**实验三**

### 实验目的

学习使用servlet和JDBC访问mysql数据库。

### 实验内容

1. **要求：将程序部署在服务器上，观察程序的运行,掌握通过jdbc的方式实现对数据库表中数据的增删改，掌握程序的实现原理。（完成1即可）**

2、（课程设置原因只做1即可，2不用做）数据库中增加“联系电话”列，在index.jsp界面增加“联系电话”输入框，修改model/User.java文件（user对象），修改dao/UserDao.java文件（实现对数据的插入sql语句），发布并运行程序。

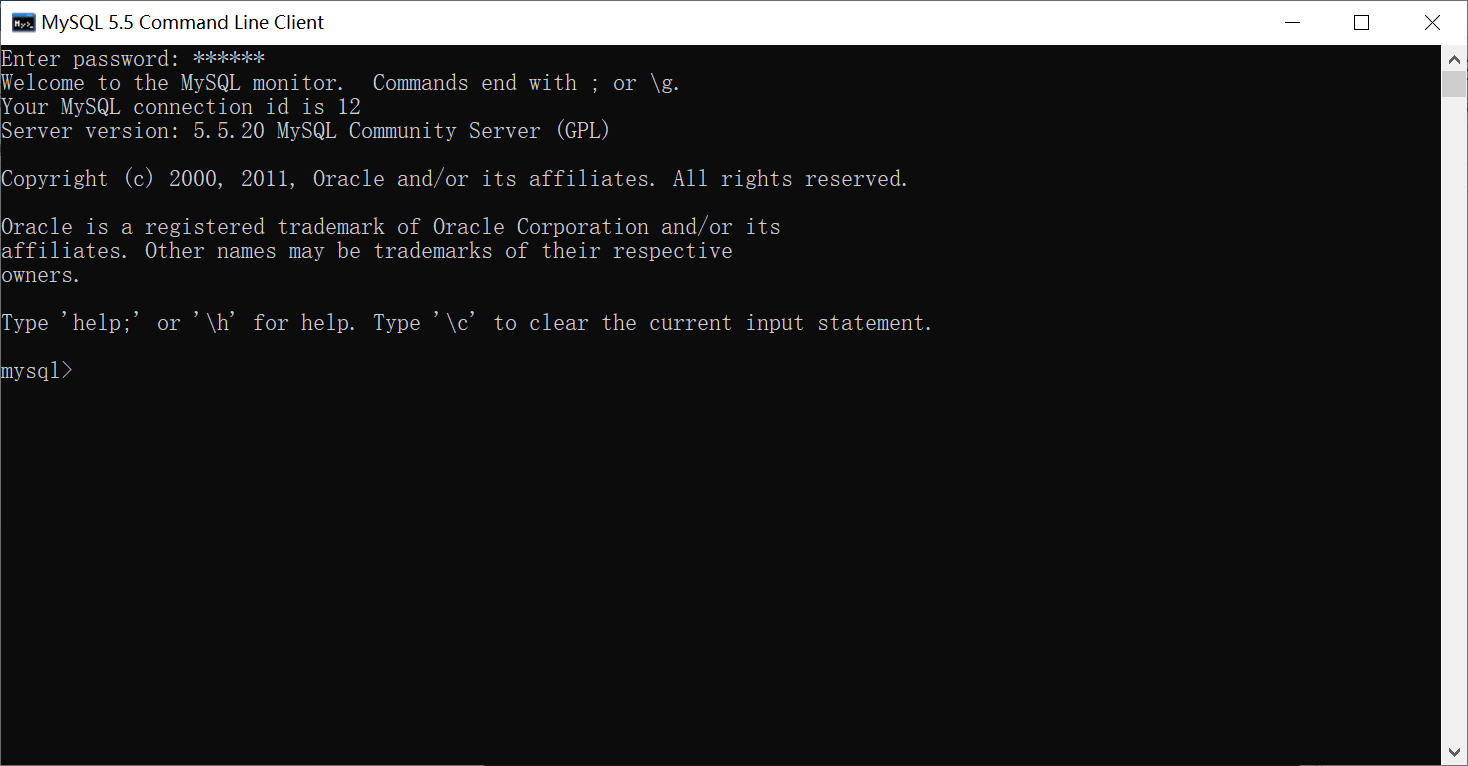
### 实验流程

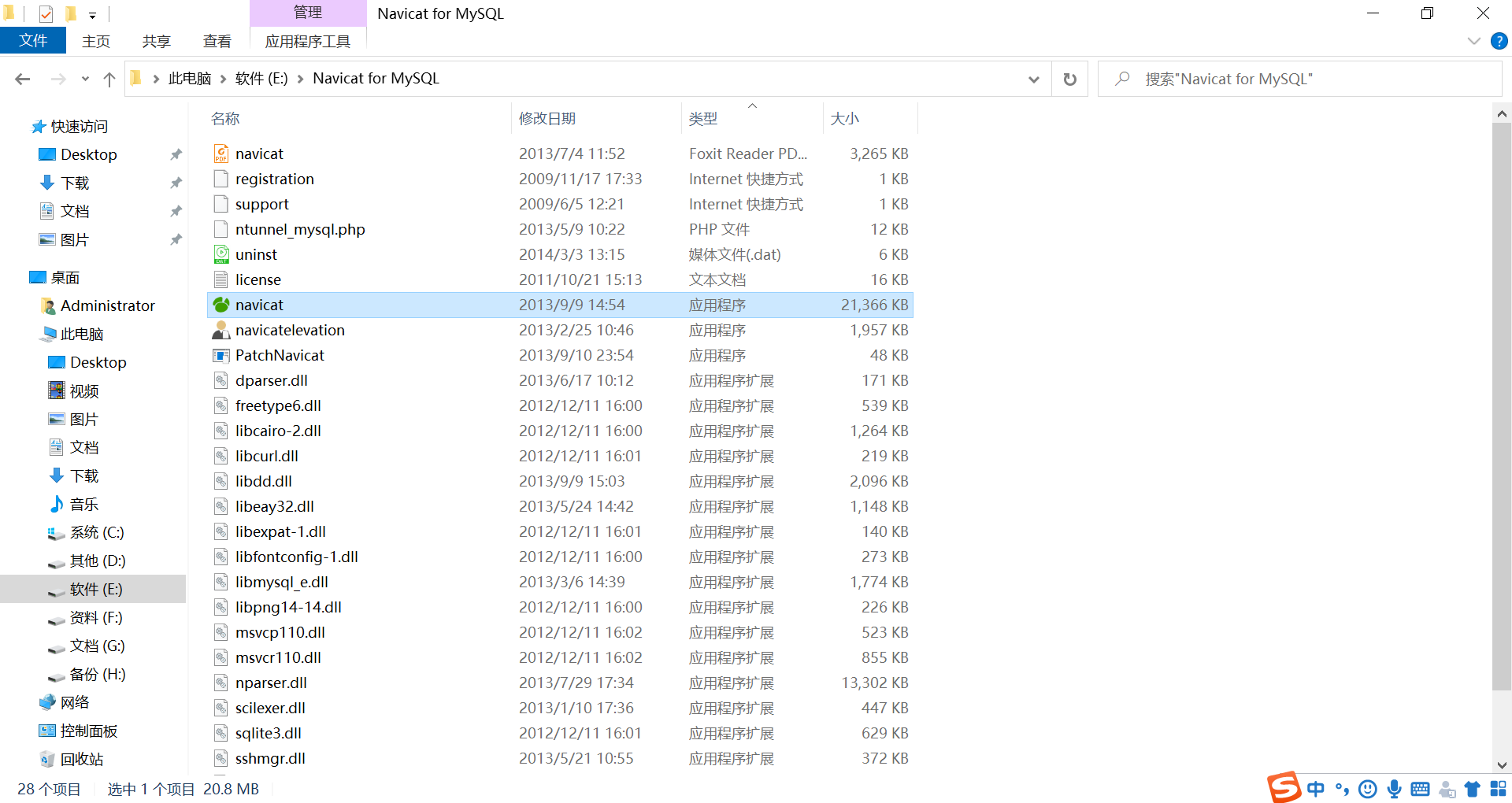
1. 将工程jdbc导入eclipse开发环境，并部署到tomcat服务器；
2. 导入工程流程：首先点击菜单file🡪import，在弹出的对话框中选择general🡪exsiting project into workspace，点击browse，选中jdbc文件夹，最后点击finish.
3. 创建数据库和表
4. 安装mysql（实验室电脑安装的是MySQL5.5 64位），注意root密码，由于jdbc驱动与mysql版本有关，因此JDBC版本与MySQL版本要匹配。（所给的工程所用的jdbc connector是以Mysql5.5版本匹配的）

(实验室环境已安装Mysql 5.5)

1. 安装mysql工具（解压Navicat for MySQL，并运行navicat.exe）（实验室环境已安装）

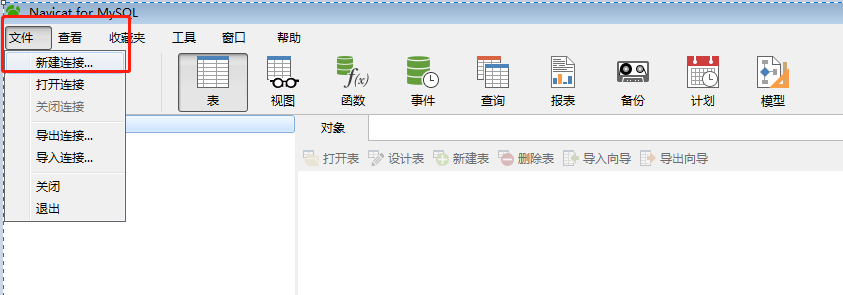
3） MySQL服务器占用的端口是3306（3306是MySQL服务器默认使用的端口号）。启动成功后（无任何提示信息），MySQL数据库服务器将占用当前MS-DOS窗口

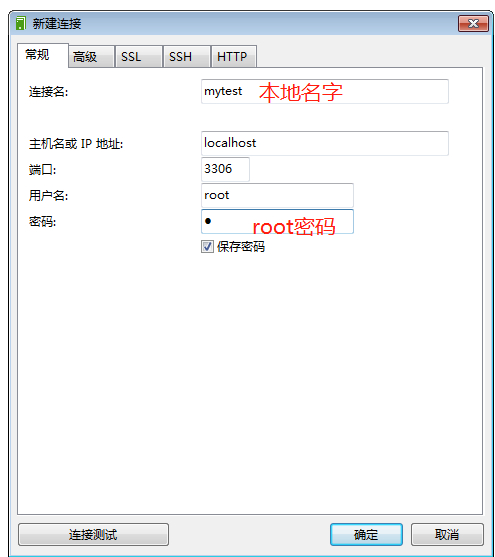




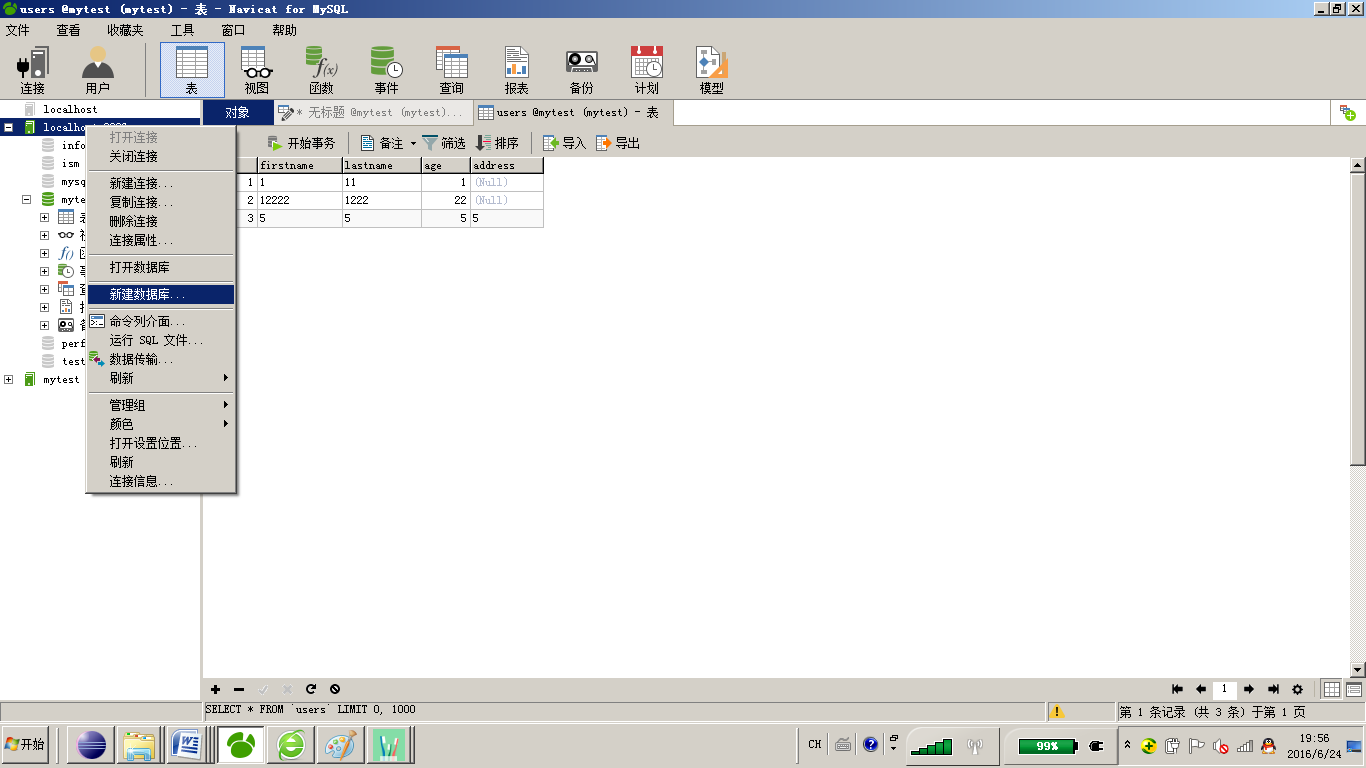
1. 建立数据库连接

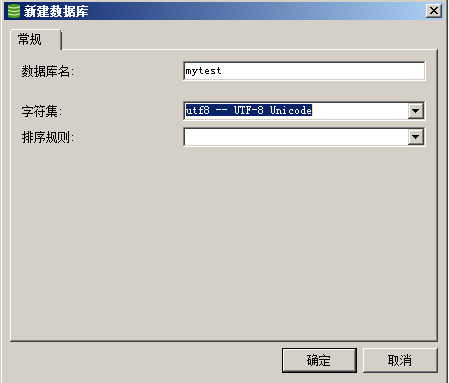
连接名自定义，root设置为MySQL的root密码。





1. 建立数据库和数据库表和列

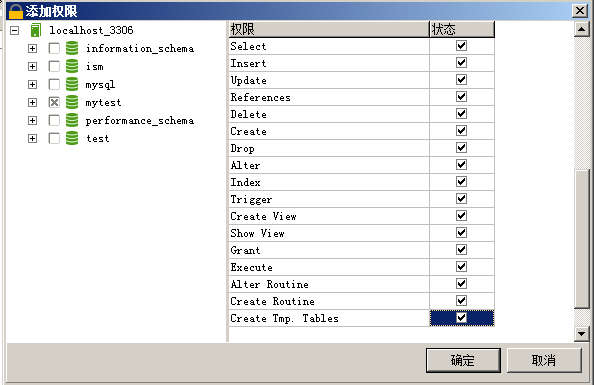




1. 点击，新建用户，密码为MySQL密码



1. 点击“权限”-“添加权限”



1. 确定，回到主界面保存
2. 点击，再点击新建查询，输入如下sql语句

USE `mytest`;

DROP TABLE IF EXISTS `user`;

CREATE TABLE `user` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`name` varchar(10) DEFAULT NULL,

`password` varchar(15) DEFAULT NULL,

`date` datetime DEFAULT NULL,

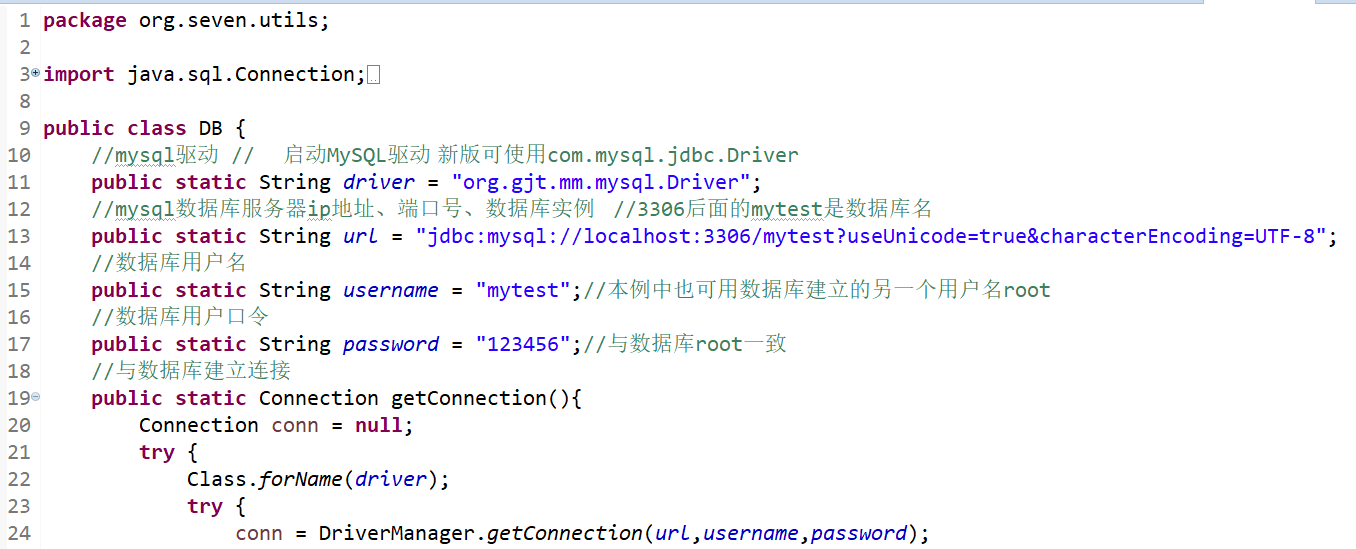
PRIMARY KEY (`id`)

) ;

1. 点击运行
2. 启动tomcat服务器，在eclipse上运行工程（运行index.jsp，选择run on server）;或在浏览器上输入“http://localhost:8080/jdbc”运行程序，查看运行效果。

### 重点内容

1. JDBC使用流程：
2. jdbc配置：src/utils/db.java



1. servlet注册

WebContent/WEB-INF/web.xml

servlet的名字为“add”

接收jsp页面的数据请求

<url-pattern>/add.do</url-pattern>

将请求的数据转向给后台java程序来处理：

<servlet-class>org.seven.web.AddServlet</servlet-class>



1. java文件说明

dao：负责与数据库交互

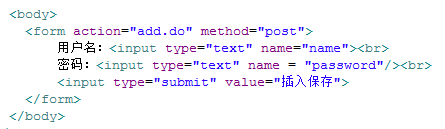
model：业务逻辑，在此文件中可以写一些业务逻辑，例如：数据转换、数据校验等

web：servlet，负责接收jsp页面请求，并将数据交由dao去处理

utils：jdbc设置（数据库驱动、实例名、用户名、口令）

1. 数据插入

index.jsp：首页文件，包含数据插入及保存



将前台页面输入的用户名和口令，通过web.xml文件中sevlet注册的“add.do”，将数据发送给src/web/addServlet.java文件去处理。

* addServlet.java的处理步骤：  
  获取jsp页面传送过来的用户名和口令
* 声明user对象，并将用户名和口令赋予该对象
* 声明userDao对象，并执行保存，参数为user对象
* 保存成功后，将结果返回给list.jsp页面

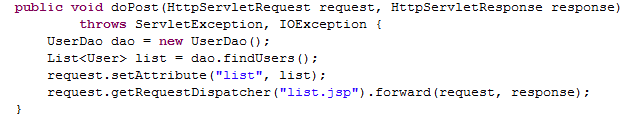
1. list.jsp：数据显示



数据显示：例如addServlet.java

resp.sendRedirect("list.do");

通过web.xml文件中的servlet注册，交由src/web/ListServlet.java去处理



直接调用dao的findUsers方法将数据获取出来，并传递给list.jsp页面显示

1. 数据修改

* 在list.jsp页面，通过“toupdate.do”找到对应的servlet：ToupdateServlet.java
* 在ToupdateServlet.java中，通过前台页面的id使用dao查询到该条数据，并赋予user对象
* 将user对象传递给update.jsp文件，显示该条数据
* 在update.jsp页面，通过“update.do”，交由updateServlet.java，去保存

1. 删除操作：delete.do

* 在list.jsp页面，通过“delete.do”找到对应的servlet：deleteServlet.java
* 在deleteServlet.java中，通过前台页面的id使用dao查询到该条数据，并赋予user对象
* 在deleteServlet.java文件中，通过dao去删除数据。

### 注意事项

* dao中的所有对数据库的操作，最终都会形成一个sql语句来执行
* 每次数据库操作的第一步都要先建立数据库连接，执行完后必须要释放连接，如果不释放，数据库连接越来越多，最后数据库会崩溃
* 大型应用系统，都会有连接池机制，交由中间件（weblogic)去处理。