昨日回顾

迭代思想

数组 索引数组和关联数组 key(下标) => 值

数组的遍历 foreach($数组名 as $key => $value) {}

数组的其他遍历 数组的指针for; key ,current ,next ,reset,prev,end;

while(条件) each函数返回数组指针指向当前元素的一个新数组.

list(变量,变量,…) = 索引数组

冒泡排序 每走访一次都能够出来一个最小值(最大值), 最多走访n-1次(n表示数列的个数)

每走访一次,都会进行两两比较,两两比较的次数 = 个数 – 走访的次数

根据条件进行两两交换

**MySql数据库**

# 第1讲 数据库简介

## 数据库的概念

存储、管理数据的仓库。英文：Database。实际项目中，一般将项目相关的数据全部存储在指定的数据库中。

存储数据的软件 execl,word.

最重要的除了 存储数据,还有一样就是管理数据

数据库系统 = 数据库+数据库管理系统(增,删,改,查)+数据库管理员

现在最主流的数据库是 关系型数据库

## 关系型数据库

当前使用范围最广的数据库

大型数据库: oracle(甲骨文), DB2(IBM)

中型数据库: MySQL(甲骨文), sql server (微软)

小型数据库: access

关系型数据库，就是基于关系模型的数据库，一个关系模型其实就是一张二维表，而一张二维表往往又对应着现实世界中的一个实体集

### 数据库概念设计的描述术语

* 实体

描述客观事物的概念，具体的事物（人，手机），抽象的事物（订单）

* 实体集

同一类的所有实体组成的整体

实体集反映到数据库中就是一张一张的二维表，数据表

* 属性

实体具有很多属性，对应数据表中的列（字段）

房间(房间号,床,电视, 空调)

服务人员(编号,入职时间,工资,性别)

账单(金额,时间)

物料(擦脚布, 香水)

### 数据库的逻辑设计术语

* 数据表（table）

表是以行和列的形式组织起来的数据的集合。存储某方面数据，如员工信息，商品信息，文章信息

一个数据库中可以有多个数据表

* 记录（row，record）

一行有意义的信息的集合

一个数据表中可以有N条记录

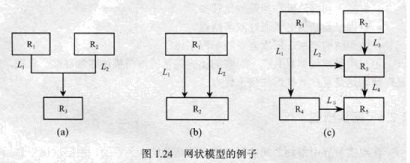
* 字段(column，field)

某一列的值的名称

## 非关系型数据库(了解)

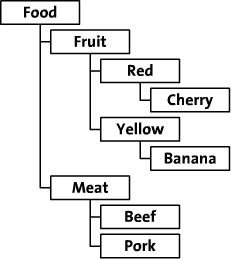
### 网状数据库

处理以记录类型为节点的网状数据模型的数据库。



### 层级数据库

层级数据库是将数据组织成有向有序的树状结构，并用一对多的关系连接不同层次的数据库。



## SQL语言

SQL：structured Query Language结构化查询语言

SQL语句。它相当于代码或者发送的命令;

### 概念

关系型数据库使用通用语言。

### 分类

* DQL-- Data Query Language，数据查询语句

SELECT（查询数据）

select 查询

* DDL-- Data Definition Language，数据定义语言

CREATE（创建）、DROP（删除）、ALTER(修改)

create 创建 表格

drop 删除

alter 修改

对表(实体集)的修改和增删操作

* DML-- Data Manipulation Language，数据操作语言

INSERT（插入数据）、UPDATE（更新数据）、DELETE（删除数据）

insert 插入

update 更新

delete 删除

对记录(数据)的增删改

* DCL-- Date Control Language，数据控制语言

GRANT（授权）、REVOKE（取消授权）

登录的用户的权限问题

# 第2讲 MySql的软件架构

mysql安装了2个东西,一个是 mysql的服务端,还有一个 客户端

c/s架构 c:client server 服务端

## 访问步骤

### 第一步:服务器开启

成功访问数据服务器的前提是：数据库服务器在运行状态。



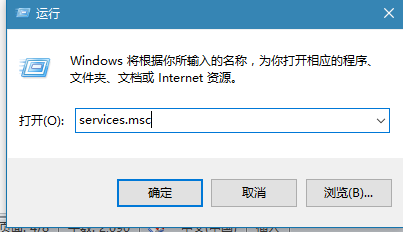
### 第二步:通过客户端发送命令



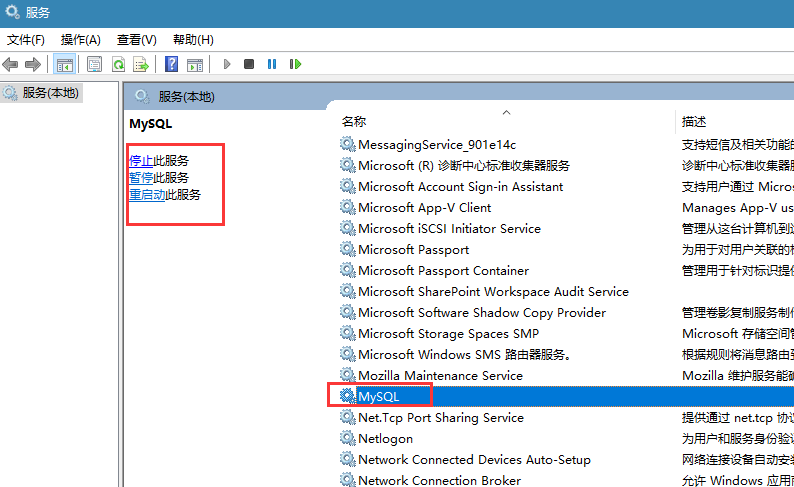
## 开启MySql服务器

### 以window服务方式

找到运行,输入services.msc

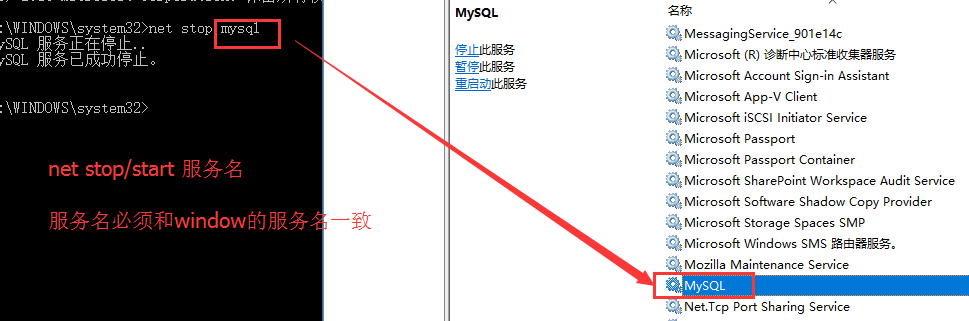


找到 msyql



### 以命令行方式

命令行一定要以管理员的权限进入

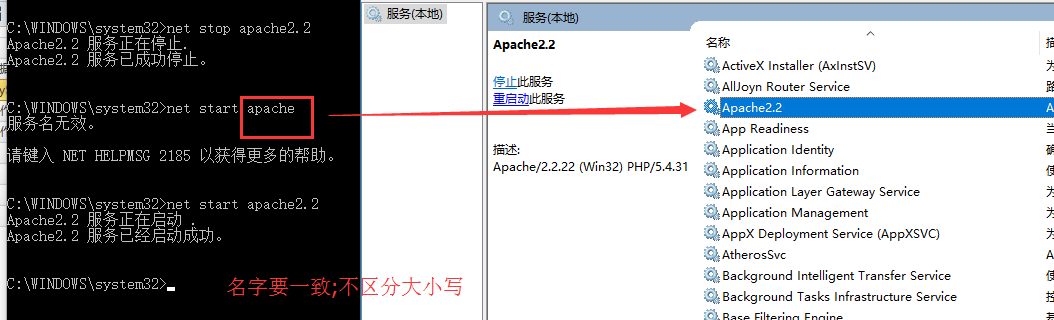


net start 服务名 – 开始服务

net stop 服务名 – 停止服务

net –h 输出帮助 输出更多的命令

除了net start/stop 能够开启 mysql对应的服务,还可以开启和关闭对应的 apache的服务

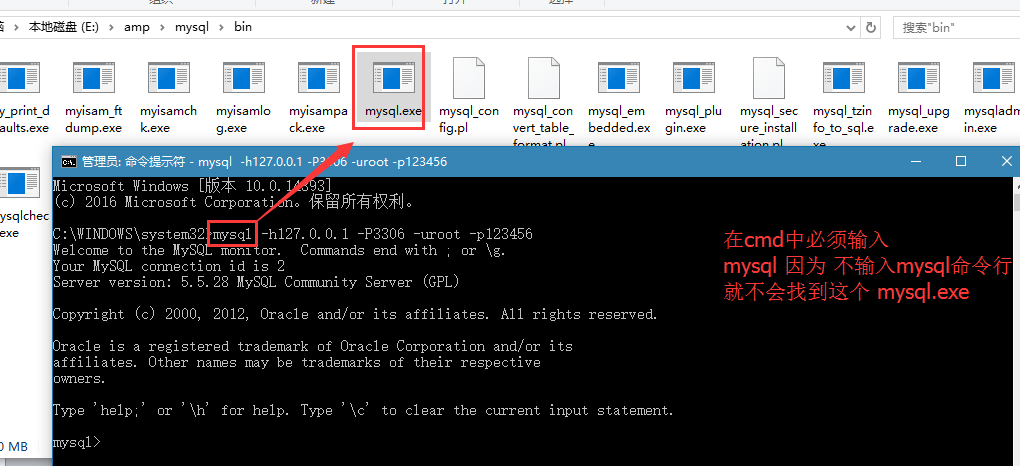


## 客户端连接服务器

### 命令行

安装mysql的时候 有一个 mysql client

还有另外一个 mysql作为了window的环境变量 cmd中操作



mysql –h127.0.0.1 –P3306 –uroot –p123456

h表示host 主机; 主机除了可以连接自己的还可以连接别人的

P表示 端口号(P一定要是大写的)

u 表示 user ,是用户名,现在我们都使用超级管理员 root

p 表示 password,密码(小写的p).

如果我们想登陆本机,并且mysql的端口号是默认的3306,我们可以 把-h 和 –P省略

### 

### 可视化界面

以后再用,我们学习的时候千万不要用.

phpmyadmin

navcate

# 第3讲 数据库操作

\s就能显示一些信息,包括服务器的编码和客户端的编码以及当前使用的数据库



quit/exit 退出客户端的登录

**sql语句中的中括号是可选的 []**

**.sql 是数据库文件的后缀名**

**数据库的注释有两种:**

**行注释: #**

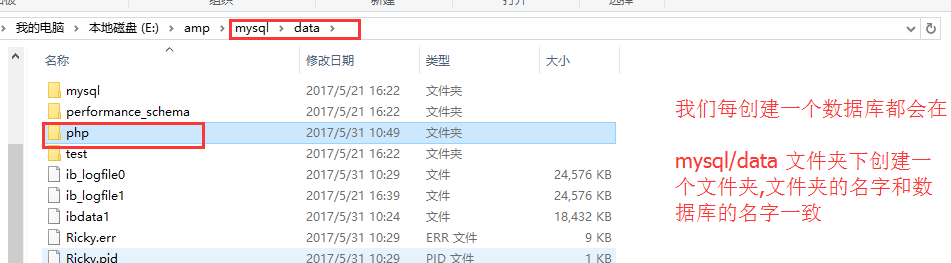
**--**

**#和—的区别是 –后边必须有一个空格才可以写注释内容**

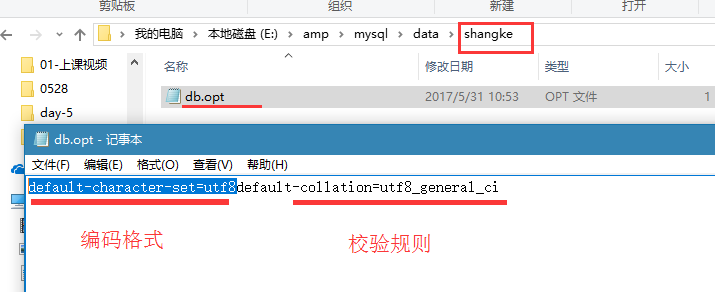
**块注释: /\*\*/**

## 创建数据库(增)

每创建一个数据库，在数据的存储路径mysql/data文件夹中创建数据库名字对应的文件夹



数据库文件夹中有一个文件



### 语法形式

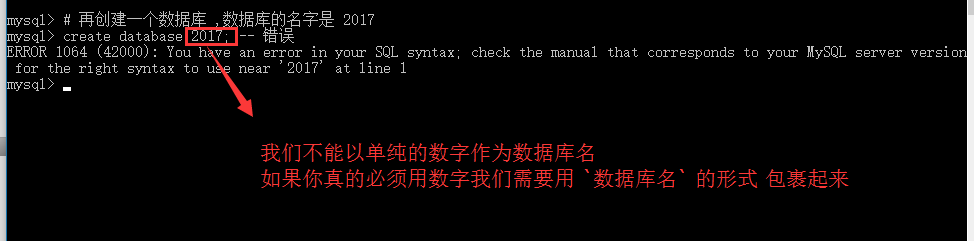
create database 数据库名[数据库选项];

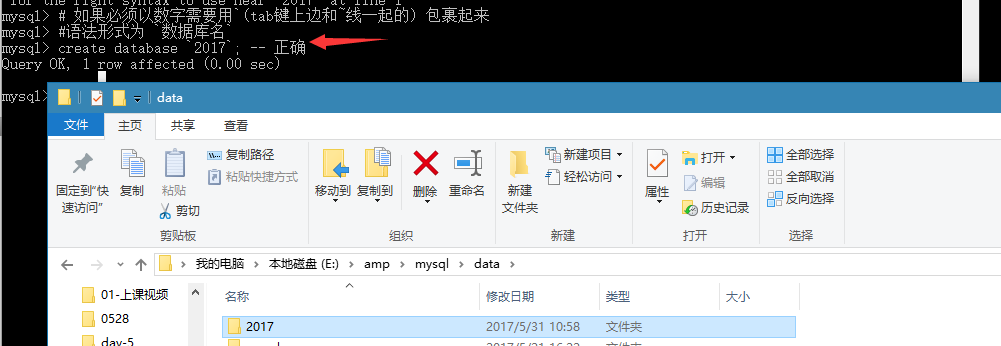
数据库选项就2个,一个是 编码格式,另外一个是 校验规则;

### 

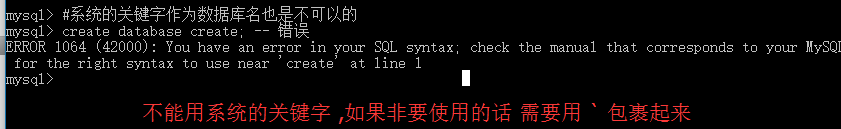
### 数据库命名规则

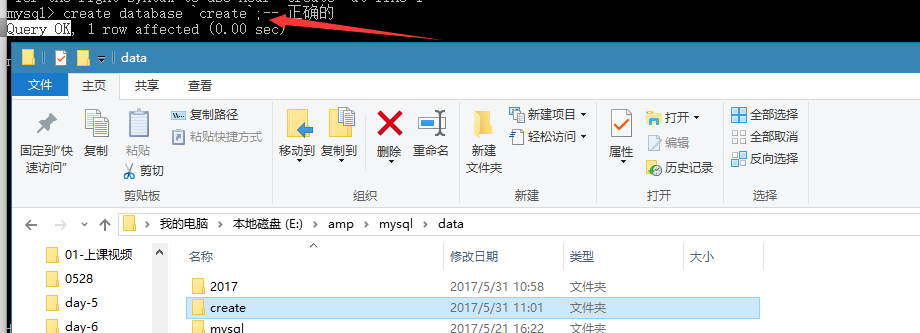
* 字母、数字、下划线。
* 不能使用纯数字的名称；



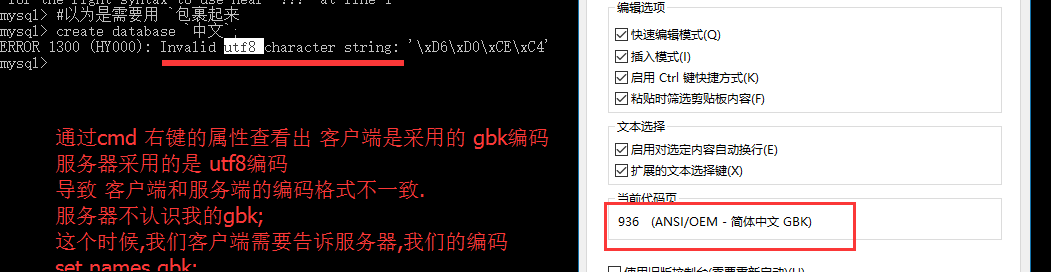


* 命名见名知意。
* 使用系统关键字、特殊字符作为表名时，使用“`”包括起来 ；

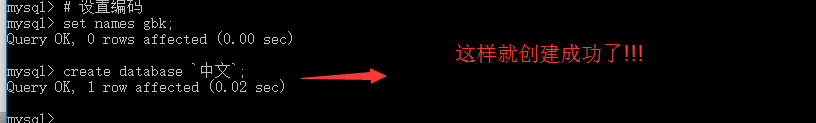




* 长度不超过64个字符。
* 如果非要使用中文,注意了



我们需要通过客户端的一条指令 set names gbk; 来告诉服务器我客户端用到编码是 gbk,这个服务器就知道怎么编码.



## 查看数据库(查)

### 查看当前有哪些数据库

show databases;

### 

### 查看数据库的创建语句

show create database 数据库名;

## 

## 删除数据库(删)

提示：不能删除系统数据库（information\_schema, mysql, performance\_schema）

drop database 数据库名;

数据库删除之后就没有了,所以删除数据库一定要慎重再慎重!!!

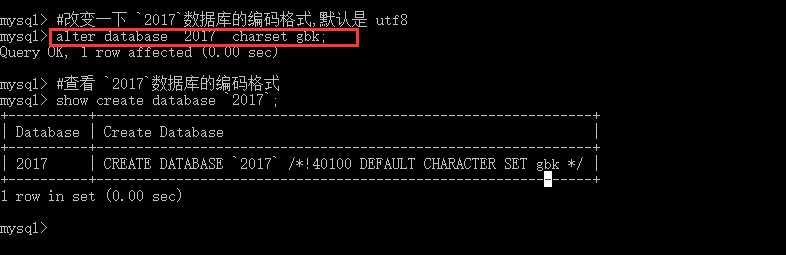
## 

## 修改数据库(改)

数据库创建的时候 就是有一个名 还有一个 数据库选项

数据库的名不能修改;

alter database 数据库名 charset gbk;



数据库增 : create database

删: drop database

改: alter database

查: show databases; show create database

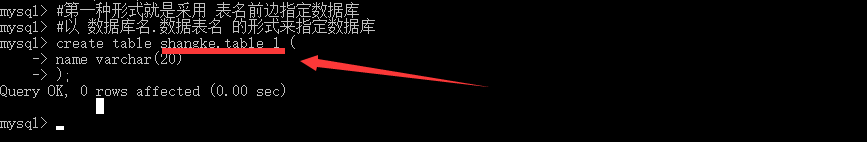
# 第4讲 数据表操作

## 数据表不能脱离数据库

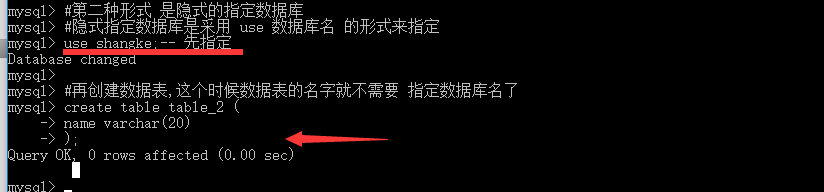


### 指定数据库的两种方式

* 显示的指定数据库,形式是 **数据库名.表名**



* 隐式的指定数据库



当我们隐式指定了数据库之后 use 数据库名 之后,我们默认的对表的操作都是 指定的这个数据库

例如: 我们再创建一个数据表之后,这个数据表只归这个数据库所有.

当我们指定了数据库之后,也可以在当前的情况下 对数据库进行增删改查的操作

## 创建数据表(增)

### 语法

create table 表名 (

字段名 数据类型,

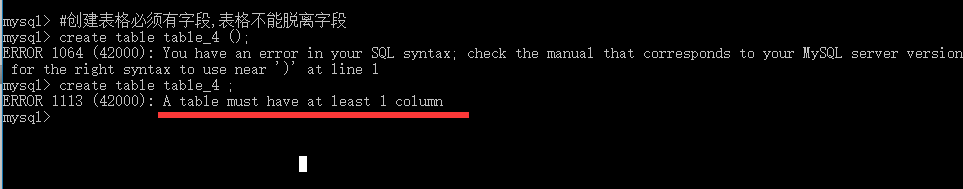
字段名 数据类型[, 字段名 数据类型,…]

)[表选项];

表选项也有2个,一个是字符编码,另外一个是数据库引擎

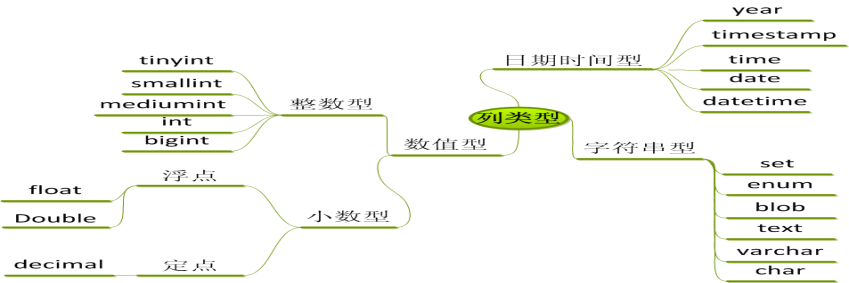
### 注意事项

①我们创建数据表的时候,必须至少有一个字段



②表名的命名规范和数据库命名规范类似(一样). 习惯上给数据表加一个前缀, t\_ 表示 table;

③ 字段名后边必须有类型,而且 字段名和数据类型之间用空格分割.



int 整形

float 浮点数

varchar/char , varchar/char(M)

④ 多个字段之间用逗号进行分割. 最后一个字段后边不需要逗号.

## 查看数据表(查)

### 查看当前有哪些数据表

show tables;

### 

### 查看数据表的创建语句

show create table 数据表名;

### 

### 查看表结构

desc 表名; (describe 表名;)

## 

有些时候我们忘记了这个表有哪些字段,或者是别人写的表我想查看都有哪些字段

## 删除数据表(删)

### 语法

drop table 表名;

表删除的时候一定要慎重,因为表里如果有数据也就跟着删除了.

### 

### if exists语法

如果我们删除一个不存在的表格会报错误,为了防止出现这样的错误我们在drop table 后边加上一句话 if exists

drop table if exists 表名

## 

## 修改数据表(改)

修改表 ,我们来看一看 表都有哪些东西可以修改.

create table 表名 (

字段名 类型 [,字段名 类型,…]

)[表选项];

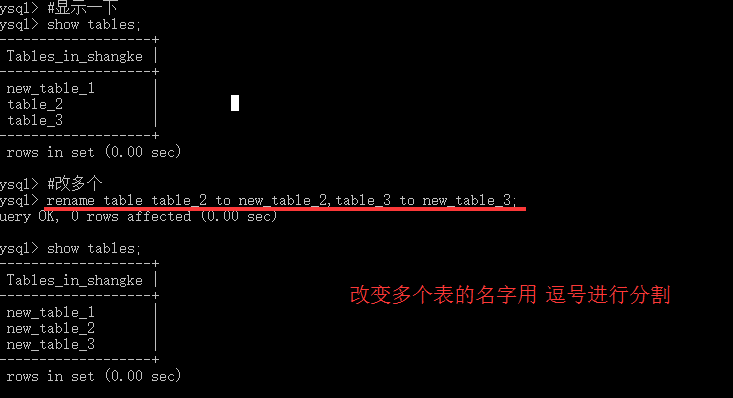
### 修改表名

rename table 旧名字 to 新名字[,旧名字 to 新名字,…];

改一个表的名字

### 

改多个表的名字



### 修改列定义

这个列可以增加,可以删除,也可以重命名,更可以修改类型

修改列,也是在修改表格,所以 修改列定义是 父指令 + 子指令

alter table 表名

add|drop|change|modify

#### 增加一列(add)

alter table 表名

add 字段名 字段类型;

#### 

#### 删除一列(drop)

alter table 表名

drop 字段名;

#### 

#### 重命名字段(change)

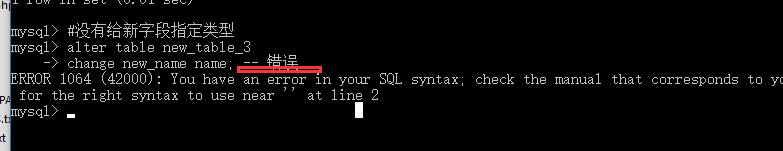
alter table 表名

change 旧字段名 新字段名 字段类型(新字段名的字段类型)

新字段名后边必须有字段类型,即便不修改旧字段的类型也需要写上字段类型

#### 

不指定类型是错误的



#### 修改字段类型(modify)

alter table 表名

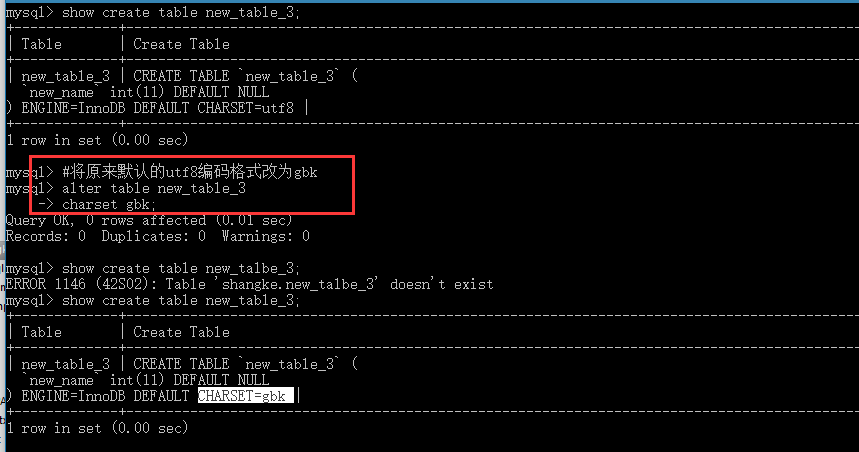
modify 字段名 新类型;

### 

### 修改表选项

alter table 表名

charset 编码格式



小规律:

**create database/table 增加数据库/表**

**drop database/table 删除数据库/表**

**alter database/table 修改数据库/表**

**show databases/tables 显示有哪些数据库/表**

**show create database/table 显示创建的语句**

# 第5讲 数据操作

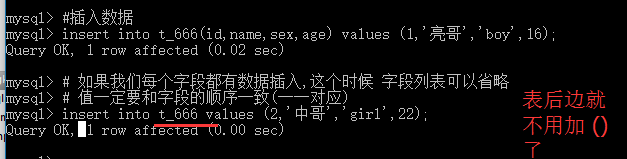
## 插入数据(增)

### 语法

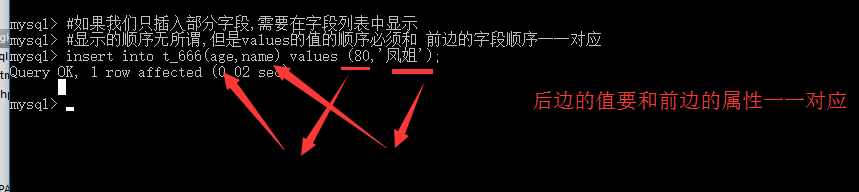
insert into 表名(字段列表) values (值列表);

### 注意事项

* 插入数据时，字段列表与数据列表要一一对应。



* 字段列表中，字段出现的顺序不需要与数据表中字段的顺序一致。



* 插入全部字段信息时，可以省略字段列表。此时，必须按照数据表中字段出现的顺序录入。

## 查询数据(查)

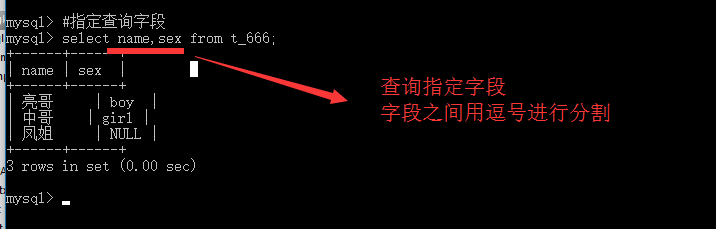
### 语法

select \*| 字段列表 from 表名 [where 条件判断];

\* 表示 该表的所有字段

字段列表中的字段以 逗号进行分割 name,age,sex

## 



## 删除数据(删)

### 语法

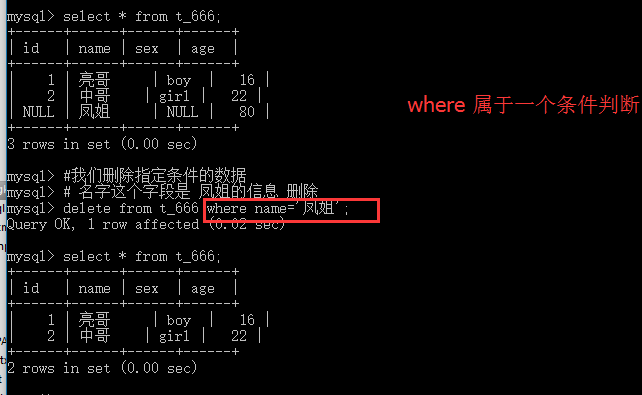
delete from 表名 [where 条件判断];

大家一定不要忘记加条件,如果不加条件,表里所有的数据都删除了.

## 

where 后边相当于一个条件

符合where添加的才去执行.

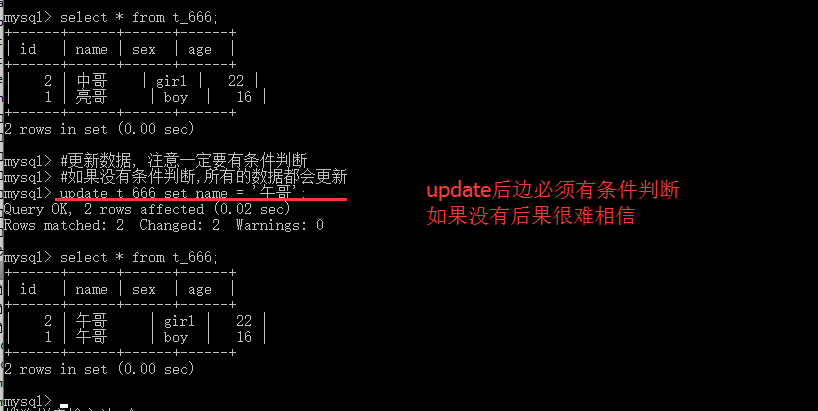


## 修改数据(改)

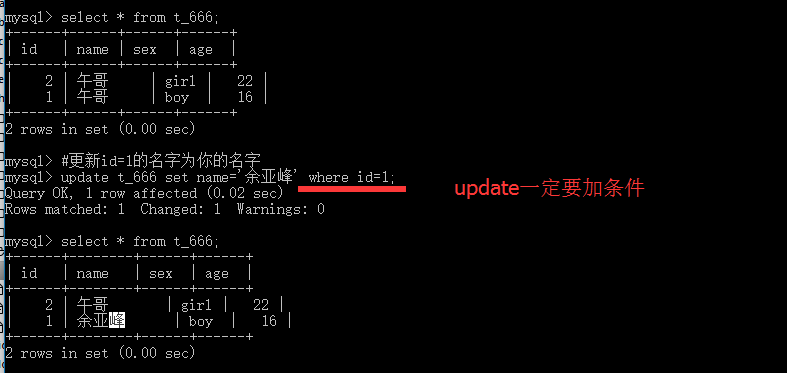
### 语法

update 表名 set 字段=值,字段=值[,字段=值] where 条件判断;

大家一定不要忘记加条件,如果不加条件,表里所有的数据都更新了.



更新数据的时候一定要加条件



# 第6讲 字符集

为什么会有这么多字符集???

世界上有很多语言,英语,汉语,德语,日语,汉语

## 字符集介绍

### 字符集概念

字符集：字符及其编码的集合。

http://www.qqxiuzi.cn/bianma/guobiaoma.php

常用的字符集：ASCII，GBK,GB2312，BIG5，UTF8

ascii: 是美国标准信息码, 用一个字节就能够表示 所有的英文字母和一些符号



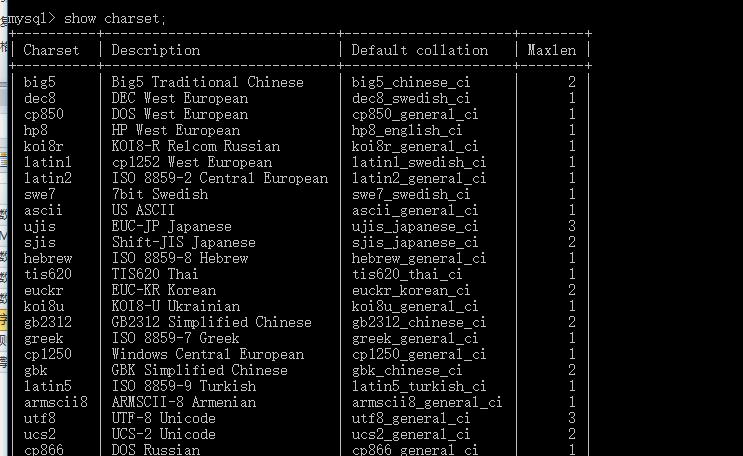
gbk:表示我们中文(简体中文)

big5:大五码(繁体中文)

utf8: 把所有的语言都进行了编码

### 查看服务器支持的全部字符集

Show charset;



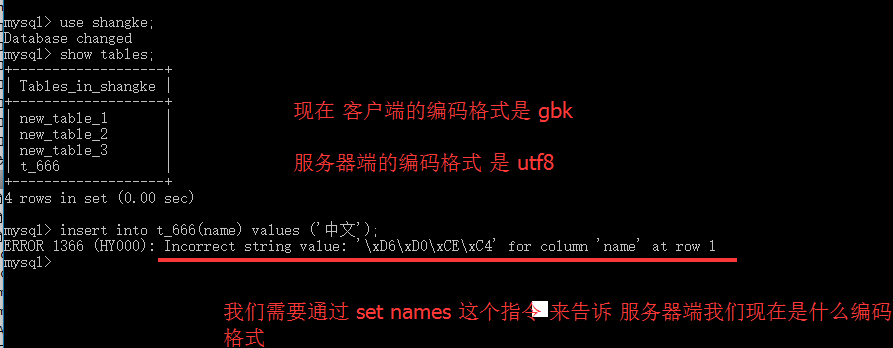
数据库支持的编码格式越多,认识的数据格式也越多.

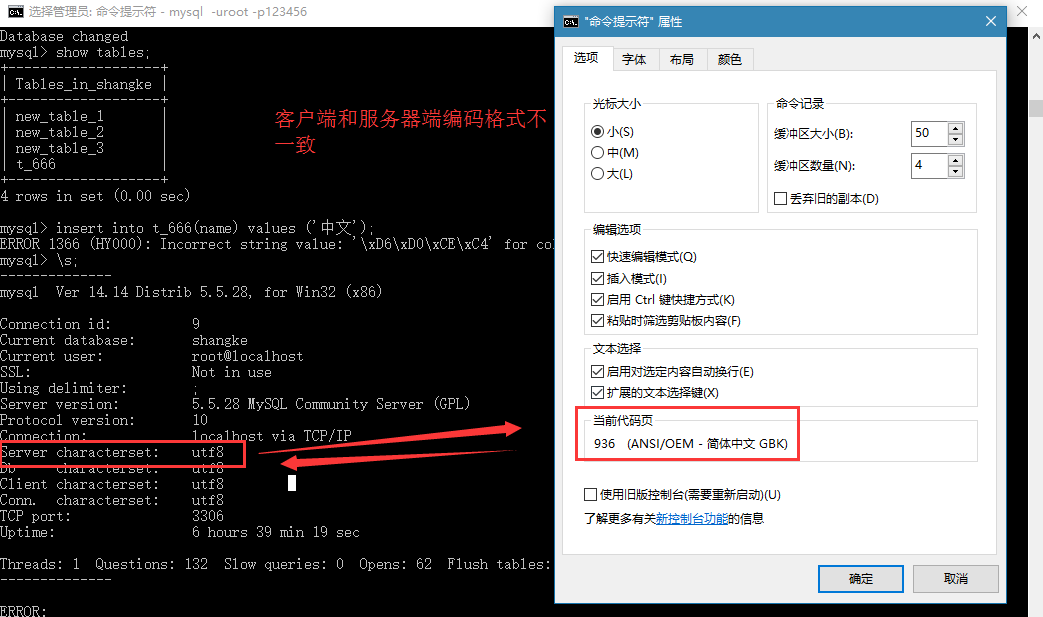
我们可以理解成为 mysql服务器可以说多少门语言.

## 乱码问题

### 乱码的本质原因：

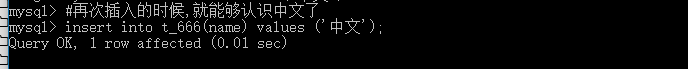
数据库服务器返回数据的字符集与客户端显示的字符集不一致。





### set names 字符集

set names gbk;



字符集设置的黄金原则---‘三码合一原则’：数据库、文件编码（PHP代码）及web页面显示尽量使用相同的字符集

数据库和数据表都会涉及到编码格式,默认都是utf8,我们用utf8就可以了.

修改编码格式在数据库的修改和数据表的修改中都有涉及

# 第7讲 校对规则

## 概念

a , B,c 我们对这三个字母进行排序.

可以按照大小写排序: B,a,c (a,c,B)

不按照大小写: a,B,c

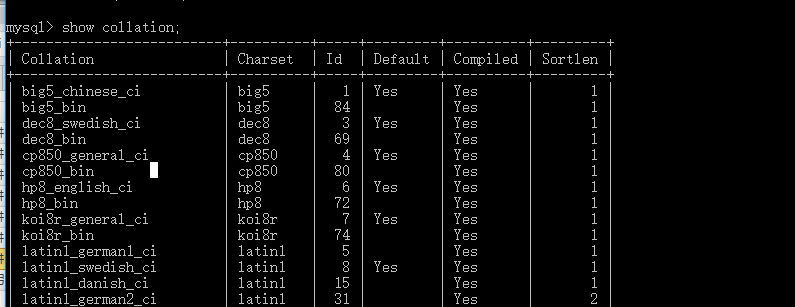
按照按照二进制 ,B 的值是 66 a是97,c是99

是指在同一字符集内字符之间的比较排序规则。校对规则是基于字符集。英文:collate，collation

## 查看服务器支持的校对规则

### 命令

show collation;



### 常见的校对规则

* ci结尾的:不区分大小写
* cs结尾的:区分大小写
* bin结尾的:二进制编码比较

一般 情况下 英文 就是按照 我们的ascii码就行一个排序

# 第8讲 存储引擎

## 概念

表在创建的时候默认是 innodb存储引擎

.

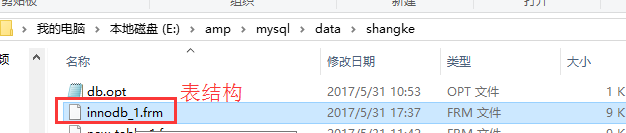
也叫做表类型，指的是数据表的存储机制，索引方案等配套相关功能，不同的引擎，由于处理方式不同，会带来不同的功能或者相应优化

创建表格的时候我们可以指定数据库的存储引擎

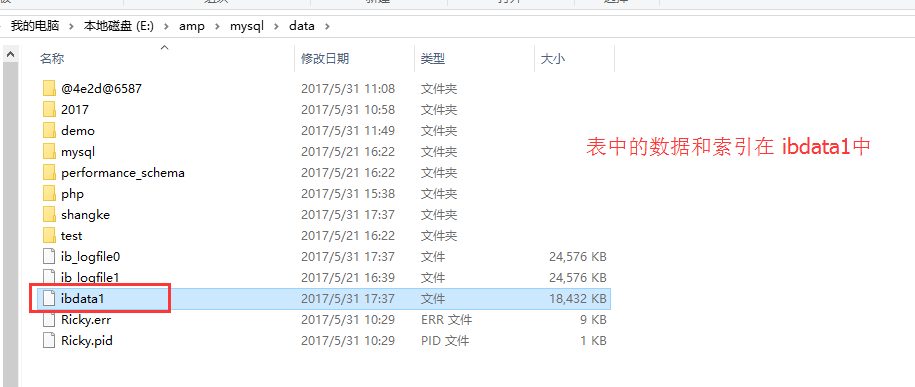
create table 表名 (字段 类型) engine = innodb|myisam;

innodb存储引擎和 myisam存储引擎除了性能和功能有区别,在创建的时候,创建的文件有也区别

innodb: 会在所对应的数据库文件夹下创建一个 表结构文件夹



innodb的数据 在



myisam: myisam会在对应的数据库文件夹中创建3个文件



## 存储引擎的选取

**选择存储引擎的依据：**

* 功能 比如外键 ，事物安全等、
* 性能

总体来说，就是在满足功能的基础上追求性能！

而从目前来看，InnoDB的性能可能越来越强大，将来有可能全面代替Myisam！

一般，用默认的InnoDB就行了

