

一、关系模式如下：

- 职工 E (姓名 **ename**, 工号 **eno**, 出生日期 **bdate**, 家庭地址 **addr**, 工资 **salary**, 所在部门编号 **dno**)
- 部门 D (部门名称 **dname**, 部门编号 **dno**, 部门负责人的工号 **mgreno**)
- 项目 P (项目名称 **pname**, 项目编号 **pno**, 所在城市 **city**, 主管部门编号 **dno**)
- 工作 W (职工工号 **eno**, 项目编号 **pno**, 工作时间 **hours**)
- 职工家属 Depend (职工工号 **eno**, 姓名 **name**, 性别 **sex**)

请用 SQL 语言来表示下述数据查询操作。

1) 检索所有部门负责人的工号和姓名；

```
select eno, ename  
from E, D  
where E.eno=D.mgreno
```

2) 检索职工 Smith 所参与的项目的编号和名称；

```
select pno, pname  
From E, W, P  
where ename = 'Smith' and E.eno = W.eno and W.pno=P.pno
```

3) 检索拥有两个或两个以上家属的职工的姓名；

(答案 1)

```
Select ename  
from E, Depend D1, Depend D2  
Where E.eno = D1. eno and D1. eno = D2. eno and D1.name <> D2.name
```

(答案 2)

```
select ename  
from E  
Where 2 <= some ( select count(*)  
from Depend  
Where E.eno = Depend.eno )
```

注：量词 some 也可以被替换为 any 或者 all

(答案 3)错误用法！

```
select ename  
from E, Depend  
Where E. eno = Depend. eno  
group by E.eno  
having Count(*) >= 2
```

注：答案 3 的错误在于 SELECT 子句和 GROUP BY 子句不相符。

4) 检索不带家属的职工的姓名；

```
select ename  
from E  
where eno NOT IN ( select eno from Depend )
```

注：也可以使用 NOT EXISTS 谓词来表示该查询

5) 检索只参加过‘p2’号项目的职工的姓名;

(答案 1)

```
Select eno from W
where eno NOT IN (select eno from W where pno<>'p2')
```

(答案 2)

```
Select eno from W W1
where NOT EXISTS ( select * from W W2
where W2.pno<>'p2' and W2.eno=W1.eno )
```

6) 检索只参加过一个项目的职工的姓名;

(答案 1)

```
Select ename from E
Where 1 = some ( select count(*) from W where W.eno=E.eno)
```

(答案 2)

```
Select ename from E
Where eno IN (select eno from W group by eno having count(*)=1)
```

注: 要避免犯第 3) 题 (答案 3) 中的错误!

(答案 3)

```
Select ename
From E, (select eno, count(*) as pnumber from W group by eno) X
Where E.eno=X.eno and X.pnumber=1
```

7) 检索参加了所有项目的职工的工号;

(答案 1)

```
Select eno from E Where NOT EXISTS (
Select * from P where NOT EXISTS (
Select * from W where W.eno=E.eno and W.pno=P.pno))
```

(答案 2) 第二个 NOT EXISTS 可以改用 NOT IN 谓词来表示!

```
Select eno from E Where NOT EXISTS (
Select * from P where P.pno NOT IN (
Select W.pno from W where W.eno=E.eno))
```

8) 检索全体 3 号部门的职工都参加了的项目的编号和名称;

(答案 1)

```
Select pno, pname from P where NOT EXISTS (
Select * from E where E.dno=3 and NOT EXISTS (
Select * from W where W.eno=E.eno and W.pno=P.pno))
```

(答案 2) 略。

9) 检索工资收入最高的职工的姓名；(请给出使用统计函数和不使用统计函数的两种写法)
下面分别给出不使用统计函数（答案 1、2、3）和使用统计函数（答案 4）的写法。

(答案 1)

```
Select  ename from  E  E1
where  NOT EXISTS ( select  *  From  E  E2  where  E2.salary > E1.salary )
```

(答案 2)

```
Select  ename from  E
where  eno NOT IN (
    select  E1.eno From  E E1, E E2  where  E1.salary < E2.salary )
```

(答案 3)

```
Select  ename  from  E
where  salary >= ALL (select  salary  From  E )
```

(答案 4)

```
Select  ename  from  E
where  salary IN (select  max(salary) From  E )
```

10) 查询每一个部门中工资收入最高的职工，结果返回部门编号以及该部门中工资收入最高的职工的工号。

(答案 1)

```
Select  dno, eno from  E  E1
where  NOT EXISTS ( select  *  From  E  E2
                    where  E2.dno=E1.dno  and  E2.salary>E1.salary )
```

(答案 2)

```
Select  dno, eno  from  E
where  eno NOT IN (select  E1.eno From  E E1,  E E2
                    where  E1.dno=E2.dno  and  E1.salary<E2.salary )
```

(答案 3)

```
Select  dno, eno  from  E E1
where  salary >= ALL ( select E2.salary  From E E2  where E2.dno=E1.dno )
```

(答案 4)

```
Select  dno, eno  from  E E1
where  salary IN ( select max(E2.salary) From E E2  where E2.dno=E1.dno )
```

二、假设存在如下的关系模式

Customers (cid, cname, city, discnt)

Agents (aid, aname, city, percent)

Products (pid, pname, city, quantity, price)

Orders (ordno, orddate, cid, aid, pid, qty, dollars)

注: orddate 是一个日期类型(timestamp)的属性, 用于记录订单的创建日期, 日期类型的值可以进行大小比较。

请用 SQL 语言来表示下述数据操作要求。

- 1) 统计每一种商品的订单数, 销售总金额, 及其单笔订单的最高金额, 并按照销售总金额的降序输出查询结果;

```
Select pid, count(*), sum(dollars), max(dollars)
```

```
From orders
```

```
Group by pid
```

```
Order by sum(dollars) DESC
```

- 2) 检索满足下述条件的商品的编号: 单价超过 100 美元, 且居住在 Dallas 的所有客户都购买过;

```
Select  pid    From  products P  Where  price>100 and not exists (  
        Select *  From  Customers C  Where  C.city='Dallas' and not exists (  
            Select *  From  orders O  Where  O.pid=P.pid and O.cid=C.cid ))
```

- 3) 检索每一个客户的最近一份订单的订单编号、订购日期和订购金额; (注: 最近一份订单是指该客户的订购日期 orddate 最大的订单)

(可以参考第一大题第 10 小题的做法)

```
Select  cid, ordno, orddate, dollars
```

```
From  orders
```

```
where  ordno NOT IN ( select  O1.ordno  
                        From  orders O1, orders O2  
                        where  O1.cid=O2.cid  and  O1.orddate<O2.orddate)
```

- 4) 检索符合下述要求的客户的编号: 在该客户的所有订单中, 每一种商品的平均每笔订单的订购数量均达到或超过 300。

```
Select  cid
```

```
From  (select cid, pid, avg(qty) as qty_avg from orders group by cid, pid) T
```

```
Group by  T.cid
```

```
Having  min(T.qty_avg)>=300
```

- 5) 设销售金额 **dollars** 达到 1 万元的订单被称为‘大订单’，‘大订单’的数目大于等于 10 的客户被称为‘大客户’。对于每一个‘大客户’，统计其‘大订单’的平均订购金额，并按照平均订购金额的降序输出查询结果。

```
Select  cid, avg(dollars)
From    orders
Where   dollars >= 10000
Group by cid
Having  count(*) >= 10
Order by avg(dollars) DESC
```

- 6) 从 **Orders** 表中删除那些订购金额为空的订单元组；

```
Delete From orders where dollars is null;
```

- 7) 从 **Agents** 表中删除没有销售过商品的供应商元组；

```
Delete From agents where aid not in (select aid from orders)
```

- 8) 将累计购买金额超过 1 万元的客户的折扣 **discnt** 增加 10%

```
Update  customers
Set     discnt = discnt * 1.1
Where   cid in ( select cid from orders
                  group by cid having sum(dollars) > 10000)
```

- 9) 请在 **Customers** 表中插入一个编号为 c0009，名字为 John 的客户元组；

```
Insert into customers(cid, cname) value('c0009', 'John')
```

- 10) 假设有个 VIP 客户表 **v_customer**(表的结构与 **Customers** 完全一样)，请查出累计购买金额超过 1 万元的客户，并将其插入到 **v_customer** 表中。

```
Insert into v_customer
select *
from customers
where cid IN (select cid from order
              group by cid having sum(dollars) > 10000 );
```