  --      请用一条SQL语句实现该查询：
3
4
5  select c_name, c_phone, c_mail
6  from client
7  order by c_id;
8  /* end of your code */
9
10
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	评测通过	1.24	46.06	2024-04-29 18:37	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 3

src/test2/test2.sql

```
1  -- 2) 查询客户表(client)中邮箱信息为null的客户的编号、名称、身份证号、手机号。
2  -- 请用一条SQL语句实现该查询:
3  select c_id, c_name, c_id_card, c_phone
4  from client
5  where c_mail is null
6
7
8  /* end of your code */
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	评测通过	1.31	46.06	2024-04-29 18:41	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 12

src/test3/test3.sql

```
1  -- 3) 查询既买了保险又买了基金的客户的名称、邮箱和电话。结果依c_id排序
2  -- 请用一条SQL语句实现该查询:
3  select c_name, c_mail, c_phone
4  from client
5  where exists(
6      select 1
7      from property
8      where c_id=pro_c_id and pro_type=2)
9      and
10     exists(
11         select 1
12         from property
13         where c_id=pro_c_id and pro_type=3)
14  order by c_id
15
16
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	1.21	46	2024-04-29 19:03	查看代码
2	共有2组测试集，其中有2组测试结果不匹配	1.21	46	2024-04-29 19:04	查看代码
3	评测通过	1.21	46	2024-04-29 19:04	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 4

src/test4/test4.sql

```
1  -- 4) 查询办理了储蓄卡的客户名称、手机号、银行卡号。 查询结果结果依客户编号排序。
2  --      请用一条SQL语句实现该查询：
3  select  c_name,c_phone,b_number
4  from client, bank_card
5  where c_id=b_c_id and b_type="储蓄卡"
6  order by c_id
7
8
9
10
11  /* end of your code */
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	评测通过	1.17	50.08	2024-04-29 19:07	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 4

src/test5/test5.sql

```
1  -- 5) 查询理财产品中每份金额在30000~50000之间的理财产品的编号,每份金额,理财年限,并按照金额升序排序,金额相同的按照理财年限降序排序。
2  -- 请用一条SQL语句实现该查询:
3  select p_id,p_amount,p_year
4  from finances_product
5  where p_amount between 30000 and 50000
6  order by p_amount asc, p_year desc
7
8
9  /* end of your code */
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	评测通过	1.22	50.56	2024-04-29 19:14	查看代码

最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 4

src/test6/test6.sql

```
1  -- 6) 查询资产表中所有资产记录里商品收益的众数和它出现的次数。
2  --  请用一条SQL语句实现该查询：
3  SELECT pro_income,COUNT(*) AS presence
4  FROM property
5  GROUP BY pro_income
6  HAVING COUNT(*) >= ALL(SELECT COUNT(*) FROM property GROUP BY pro_income);
7
8  /* end of your code */
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	评测通过	1.4	46.06	2024-04-29 20:46	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 9

src/test7/test7.sql

```
1  -- 7) 查询身份证隶属武汉市没有买过任何理财产品的客户的名称、电话号、邮箱。
2  -- 请用一条SQL语句实现该查询：
3  select c_name, c_phone, c_mail
4  from client
5  where c_id_card like "4201%" and
6  not exists (
7      select 1
8      from property
9      where pro_c_id=c_id and pro_type=1
10 )
11 order by c_id
12
13
14
15
16 /* end of your code */
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	1.3	45.96	2024-05-06 15:22	查看代码
2	共有2组测试集，其中有2组测试结果不匹配	1.3	45.96	2024-05-06 15:24	查看代码
3	评测通过	1.3	45.96	2024-05-06 15:25	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 10

src/test8/test8.sql

```
1  -- 8) 查询持有两张(含)以上信用卡的用户的名称、身份证号、手机号。
2  -- 请用一条SQL语句实现该查询:
3  select c_name, c_id_card, c_phone
4  from client
5  where c_id in(
6      select b_c_id
7      from bank_card
8      where b_type="信用卡"
9      group by b_c_id
10     having count(*)>=2
11 )
12 order by c_id
13
14
15
16
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	评测通过	1.19	50.28	2024-05-06 15:31	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 8

src/test9/test9.sql

```
1  -- 9) 查询购买了货币型(f_type='货币型')基金的用户的名称、电话号、邮箱。
2  -- 请用一条SQL语句实现该查询:
3  select c_name, c_phone, c_mail
4  from client
5  where c_id in(
6      select pro_c_id
7      from property, fund
8      where f_type="货币型" and pro_pif_id=f_id and pro_type=3
9  )
10 order by c_id;
11
12
13 /* end of your code */
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	1.2	46.1	2024-05-06 15:37	查看代码
2	共有2组测试集，其中有2组测试结果不匹配	1.2	46.1	2024-05-06 15:42	查看代码
3	共有2组测试集，其中有2组测试结果不匹配	1.2	46.1	2024-05-06 15:43	查看代码
4	评测通过	1.2	46.1	2024-05-06 15:43	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 9

src/test10/test10.sql

```
1  -- 10) 查询当前总的可用资产收益(被冻结的资产除外)前三名的客户的名称、身份证号及其总收益，按收益降序输出，总收益命名为total_income。不考虑并列排名形。
2  -- 请用一条SQL语句实现该查询：
3  select c_name, c_id_card, total_income
4  from client,
5       (select sum(pro_income) as total_income,pro_c_id
6        from property
7        where pro_status="可用"
8        group by pro_c_id) c_income
9  where c_id=pro_c_id
10 order by total_income desc
11 limit 0,3;
12
13
14 /* end of your code */
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	1.24	49.96	2024-05-06 15:56	查看代码
2	评测通过	1.24	49.96	2024-05-06 15:58	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 5

src/test11/test11.sql

```
1  -- 11) 给出黄姓用户的编号、名称、办理的银行卡的数量(没有办卡的卡数量计为0),持卡数量命名为number_of_cards,
2  --      按办理银行卡数量降序输出,持卡数量相同的,依客户编号排序。
3  -- 请用一条SQL语句实现该查询:
4  select c_id, c_name, count(b_c_id) as number_of_cards
5  from client left join bank_card on c_id=b_c_id
6  where c_name like "黄%"
7  group by c_id
8  order by number_of_cards desc, c_id;
9
10
11  /* end of your code */
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	1.4	46.06	2024-05-06 19:07	查看代码
2	共有2组测试集，其中有2组测试结果不匹配	1.4	46.06	2024-05-06 19:08	查看代码
3	共有2组测试集，其中有2组测试结果不匹配	1.4	46.06	2024-05-06 19:08	查看代码
4	评测通过	1.4	46.06	2024-05-06 19:13	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 19

src/test12/test12.sql

```
1  -- 12) 综合客户表(client)、资产表(property)、理财产品表(finances_product)、保险表(insurance)和
2  -- 基金表(fund)，列出客户的名称、身份证号以及投资总金额（即投资本金，
3  -- 每笔投资金额=商品数量*该产品每份金额），注意投资金额按类型需要查询不同的表，
4  -- 投资总金额是客户购买的各类资产(理财,保险,基金)投资金额的总和，总金额命名为total_amount。
5  -- 查询结果按总金额降序排序。
6  -- 请用一条SQL语句实现该查询：
7  select c_name, c_id_card, ifnull(sum(proper_total.total),0) as total_amount
8  from client left join(
9      select pro_c_id, sum(p_amount*pro_quantity) as total
10     from finances_product, property
11     where p_id=pro_pif_id and pro_type=1
12     group by pro_c_id
13     union all
14     select pro_c_id, sum(i_amount*pro_quantity) as total
15     from insurance, property
16     where i_id=pro_pif_id and pro_type=2
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	1.22	46.04	2024-05-06 19:29	查看代码
2	共有2组测试集，其中有2组测试结果不匹配	1.22	46.04	2024-05-06 19:30	查看代码
3	评测通过	1.22	46.04	2024-05-06 19:35	查看代码
4	评测通过	1.22	46.04	2024-05-06 19:36	查看代码

最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 33

src/test13/test13.sql

```
1  -- 13) 综合客户表(client)、资产表(property)、理财产品表(finances_product)、
2  --     保险表(insurance)、基金表(fund)和投资资产表(property),
3  --     列出所有客户的编号、名称和总资产，总资产命名为total_property。
4  --     总资产为储蓄卡余额，投资总额，投资总收益的和，再扣除信用卡透支的金额
5  --     (信用卡余额即为透支金额)。客户总资产包括被冻结的资产。
6  --     请用一条SQL语句实现该查询：
7  select c_id, c_name, ifnull(sum(proper_total.total),0) as total_property
8  from client left join(
9      select pro_c_id, sum(p_amount*pro_quantity) as total
10     from finances_product, property
11     where p_id=pro_pif_id and pro_type=1
12     group by pro_c_id
13     union all
14     select pro_c_id, sum(i_amount*pro_quantity) as total
15     from insurance, property
16     where i_id=pro_pif_id and pro_type=2
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	1.19	45.86	2024-05-06 19:41	查看代码
2	评测通过	1.19	45.86	2024-05-06 19:42	查看代码

最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 7

src/test14/test14.sql

```
1  -- 14) 查询每份保险金额第4高保险产品的编号和保险金额。
2  --      在数字序列8000,8000,7000,7000,6000中，
3  --      两个8000均为第1高，两个7000均为第2高,6000为第3高。
4  -- 请用一条SQL语句实现该查询：
5  select i_id, i_amount
6  from insurance
7  where i_amount=(select distinct i_amount
8                  from insurance
9                  order by i_amount desc
10                 limit 3,1);
11
12
13
14
15
16
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	1.17	46.18	2024-05-06 19:44	查看代码
2	共有2组测试集，其中有2组测试结果不匹配	1.17	46.18	2024-05-06 19:46	查看代码
3	评测通过	1.17	46.18	2024-05-06 19:47	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 33

src/test15/test15.sql

```
1  -- 15) 查询资产表中客户编号，客户基金投资总收益,基金投资总收益的排名(从高到低排名)。  
2  -- 总收益相同时名次亦相同(即并列名次)。总收益命名为total_revenue，名次命名为rank。  
3  -- 第一条SQL语句实现全局名次不连续的排名，  
4  -- 第二条SQL语句实现全局名次连续的排名。  
5  
6  -- (1) 基金总收益排名(名次不连续)  
7  select pro_c_id,  
8         total_revenue,  
9         rank() over(order by total_revenue desc) as "rank"  
10 from (  
11     select pro_c_id, sum(pro_income) as total_revenue  
12     from property  
13     where pro_type=3  
14     group by pro_c_id  
15 ) a  
16 order by total revenue desc, pro c id;
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	评测通过	1.23	49.36	2024-05-06 20:16	查看代码
2	共有2组测试集，其中有2组测试结果不匹配	1.23	49.36	2024-05-09 16:05	查看代码
3	共有2组测试集，其中有2组测试结果不匹配	1.23	49.36	2024-05-09 16:06	查看代码
4	共有2组测试集，其中有2组测试结果不匹配	1.23	49.36	2024-05-09 16:07	查看代码
5	共有2组测试集，其中有2组测试结果不匹配	1.23	49.36	2024-05-09 16:12	查看代码
6	共有2组测试集，其中有2组测试结果不匹配	1.23	49.36	2024-05-09 16:22	查看代码
7	共有2组测试集，其中有2组测试结果不匹配	1.23	49.36	2024-05-09 16:23	查看代码



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 26

src/test16/test16.sql

```
1  -- 16) 查询持有相同基金组合的客户对，如编号为A的客户持有的基金，编号为B的客户也持有，反过来，编号为B的客户持有的基金，编号为A的客户也持有，则(A,B)
   为持有相同基金组合的二元组，请列出这样的客户对。为避免过多的重复，如果(1,2)为满足条件的元组，则不必显示(2,1)，即只显示编号小者在前的那一对，这一组
   户编号分别命名为c_id1,c_id2。
2
3  -- 请用一条SQL语句实现该查询：
4  select distinct A.pro_c_id as c_id1, B.pro_c_id as c_id2
5  from property A, property B
6  where A.pro_c_id < B.pro_c_id
7         and not exists(
8             select 1
9             from property C
10            where A.pro_c_id=C.pro_c_id and C.pro_type=3 and C.pro_pif_id not in (
11                select pro_pif_id
12                from property D
13                where D.pro_c_id=B.pro_c_id and D.pro_type=3
14            )
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	2.8	50.04	2024-05-06 20:41	查看代码
2	共有2组测试集，其中有2组测试结果不匹配	2.8	50.04	2024-05-06 20:42	查看代码
3	共有2组测试集，其中有2组测试结果不匹配	2.8	50.04	2024-05-06 20:43	查看代码
4	共有2组测试集，其中有2组测试结果不匹配	2.8	50.04	2024-05-06 20:44	查看代码
5	评测通过	2.8	50.04	2024-05-06 20:45	查看代码



最新代码 未通过本题，展示最新编写的代码详情

src/test17/test17.sql



```
1  -- 17 查询2022年2月购买基金的高峰期。至少连续三个交易日，所有投资者购买基金的总金额超过100万(含)，则称这段连续交易日为投资者购买基金的高峰期。只有
   交易日才能购买基金,但不能保证每个交易日都有投资者购买基金。2022年春节假期之后的第1个交易日为2月7日,周六和周日是非交易日，其余均为交易日。请列出高峰
   的日期和当日基金的总购买金额，按日期顺序排序。总购买金额命名为total_amount。
2  --      请用一条SQL语句实现该查询：
3
4
5
6
7
8
9  /* end of your code */
```

本关无评测记录



最后通关代码 通过了本题，在作业时间内最后一次提交的通关代码

代码修改行数 13

src/test18/test18.sql

```
1  -- 18) 查询至少有一张信用卡余额超过5000元的客户编号，以及该客户持有的信用卡总余额，总余额命名为credit_card_amount。
2  -- 请用一条SQL语句实现该查询：
3  select b_c_id, credit_card_amount
4  from (
5      select b_c_id, sum(b_balance) as credit_card_amount
6      from bank_card
7      where b_type="信用卡"
8      group by b_c_id
9  ) b_amount
10 where exists(
11     select *
12     from bank_card
13     where b_amount.b_c_id = bank_card.b_c_id and b_type="信用卡" and b_balance > 5000
14 )
15 order by b_c_id asc;
16
```

评测次数	详细信息	评测耗时(S)	内存消耗(MB)	评测时间	操作
1	共有2组测试集，其中有2组测试结果不匹配	1.81	50.46	2024-05-10 15:06	查看代码
2	评测通过	1.81	50.46	2024-05-10 15:07	查看代码



第19关 以日历表格式显示每日基金购买总金额

最新代码 未通过本题，展示最新编写的代码详情

代码修改行数 1

src/test19/test19.sql



```
1  -- 19) 以日历表格式列出2022年2月每周每日基金购买总金额，输出格式如下：
2  -- week_of_trading Monday Tuesday Wednesday Thursday Friday
3  --                1
4  --                2
5  --                3
6  --                4
7  --  请用一条SQL语句实现该查询：
8  select week_of_trading,
9
10
11
12
13
14
15  /* end of your code */
```

本关无评测记录