

## mysql开发技巧 2

如何进行行列转换

如何生成唯一序列化

如何删除重复数据

行列转换的场景

### 1 报表统计

日期	销售金额
2015/1/1	111422.63
2015/1/2	85516.79
2015/1/3	95423.23
2015/1/4	160989.23
2015/1/5	152318.29
2015/1/6	165338.73
2015/1/7	166051.35
...	...
...	...
2015/3/31	105285.37
2015/4/1	104659.66
2015/4/2	94135.71
2015/4/3	89632.3
2015/4/4	84380.02
2015/4/5	79656.63

月份	1月	2月	3月
销售额	7043816.7	8057446	3176372.95

### 2 汇总显示

汇总显示

姓名	学科	成绩
张三	数学	78
张三	英语	85
张三	语文	90
李四	数学	90
李四	英语	75
李四	语文	80

使用自连接（使用这种行转列的方式 效率低 如果用户输入越多，我们查询的表越多）

```
Select *
```

```
From (
```

```
select sum(kills) as '沙僧' from user1 a join  
user_kills b on a.id =b.user_id and  
a.`user_name`='沙僧'
```

```
) a cross join (
```

```
select sum(kills) as '猪八戒' from user1 a join  
user_kills b on a.id =b.user_id and  
a.`user_name`='猪八戒'
```

```
) b cross join (
```

```
select sum(kills) as '孙悟空' from user1 a join  
user_kills b on a.id =b.user_id and  
a.`user_name`='孙悟空'
```

## 使用case语句进行行列转换

```
select sum(case when user_name='孙悟空' then kill end) as '孙悟空',  
sum(case when user_name='猪八戒' then kill end) as '猪八戒',  
sum(case when user_name='沙僧' then kill end) as '沙僧'  
from user1 a join user_kills b on a.id = b.user_id;
```

## 单列转多行

### 属性拆分

operator_id	op_userid	permission
1	10001	10070501,13011104,13011105,13010403
70	10002	14040101,14050202,14090104,14090101
72	10003	4130104,14130107,14130108
73	10004	10100904,10100905
74	10005	14010101

operator_id	op_userid	permission
1	10001	10070501
1	10001	13011104
1	10001	13011105
1	10001	13010403

还需要列转多行的场景

ETL数据处理

### ETL数据处理

user_name	mobile
唐僧	12112345678,14112345678,16112345678
猪八戒	12144643321,14144643321
孙悟空	12166666666,14166666666,16166666666,18166666666
沙僧	12198343214,14198343214

user_name	mobile
唐僧	12112345678
唐僧	14112345678
唐僧	16112345678
孙悟空	12166666666
孙悟空	14166666666

## 进行行列转换

### 如何进行行列转换

#### 利用序列表处理列转行的数据

```
SELECT user_name  
      ,REPLACE(SUBSTRING(SUBSTRING_INDEX(mobile,',',a.id),CHAR_LENGTH(  
SUBSTRING_INDEX(mobile,',',a.id-1)) + 1),',', '') AS mobile  
FROM tb_sequence a CROSS JOIN (  
      SELECT user_name,CONCAT(mobile,',') AS mobile,LENGTH(mobile)-  
LENGTH(REPLACE(mobile,',',''))+1 size  
FROM user1 b ) b ON a.id <= b.size
```

## 列转行

```
select user_name,replace(substring(substring_index(mobile,',',a.id),char_length(substring_index(mobile,',',a.id-1))+1),',', '') as mobile  
from tb_sequence as a  
cross join(  
select user_name,concat(mobile,',') as mobile,length(mobile)-length(replace(mobile,',',''))+1 as size from user1 as b) as b on  
a.id<=b.size
```

## 使用UNION方法实现列转行

```
select user_name,'arms' as equipment, arms from user1 a join user1_equipment b on a.id=user_id  
union all  
select user_name,'clothing' as equipment, clothing from user1 a join user1_equipment b on a.id=user_id;  
union all  
select user_name,'shoe' as equipment, shoe from user1 a join user1_equipment b on a.id=user_id;
```

### 需要进行列转行的另一种场景

username	arms	clothing	shoe
唐僧	九环锡杖	锦斓袈裟	僧鞋
猪八戒	九齿钉耙	僧衣	僧鞋
孙悟空	金箍棒	锁子黄金甲	藕丝步云履
沙僧	降妖宝杖	僧衣	僧鞋

username	equipment	eq_name
唐僧	arms	九环锡杖
唐僧	clothing	锦斓袈裟
唐僧	shoe	僧鞋
孙悟空	arms	金箍棒
孙悟空	clothing	锁子黄金甲
孙悟空	shoe	藕丝步云履

tb\_equipment我们自己建立的序列化表  
这个模样

id
1
2
3
4
5

使用序列化表的方式来实现列行转换

```
select user_name,  
case when c.id=1 then 'arms'  
when c.id=2 then 'clothing'  
when c.id=3 then 'shoe'  
end as equipment,  
coalesce(case when c.id=1 then arms end,  
case when c.id=2 then clothing end,  
case when c.id=3 then shoe end) as eq_name  
from user1 a  
join user1_equipment b on a.id=b.user_id  
cross join tb_sequence c where c.id<=3 order by user_name;
```

一行转三行的方法思路 cross join tb\_sequence c where c.id<=3

coalesce ( ) 函数的作用： 如果第一组为空，就取第二组， 如果第二组非空，就取第二组， 如果第二组为空，就取第三组

生成唯一序列号

- 1 数据库主键 (尽可能小，并且保证唯一不重复)
- 2 业务序列号 如：发票号，车票号，订单号 等。。。

## 生成序列号的方法



优先选择系统提供的序列号生成的方式（对并发的控制，有很好的并发效果 可以保持它的不重复与唯一，但是系统本身生成的序列化的生成方式呢，1,2 跳过3 直接4,5,6）

另外一种生成方式 就是使用sql语句来生成序列化的方式  
在特殊的情况下使用sql方式生成序列号

需求：生成订单序列号，并且订单号的格式如下  
YYYYMMDDNNNNNNNN 如201505120000003  
不会重复的  
由于会累加，所以不会重复

```
UPDATE order_seq SET order_sn=order_sn+v_cnt V
```

## 用SQL来生成特殊的序列号

```
DECLARE v_cnt INT;
DECLARE v_timestr INT;
DECLARE rowcount BIGINT;
SET v_timestr=DATE_FORMAT(NOW(),'%Y%m%d');
SELECT ROUND(RAND()*100,0)+1 INTO v_cnt;
START TRANSACTION;
    UPDATE order_seq SET order_sn=order_sn+v_cnt WHERE timestr=v_timestr;
    IF ROW_COUNT()=0 THEN
        INSERT INTO order_seq(timestr,order_sn) VALUES(v_timestr,v_cnt);
    END IF;
    SELECT CONCAT(v_timestr,LPAD(order_sn,7,0)) AS order_sn
    FROM order_seq WHERE timestr=v_timestr;
COMMIT;
```

可以每秒生成1000个订单号

删除重复数据  
查看重复的sql

```
SELECT user_name,COUNT(*)
FROM user1_test
GROUP BY user_name HAVING COUNT(*) > 1
```

删除重复数据

### 删除重复数据，对于相同数据保留ID最大的

```
DELETE a
FROM user1_test a JOIN (
    SELECT user_name,COUNT(*),MAX(id) AS id
    FROM user1_test
    GROUP BY user_name HAVING COUNT(*) > 1
) b ON a.user_name = b.user_name
WHERE a.id < b.id
```