数据库mysql

显示数据库：Show databases；

创建自己的用户数据库：create database 数据库名字；

用创建好的数据库：use 数据库名字；

看数据库中的表格：show tables;

建表语句（列名就是字段）(自动递增)：

create table 表名字(

Stuid int primary key auto\_increment,

Stuname varchar(10) not null,

Stuscore float

);

看表的结构：desc 表名；

删除表：drop table 表名；

查询表数据：select \* from 表名；

增（向表里面写数据）：insert into 表名 values(1,’Tom’,9);

向表中插入多行数据：

Insert into 表名

Select 1,’dudu’,78 union

Select 1,’dudu’,78 union

Select 1,’dudu’,78 ;

查询其他表中的数据插入到此表中：

Insert into 表名

Sleect stuid,stuname,stuscore from students;

删除表数据：delete from 表名 ；（表里面数据全删除）

不想全部删除要加条件：

Delete from 表名 where stuid=1；删除stuid为1的数据

更新表：update 表名 set stuname=‘xyz’，stuscore=80 where stuid=3;

设置字符集简体中文：Set charset gbk;

主键：唯一不重复的字段。

创建外键：

foreign key(stucourse) references course(cid);

模糊查询：

like：%表示任意多个字符，下划线表示任意

一个字符，like用在varchar数据类型上面。

表示查询包含肉字的数据：



表示查询一共三个字，最后一个字为肉的数据：





范围查询：

between and 从多少到多少

is：空和非空的查询

in :在什么什么里面



查询函数：

Count（\*）查询所有数据的总行数。

Sum()求和，括号里面写上要算的字段名



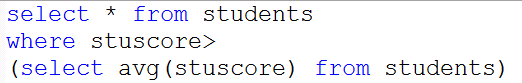
avg（）求平均值



max()求最大值

mix()求最小值

查找成绩比平均成绩高的所有学生的信息



Order by 字段名；根据字段名排序



a表示开始位置下标。b表示找几个数据。Limit a,b



查询语句例子：查询川菜粤菜当中，价格在20到50 的最便宜的三个菜。

select \* from dish where dishcookstyle in ('川菜','粤菜') and dishprice between 20 and 50

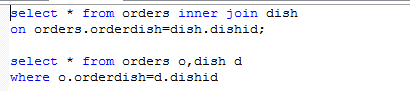
order by dishprice limit 0,3;

group by having count(\*);分组

查询不同表的数据：

表名1.外键=表名2.主键，这样来连接表。

两种方法查询有练习的表。

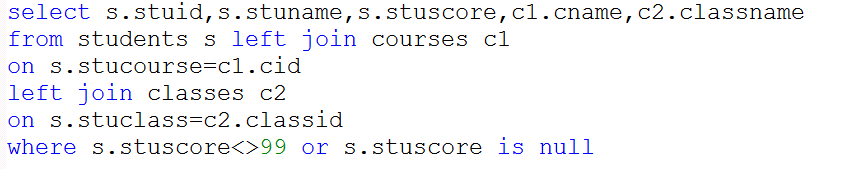


外连接：

右外连接，以右表为主。左外连接，以左表为主。



三表连接查询



排除重复：distinct



交叉连接：



database administrator

dba

去IOE

IBM小型机

Oracle数据库

EMC存储设备

MySQL

DB2

SQL Server

1.show databases;-显示数据库

2.create database mydb;-创建数据库

3.use mydb;-使用数据库

4.show tables;-显示表

5.-建表:列名（字段）行（记录）

create table students(

stuid int primary key auto\_increment,

stuname varchar(50) not null,

stuscore float,

foreign key(stucourse) references courses(cid)

);

6.desc students;-描述表

7.drop table students;-删表

drop database mydb;-删数据库

8.select \* from students;-查询表数据

9.insert into students values(null,'Tom',null);-往表里加数据(单条)

insert into persons（指定多条）

select 5,'abc',2 union

select 6,'xyz',3 union

select 7,'Marry',1

insert into persons（从表查出来的数据加多条）

select id,stuname,stuclass from students where id>0;

10.delete from students where stuname='Tom' or stuname='Jack';-删除数据（通过条件用where）

11.update students set stuname='xyz',stuscore=88.8 where stuid=3;-更新数据（通过条件用where）

12.set charset gbk;-设置字符集简体中文

主键：用来表示每一条记录是不同，primary key=unique+not null,不能为空，主键值唯一不能重复,每一张表都要有个主键

模糊查询：

and or not

1 like: %（任意多个字符） \_(任意一个字符) 用在varchar列

（1）查姓王的学java的学生的信息

（2）名字含有王字的成绩及格的学生的信息

（3）查姓王的，并且名字三个字的学生的信息

（4）没成绩和不及格的学生放到补考表里

（5）学java和数据库的成绩在90分以上的学生信息

2 between and:范围查询

3 is:空和非空查询

（1）川菜和粤菜价格20~50并且类型不能为空的菜品

（2）用微信支付并且已经支付完成的订单

4 in:

常用函数：

1 count():求个数，写字段名，得到非空的个数。得到表的数据的总行数：count(\*)

2 sum():求和

3 avg():求平均值

select \* from students where stuscore>(select avg(stuscore) from students)

4 max():求最大

5 min():求最小

排序：order by ,asc(升序)，desc(降序) 有where先写，再写order by

limit 第一个参数-(开始位置，从0开始)，第二个参数-(从该位置往下找几条)

分页的开始位置：start=(page-1)\*rows：start-(每个分页的开始位置),page-(页号，从1开始)，rows-(每一个分页要显示的行数)

分组：group by where-group by-having

表连接：内连接：表的地位相等，查询结果是匹配的

1.select s.stuid as 学号,s.stuname 名字,s.stuscore 成绩,c.cname 课程名

from courses c,students s

where c.cid=s.stucourse

2.select s.stuid as 学号,s.stuname 名字,s.stuscore 成绩,c.cname 课程名

from students s inner join courses c

on c.cid=s.stucourse

where s.stuscore is not null

select s.stuid 学号,s.stuname 名字,s.stuscore 成绩,

c2.classname 班级名,c1.cname 课程名

from students s,courses c1,classes c2

where s.stucourse=c1.cid and s.stuclass=c2.classid

select s.stuid 学号,s.stuname 名字,s.stuscore 成绩,

c1.cname 课程名,c2.classname 班级名

from students s inner join courses c1

on s.stucourse=c1.cid

inner join classes c2

on s.stuclass=c2.classid

自连接：

select s1.menuid 编号,s1.menname 菜单名,s2.menname 上级菜单名

from systemmenu s1,systemmenu s2

where s1.superid=s2.menuid

select s1.menuid 编号,s1.menname 菜单名,s2.menname 上级菜单名

from systemmenu s1 inner join systemmenu s2

on s1.superid=s2.menuid

where s2.menname='文件'

外连接：

左外连接-左表为主，地位高，查出左表所有的数据，有匹配的显示匹配的数据，没有匹配上的显示空

left join

select s1.menname,s2.menname from systemmenu s1 left join systemmenu s2 on s1.superid=s2.menuid

右外连接：right join-与左外连接相对应

交叉连接：cross join-笛卡尔积

distinct:排除重复