2019-3-27

MySql基础操作

讲师:BRUCELIU

# 今日任务

* SQL语句概念
* 创建数据库
* 创建表
* 熟悉INSERT DELETE UPDATE语句

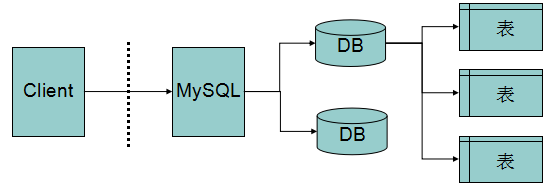
熟悉各种**SQL**语句！ SQL在不同的数据库中相似度很高，但是还是有细微差别！

# 1.MySQL数据库服务器、数据库和表的关系

所谓安装数据库服务器，只是在机器上装了一个数据库管理程序，这个管理程序可以管理多个数据库，一般开发人员会针对每一个应用创建一个数据库。

为保存应用中实体的数据，一般会在数据库创建多个表，以保存程序中实体的数据。

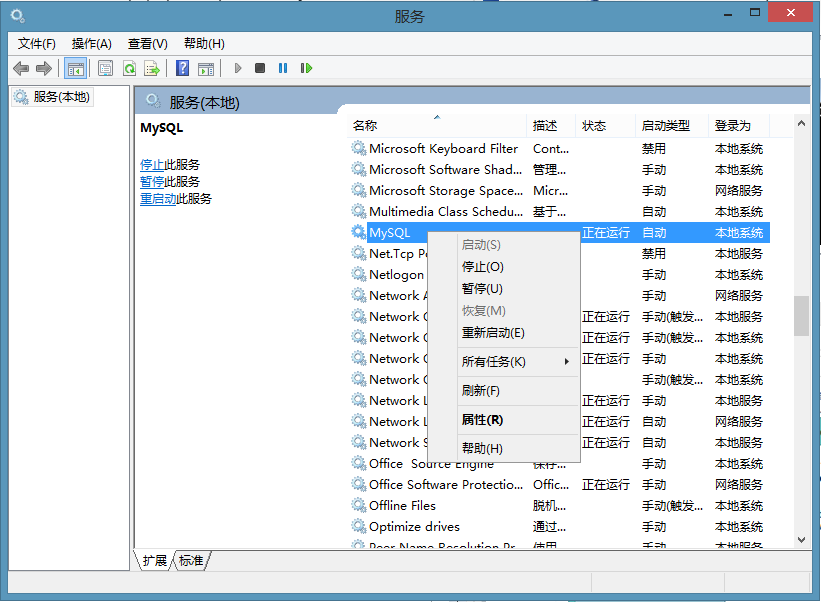
数据库服务器、数据库和表的关系如图所示：



# 连接Mysql

## 2.1 启动Mysql服务

安装后，MySQL会以windows服务的方式为我们提供数据存储功能。开启和关闭服务的操作：右键点击我的电脑→管理→服务→可以找到MySQL服务开启或停止。

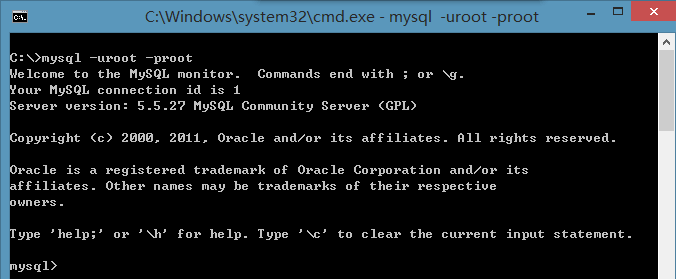


## 2.2命令行连接并登录Mysql服务器

MySQL是一个需要账户名密码登录的数据库，登陆后使用，它提供了一个默认的root账号，使用安装时设置的密码即可登录。

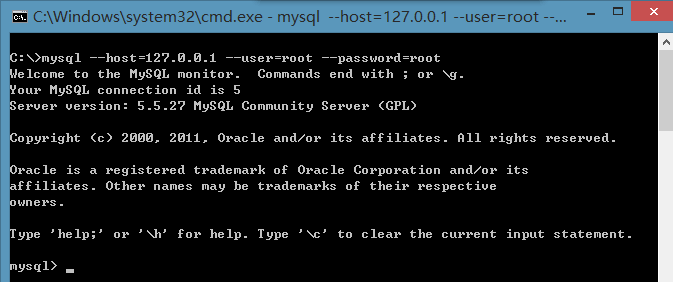
格式1：cmd> mysql –u用户名 –p密码

例如：mysql -uroot –proot

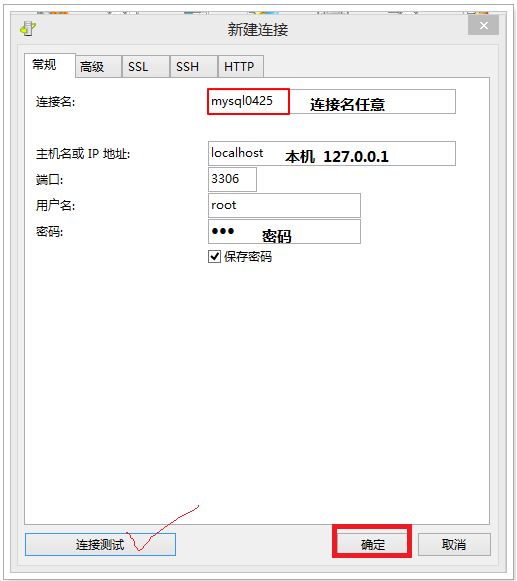


格式2：cmd> mysql --host=ip地址 --user=用户名 --password=密码

例如：mysql --host=127.0.0.1 --user=root --password=root



## 2.3使用可视化的客户端（例如navicat）



# SQL语句简介

Mysql数据库的操作需要SQL支持！

* 数据库是不认识JAVA语言的，但是我们同样要与数据库交互，这时需要使用到数据库认识的语言**SQL语句，它是数据库的代码**。
* 结构化查询语言(**S**tructured **Q**uery **L**anguage)简称SQL，是一种数据库查询和[程序设计语言](http://baike.baidu.com/view/128511.htm" \t "_blank)，用于存取数据以及查询、更新和管理[关系数据库系统](http://baike.baidu.com/view/549699.htm" \t "_blank)。
* 创建数据库、创建数据表、向数据表中添加一条条数据信息均需要使用SQL语句。
* 数库都是使用SQL语句进行操作的，不同的数据库，SQL语句有细微的差别！！
* **SQL语句不区分大小写！！！一般推荐用大写！**
* **SQL语句英文分号结尾！**

# SQL语句分类

* 数据定义语言：简称DDL(Data Definition Language)，用来定义数据库对象：数据库，表，列等。关键字：**create**，**alter**，**drop**等
* 数据操作语言：简称DML(Data Manipulation Language)，用来对数据库中表的记录进行更新。关键字：insert，delete，update等
* 数据控制语言：简称DCL(Data Control Language)，用来定义数据库的访问权限和安全级别，及创建用户。
* 数据查询语言：简称DQL(Data Query Language)，用来查询数据库中表的记录。关键字：select，from，where等



# DDL应用

## 数据库操作

### 5.1.1创建数据库

格式:

\* **create** database 数据库名;

\* **create** database 数据库名 character set 字符集 COLLATE 排序规则;

例如：

#创建数据库 数据库中数据的编码采用的是安装数据库时指定的默认编码 utf8,内部字符排序规则默认 utf8\_general\_ci

CREATE DATABASE day0101;

#创建数据库 并指定数据库中数据的编码

CREATE DATABASE day0101 CHARACTER SET utf8;

练习:

创建一个名称为mydb1的数据库。

Create database mydb1;

创建一个使用utf8字符集的mydb2数据库。

Create database mydb2 character set utf8;

创建一个使用utf8字符集，并带排序规则的mydb3数据库。

Create database mydb3 character set utf8 collate utf8\_general\_ci;

* 查看数据库

查看数据库MySQL服务器中的所有的数据库:

show databases;

查看某个数据库的定义的信息:

show create database 数据库名;

例如：

show create database day0101;



* 删除数据库

drop database 数据库名称;

例如：

drop database day0101;

* 其他的数据库操作命令

切换数据库：

use 数据库名;

例如：

use day0101;

* 查看正在使用的数据库:

select database();

### 5.1.2 修改数据库

不能修改数据库名，只能**字符集**和字符集排序较方式

ALTER DATABASE db\_name

[DEFAULT] CHARACTER SET charset\_name

| [DEFAULT] COLLATE collation\_name

* 切换数据库 use db\_name;
* 查看当前使用的数据库 select database();

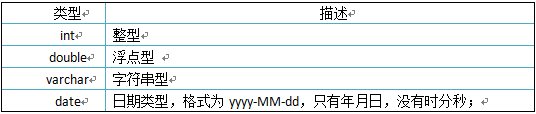
练习：

查看服务器中的数据库，并把其中某一个库的字符集修改为utf8;

alter database mydb1 character set utf8

## Mysql**常见**数据类型

MySQL中的我们常使用的数据类型如下



* **字符串型** VARCHAR(可变长度)、CHAR(定长)

大数据类型 BLOB（字节型大数据）、TEXT（字符型大数据）

数值型 TINYINT(**Byte**) 、SMALLINT(**Short**)、**INT**、BIGINT(**Long**)、FLOAT、**DOUBLE**

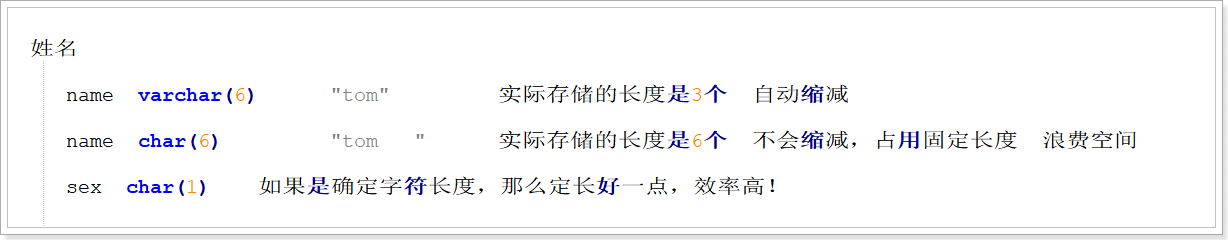
日期型

DATE（年月日）、TIME（时分秒）、DATETIME（年月日时分秒）

* **varchar和char的区别**

varchar 变长字符串 省空间

Char 定长字符串 效率高



## 创建数据表(不带约束)

### 5.3.1 SQL语句创建



注意：创建表前，要先使用use db语句使用库。

|  |
| --- |
| -- 学生表  CREATE TABLE STUDENT (  ID INT (8),  NAME VARCHAR (4),  SEX CHAR (1),  AGE INT (3),  BIRTHDAY DATE,  WEIGHT DOUBLE,  EMAIL VARCHAR (20)  ); |

### 5.3.1 查看表

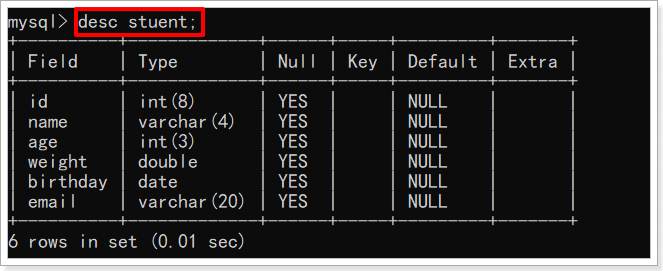
* 查看数据库中的所有表： 查询某一数据库下面所有的表

格式：**show tables;**

* 查看表结构：

格式：desc 表名;

例如：desc student;



### 5.3.2 可视化界面创建

## 表的约束（对这一列数据的限定条件）

数据库中的约束是加在列上的，是对整个列数据进行约束。

数据库的约束包括5种：

* 主键约束(表每一行数据的唯一标识，学号、订单号、工号、快递号...)： **primary key**:**不允许为空，不允许重复**。

一个表都需要做一个主键，如果找不到什么是主键的话，就生写一个**编号**主键！

一般主键都是自动增长的，主键一般都是数字类型！

如果主键想自动增长，那么设置

自动增长 **auto\_increment** （必须是整数类型字段）

如果主键是字符串，那么就需要手动填！

* 非空约束：**not null** 某一字段不能为空, 如果不设置，默认是可以为null！

例如：salary double **not null**

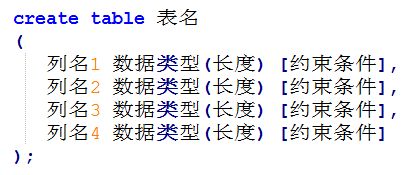
* 唯一性约束： **unique**  某一个字段的值是否是唯一的，例如身份证号码、手机号、邮箱....

例如：name varchar(20) **unique**

* 外键约束（后面多表时讲）
* 用户自定义约束（检查约束）--**MySQL不支持**

## 创建数据表(带约束)

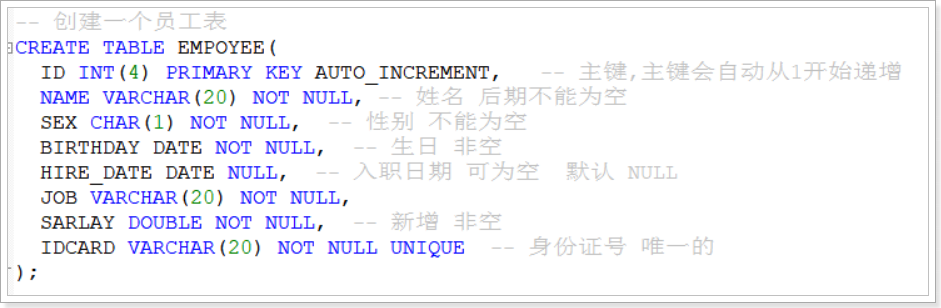
标准创建语法：



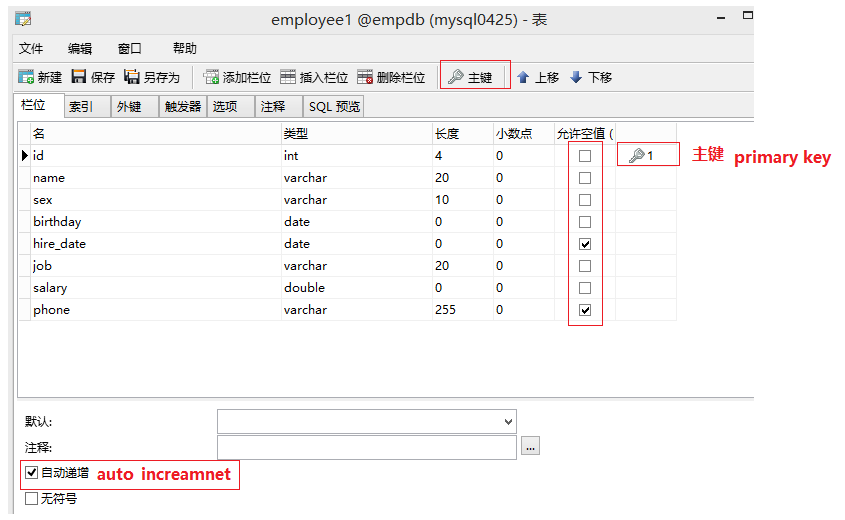
> 例题：在员工数据库**EmpDB**中创建一个员工表**Employee**



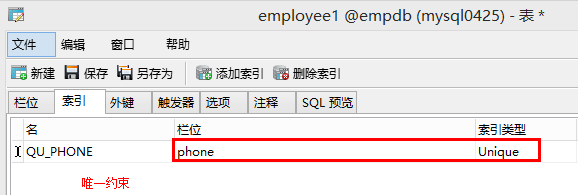
create database empdb; -- 创建数据库



* 使用可视化的界面创建表：



唯一约束：



## 删除表

格式：drop table 表名;

例如：drop table student;

## 修改表结构(了解)

* alter table 表名 add 列名 类型(长度) 约束;

作用：修改表添加列.

例如：

#1，为student 表添加一个新的字段为 分类描述 varchar(20)

**ALTER TABLE** student ADD sdesc VARCHAR(20);

* alter table 表名 modify 列名 类型(长度) 约束;

作用：修改表修改列的类型长度及约束.

例如：

#2, 为student 表的名称字段进行修改，类型varchar(50) 添加约束 not null

**ALTER TABLE** student **MODIFY** name VARCHAR(50) NOT NULL;

* alter table 表名 change 旧列名 新列名 类型(长度) 约束;

作用：修改表修改列名.

例如：

#3, 为student表名称字段进行更换 更换为 sname varchar(30)

ALTER TABLE student **CHANGE** name sname VARCHAR(30);

* alter table 表名 drop 列名;

作用：修改表删除列.

例如：

#4, 删除表中name这列

ALTER TABLE Student **DROP** name;

* rename table 表名 to 新表名;

作用：修改表名

例如：

#5, 为表student 改名成 student

RENAME TABLE student TO stu;

# DML应用

## 插入表数据 **INSERT**

* 语法：

**insert** into 表 (列名1,列名2,列名3..) values (值1,值2,值3..); -- 向表中插入某些列

**insert** into 表 values (值1,值2,值3..); --向表中插入**所有列，可以省列名**

注意：

插入的数据应与字段的数据类型相同

数据的大小应该在列的长度范围内

在values中列出的数据位置必须与被加入列的排列位置相对应。

除了数值类型外，其它的字段类型的值必须使用引号引起。

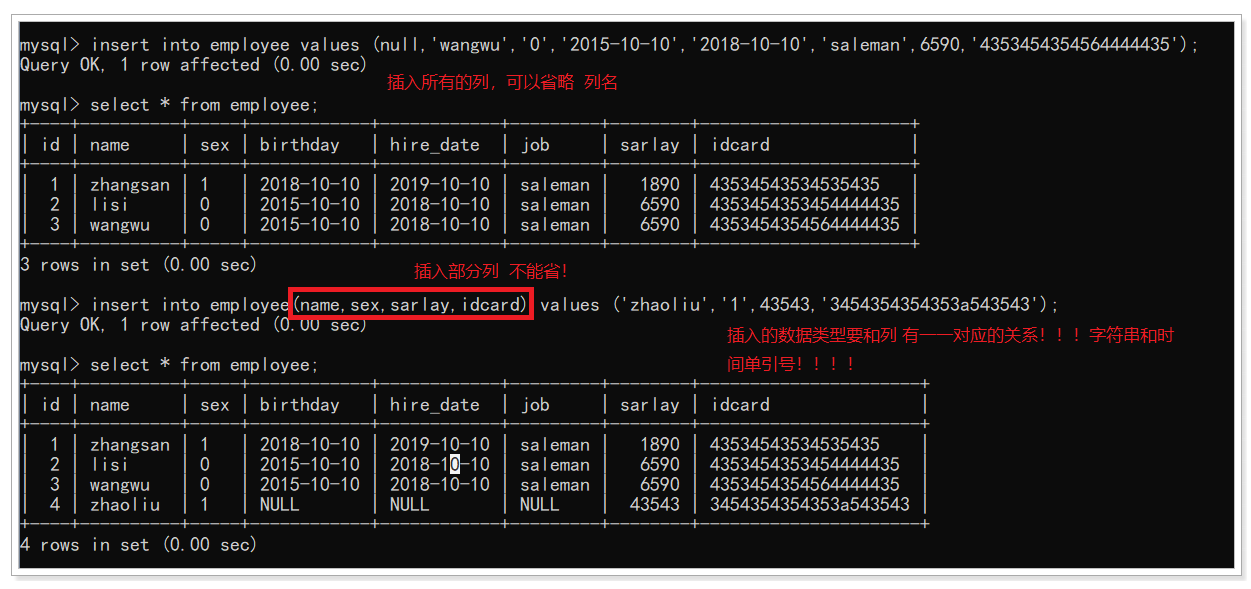
如果要插入空值，可以不写字段，或者插入 null。

对于自动增长的列在操作时，直接插入null值即可，后期会自动编号。

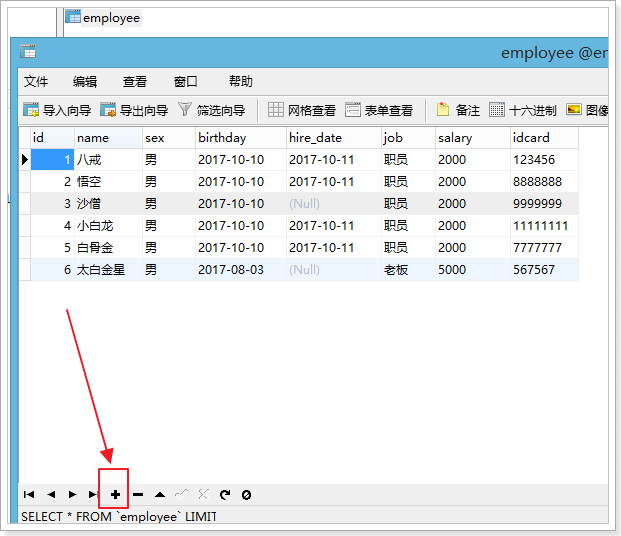
> VARCHAR和DATE的值需要用英文的**单引号引**起来！

> **INT和DOUBEL**不需要单引号！

示例：



* 可视化界面新增数据库表



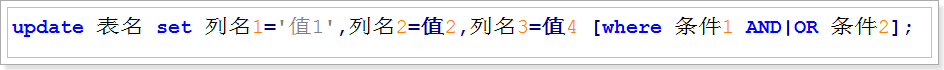
## 更新表记录 **UPDATE**

用来修改指定条件的数据，将满足条件的记录指定列修改为指定值

* 语法：

**update 表名** set 列名1=值1,列名2=值2; --如果没有WHERE 表示更新整个表数据

update 表名 set 字段名=值,字段名=值 where 条件; --如果有WHERE表示更新指定的数据



* 注意：
  + - 列名的类型与修改的值要一致.
    - 修改值得时候不能超过最大长度.
    - 值如果是字符串或者日期需要加’’.

如果没有Where条件，表示更新整张表数据！！

|  |
| --- |
| -- 1.将表中所有性别为1的那些人性别改为 '男';  update employee set sex='男' where sex='1';  update employee set sex='女' where sex='0';  -- 2.将表中所有的男人薪资涨10000元  update employee set sarlay=sarlay+10000 where sex='男';  -- 3.将表中 '2018-10-10'以前出生的人，职业改为 '老油条',薪资降1000元  update employee set job='老油条',sarlay=sarlay-1000 where birthday<'2018-10-10';  -- 4.把ID大于等于4的女人，职业改为 '待业'，薪资归0  update employee set job='待业',sarlay=0 where id>=4 and sex='女'; |

**> 示例：**



## 删除记录 DELETE

* 语法：

delete from 表名 **[where 条件]**;

或者

truncate table 表名; 不能有 **where 条件，作用就是清空表，自增归0！！！**

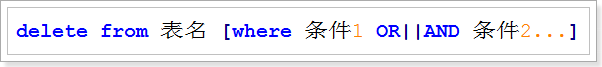
* 面试题：

删除表中所有记录使用delete from 表名; 还是用truncate table 表名;

删除方式：delete 一条一条删除，不清空auto\_increment记录数。

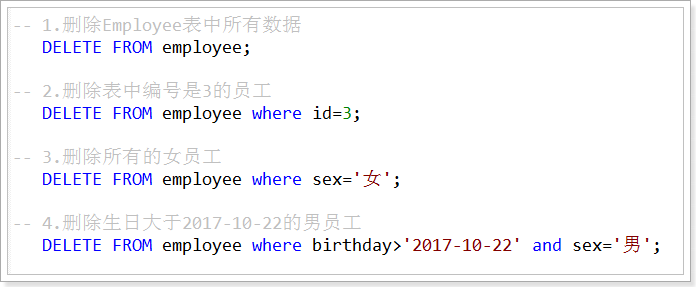
truncate 直接将表数据全部删除，重新建表，auto\_increment将置为零，从新开始。

* 使用SQL语句删除数据库表数据



> 如果删除的时候，**不带where条件，那么删除的是所有的数据**！！！

示例：



|  |
| --- |
| -- 1.删除所有数据  delete from employee;  -- 2.删除所有薪资为0的女员工  delete from employee where sex='女' AND sarlay=0;  -- 3.删除ID小于3的 或者 薪资在80000以下的  DELETE from employee where sarlay<=80000 OR id<3;  -- 新增员工  INSERT INTO employee VALUES (null,'小王','男','2000-1-1','2015-1-1','网红',35454,'8797897897788786788');  SELECT \* from employee; |

# DOS操作数据乱码解决

我们在dos命令行操作中文时，会报错

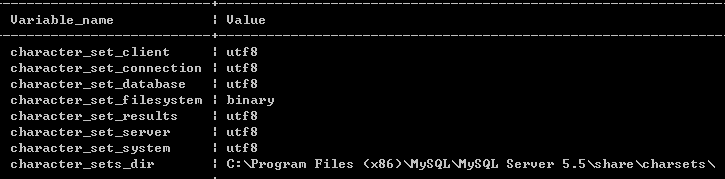
insert into user(username,password) values(‘张三’,’123’);

ERROR 1366 (HY000): Incorrect string value: '\xD5\xC5\xC8\xFD' for column 'username' at row 1

原因:因为mysql的客户端编码的问题我们的是utf8,而系统的cmd窗口编码是gbk

解决方案（临时解决方案）:修改mysql客户端编码。

show variables like 'character%'; 查看所有mysql的编码



在图中与客户端有关的编码设置:

* 将客户端编码修改为gbk.

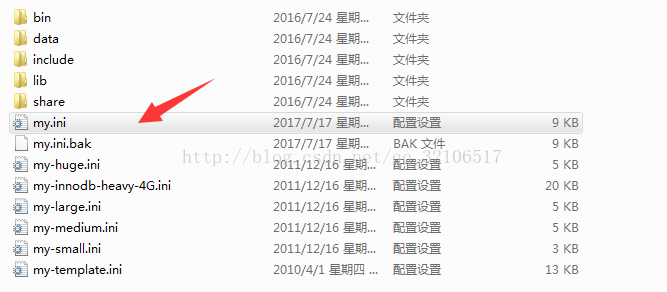
set character\_set\_client=gbk;

set character\_set\_results=gbk;

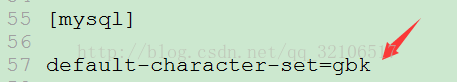
以上操作，只针对当前窗口有效果，如果关闭了服务器便失效。如果想要永久修改，通过以下方式:

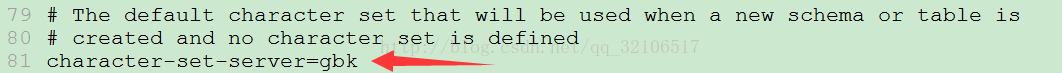
* 在mysql安装目录下有my.ini文件

在MySQL的安装目录下的配置文件my.ini里面的参数：default-character-set和character-set-server



使用高级记事本打开my.ini ,我们可以修改default-character-set和character-set-server 将其改为自己想要的字符集类型，如：gbk,gb2312等等，这里我是修改为gbk;





**关于命令行界面对齐问题:**

执行以下下面得命令：

登录上MySql之后执行：

set names gbk;

charset gbk;