**Experiment7**

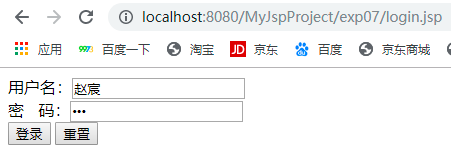
1.利用JavaBean、数据库等技术模拟用户登陆程序

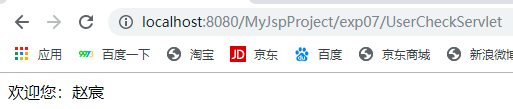
P217页程序改进，建议使用连接池技术，比如druid

部分代码，见第十二章压缩包，该程序没建立数据库，没使用连接池，请重写并完善。

**运行截图：**

**登录成功：**

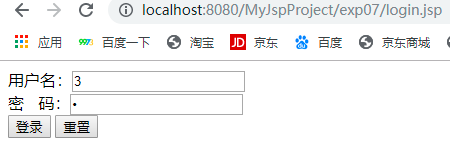


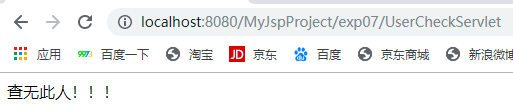


**密码不正确：**



**数据库中没有此人信息：**





**代码：**

**c3p0数据库连接池，配置文件：cn/spring/c3p0/beans.xml**

**JavaBean:** **UserBean.java**

**Servlet: UserCheckServlet.java**

**数据库操作：UserDao.java**

**Springc3p0读取文件： cn/spring/c3p0/jdbcInfo.properties**

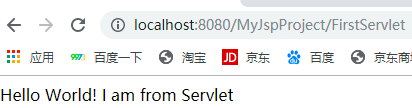
**jsp文件：login.jsp, welcome.jsp, error.jsp**

2.利用Servlet技术，重写HttpServlet类中的doPost()方法, 生成"Hello World！ I am from Servlet"

参考：<https://blog.csdn.net/evankaka/article/details/45151569>

<https://blog.csdn.net/u011457627/article/details/50172307>

**运行截图：**



**代码：FirstServlet.java**

3.请在1题的基础上，利用JavaBean、Servlet、数据库等技术，采用MVC模型设计用户登陆程序

<https://www.cnblogs.com/biehongli/p/6078740.html>

<https://www.cnblogs.com/optblogs/p/7880996.html>

<https://blog.csdn.net/csdn_zsdf/article/details/76456054\>

MVC全名是Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。

M = JavaBean：用于传递数据，拥有与数据相关的逻辑处理

V = JSP：从Model接收数据并生成HTML

C = Servlet：接收HTTP请求并控制Model和View

**代码和运行截图同题目1：**

**M（模型）：UserBean.java**

**V（视图）：jsp文件**

**C（控制器）：UserDao.java和UserCheckServlet.java**