

第六章：秋招秘籍 A

练习一：各部门工资最高的员工（难度：中等）

创建Employee 表，包含所有员工信息，每个员工有其对应的 Id, salary 和 department Id。

Id	Name	Salary	DepartmentId
1	Joe	70000	1
2	Henry	80000	2
3	Sam	60000	2
4	Max	90000	1

创建Department 表，包含公司所有部门的信息。

Id	Name
1	IT
2	Sales

编写一个 SQL 查询，找出每个部门工资最高的员工。例如，根据上述给定的表格，Max 在 IT 部门有最高工资，Henry 在 Sales 部门有最高工资。

Department	Employee	Salary
IT	Max	90000
Sales	Henry	80000

练习二：换座位（难度：中等）

小美是一所中学的信息科技老师，她有一张 seat 座位表，平时用来储存学生名字和与他们相对应的座位 id。

其中纵列的id是连续递增的

小美想改变相邻俩学生的座位。

你能不能帮她写一个 SQL query 来输出小美想要的结果呢？

请创建如下所示seat表：

示例：

id	student
1	Abbot
2	Doris
3	Emerson
4	Green
5	Jeames

假如数据输入的是上表，则输出结果如下：

id	student
1	Doris
2	Abbot
3	Green
4	Emerson
5	Jeames

注意：

如果学生人数是奇数，则不需要改变最后一个同学的座位。

练习三：分数排名（难度：中等）

假设在某次期末考试中，二年级四个班的平均成绩分别是 93、93、93、91，请问可以实现几种排名结果？分别使用了什么函数？排序结果是怎样的？（只考虑降序）

class	score_avg
1	93
2	93
3	93
4	91

练习四：连续出现的数字（难度：中等）

编写一个 SQL 查询，查找所有至少连续出现三次的数字。

Id	Num
1	1
2	1
3	1
4	2
5	1
6	2
7	2

例如，给定上面的 Logs 表，1 是唯一连续出现至少三次的数字。

ConsecutiveNums
1

练习五：树节点（难度：中等）

对于tree表，id是树节点的标识，p_id是其父节点的id。

id	p_id
1	null
2	1
3	1
4	2
5	2

每个节点都是以下三种类型中的一种：

- Root: 如果节点是根节点。
- Leaf: 如果节点是叶子节点。
- Inner: 如果节点既不是根节点也不是叶子节点。

写一条查询语句打印节点id及对应的节点类型。按照节点id排序。上面例子的对应结果为：

id	Type
1	Root
2	Inner
3	Leaf
4	Leaf
5	Leaf

说明

- 节点'1'是根节点，因为它的父节点为NULL，有'2'和'3'两个子节点。
- 节点'2'是内部节点，因为它的父节点是'1'，有子节点'4'和'5'。
- 节点'3'，'4'，'5'是叶子节点，因为它们有父节点但没有子节点。

下面是树的图形：

```

      1
     / \
    2   3
   / \
  4   5

```

注意

如果一个树只有一个节点，只需要输出根节点属性。

练习六：至少有五名直接下属的经理（难度：中等）

Employee表包含所有员工及其上级的信息。每位员工都有一个Id，并且还有一个对应主管的Id（ManagerId）。

Id	Name	Department	ManagerId
101	John	A	null
102	Dan	A	101
103	James	A	101
104	Amy	A	101
105	Anne	A	101
106	Ron	B	101

针对**Employee**表，写一条SQL语句找出有5个下属的主管。对于上面的表，结果应输出：

Name
John

注意：

没有人向自己汇报。

练习七：查询回答率最高的问题（难度：中等）

求出survey_log表中回答率最高的问题，表格的字段有：uid, action, question_id, answer_id, q_num, timestamp。

uid是用户id；action的值为：“show”，“answer”，“skip”；当action是“answer”时，answer_id不为空，相反，当action是“show”和“skip”时为空（null）；q_num是问题的数字序号。

写一条sql语句找出回答率最高的 question_id。

举例：

输入

uid	action	question_id	answer_id	q_num	timestamp
5	show	285	null	1	123
5	answer	285	124	1	124
5	show	369	null	2	125
5	skip	369	null	2	126

输出

question_id
285

说明

问题285的回答率为1/1，然而问题369的回答率是0/1，所以输出是285。

注意：

最高回答率的意思是：同一个问题出现的次数中回答的比例。

练习八：各部门前3高工资的员工（难度：中等）

将练习一中的 `employee` 表清空，重新插入以下数据（也可以复制练习一中的 `employee` 表，再插入第5、第6行数据）：

Id	Name	Salary	DepartmentId
1	Joe	70000	1
2	Henry	80000	2
3	Sam	60000	2
4	Max	90000	1
5	Janet	69000	1
6	Randy	85000	1

编写一个 SQL 查询，找出每个部门工资前三高的员工。例如，根据上述给定的表格，查询结果应返回：

Department	Employee	Salary
IT	Max	90000
IT	Randy	85000
IT	Joe	70000
Sales	Henry	80000
Sales	Sam	60000

此外，请考虑实现各部门前N高工资的员工功能。

练习九：平面上最近距离 (难度：困难)

`point_2d`表包含一个平面内一些点（超过两个）的坐标值（x, y）。

写一条查询语句求出这些点中的最短距离并保留2位小数。

x	y
-1	-1
0	0
-1	-2

最短距离是1，从点 (-1, -1) 到点 (-1, -2)。所以输出结果为：

| shortest |

1.00

```
+-----+
|shortest|
+-----+
|1.00    |
+-----+
```

****注意：**所有点的最大距离小于10000。

练习十：行程和用户（难度：困难）

Trips 表中存所有出租车的行程信息。每段行程有唯一键 Id，Client_Id 和 Driver_Id 是 Users 表中 Users_Id 的外键。Status 是枚举类型，枚举成员为 ('completed', 'cancelled_by_driver', 'cancelled_by_client')。

Id	Client_Id	Driver_Id	City_Id	Status	Request_at
1	1	10	1	completed	2013-10-1
2	2	11	1	cancelled_by_driver	2013-10-1
3	3	12	6	completed	2013-10-1
4	4	13	6	cancelled_by_client	2013-10-1
5	1	10	1	completed	2013-10-2
6	2	11	6	completed	2013-10-2

Id	Client_Id	Driver_Id	City_Id	Status	Request_at
7	3	12	6	completed	2013-10-2
8	2	12	12	completed	2013-10-3
9	3	10	12	completed	2013-10-3
10	4	13	12	cancelled_by_driver	2013-10-3

Users 表存所有用户。每个用户有唯一键 Users_Id。Banned 表示这个用户是否被禁止，Role 则是一个表示（‘client’，‘driver’，‘partner’）的枚举类型。

Users_Id	Banned	Role
1	No	client
2	Yes	client
3	No	client
4	No	client
10	No	driver
11	No	driver
12	No	driver
13	No	driver

写一段 SQL 语句查出**2013年10月1日至2013年10月3日**期间非禁止用户的取消率。基于上表，你的 SQL 语句应返回如下结果，取消率（Cancellation Rate）保留两位小数。

Day	Cancellation Rate
2013-10-01	0.33
2013-10-02	0.00
2013-10-03	0.50