

题目一：

三段 SQL 语句均不合法，分别运行这三段 SQL 语句，得到报错如下：

```
[2025-03-30 21:42:17] 已连接
university.public> SELECT dept_name, min(salary)
                    FROM instructor
[2025-03-30 21:42:17] [42803] 错误：字段 "instructor.dept_name" 必须出
现在 GROUP BY 子句中或者在聚合函数中使用
[2025-03-30 21:42:17] 位置： 8
university.public> SELECT dept_name, min(salary)
                    FROM instructor
                    GROUP BY dept_name
                    HAVING name LIKE '%at%'
[2025-03-30 21:43:59] [42803] 错误：字段 "instructor.name" 必须出现在
GROUP BY 子句中或者在聚合函数中使用
[2025-03-30 21:43:59] 位置： 76
university.public> SELECT dept_name
                    FROM instructor
                    WHERE AVG(salary) > 20000
[2025-03-30 21:44:20] [42803] 错误：聚合函数不允许出现在WHERE中
[2025-03-30 21:44:20] 位置： 42
```

可以得出结论：1. 出现在 SELECT 和 HAVING 子句中但没有被聚集的属性只能是出现在 GROUP BY 子句中的属性；2. 聚合函数不能出现在 WHERE 子句中

题目二：

1、工资最高员工只有一位，使用四种方法找到工资最高员工的姓名：

```
15  -- 1.方法一：
16  SELECT name FROM instructor
17  WHERE salary ≥ ALL(SELECT salary
18                      FROM instructor);
19  --1.方法二：
20  SELECT name FROM instructor
21  WHERE salary IN (SELECT max(salary)
22                  FROM instructor);
```

```

23 --1. 方法三:
24 ✓ SELECT name FROM instructor
25 ✓ WHERE salary = (SELECT max(salary)
26                      FROM instructor);
27 --1. 方法四:
28 ✓ SELECT name FROM instructor
29   ORDER BY salary DESC
30   LIMIT 1;

```

运行结果都如下:

	name
1	Wieland

2、假设工资最高的员工有多位，用四种方法查询工资最高员工的名字。

前三种方法与第一题一样，第四种方法在第一题的方法四的基础上改变:

```

32 -- 2. 方法一:
33 SELECT name FROM instructor
34 WHERE salary ≥ ALL(SELECT salary
35                      FROM instructor);
36 --2. 方法二:
37 SELECT name FROM instructor
38 WHERE salary IN (SELECT max(salary)
39                  FROM instructor);
40 --2. 方法三:
41 SELECT name FROM instructor
42 WHERE salary = (SELECT max(salary)
43                 FROM instructor);
44 --2. 方法四:
45 SELECT name FROM instructor
46 WHERE salary = (SELECT salary FROM instructor
47                 ORDER BY salary DESC
48                 LIMIT 1);

```

3、解释四个 SQL 语句:

`SELECT 1 IN (1);` 检查数字 1 是否在集合(1)中，因为 1 确实在集合(1)中，所以返回结果为布尔值 `true`

`SELECT 1 = (1);` 比较等号左右两边的值，等号左边是 1，后边的括号在这里只起分组作用，不影响值本身，所以等号右边本质上也是 1，因此返回结果为布尔值 • **true**

`SELECT (1, 2) = (1, 2);` 比较两个行元组，行元组的比较是组个字段进行的，因为两个行元组的所有字段都相等（第一个元素 1=1，第二个元素 2=2），所以整个行元组相等，返回 • **true**

`SELECT (1) IN (1, 2);` 检查左侧的 (1) 是否在右侧的集合中，(1) 等同于简单的 1，确实存在于集合 (1, 2) 中，因此返回 • **true**