**《Java语言程序设计基础教程》**

**上机实验指导手册**

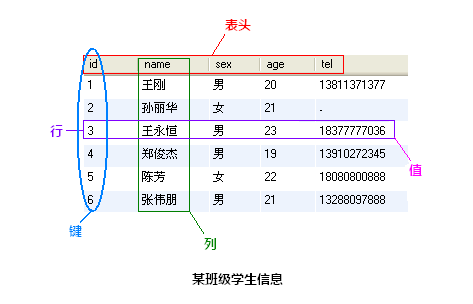
**实验十一 MySQL数据库**

## 【目的】

1. 安装mysql数据库
2. 对mysql数据库进行常规的操作

## MySQL的相关概念介绍

MySQL 为关系型数据库(Relational Database Management System), 这种所谓的"关系型"可以理解为"表格"的概念, 一个关系型数据库由一个或数个表格组成, 如图所示的一个表格:



* **表头(header)**: 每一列的名称;
* **列(row)**: 具有相同数据类型的数据的集合;
* **行(col)**: 每一行用来描述某个人/物的具体信息;
* **值(value)**: 行的具体信息, 每个值必须与该列的数据类型相同;

键(key): 表中用来识别某个特定的人\物的方法, 键的值在当前列中具有唯一性。

## Windows下MySQL的配置

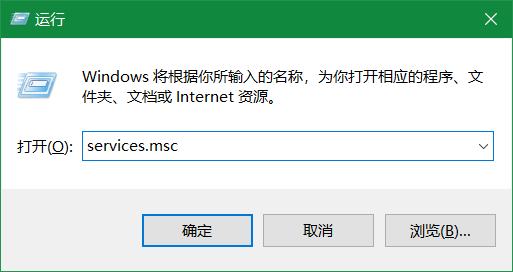
以MySQL 5.5为例，下载windows msi安装包，官网地址：

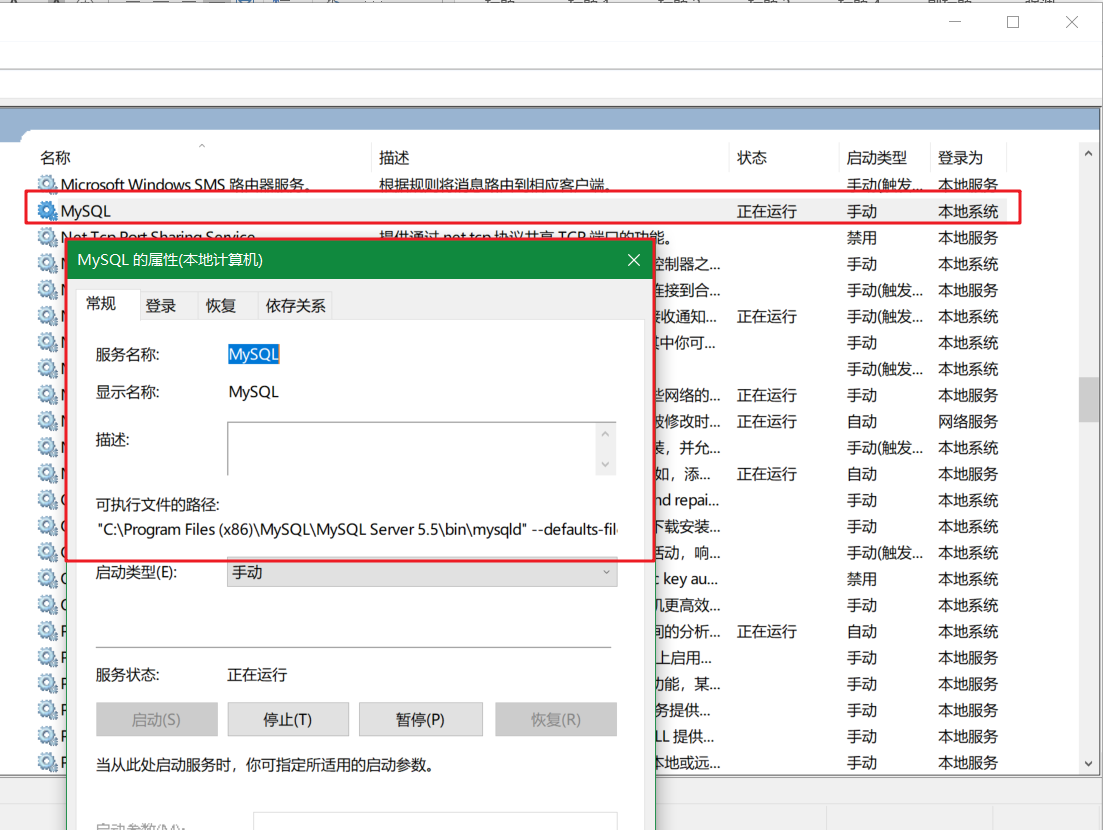
[https://dev.mysql.com/downloads/mysql/5.5.html#downloads](https://dev.mysql.com/downloads/mysql/5.5.html%23downloads)

按照说明进行软件的安装和配置：

<https://jingyan.baidu.com/article/335530daea392519cb41c326.html>

在命令窗口中键入services.msc命令，查看服务是否启动成功：





MySQL服务的启动、停止与卸载

在 Windows 命令提示符下运行:

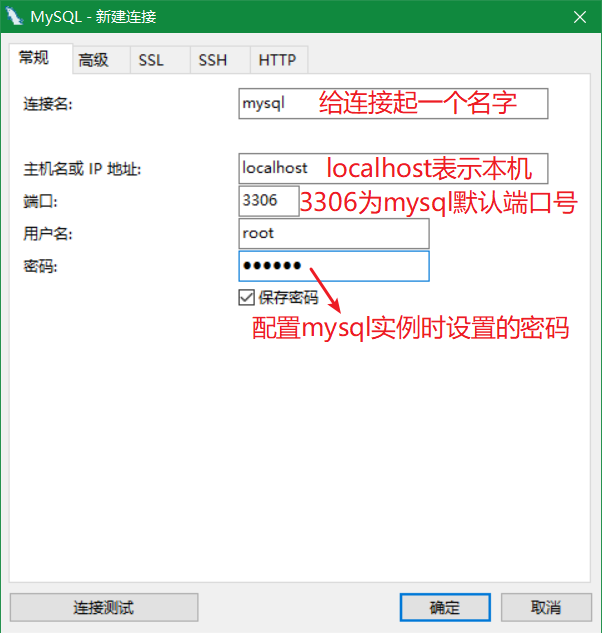
启动: net start MySQL

停止: net stop MySQL

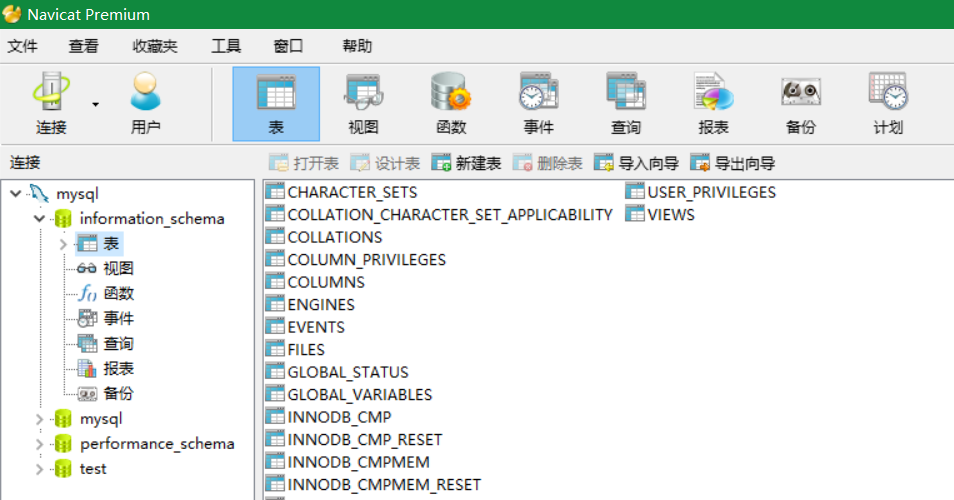
卸载: sc delete MySQL

## 数据库管理工具Navicat使用

安装并使用navicat，连接mysql数据库实例，新建mysql连接，如下图所示：



连接成功如下所示：



## MySQL中的数据类型

MySQL有三大类数据类型, 分别为数字、日期\时间、字符串, 这三大类中又更细致的划分了许多子类型:

**数字类型**

整数: tinyint、smallint、mediumint、int、bigint

浮点数: float、double、real、decimal

**日期和时间**: date、time、datetime、timestamp、year

**字符串类型**

字符串: char、varchar

文本: tinytext、text、mediumtext、longtext

二进制(可用来存储图片、音乐等): tinyblob、blob、mediumblob、longblob

这里不能详细对这些类型进行介绍了, 篇幅可能会很长, 详细介绍参见:

[《MySQL数据类型》](http://www.cnblogs.com/zbseoag/archive/2013/03/19/2970004.html) :<http://www.cnblogs.com/zbseoag/archive/2013/03/19/2970004.html>

## 创建数据库表

使用 create table 语句可完成对表的创建, create table 的常见形式:

create table 表名称(列声明);

以创建 student 表为例, 表中将存放 学号(id)、姓名(name)、性别(sex)、年龄(age)、联系电话(tel) 这些内容:

create table student

（

id int unsigned not null auto\_increment primary key,

name char(8) not null,

sex char(4) not null,

age tinyint unsigned not null,

tel char(13) null default "-"

);

**语句解说:**

create table tablename(columns) 为创建数据库表的命令, 列的名称以及该列的数据类型将在括号内完成;

括号内声明了5列内容, id、name、sex、age、tel为每列的名称, 后面跟的是数据类型描述, 列与列的描述之间用逗号(,)隔开;

以 "id int unsigned not null auto\_increment primary key" 行进行介绍:

* "id" 为列的名称;
* "int" 指定该列的类型为 int(取值范围为 -8388608到8388607), 在后面我们又用 "unsigned" 加以修饰, 表示该类型为无符号型, 此时该列的取值范围为 0到16777215;
* "not null" 说明该列的值不能为空, 必须要填, 如果不指定该属性, 默认可为空;
* "auto\_increment" 需在整数列中使用, 其作用是在插入数据时若该列为 NULL, MySQL将自动产生一个比现存值更大的唯一标识符值。在每张表中仅能有一个这样的值且所在列必须为索引列。
* "primary key" 表示该列是表的主键, 本列的值必须唯一, MySQL将自动索引该列。

下面的 char(8) 表示存储的字符长度为8, tinyint的取值范围为 -127到128, default 属性指定当该列值为空时的默认值。

更多的数据类型请参阅 [《MySQL数据类型》](http://www.cnblogs.com/zbseoag/archive/2013/03/19/2970004.html) :

<http://www.cnblogs.com/zbseoag/archive/2013/03/19/2970004.html>

## 操作MySQL数据库

### 向表中插入数据

insert 语句可以用来将一行或多行数据插到数据库表中, 使用的一般形式如下:

insert [into] 表名 [(列名1, 列名2, 列名3, ...)] values (值1, 值2, 值3, ...);

其中 [] 内的内容是可选的, 例如, 要给 samp\_db 数据库中的 student 表插入一条记录, 执行语句:

insert into student values(NULL, "王刚", "男", 20, "13811371377");

按回车键确认后若提示 如下 表示数据插入成功。



有时我们只需要插入部分数据, 或者不按照列的顺序进行插入, 可以使用这样的形式进行插入:

insert into student (name, sex, age) values("孙丽华", "女", 21);

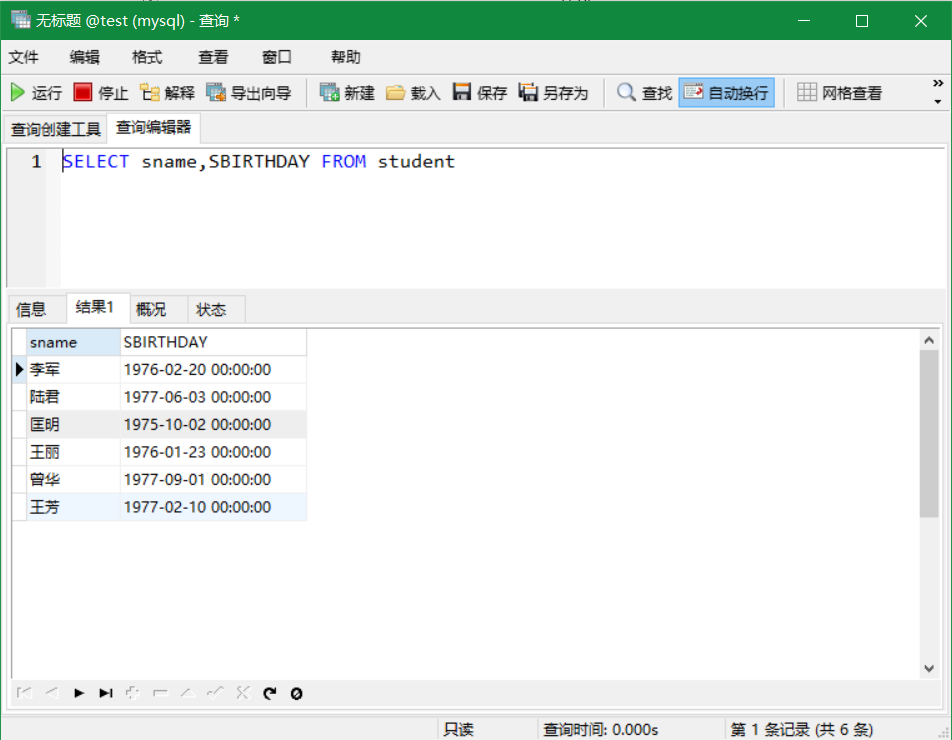
### 查询表中的数据

select 语句常用来根据一定的查询规则到数据库中获取数据, 其基本的用法为:

select 列名称 from 表名称 [查询条件];

例如要查询 student 表中所有学生的名字和年龄, 输入语句

SELECT sname,SBIRTHDAY FROM student; 执行结果如下:



也可以使用通配符 \* 查询表中所有的内容, 语句: select \* from student;

### **按特定条件查询:**

where 关键词用于指定查询条件, 用法形式为: select 列名称 from 表名称 where 条件;

以查询所有性别为女的信息为例, 输入查询语句:

select \* from student where ssex="女";

where 子句不仅仅支持 "where 列名 = 值" 这种名等于值的查询形式, 对一般的比较运算的运算符都是支持的, 例如 =、>、<、>=、<、!= 以及一些扩展运算符 is [not] null、in、like 等等。 还可以对查询条件使用 or 和 and 进行组合查询, 以后还会学到更加高级的条件查询方式, 这里不再多做介绍。

**示例:**

查询年龄在21岁以上的所有人信息:

select \* from student where age > 21;

查询名字中带有 "王" 字的所有人信息:

select \* from student where name like "%王%";

查询id小于5且年龄大于20的所有人信息:

select \* from student where id<5 and age>20;

### 更新表中的数据

update 语句可用来修改表中的数据, 基本的使用形式为:

update 表名称 set 列名称=新值 where 更新条件;

**使用示例:**

将id为5的手机号改为默认的"-":

update student set tel=default where id=5;

将所有人的年龄增加1:

update student set age=age+1;

将手机号为 13288097888 的姓名改为 "张伟鹏", 年龄改为 19:

update student set name="张伟鹏", age=19 where tel="13288097888";

### 删除表中的数据

delete 语句用于删除表中的数据, 基本用法为:

delete from 表名称 where 删除条件;

**使用示例:**

删除id为2的行:

delete from student where id=2;

删除所有年龄小于21岁的数据:

delete from student where age<20;

删除表中的所有数据:

delete from student;

## 创建后表的修改

alter table 语句用于创建后对表的修改, 基础用法如下:

### 添加列

基本形式:

alter table 表名 add 列名 列数据类型 [after 插入位置];

**示例:**

在表的最后追加列 address:

alter table student add address char(60);

在名为 age 的列后插入列 birthday:

alter table student add birthday date after age;

### 修改列

基本形式: alter table 表名 change 列名称 列新名称 新数据类型;

**示例:**

将表 tel 列改名为 telphone:

alter table student change tel telphone char(13) default "-";

将 name 列的数据类型改为 char(16):

alter table student change name name char(16) not null;

### 删除列

基本形式: alter table 表名 drop 列名称;

**示例:**

删除 birthday 列:

alter table student drop birthday;

### 重命名表

基本形式: alter table 表名 rename 新表名;

**示例:**

重命名 student 表为 workmates:

alter table student rename workmates;

### 删除整张表

基本形式: drop table 表名;

**示例:** 删除 workmates 表: drop table workmates;

### 删除整个数据库

基本形式: drop database 数据库名;

**示例:** 删除 samp\_db 数据库: drop database samp\_db;

## 附录

### 可视化管理工具 MySQL Workbench

尽管我们可以在命令提示符下通过一行行的输入或者通过重定向文件来执行mysql语句, 但该方式效率较低, 由于没有执行前的语法自动检查, 输入失误造成的一些错误的可能性会大大增加, 这时不妨试试一些可视化的MySQL数据库管理工具, MySQL Workbench 就是 MySQL 官方 为 MySQL 提供的一款可视化管理工具, 你可以在里面通过可视化的方式直接管理数据库中的内容, 并且 MySQL Workbench 的 SQL 脚本编辑器支持语法高亮以及输入时的语法检查, 当然, 它的功能强大, 绝不仅限于这两点。

MySQL Workbench官方介绍: <http://www.mysql.com/products/workbench/>

MySQL Workbench 下载页: <http://dev.mysql.com/downloads/tools/workbench/>

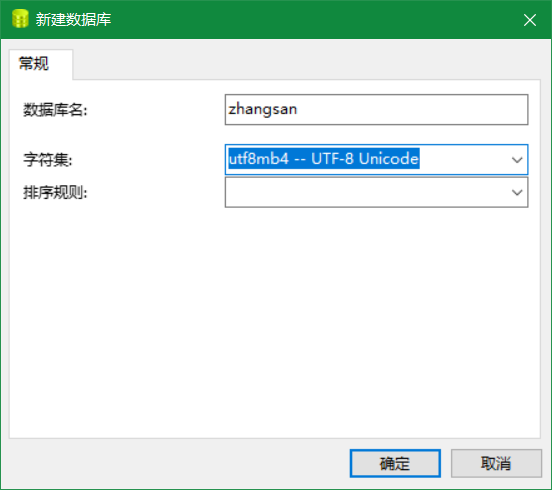
以上内容来自于网络（<http://www.cnblogs.com/mr-wid/archive/2013/05/09/3068229.html>）

在教室通过FTP或者百度云进行下载安装资源包：

链接: <https://pan.baidu.com/s/1_EB9egssto9ulx2LFharCQ> 提取码: esrd

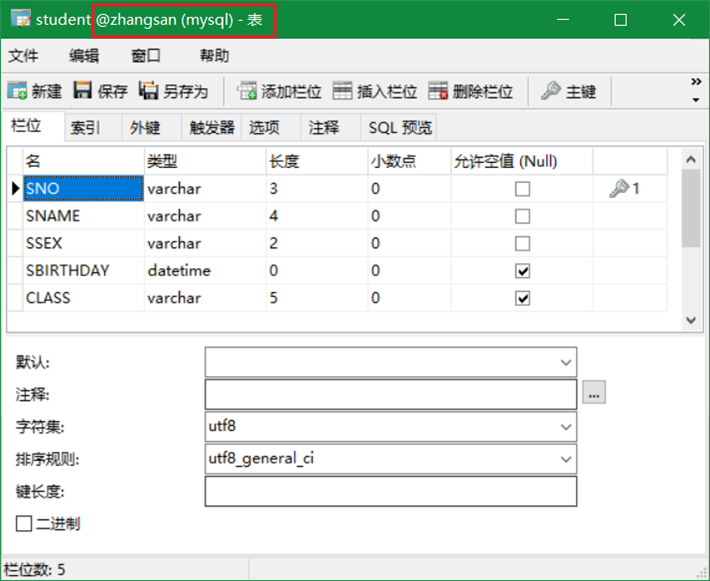
## 【内容】

以自己姓名的全拼作为数据库名，数据库编码为【utf8mb4 -- UTF-8 Unicode】

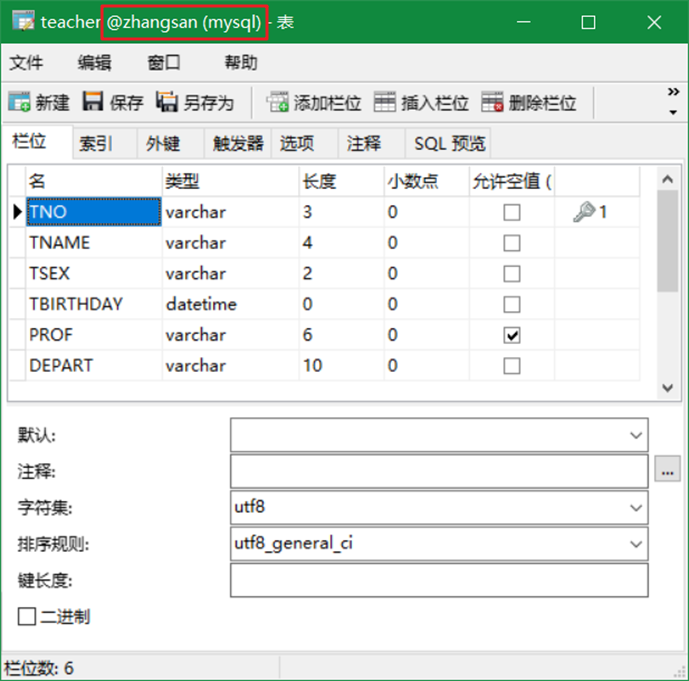


创建三张表，课程【course】，学生【student】，教师【teacher】

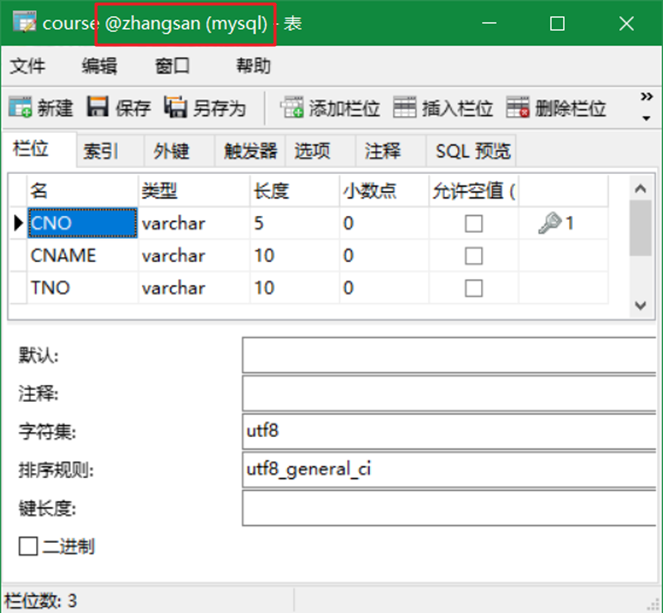
学生表【student】结构如下图所示：



教师表【teacher】结构如下图所示：



课程表【course】结构如下：



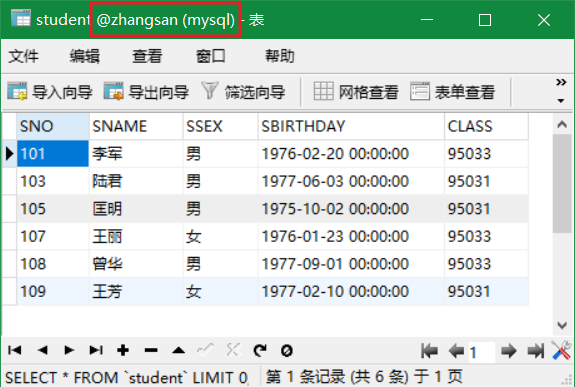
每张表添加不少于5条记录（记录内容自拟）

提交产物包括：

1、每张表数据的截图（如下图示例）

2、创建表的SQL脚本（create table……）

3、插入记录的SQL脚本（insert into…….）



## 【注意事项】

1. 提交产物为相关问题的回答以及java源代码和脚本（xxx.java文件和xxx.sql）
2. 提交方式，通过博思平台进行提交，截止时间详见平台