|  |
| --- |
| 获取所有非manager的员工emp\_no  我的做法：子查询  SELECT emp\_no FROM employees  WHERE emp\_no NOT IN (SELECT emp\_no FROM dept\_manager)  知道不做的地方，用in的话效率太低。能够用其他方法，exists关键字有没有作用。 |
| 别人的做法：  先使用LEFT JOIN连接两张表，再从此表中选出dept\_no值为NULL对应的emp\_no记录  SELECT emp\_no FROM (SELECT \* FROM employees LEFT JOIN dept\_manager  ON employees.emp\_no = dept\_manager.emp\_no)  WHERE dept\_no IS NULL  这里是用了一个格式上的区别。就是有一张表的特殊字段。这是利用了数据的特性。挺好。  这种方式，能够给人一些提示，不是本题能体现的。 |
| 刚才我在【[https://www.w3schools.com](https://www.w3schools.com/" \t "_blank)】在线数据库测试了一下，发现(B.emp\_no并不是只包含dept\_manager表中第一条记录的emp\_no，因为A中有6条数据，B中有11条数据，所以【select A.emp\_no from employees A,dept\_manager B where A.emp\_no not in (B.emp\_no);】会产生60条数据，【select A.emp\_no from employees A,dept\_manager B where A.emp\_no in (B.emp\_no);】会产生6条数据，总共6\*11=66条  **又是一种数据库的学习方式，这样的话，可以用网上的环境做测试，不过应该要用数据库创建数据库。** |
| Sql优化的一个最常用的方式就是不用not in和in  而是改用连接。  [**sql语句优化：用join取代not in**](http://www.cnblogs.com/rising-fay/archive/2012/11/13/2767538.html)  写了好几个页面，速度都上不去，瓶颈在于SQL查询。太多的表，太多的not in，总是从一大推表和数据中筛选出一点数据。看了很多关于SQL优化的文章，都强烈要求不要太多使用not in查询，最好用表连接来取代它。如： select ID,name from Table\_A where ID not in (select ID from Table\_B) 这句是最经典的not in查询了。改为表连接代码如下： select Table\_A.ID,Table\_A.name from Table\_A left join Table\_B on Table\_A.ID=Table\_B.ID and **Table\_B.ID is null** 或者： select Table\_A.ID,Table\_A.name from Table\_A left join Table\_B on Table\_A.ID=Table\_B.ID **where Table\_B.ID is null** |
| 【用exists优化】exists子句不在乎返回什么，而是在乎是不是有结果集返回。  EXISTS与IN的使用效率的问题，通常情况下采用exists要比in效率高，因为IN不走索引，但要看实际情况具体使用： **IN适合于外表大而内表小的情况**；**EXISTS适合于外表小而内表大的情况**。  [**http://www.cnblogs.com/netserver/archive/2008/12/25/1362615.html**](http://www.cnblogs.com/netserver/archive/2008/12/25/1362615.html)  select emp\_no from employees  em where not exists(select emp\_no from dept\_manager where emp\_no = em.emp\_no)  直接用的是not exist的方式。而不是not in。这样的方式emp\_no not exists方式。 |
| 1.exists的优点是：只要存在就返回了，这样的话很有**可能不需要**扫描整个表。  in需要扫描完整个表，并返回结果整个结果集。  这个解释是最好的，nice。  2. #3 字表查询的结果  exists判断子查询的结果是不是存在，但查到什么结果，什么字段，并不关心；  in      需要子查询查得的结果给主查询使用。返回的的是结果集。  <http://www.cnblogs.com/jerome-rong/archive/2013/03/27/2985505.html>  这个总结的最好了。 |
| select emp\_no from employees  em where not exists(select emp\_no from dept\_manager where emp\_no = em.emp\_no)  EXISTS语法**并没有说哪个字段落在了子查寻的结果中**，而是说exists后面的语句执行的结果是不是有记录，只要有记录，则**主查询语句就成立**。注意到，外层的where后只用写not exists即可，不用写其他的。那么可以用这种方式来解决的。选择什么并不重要。  select e.emp\_no  from employees e  where not exists(  select **null**  from dept\_manager d  where **d.emp\_no = e.emp\_no**  ) |
|  |

第二题：

牛客网选定后，总是回不来，总是跳。解决方法是用ctrl+z解决。

|  |
| --- |
| 获取所有员工当前的manager |
| 不用连接的做法。解法一：  select m.emp\_no , d.emp\_no as manager\_no  from dept\_emp m, dept\_manager d  where m.dept\_no = d.dept\_no and m.emp\_no <> d.emp\_no and m.to\_date='9999-01-01' and d.to\_date='9999-01-01'  运行时间：24ms  占用内存：3196k  用连接和用from table1, table2并列查询，有时候都可以解题。只是前者将几张表连接成一张大表，后者是并列查询若干表 |
| 运行时间：24ms  占用内存：3072k  select m.emp\_no , d.emp\_no as manager\_no  from dept\_emp m inner join dept\_manager d on m.dept\_no = d.dept\_no  where m.emp\_no <> d.emp\_no and m.to\_date='9999-01-01' and d.to\_date='9999-01-01' |
| 运行时间：20ms  占用内存：3196k  select m.emp\_no , d.emp\_no as manager\_no  from dept\_emp m join dept\_manager d on m.dept\_no = d.dept\_no  where m.emp\_no <> d.emp\_no and m.to\_date='9999-01-01' and d.to\_date='9999-01-01' |
| 运行时间：28ms  占用内存：3080k  select m.emp\_no , d.emp\_no as manager\_no  from dept\_emp m **left join** dept\_manager d on m.dept\_no = d.dept\_no  where m.emp\_no <> d.emp\_no and m.to\_date='9999-01-01' and d.to\_date='9999-01-01' |
|  |

第三题：获取所有部门中当前员工薪水最高的相关信息

【收获】

<http://w3schools.bootcss.com/sql/default.html>

<http://w3schools.bootcss.com/sql/trysql_select_all.html?filename=trysql_select_all>

这个是在线工具，挺好用的。

|  |
| --- |
| 【方法1】select de.dept\_no ,de.emp\_no ,max(s.salary) as salary  from dept\_emp de left join salaries s on de.emp\_no = s.emp\_no  where de.to\_date = s.to\_date  group by de.dept\_no  AC的 |
| select de.dept\_no ,de.emp\_no ,s.salary  from dept\_emp de left join salaries s on de.emp\_no = s.emp\_no  where de.to\_date = s.to\_date  group by de.dept\_no  order by s.salary  为啥不行。这个应该是排序好的最大数据，而不是最终的数据。关键。 |
| 不允许select后面有不包含在group by里的非聚合函数的 |
| 记住：**所有的连接条件都必需要放在ON后面**，不然前面的所有LEFT,和RIGHT关联将作为摆设，而不起任何作用。 |
| 子查询的效率毕竟不如关联查询，所以多用连接查询。 |
| 【总结】可以用group by和聚合函数一起使用。 |
| 【方法二】select d.dept\_no, d.emp\_no , salary  from salaries as s inner join dept\_emp as d on s.emp\_no=d.emp\_no  and d.to\_date='9999-01-01' and s.to\_date='9999-01-01'  group by dept\_no **having salary=max(salary)**  这个方案是挺好的。 |
| select d.dept\_no,d.emp\_no,s.salary  from dept\_emp as d,salary as s  where d.emp\_no=s.emp\_no and d.to\_date='9999-01-01'and s.to\_date='9999-01-01'and select distinct salary from s order by salary desc limit0,1; |
| 如何选出每个部门中第二高的人。  **获取所有部门中当前员工薪水第二高的相关信息。**【思路，用子查询。就应该可以。小于最大的再用一次这个逻辑。】  select emp\_no,max(salary)  from salaries  where salary < (select max(salary) from salaries) |
| 查询高于自己部门平均工资的员工信息  select b.\* from (  select deptno,[avg](http://www.baidu.com/s?wd=avg&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3uAPWPvnYPjIBrAfsnAn30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHRYnjf1rjnLrHDYn1bLrj6vPs)([sal](http://www.baidu.com/s?wd=sal&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3uAPWPvnYPjIBrAfsnAn30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHRYnjf1rjnLrHDYn1bLrj6vPs" \t "_blank)) as [avg](http://www.baidu.com/s?wd=avg&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3uAPWPvnYPjIBrAfsnAn30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHRYnjf1rjnLrHDYn1bLrj6vPs)[sal](http://www.baidu.com/s?wd=sal&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3uAPWPvnYPjIBrAfsnAn30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHRYnjf1rjnLrHDYn1bLrj6vPs) from [emp](http://www.baidu.com/s?wd=emp&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3uAPWPvnYPjIBrAfsnAn30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHRYnjf1rjnLrHDYn1bLrj6vPs) where deptno='部门名称' group by deptno ) a,[emp](http://www.baidu.com/s?wd=emp&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3uAPWPvnYPjIBrAfsnAn30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHRYnjf1rjnLrHDYn1bLrj6vPs) b where b.[sal](http://www.baidu.com/s?wd=sal&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3uAPWPvnYPjIBrAfsnAn30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHRYnjf1rjnLrHDYn1bLrj6vPs)>a.sal and a.deptno=b.deptno |

第四题：获取当前（to\_date='9999-01-01'）薪水第二多的员工的emp\_no以及其对应的薪水salary

|  |
| --- |
| 1注意是desc，简单。  运行时间：27ms  占用内存：3196k |
| select emp\_no , salary  from salaries  where to\_date = '9999-01-01'  order by salary desc  limit 1,1 |
| 【优化】  注意: 钱最多的人可能存在多个！  select emp\_no, salary   from salaries  where salary = (select salary from salaries group by salary order by salary desc limit 1,1) |
| select emp\_no,max(salary)  from salaries  where salary < (select max(salary) from salaries) |
| 【一些需要注意的问题】  1假设这题salary最高的员工有两个a 10000,b 10000，如果用select emp\_no,salary from salaries where to\_date='9999-01-01' order by salary desc limit 1,1;去做的话，那么这条记录取出来是一样的salary 10000（亲自测试）；  下面的这条语句可以避免这个问题：  select emp\_no, salary from salaries where salary = (select salary from salaries group by salary order by salary desc limit 1,1)  2第二多到底指什么还不明确吧。比如100,100，99。如果按照这个顺序的话，直接返回第二个100；但是如果按照相同的数都在一个等级，那么第二多应该是99。 |

第5题：

|  |
| --- |
| 查找当前薪水(to\_date='9999-01-01')排名第二多的员工编号emp\_no、薪水salary、last\_name以及first\_name，不准使用order by |
| 运行时间：31ms  占用内存：3068k |
| select em.emp\_no , max(s.salary) as salary ,last\_name, first\_name  from employees em left join salaries s on em.emp\_no = s.emp\_no  where salary <  (select max(salary)  from salaries  where to\_date = '9999-01-01'  ) |
| 剔除可能不是一个的情况。  select e.emp\_no, s.salary, e.last\_name, e.first\_name from employees as e, salaries as s where e.emp\_no  = s.emp\_no and s.salary = (select max(salary) from salaries where salary != (select max(salary) from salaries where to\_date='9999-01-01') )and s.to\_date = '9999-01-01' |
|  |
|  |