

总体评价



王文涛 按时通关

学号：U202215357

分班：计算机科学2201班

截止前完成关卡：6/6

最新完成关卡：6/6

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
2024-06-03 20:24	页面停留时长	5时 49分 10秒	7	--	--	6.0/6.0	优秀

阶段成绩

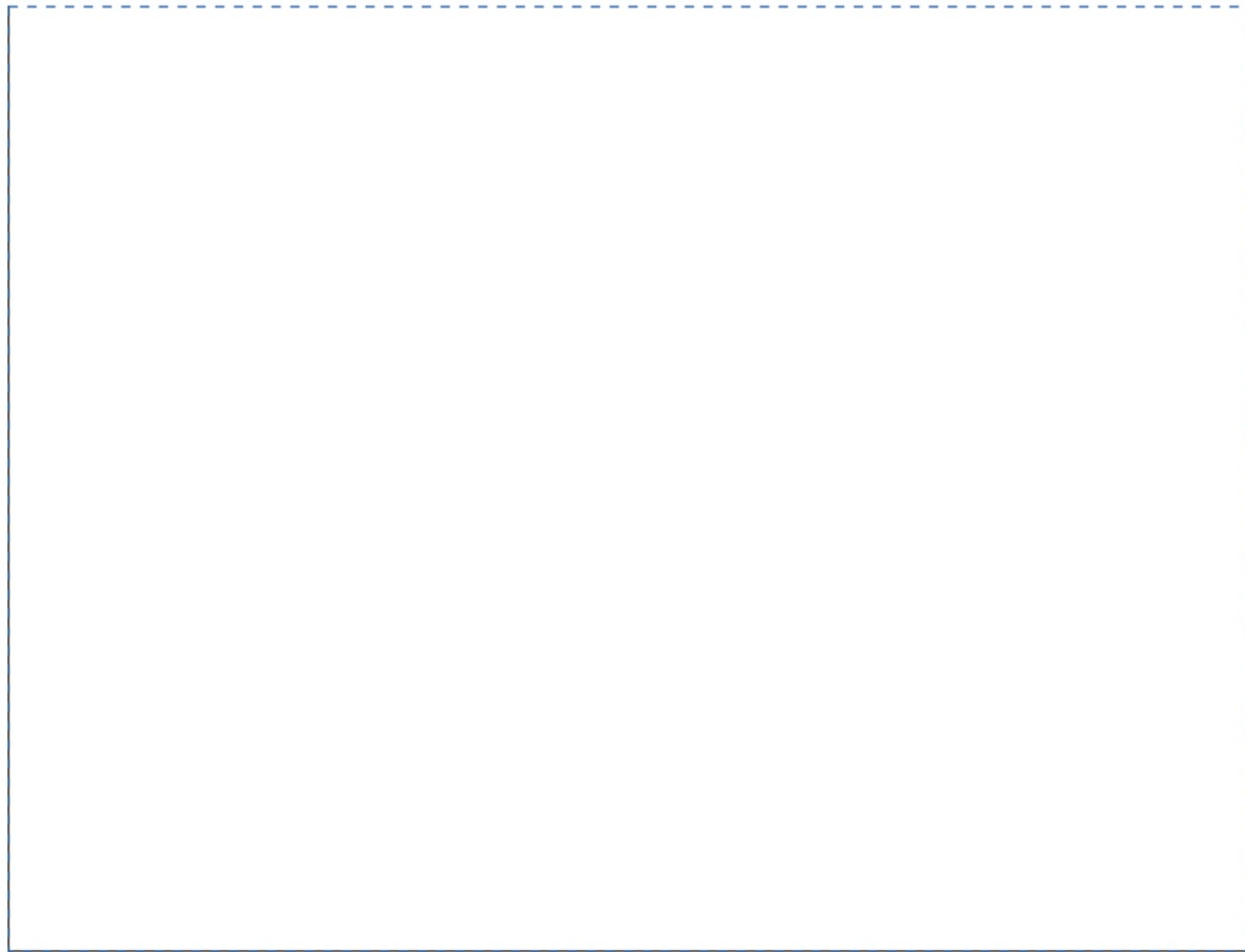
关卡	任务名称	开启时间	代码修改行数	评测次数	完成时间	实训耗时	是否查看答案	经验值	关卡得分	调分
1	并发控制与事务的隔离级别	2024-05-10 21:11	2	1	2024-06-03 14:57	23分 57秒	否	200/200	1.00/1	1.00
2	读脏	2024-06-03 14:57	4	1	2024-06-03 15:01	4分 4秒	否	900/900	1.00/1	1.00
3	不可重复读	2024-06-03 14:59	7	1	2024-06-03 15:06	3分 7秒	否	900/900	1.00/1	1.00
4	幻读	2024-06-03 15:06	2	1	2024-06-03 15:08	1分 53秒	否	500/500	1.00/1	1.00
5	主动加锁保证可重复读	2024-06-03 15:08	3	1	2024-06-03 15:12	3分 57秒	否	500/500	1.00/1	1.00
6	可串行化	2024-06-03	4	2	2024-06-03	5时 12分 12	否	300/300	1.00/1	1.00

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

能力值 (实训获得经验值/实训评测次数)

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
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

$\log(\text{实训的总时间})$

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 2

src/step1/test1.sql

```
1  -- 请不要在本代码文件中添加空行!!!
2  use testdb1;
3  # 设置事务的隔离级别为 read uncommitted
4  set session transaction isolation level read uncommitted;
5  -- 开启事务
6  start transaction;
7  insert into dept(name) values('运维部');
8  # 回滚事务:
9  rollback;
10 /* 结束 */
```

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	3.44	46.24	2024-06-03 14:57	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/step3/t1.sql



```
1  -- 事务1:
2  use testdb1;
3  ## 请设置适当的事务隔离级别
4  set session transaction isolation level read uncommitted;
5
6  start transaction;
7
8  -- 时刻2 - 事务1读航班余票,发生在事务2修改之后
9  ## 添加等待代码，确保读脏
10 set @n = sleep(1);
11 select tickets from ticket where flight_no = 'CA8213';
12 commit;
13
```

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	6.2	46.46	2024-06-03 15:01	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/step2/t1.sql

```
1  -- 事务1:
2  ## 请设置适当的事务隔离级别
3  set session transaction isolation level read uncommitted;
4
5  -- 开启事务
6  start transaction;
7  -- 时刻1 - 事务1读航班余票:
8  insert into result
9  select now(),1 t, tickets from ticket where flight_no = 'CZ5525';
10
11 ## 添加等待代码，确保事务2的第一次读取在事务1修改前发生
12 set @n = sleep(2);
13
14 -- 时刻3 - 事务1修改余票，并立即读取:
15 update ticket set tickets = tickets - 1 where flight_no = 'CZ5525';
16 insert into result
```

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	14.6	50.14	2024-06-03 15:06	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 2

src/step4/t1.sql



```
1  -- 事务1（采用默认的事务隔离级别- repeatable read）：
2  use testdb1;
3  select @@transaction_isolation;
4  start transaction;
5  ## 第1次查询余票超过300张的航班信息
6  select * from ticket where tickets > 300;
7  set @n = sleep(2);
8  -- 修改航班MU5111的执飞机型为A330-300:
9  update ticket set aircraft = 'A330-300' where flight_no = 'MU5111';
10 -- 第2次查询余票超过300张的航班信息
11 select * from ticket where tickets > 300;
12 commit;
```

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	4.18	45.7	2024-06-03 15:08	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/step5/t1.sql

```
1  -- 事务1:
2  use testdb1;
3  set session transaction isolation level read uncommitted;
4  start transaction;
5  # 第1次查询航班'MU2455'的余票
6  select tickets from ticket where flight_no = "MU2455" for update;
7  set @n = sleep(5);
8  # 第2次查询航班'MU2455'的余票
9  select tickets from ticket where flight_no = "MU2455";
10 commit;
11 -- 第3次查询所有航班的余票，发生在事务2提交后
12 set @n = sleep(1);
13 select * from ticket;
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	评测通过	12.18	46.24	2024-06-03 15:12	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/step6/t1.sql



```
1  -- 事务1:
2  use testdb1;
3  start transaction;
4  set @n = sleep(1);
5  select tickets from ticket where flight_no = 'MU2455';
6
7  select tickets from ticket where flight_no = 'MU2455';
8  commit;
9
```

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	0.2	46.18	2024-06-03 20:23	查看代码
2	评测通过	0.2	46.18	2024-06-03 20:24	查看代码