**插入数据：**

insert into department (name,date) values ('department3',str\_to\_date('2012-05-01 23:59:59','%Y-%m-%d %T'));

**注意：**

1、以上日期格式是针对mysql,oracle日期格式应该是to\_date('2014-02-14 20:47:00','yyyy-mm-dd hh:mm:ss')

2、插入语句中列明不需要引号，只有列值需要，跟查询语句类似

**删除数据：**

delete from department where department\_id='3';

truncate table department;

**注意：**

删除列的时候可以用update直接将列值置成空。

**修改数据 + 3种模糊限制条件：**

update addresstable set phone='07104272553' where first\_name like '阿%' and score between 80 and 82 and job\_title in ('中级测试工程师07','中级测试工程师08');

**限制查询 + 更改列名 + 查询空行 ：**

mysql> select first\_name as 小名,last\_name as 姓 from addresstable where phone is null limit 0,3;

+--------+----+

| 小名 | 姓 |

+--------+----+

| 雪莲03 | 朱 |

| 雪莲04 | 朱 |

| 阿雪01 | 张 |

+--------+----+

3 rows in set

**查询中使用常量：**

mysql> select first\_name as 小名,'朱' as 姓 from addresstable where phone is null limit 0,3;

+--------+----+

| 小名 | 姓 |

+--------+----+

| 雪莲03 | 朱 |

| 雪莲04 | 朱 |

| 阿雪01 | 朱 |

+--------+----+

3 rows in set

**注意：**

1、查询中使用常量的意思是新增加一列查询结果全是常量的值

2、只有sqlserver支持top关键字体，Oracle和mysql不支持top，所以上面是用的limit

**联接查询 + 排序查询 + 分组查询：**

**1、含函数时，分组查询和非分组查询区别**

mysql> Select c.name,a.first\_name,b.name,sum(a.score) from addresstable a,department b,employee c where a.id=c.address\_id and b.department\_id=c.department\_id;

+----------+------------+--------------+--------------+

| name | first\_name | name | sum(a.score) |

+----------+------------+--------------+--------------+

| 朱雪莲01 | 雪莲01 | Department 1 | 668 |

+----------+------------+--------------+--------------+

1 row in set

mysql> Select c.name,a.first\_name,b.name,sum(a.score) from addresstable a,department b,employee c where a.id=c.address\_id and b.department\_id=c.department\_id group by c.name,a.first\_name,b.name;

+----------+------------+--------------+--------------+

| name | first\_name | name | sum(a.score) |

+----------+------------+--------------+--------------+

| 张雪莲05 | 阿雪01 | Department2 | 87 |

| 张雪莲06 | 阿雪02 | Department2 | 82 |

| 张雪莲07 | 阿雪03 | Department2 | 81 |

| 张雪莲08 | 阿雪04 | Department2 | 80 |

| 朱雪莲01 | 雪莲01 | Department 1 | 83 |

| 朱雪莲02 | 雪莲02 | Department 1 | 84 |

| 朱雪莲03 | 雪莲03 | Department 1 | 85 |

| 朱雪莲04 | 雪莲04 | Department 1 | 86 |

+----------+------------+--------------+--------------+

8 rows in set

**2、同一个字段，即两行的和或者平均值排序查询**

mysql> Select c.name,a.first\_name,b.name,sum(a.score) as 成绩综合 from addresstable a,department b,employee c where a.id=c.address\_id and b.department\_id=c.department\_id group by c.name,a.first\_name,b.name order by 成绩综合 desc;

+----------+------------+--------------+----------+

| name | first\_name | name | 成绩综合 |

+----------+------------+--------------+----------+

| 张雪莲05 | 阿雪01 | Department2 | 87 |

| 朱雪莲04 | 雪莲04 | Department 1 | 86 |

| 朱雪莲03 | 雪莲03 | Department 1 | 85 |

| 朱雪莲02 | 雪莲02 | Department 1 | 84 |

| 朱雪莲01 | 雪莲01 | Department 1 | 83 |

| 张雪莲06 | 阿雪02 | Department2 | 82 |

| 张雪莲07 | 阿雪03 | Department2 | 81 |

| 张雪莲08 | 阿雪04 | Department2 | 80 |

+----------+------------+--------------+----------+

8 rows in set

**3、不同字段，即两列和或者两列的平均值的查询语句**

mysql> Select c.name,a.first\_name,b.name,sum(a.score+a.score2) from addresstable a,department b,employee c where a.id=c.address\_id and b.department\_id=c.department\_id group by c.name,a.first\_name,b.name order by sum(a.score+a.score2) desc;

+----------+------------+--------------+-----------------------+

| name | first\_name | name | sum(a.score+a.score2) |

+----------+------------+--------------+-----------------------+

| 朱雪莲03 | 雪莲03 | Department 1 | 175 |

| 张雪莲06 | 阿雪02 | Department2 | 172 |

| 张雪莲05 | 阿雪01 | Department2 | 167 |

| 朱雪莲02 | 雪莲02 | Department 1 | 164 |

| 张雪莲07 | 阿雪03 | Department2 | 161 |

| 张雪莲08 | 阿雪04 | Department2 | 160 |

| 朱雪莲04 | 雪莲04 | Department 1 | 156 |

| 朱雪莲01 | 雪莲01 | Department 1 | 153 |

+----------+------------+--------------+-----------------------+

mysql> Select c.name,a.first\_name,b.name,(a.score+a.score2)/2 from addresstable a,department b,employee c where a.id=c.address\_id and b.department\_id=c.department\_id group by c.name,a.first\_name,b.name order by (a.score+a.score2)/2 desc;

+----------+------------+--------------+----------------------+

| name | first\_name | name | (a.score+a.score2)/2 |

+----------+------------+--------------+----------------------+

| 朱雪莲03 | 雪莲03 | Department 1 | 87.5000 |

| 张雪莲06 | 阿雪02 | Department2 | 86.0000 |

| 张雪莲05 | 阿雪01 | Department2 | 83.5000 |

| 朱雪莲02 | 雪莲02 | Department 1 | 82.0000 |

| 张雪莲07 | 阿雪03 | Department2 | 80.5000 |

| 张雪莲08 | 阿雪04 | Department2 | 80.0000 |

| 朱雪莲04 | 雪莲04 | Department 1 | 78.0000 |

| 朱雪莲01 | 雪莲01 | Department 1 | 76.5000 |

+----------+------------+--------------+----------------------+

**2、下面是查某个字段值相同时出现的次数，即字段值相同的时候的行数**

mysql> Select phone,count(phone) from addresstable group by phone having count(phone)>1;

+-------------+--------------+

| phone | count(phone) |

+-------------+--------------+

| 07104272553 | 2 |

+-------------+--------------+

mysql> Select phone,count(phone) from addresstable group by phone order by count(phone) desc;

+-------------+--------------+

| phone | count(phone) |

+-------------+--------------+

| 07104272553 | 2 |

| 0214272558 | 1 |

| | 1 |

| NULL | 0 |

+-------------+--------------+

**注意：**

因为有函数，所以不能用where只能用having，此时的having就相当于where

**一张表同一个字段比较大小，或不同字段比较大小**