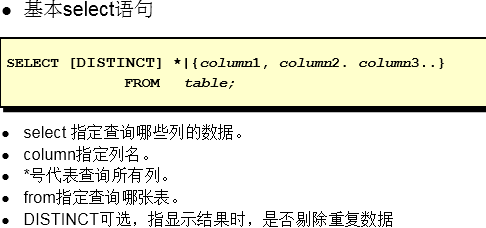
# Select简单查询 s – f

## 基本关键字概述



语法：

select \* from 表名; 查询所有（字段）

select 字段名1,字段名2,字段名3 from 表名; 显示查询字段名

select DISTINCT 字段名 from 表名; 去除重复的数据。

练习：

create table stu(

id int primary key auto\_increment,

name varchar(20),

math int,

english int,

chinese int

);

insert into stu values(null , '老兰' , 52 , 62 , 62);

insert into stu values(null , '美甲' , 12 , 62 , 62);

insert into stu values(null , '美女' , 42 , 62 , 62);

insert into stu values(null , '美美' , 12 , 62 , 62);

insert into stu values(null , '猪' , 52 , 62 , 62);

查询表中所有学生的信息。

select \* from stu;

查询表中所有学生的姓名和对应的英语成绩。

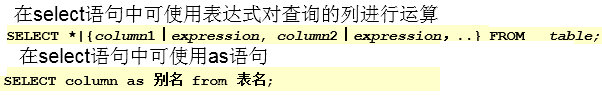
select name,english from stu;

过滤表中重复数据。（只是单纯的去除结果中的重复元素，而且联合重复）

select distinct english from stu;

## 查询结果

1.所有 \* 2.指定列 3.计算后的值（函数 、 运算符）



练习：

在所有学生分数上加10分特长分。

select name,math+10,english+10,chinese+10 from stu;

统计每个学生的总分。

select name,math+english+chinese from stu;

使用别名表示学生分数。

select name,(math+english+chinese) as sum from stu;

## distinct

DISTINCT不同，DISTINCT不能部分使用。换句话说，DISTINCT关键字应用于所有列而不 仅是它后面的第一个指定列。例如，查询3个字段s\_id，f\_name，f\_price，如果不同记录的这3个字段的组合值都不同，则所有记录都会被查询出来。

# where条件 w –

## 条件

查询姓名为班长的学生成绩

select \* from stu where name='班长';

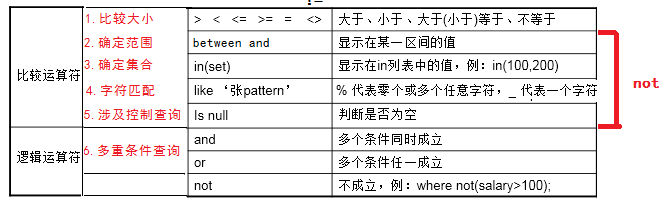
查询英语成绩大于90分的同学

select name,english from stu where english < 15;

查询总分大于200分的所有同学

select name,math+english+chinese from stu where (math+english+chinese) > 200;

## 运算符



练习

查询英语分数在 80－90之间的同学。

select \* from stu where english >80 and english <90;

select \* from stu where english between 80 and 90;

查询数学分数为18,78,46的同学。（in）

select \* from stu where math in(18,78,46);

查询所有姓班的学生成绩。

select \* from stu where name like '班%';

查询数学分>80，语文分>80的同学。

select \* from stu where math >80 or chinese > 80;

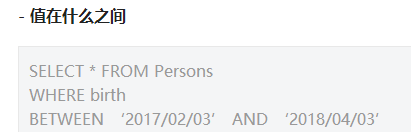
### 模糊查询%和\_



### in

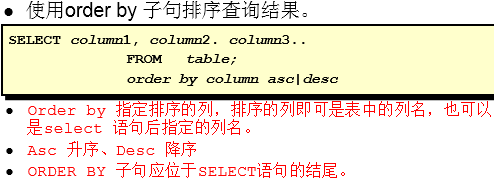


### Between and



# 排序

## Order by



升序默认的(asc)/降序(desc)， 出现select的语句末尾。

练习：

对数学成绩排序后输出。

select name,math from stu order by math;

对总分排序按从高到低的顺序输出

select name,math+english+chinese from stu order by (math+english+chinese) desc;

对学生成绩按照英语进行降序排序，英语相同学员按照数学降序(先满足第一个条件)

select \* from stu order by english desc,math desc;

对姓美的学生成绩排序输出

select \* from stu where name like '美%' order by english desc;

## desc

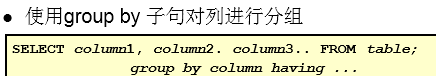
默认asc

只对一个生效

# 分组查询

通过那个和哪些字段进行分组

group by 分组（一起使用） 条件过滤需要是having，不能使用where



练习：

create table orders(

id int,

product varchar(20),

price float

);

insert into orders(id,product,price) values(1,'电视',900);

insert into orders(id,product,price) values(2,'洗衣机',100);

insert into orders(id,product,price) values(3,'洗衣粉',90);

insert into orders(id,product,price) values(4,'桔子',9);

insert into orders(id,product,price) values(5,'洗衣粉',90);

insert into orders(id,product,price) values(6,'电视',900);

对订单表中商品归类后，显示每一类商品的总价.

select product,count(\*),sum(price) from orders group by product;

使用having 子句 对分组结果进行过滤

练习：

查询购买了几类商品，并且每类总价大于100的商品

select product,sum(price) from orders group by product having sum(price) > 100;

# 分页查询—mysql独有

## limit

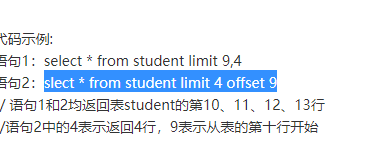






## offset

https://blog.csdn.net/l1212xiao/article/details/80520330



# 查询顺序

