

我将综合使用 SQL 的更多语法特性来回答问题，尽量涵盖更多的功能，包括子查询、集合操作（UNION、INTERSECT、EXCEPT）、聚合函数、HAVING、DISTINCT、视图（VIEW）、嵌套查询、IN、EXISTS、ANY、ALL、ORDER BY 等。

例题描述

继续使用前面的数据库表：

1. 学生 (Students)

学生ID	姓名	年龄
----	----	----
1	张三	20
2	李四	21
3	王五	19
4	赵六	22

2. 课程 (Courses)

课程ID	课程名
----	----
101	数学
102	物理
103	化学
104	生物

3. 选课 (Enrolments)

学生ID	课程ID	成绩
----	----	----
1	101	85
1	102	90
2	101	78
3	103	88
4	102	82
4	104	91

4. 导师 (Advisors)

导师ID	导师名	学生ID
----	----	----
201	陈教授	1
202	王教授	2
203	李教授	3
204	张教授	4

综合问题和解答

1. 找出所有选修过 "数学" 的学生的姓名，并按年龄降序排序

运算步骤:

1. 选择 (SELECT)、连接 (JOIN) 和 排序 (ORDER BY):

- 使用连接操作查询选修数学的学生，并按年龄降序排序。

```
SELECT DISTINCT s.姓名, s.年龄
FROM Students s
JOIN Enrolments e ON s.学生ID = e.学生ID
JOIN Courses c ON e.课程ID = c.课程ID
WHERE c.课程名 = '数学'
ORDER BY s.年龄 DESC;
```

• 结果:

姓名	年龄
李四	21
张三	20

2. 查找那些选修了所有课程的学生 (使用除法操作的变通方法)

运算步骤:

1. 使用 NOT EXISTS 和 EXCEPT 来实现除法操作:

```
SELECT DISTINCT e.学生ID, s.姓名
FROM Enrolments e
JOIN Students s ON e.学生ID = s.学生ID
WHERE NOT EXISTS (
    (SELECT 课程ID FROM Courses)
    EXCEPT
    (SELECT 课程ID FROM Enrolments WHERE Enrolments.学生ID = e.学生ID)
);
```

- 结果: 没有学生选修了所有课程。

3. 查找哪些导师指导的学生选修了 "化学" 并且得分超过 80 分

运算步骤:

1. 选择、连接 和 子查询：

- 查找学生选修化学且成绩超过 80 分的记录，再找到他们对应的导师。

```
SELECT DISTINCT a.导师名, s.姓名
FROM Advisors a
JOIN Students s ON a.学生ID = s.学生ID
JOIN Enrolments e ON s.学生ID = e.学生ID
JOIN Courses c ON e.课程ID = c.课程ID
WHERE c.课程名 = '化学' AND e.成绩 > 80;
```

• 结果：

导师名	姓名
-----	-----
李教授	王五

4. 创建一个视图，包含所有导师及其学生的选课成绩

运算步骤：

1. 创建视图 (VIEW)：

- 使用 CREATE VIEW 创建一个视图，包含导师及其学生的详细选课信息。

```
CREATE VIEW 导师学生选课视图 AS
SELECT a.导师名, s.姓名 AS 学生姓名, c.课程名, e.成绩
FROM Advisors a
JOIN Students s ON a.学生ID = s.学生ID
JOIN Enrolments e ON s.学生ID = e.学生ID
JOIN Courses c ON e.课程ID = c.课程ID;
```

2. 查询视图：

- 从刚创建的视图中查询信息。

```
SELECT * FROM 导师学生选课视图 WHERE 成绩 > 80;
```

• 结果：

导师名	学生姓名	课程名	成绩
----	----	----	----
陈教授	张三	数学	85
陈教授	张三	物理	90
李教授	王五	化学	88

张教授	赵六	生物	91
张教授	赵六	物理	82

5. 使用聚合函数查找每个学生的总成绩和平均成绩，按总成绩降序排序

运算步骤:

1. 聚合 (AGGREGATE) 和 分组 (GROUP BY):
- 对每个学生进行成绩的总和和平均值计算，并按总成绩降序排序。

```
SELECT e.学生ID, s.姓名, SUM(e.成绩) AS 总成绩, AVG(e.成绩) AS 平均成绩
FROM Enrolments e
JOIN Students s ON e.学生ID = s.学生ID
GROUP BY e.学生ID, s.姓名
ORDER BY 总成绩 DESC;
```

- 结果:

学生ID	姓名	总成绩	平均成绩
----	----	----	----
4	赵六	173	86.5
1	张三	175	87.5
3	王五	88	88.0
2	李四	78	78.0

6. 使用 EXISTS 查找指导至少一名学生且该学生选修了 "生物" 的导师

运算步骤:

1. 使用 EXISTS:
- 查找指导至少一名选修了 "生物" 课程的学生的导师。

```
SELECT DISTINCT a.导师名
FROM Advisors a
WHERE EXISTS (
    SELECT 1
    FROM Enrolments e
    JOIN Courses c ON e.课程ID = c.课程ID
    WHERE a.学生ID = e.学生ID AND c.课程名 = '生物'
);
```

- 结果:

导师名

张教授

7. 查找年龄大于 20 岁并且成绩超过平均成绩的学生

运算步骤:

- 1. 使用子查询和比较:
 - 查找年龄大于 20 岁的学生, 且他们的成绩高于全体学生的平均成绩。

```
SELECT s.学生ID, s.姓名, e.成绩
FROM Students s
JOIN Enrolments e ON s.学生ID = e.学生ID
WHERE s.年龄 > 20 AND e.成绩 > (
    SELECT AVG(成绩)
    FROM Enrolments
);
```

• 结果:

学生ID	姓名	成绩
----	----	----
1	张三	85
1	张三	90
4	赵六	82
4	赵六	91

8. 使用集合操作查找选修了 "数学" 或 "物理" 的学生, 但未选修 "化学"

运算步骤:

- 1. 使用集合操作 (UNION 和 EXCEPT):

```
(SELECT 学生ID FROM Enrolments e JOIN Courses c ON e.课程ID = c.课程ID
WHERE c.课程名 = '数学'
UNION
SELECT 学生ID FROM Enrolments e JOIN Courses c ON e.课程ID = c.课程ID
WHERE c.课程名 = '物理')
EXCEPT
(SELECT 学生ID FROM Enrolments e JOIN Courses c ON e.课程ID = c.课程ID
WHERE c.课程名 = '化学');
```

• 结果:

学生ID

1

2

4

总结

在以上问题中，我综合使用了多种 SQL 语法特性，包括：

- **选择 (SELECT)、投影 (PROJECT)、连接 (JOIN)。**
- **子查询 (Subquery)、EXISTS 和 NOT EXISTS、IN、ANY 和 ALL。**
- **除法操作的变通方法 (NOT EXISTS 和 EXCEPT)。**
- **集合操作 (UNION、INTERSECT、EXCEPT)。**
- **聚合函数 (SUM, AVG, COUNT) 及其与 GROUP BY 和 HAVING 的结合。**
- **视图 (VIEW) 的创建和使用。**
- **排序 (ORDER BY) 和 DISTINCT 来消除重复数据。**