1、**数据库**：长期存储在计算机内，存储数据的仓库。

2、**数据库-关系型数据库：**

RDMS:关系型数据库将数据的关系已数据库表的形式加以表达，并将数据存储在表格中。

分类：Oracle、Mysql、SQL Server、SQLite

核心要素：

数据行（一条记录）

数据列（字段）

数据表（数据行的集合）

数据库（数据表的集合，一个数据库中能够有n多个数据表）

3、**数据库-非关系型数据库：**

NoSQL:非关系型数据库，以key-value键值对形式存储。

分类：Redis、Mongodb、Hbase

4、**SQL的分类**

SQL是一个结构化的查询语言，通过SQL能够对数据库进行相关的操作。

DQL：数据查询语言，用于对数据进行查询，例如：select

DML：数据操作语言，对数据进行增加、删除，例如：insert、update、delete

TPL：事务处理语言，对事物进行处理，例如：begin transaction、commit、rollback

DCL：数据控制语言，进行授权与权限回收，例如：grant、revoke

DDL：数据定义语言，进行数据库、表的管理等，例如：create、drop

CCL：指针控制语言，通过控制指针完成的表的操作，例如：declare cursor

注意：主流的关系型数据库都支持SQL，在MuSql中SQL语言不区分大小写。

5、**Mysql特点**

可移植性、支持多操作系统、支持多种编程语言、开源、支持多线程、支持多语言编码格式。

6、**Mysql组成及连接**

Mysql服务器：存储数据并解析编译后的SQL语句，将执行结果返回给客户端

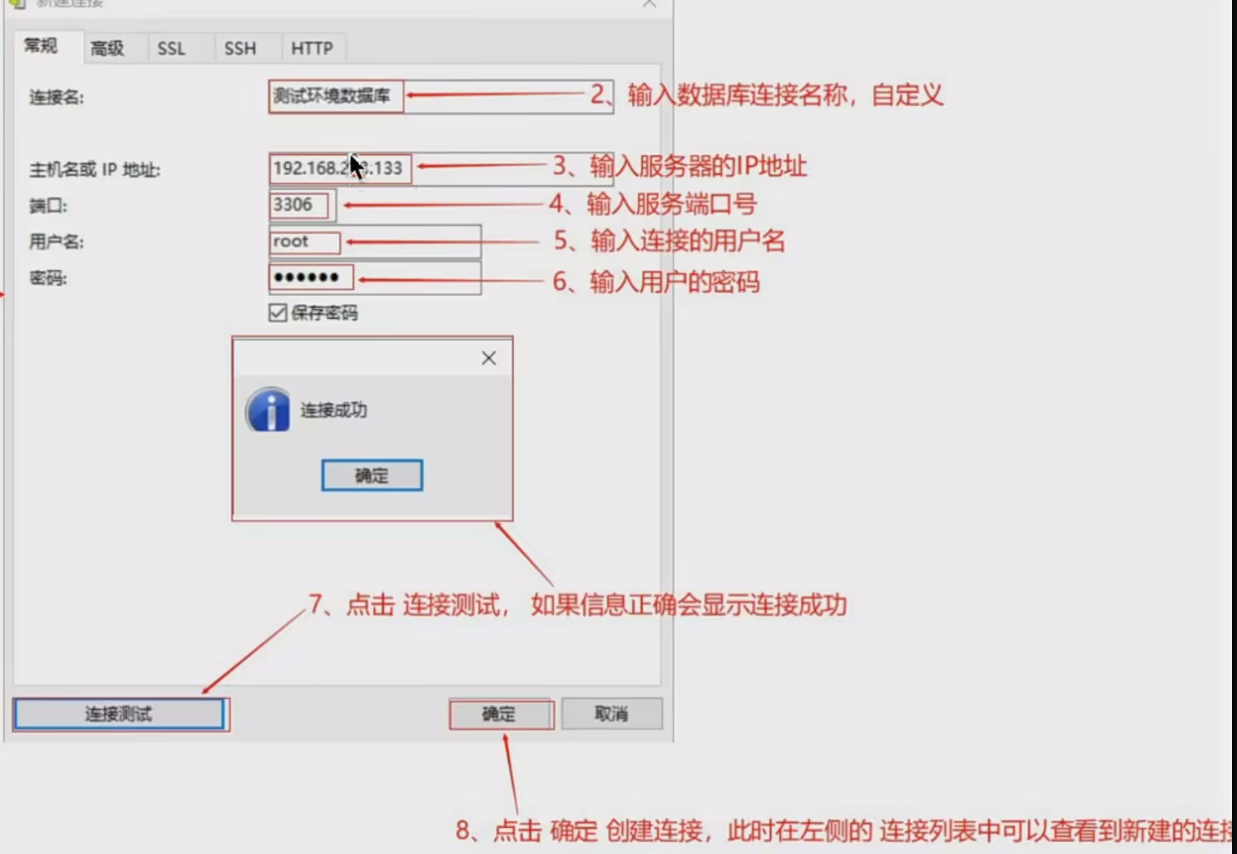
Mysql客户端：下发用户要执行的SQL语句，并显示服务器返回的执行结果

连接Mysql： mysql -h数据库IP -P端口号 -u数据库登录用户名 -p数据库登录密码

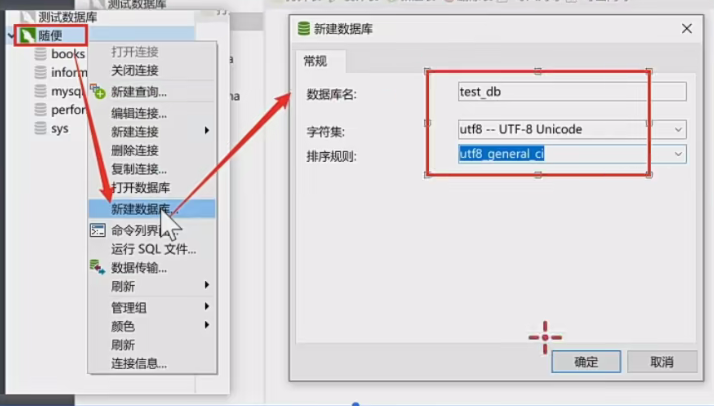
-h不加表示为本机-P不加表示默认3306端口

7、**navicat**

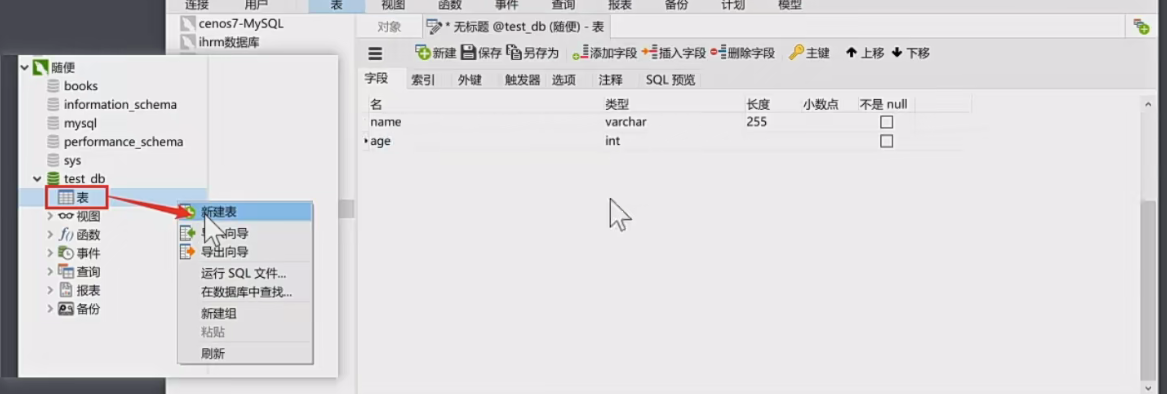
1.连接mysql



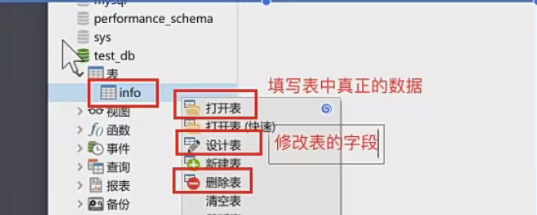
2.新建数据库



3.新建操作表



4.删除和修改



8、**mysql常用数据类型**

整型：int

字符串型：varchar，varchar(3)表示最多三个字符，一个字母或一个中文点一个字符

小数：decimal，decimal(5,2)表示存5位数，其中小数位数为2

日期型：datetime

9、**数据约束**

主键（prrimary key）：物理存储顺序

该字段唯一，且不能为空

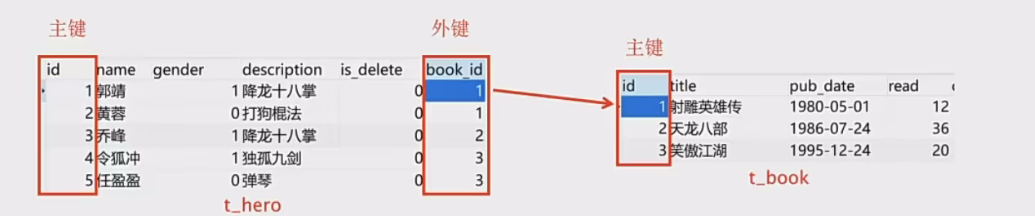
非空（not null）：此字段不允许填写空值

唯一（unique）：此字段的值不允许重复

唯一的字段可以是多个

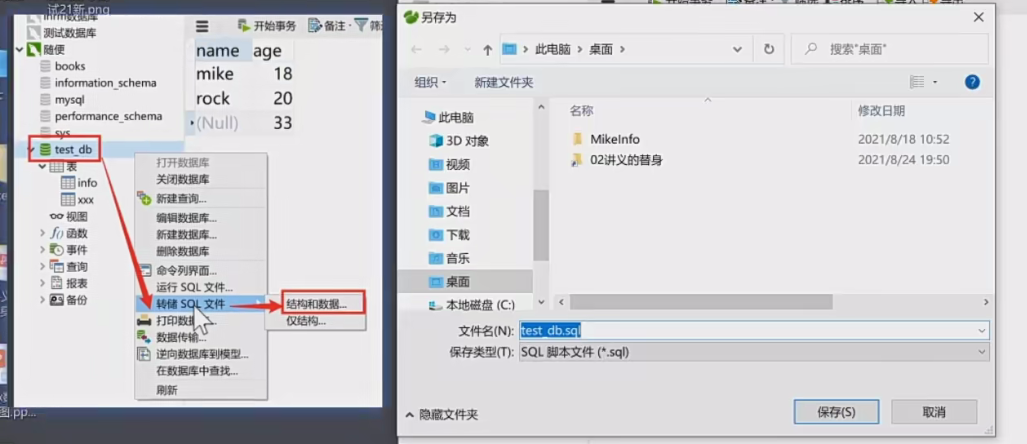
默认值（default）：当不填写此值时默认值，填写时按填写值

外键（foreign key）：维护两个表之间的关联关系

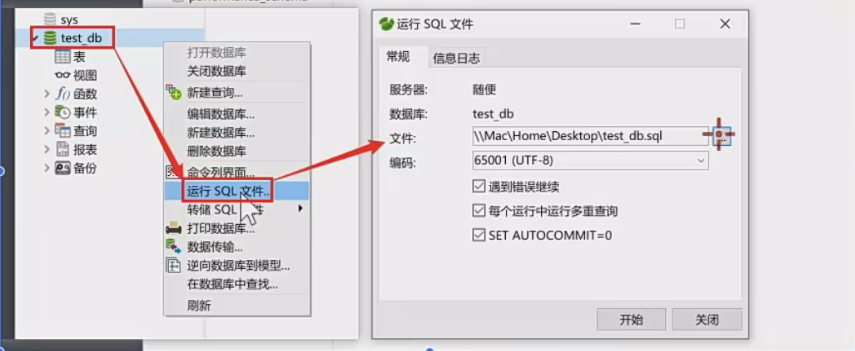


10、**备份与还原**

备份



还原要还原的数据库不存在时，要先创建数据库



11、命令行操作数据库

查看所有数据库：show databases；

使用数据库：use 数据库名；

查看当前使用的数据库：select database();

创建数据库：create database 数据库名 charset=utf8；

删除数据库：drop database 数据库名；

12、命令行操作表

查看当前所有数据库的表：show tables；

查看表结构：desc 表名；

查看表的创建语句：show create table 表名；

13、数据库表的操作

1.创建数据库表

create table 表名（字段名1 类型 约束，字段名2 类型 约束，... ... ...）

例：create table students（name varchar(10)）

2.删除数据表

drop table students；

drop table if exists students；

3、简单查询

select \* from 表名；

例：select \* from student；

4、添加一行数据

insert into 表名 values（...）

例：insert into students values（0，’亚瑟‘，22，177.56）

insert into 表名（字段1，字段2，...）value（值1，值2）

例：insert into students（name）values（’老夫子‘）

5、添加多行数据

方式一、多条insert语句，多条语句之间用英文分号分隔

insert into students（name）value（’老夫子2‘）；

insert into students（name）value（’老夫子2‘）；

方式二、通过一条insert语句插入多条数据，数据间用逗号隔开

insert into 表名 values（...）,（...）

例：insert into students values（0，’亚瑟‘，22，177.56）,（0，’亚瑟‘，22，177.56）

insert into 表名（字段名1，...） value （值1，....）

例：insert into 表名（字段名1，...） value （’老夫子5’）

6、**修改数据**

update 表名 set 字段名1=值1，字段名2=值2... where 条件

例：update students set name=‘狄仁杰’，age=20 where id=5

7、删除数据

delete from 表名 where 条件

例：delete from students where id=6；

此方法为物理删除，逻辑删除值设定一个字段来表示当前记录已经删除

truncate table 表名；

清除表里面所有数据，但表结构保留，自增长字段从1开始

drop table 表名

删除数据表，包括数据和表结构，删除数据效率最高

8、数据查询

1.查询部分字段数据

select 字段1，字段2，.... from 表名

2.起别名

selectet 别名.字段1，别名.字段2，...from 表名 as 别名

selectet 字段1 as 别名1 ，... from 表名 as 别名

3.去重

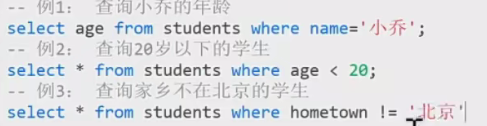
selectet distinct 字段1，... from 表名

4.条件查询

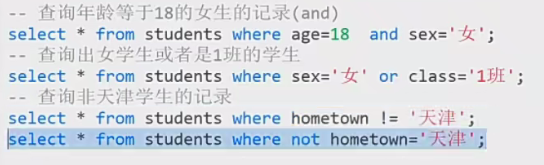
select 字段1，字段2 ... from 表名 where 条件；

例：select \* from students where id=1；

①比较运算符 = > < >= <= != <>



②逻辑运算符 or and not



③模糊查询 like

%：匹配任意多个字符

-：匹配一个任意字符

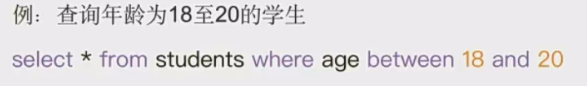


④范围查询in between...and...

in 表示一个非连续的范围

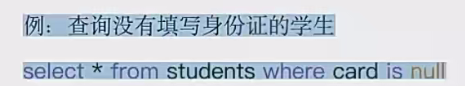


between..and..表示连续的范围

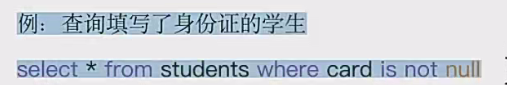


⑤空判断 null和空字符不一样

判断为空is null



判断非空is not null



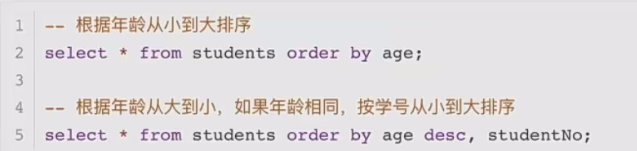
⑥排序

select \* from 表名 order by 字段1 asc|desc，字段2 asc|desc

默认为升序排序（从小到大）

asc：表示从小到大排序

desc：表示从大到小排序

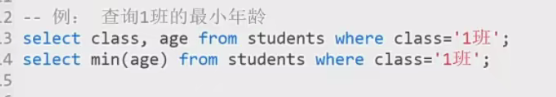


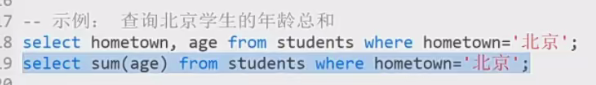
⑦聚合函数 聚合函数不能再where中使用

count()；max()；min()；sum()；avg();

image.png







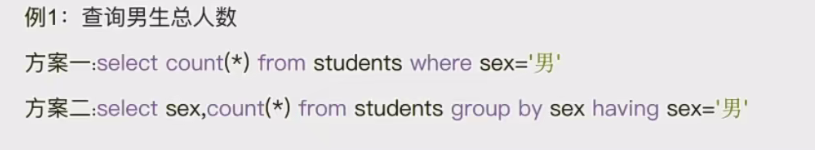
⑧分组查询

按照字段分组，此字段相同的会在同一个组中，分组为了对每一组的数据进行通知（使用聚合函数）

select 字段1，字段2，聚合函数... from 表名group by 字段1，字段2...



select 字段1，碲酸2，聚合... from 表名 group by 字段1，字段2，字段3 ... having 条件

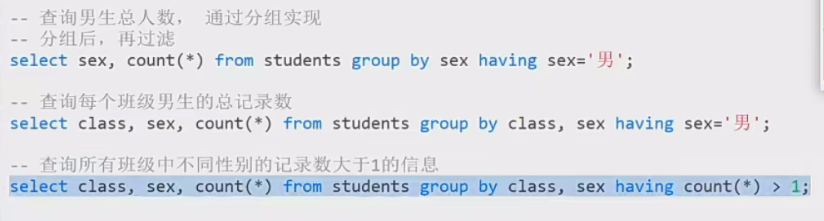


having与where对比

where是对from后面指定的表进行数据筛选，属于对原始数据的筛选

having是对group by的结果进行筛选

having后面条件中可以用聚合函数，where后面不可以

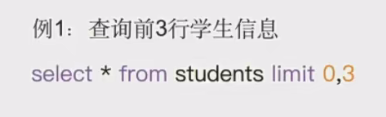


⑨分页查询

用来获取一部分的数据或者来分页

select \* from 表名 limit start，count

从start开始，获取count条数据；start索引从0开始；

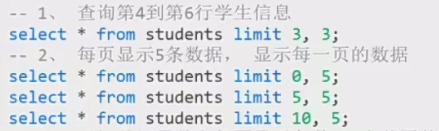


select \* from students limit （n-1）\*m，m；

n为页数n=1，2，3，4

m为每页显示的记录数m=3

（n-1）\*m,m



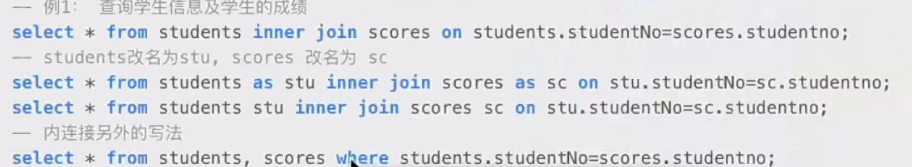
9、连接查询常用方式

1.内连接：结果为两个表匹配的数据（交集）

select \* from 表1

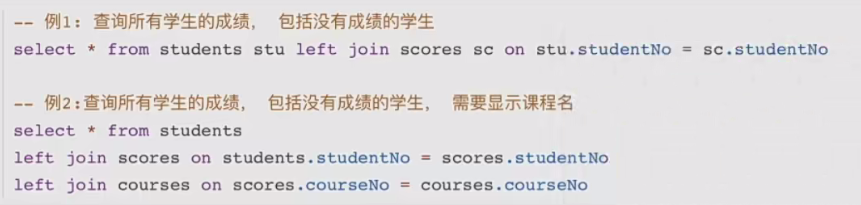
方法一：inner join 表2 on 表1.列=表2.列

方法二：select \* from 表1，表2 where 表1.列=表2.列



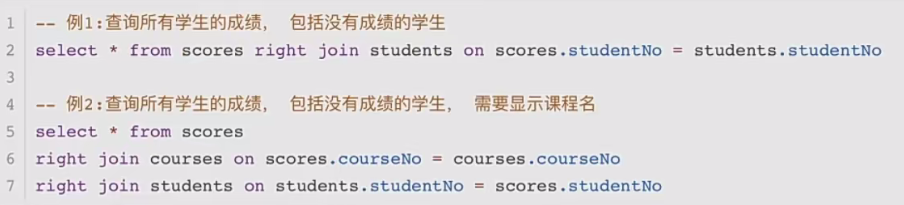
2.左连接：结果为两个表匹配到的数据加坐标特有的数据，对于右边中不存在的数据使用null填充

select \* from 表1 left join 表2 on 表1.列=表2.列



3.右连接：结果为两个表匹配到的数据加右表特有的数据，对于左表中不存在的数据使用null填充

select \* from 表1 right join 表2 on 表1.列=表2.列

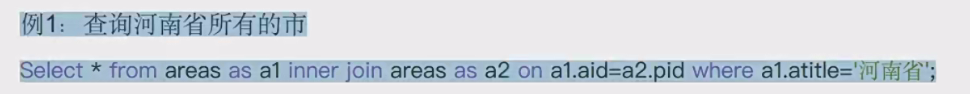


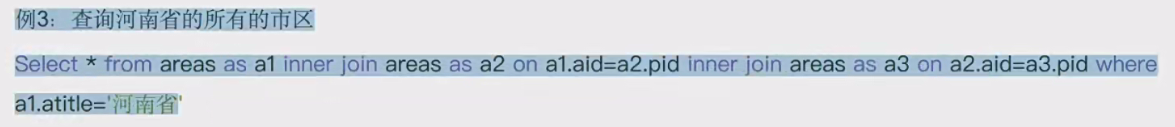
10、自关联

省市区的信息，一般不会分开放在不同的表里面进行存储，而是放在同一个表当中。

inner join 关联同一个表，不同的字段

自关联要用别名





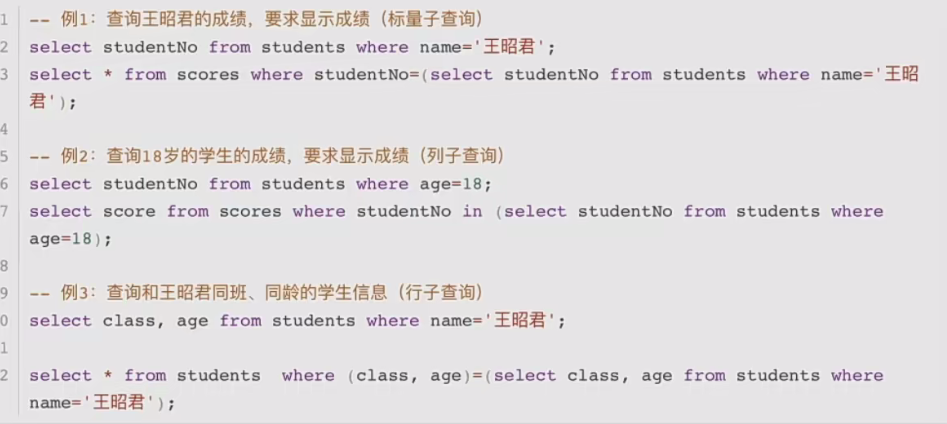
11、子查询

将SQL查询的语句嵌入在其他的SQL语句中，被嵌入的SQL语句称之为子查询，其他的SQL称之为主查询

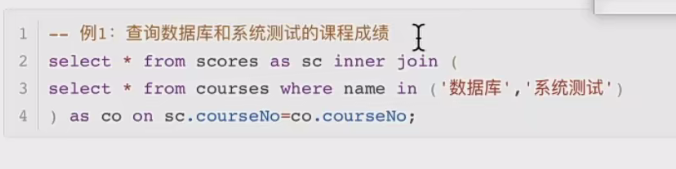
子查询select语句，要么充当条件，要么充当数据源

子查询语句是一条完整的select语句，且单独执行

子查询充当条件



子查询充当数据源



子查询特定关键字

in：主查询 where 条件 in （列子查询）

any/some：主查询 where 列 = all （列子查询）

all所有：主查询 where 列 = any （列子查询）

