

《数据库系统》2023-2024 第二学期 SQL 考试

A 卷

学号：_____ 姓名：_____ 专业：_____

机房号：_____ 座位号：_____ 分数：_____

注意事项：

1. 答案以 Word 文件的方式提交，文件名为“学号_姓名.docx”，文件放在 D 盘根目录下，未按照规定方式命名会影响考试成绩。
2. 每个题目的 SQL 语句都必须在查询分析器中调试，运行无误后提交查询的 SQL 语句(文字)和查询结果截图(需将 SQL 语句和运行结果保存在同一张截图中提交)。
3. 查询结果集中的列名必须采用查询需求中给出的列名。

数据库模式

学院 (学院编号*, 学院名, 负责人)

学生 (学号*, 姓名, 性别, 年龄, 身份证, 学院编号*, 入学日期)

课程 (课号*, 课程名, 学院编号*, 学分, 星期, 开始时间, 结束时间)

选课 (学号*, 课号*, 平时成绩, 期末成绩)

注：*为主键，+为外键

部分示例数据如下 (含表名、属性名)：

col_id	col_name	charge	stu_id	stu_name	gender	age	idcard	col_id	inday
AMTD	高等数学教学部	徐兵	0110058	郭美莉	F	19	440120205806068728	LAWS	2076-09-01
COMP	计算机学院	刘菲菲	0110114	周冲	M	19	12010320580829113X	CSSE	2076-08-31
CSSE	网络空间安全学院	刘菲菲	0110233	张笑语	F	18	430922205909100721	CSSE	2076-09-01
ENTD	公共英语教学部	杨彬	0110514	王安	M	19	220281205804291114	CSSE	2076-09-01
FINA	金融学院	胡金锁	0111010	邱依玲	F	19	410822205802120184	COMP	2076-09-02
IPTD	马克思主义基础理论教学部	谷庆丰	0111100	阮建国	M	18	131103205912050857	SOFT	2076-09-02

学院 (college)

学生 (student)

cor_id	cor_name	col_id	credits	day	start_time	end_time	stu_id	cor_id	mid_score	fin_score
0009	操作系统	COMP	3.5	W	18:30:00	20:10:00	0110058	0113	90.5	83.0
0010	计算机网络	COMP	3.5	R	08:00:00	09:40:00	0110058	0310	86.0	88.0
0113	基础英语	ENTD	2.5	M	08:00:00	09:40:00	0110058	0513	95.0	98.0
0211	高级英语	ENTD	2.0	T	08:00:00	09:40:00	0110058	2177	90.0	84.0
0309	毛概	IPTD	3.5	F	16:00:00	17:40:00	0110058	3076	90.5	87.0
0310	马原	IPTD	3.5	W	14:00:00	15:40:00	0110114	0113	98.5	85.5

课程 (course)

选课 (takes)

注：以上部分示例数据仅供参考，具体的 SQL 语句不应该和具体的数据有关，请在答题过程中合理考虑空值、重复值等。题目中给出了结果模式的属性

1. 给出任意课程中成绩和期末成绩均获得 95 分以上的学生的学号 (stu_id) (10 分)

参考答案：

```
SELECT DISTINCT stu_id FROM takes WHERE mid_score > 95 AND
```

fin_score > 95

1 SELECT DISTINCT stu_id FROM takes WHERE mid_score > 95 AND fin_score > 95			
信息	Result 1	剖析	状态
stu_id			
▶ 0210535			
0211258			
0211377			
0211666			
0310524			
0311423			
0411412			

2. 给出每门课程的总分平均分(avg_score), 总分=0.3*期中成绩 + 0.7*期末成绩 (cor_id, avg_score) (10 分)

参考答案:

```
SELECT cor_id, AVG(mid_score*0.3 + fin_score*0.7) AS avg_score FROM
takes GROUP BY cor_id;
```

1 select cor_id, AVG(mid_score*0.3 + fin_score*0.7) AS avg_score FROM takes GROUP BY cor_id;			
信息	Result 1	剖析	状态
cor_id avg_score			
▶ 0009 93			
0010 91.466667			
0113 88.015217			
0211 90.453571			
0309 90.955			
0310 89.336111			
0358 83.733333			
0372 82.402273			
0513 96.519231			
0912 92.908333			
0917 93.09			
1012 94.22			
1016 94.22			
1019 89.8			
1027 87.775			
1476 85.86			
2177 85.695			
2841 91.239286			
2879 89.75			
2913 88.95			
3076 88.792857			
3332 91.825			

3. 查询选课人数大于等于 10 的课程, 并按选课人数从多到少倒序排列 (cor_id, cor_select_num) (10 分)

参考答案:

SELECT cor_id, COUNT(stu_id) AS cor_select_num

FROM takes

GROUP BY cor_id

HAVING COUNT(stu_id) >= 10

ORDER BY cor_select_num DESC;

```
1 SELECT cor_id, COUNT(stu_id) AS cor_select_num
2 FROM takes
3 GROUP BY cor_id
4 HAVING COUNT(stu_id) >= 10 |
5 ORDER BY cor_select_num DESC;
```

信息	Result 1	剖析	状态																				
	<table><tr><th>cor_id</th><th>cor_select_num</th></tr><tr><td>0113</td><td>23</td></tr><tr><td>0372</td><td>22</td></tr><tr><td>0310</td><td>18</td></tr><tr><td>0211</td><td>14</td></tr><tr><td>2841</td><td>14</td></tr><tr><td>▶ 0513</td><td>13</td></tr><tr><td>0358</td><td>12</td></tr><tr><td>0309</td><td>10</td></tr><tr><td>2177</td><td>10</td></tr></table>	cor_id	cor_select_num	0113	23	0372	22	0310	18	0211	14	2841	14	▶ 0513	13	0358	12	0309	10	2177	10		
cor_id	cor_select_num																						
0113	23																						
0372	22																						
0310	18																						
0211	14																						
2841	14																						
▶ 0513	13																						
0358	12																						
0309	10																						
2177	10																						

4. 给出身份证号以'12'开头或身份证号为空的学生的学号 (stu_id) (10 分)

参考答案:

(1) SELECT stu_id FROM student WHERE idcard LIKE '12%' OR idcard IS NULL;

(2) SELECT stu_id FROM student WHERE LEFT(idcard, 2) = '12' OR idcard IS NULL;

1 SELECT stu_id FROM student WHERE idcard LIKE '12%' OR idcard IS NULL;
2 SELECT stu_id FROM student WHERE LEFT(idcard, 2) = '12' OR idcard IS NULL;
2

信息	Result 1	剖析	状态
	stu_id		
	0110114		
	0111123		
	0111871		
	0210535		
	0210655		
	0211258		
	0310523		
	0311423		

5. 给出所有金融学院学生的平均期末成绩 (stu_id, stu_name, avg_fin_score) (10分)

参考答案:

```
SELECT s.stu_id, s.stu_name, AVG(t.fin_score) AS avg_fin_score
FROM
    student s natural join takes t
WHERE
    s.col_id = (
        SELECT col_id
        FROM college
        WHERE col_name = '金融学院'
    )
GROUP BY
    s.stu_id;
```

1 SELECT s.stu_id, s.stu_name , AVG(t.fin_score) AS avg_score
2 FROM student s
3 JOIN takes t ON s.stu_id = t.stu_id
4 WHERE s.col_id = (SELECT col_id FROM college where col_name='金融学院')
5 GROUP BY s.stu_id;

信息	Result 1	剖析	状态
stu_id	stu_name	avg_score	
▶0111917	钱曼曼	83.91667	
0211168	姚童茗	92	

6. 给出所有同名学生的学号和姓名 (stu_id, stu_name) (10分)

参考答案:

```
select s1.stu_id, s1.stu_name
from student s1, student s2
```

where s1.stu_name=s2.stu_name

and s1.stu_id<s2.stu_id

```
1 SELECT s.stu_id, s.stu_name
2 FROM student s
3 JOIN (
4     SELECT stu_name
5     FROM student
6     GROUP BY stu_name
7     HAVING COUNT(*) > 1
8 ) s1 ON s.stu_name = s1.stu_name
9 ORDER BY s.stu_name, s.stu_id;
```

信息	Result 1	剖析	状态
----	----------	----	----

stu_id	stu_name
▶ 0110514	王安
0210535	王安
0211666	王安

7. 查询所有未选修过学分为 3.5 的课程的学生信息 (stu_id, stu_name) (10 分)

参考答案:

(1) SELECT

s.stu_id,

s.stu_name

FROM

student s

LEFT JOIN

takes t ON s.stu_id = t.stu_id

WHERE

s.stu_id NOT IN (

SELECT t.stu_id

```
        FROM takes t
        JOIN course c ON t.cor_id = c.cor_id
        WHERE c.credits = 3.5
    )
GROUP BY
    s.stu_id, s.stu_name;
```

(2) SELECT DISTINCT

```
    s.stu_id,
    s.stu_name
FROM
    student s
LEFT JOIN
    takes t ON s.stu_id = t.stu_id
WHERE
    s.stu_id NOT IN (
        SELECT t.stu_id
        FROM takes t
        JOIN course c ON t.cor_id = c.cor_id
        WHERE c.credits = 3.5
    );
```

2	s.stu_id,
3	s.stu_name
4	FROM
5	student s
6	LEFT JOIN
7	takes t ON s.stu_id = t.stu_id
8	WHERE
9	s.stu_id NOT IN (
10	SELECT t.stu_id
11	FROM takes t
12	JOIN course c ON t.cor_id = c.cor_id
13	WHERE c.credits = 3.5
14)
15	GROUP BY

信息	Result 1	剖析	状态
	stu_id	stu_name	
▶	0110114	周冲	
	0111100	阮建国	
	0111168	姚同铭	
	0111871	魏前程	
	0111917	钱曼曼	
	0111996	张眯	

8. 给出选修过全部由计算机学院开设的课程（col_name = '计算机学院'）的学生（stu_id, stu_name）(10 分)

参考答案：

```
SELECT student.stu_id, student.stu_name
FROM student
WHERE NOT EXISTS (
    SELECT *
    FROM course
    WHERE col_id = (SELECT col_id FROM college WHERE col_name = '计算机学院')
    AND NOT EXISTS (
```

```

SELECT *
FROM takes
WHERE takes.stu_id = student.stu_id AND takes.cor_id =
course.cor_id
)
)

```

```

1 SELECT student.stu_id, student.stu_name
2 FROM student
3 WHERE NOT EXISTS (
4     SELECT *
5     FROM course
6     WHERE col_id = (SELECT col_id FROM college WHERE col_name = '计算机学院')
7     AND NOT EXISTS (
8         SELECT *
9         FROM takes
10        WHERE takes.stu_id = student.stu_id AND takes.cor_id = course.cor_id
11    )
12 )

```

stu_id	stu_name
0211011	林伊伊
0311423	董参商
0411412	齐达雷

9. 给出人数小于 3 人（包括 0 人）的学院或部门的 id、名称（col_id, col_name）（10 分）

参考答案：

```

SELECT
college.col_id,
college.col_name
FROM college LEFT JOIN
(SELECT col_id, COUNT(stu_id) AS stu_num FROM student GROUP BY
col_id) AS stu_count
ON college.col_id = stu_count.col_id
WHERE stu_num < 3 OR stu_num IS NULL

```



```

1 SELECT
2 college.col_id,
3 college.col_name
4 FROM college LEFT JOIN
5 (SELECT col_id, COUNT(stu_id) AS stu_num FROM student GROUP BY col_id) AS stu_count |
6 ON college.col_id = stu_count.col_id
7 WHERE stu_num < 3 OR stu_num IS NULL

```

信息	Result 1	剖析	状态
	col_id	col_name	
	AMTD	高等数学教学部	
	ENTD	公共英语教学部	
	FINA	金融学院	
	IPTD	马克思主义基础理论	
	PETDS	体育部	
	SOFT	软件学院	
	UPEC	通识选修课	

10. 给出每门课程的课程 id、课程名和该门课程获得最高分的学生的学号、姓名，以及该学生所获分数并分数按倒序排列，其中课程最终成绩=0.3 平时成绩+0.7 期末成绩 (cor_id, cor_name, stu_id, stu_name, final_score) (10 分)

参考答案：

```

SELECT

    cr.cor_id,

    cr.cor_name,

    t.stu_id,

    s.stu_name,

    (0.3 * t.mid_score + 0.7 * t.fin_score) AS final_score

FROM

    (

        SELECT

            cor_id,

            MAX(0.3 * mid_score + 0.7 * fin_score) AS max_score

        FROM

            takes

        GROUP BY

            cor_id

    ) max_scores

```

JOIN

takes t ON max_scores.cor_id = t.cor_id

AND max_scores.max_score = 0.3 * t.mid_score + 0.7 * t.fin_score

JOIN

course cr ON t.cor_id = cr.cor_id

JOIN

student s ON t.stu_id = s.stu_id

ORDER BY

final_score DESC;

```

1  SELECT
2      cr.cor_id,
3      cr.cor_name,
4      t.stu_id,
5      s.stu_name,
6      (0.3 * t.mid_score + 0.7 * t.fin_score) AS final_score
7  FROM
8      (
9          SELECT
10             cor_id, |
11             MAX(0.3 * mid_score + 0.7 * fin_score) AS max_score
12          FROM

```

信息	Result 1	剖析	状态		
	cor_id	cor_name	stu_id	stu_name	final_score
▶	0513	健康教育	0211666	王安	98.65
	0211	高级英语	0211377	张谦	98.3
	1012	数据库系统	0210119	霍京	97.7
	0358	高等数学B	0311423	董参商	96.85
	0912	数据库系统	0211258	艾之章	96.85
	0917	算法导论	0211377	张谦	96.2
	2841	体育理论	0211011	林伊伊	96
	1016	算法设计	0310523	徐童帆	95.45
	0372	高等数学A	0110514	王安	95
	0309	毛概	0211168	姚童茗	94.75
	0009	操作系统	0311423	董参商	94.65
	3332	软件测试	0311423	董参商	94.25
	0010	计算机网络	0311423	董参商	94.05
	0310	马原	0110233	张笑语	93.7
	2913	排球	0110114	周冲	93.2
	3076	足球	0210120	衣胜男	93.1
	2879	篮球	0311423	董参商	92.45
	0113	基础英语	0211168	姚童茗	92.3
	1019	嵌入式系统	0310523	徐童帆	92.3
	2177	法律方法	0210655	王平	90.8
	1027	网络技术	0311423	董参商	90.3
	1476	专业规划	0111100	阮建国	87.5