**Java Web开发**

**实验八**

**作业报告**

|  |  |
| --- | --- |
| 学 院 | 计算机 |
| 所在系 班 级 | 2017软件工程系9班 |
| 学 号 姓 名 | 苟俊杰 |
| 指 导 教 师 | 郭晓燕 |
| 完 成 时 间 | 2019年5月 |

# 实验题目

1、（必做）请用模型2MVC模式完成1个功能，功能自选，例如：登录、注册、发帖、计算税收等；

2、（选做）继续完善网上购图书程序，采用MVC模式，完成增加计算商品总价及添加商品数量等功能，用JavaBean改写，其中购物车和商品价格等内容都用JavaBean实现，JSP页面实现增加、删除所选图书，显示所选图书及其价格的功能。尽量用EL表达式或JSTL标签库，少用程序片及表达式。

购物车提示：javaBean：

Book.java：存储一本图书的信息

BookList.java：一条记录，用于记录订单的一条信息(可省略)

CartBook.java：购物车，用于存储用户所有购物信息

JSP：

index.jsp：购图书首页

addbook.jsp:添加一条图书信息至购物车。

delbook.jsp:删除一条图书信息至购物车

showCart.jsp:显示购物车的信息

# 分析与设计

**此题需要设计4个文件完成。**

**首先需要一个JSP叫Login.jsp做为表单用来实现用户输入用户名和密码实现登陆。**

**点击登陆按钮后跳转到**LoginVerify**.java中。**

<%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"UTF-8"*%>

<%

String path = request.getContextPath();

String basePath = request.getScheme()+"://"+request.getServerName()+":"+request.getServerPort()+path+"/";

%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

<head>

<base href=*"*<%=basePath%>*"*>

<title>My JSP 'Login.jsp' starting page</title>

<meta http-equiv=*"pragma"* content=*"no-cache"*>

<meta http-equiv=*"cache-control"* content=*"no-cache"*>

<meta http-equiv=*"expires"* content=*"0"*>

<meta http-equiv=*"keywords"* content=*"keyword1,keyword2,keyword3"*>

<meta http-equiv=*"description"* content=*"This is my page"*>

<!--

<link rel="stylesheet" type="text/css" href="styles.css">

-->

</head>

<body>

<% request.setCharacterEncoding("UTF-8"); %>

<form action = *"LoginVerify.do"* method = *"post"*>

用户名<input type = *"text"* name = *"name"*><br>

密码<input type = *"password"* name = *"pwd"*>

<input type = *"submit"* value = *"登陆"*>

</body>

</html>

**第二个文件是一个VerifyData.java的文件用来实现封装业务逻辑和数据的javaBean。此页面用以根据用户输入的用户名和密码，进行判断是否一致。**

**package** Mlevel;

**public** **class** VerifyData {

**private** String name;

**private** String pwd;

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getPwd() {

**return** pwd;

}

**public** **void** setPwd(String pwd) {

**this**.pwd = pwd;

}

**public** **boolean** Verify(String name, String pwd) {

**if**(name.equals("gjj") && pwd.equals("goujunjie"))

**return** **true**;

**else**

**return** **false**;

}

}

**第三个文件是LoginVerify.java的servlet文件，用来调用javabean的方法，并根据情况实现跳转。若判断正确跳转到success.jsp，若失败重定向回Login.jsp的登陆页面。**

**package** LoginVerify;

**import** java.io.IOException;

**import** javax.servlet.ServletException;

**import** javax.servlet.annotation.WebServlet;

**import** javax.servlet.http.HttpServlet;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**import** Mlevel.VerifyData;

@WebServlet("/LoginVerify.do")

**public** **class** LoginVerify **extends** HttpServlet {

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

// **TODO** Auto-generated method stub

doPost(request,response);

}

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

// **TODO** Auto-generated method stub

String name = request.getParameter("name");

String pwd = request.getParameter("pwd");

VerifyData verifyData = **new** VerifyData();

**boolean** flag = verifyData.Verify(name, pwd);

**if**(flag)

request.getRequestDispatcher("success.jsp").

forward(request, response);

**else**

response.sendRedirect("Login.jsp");

}

}

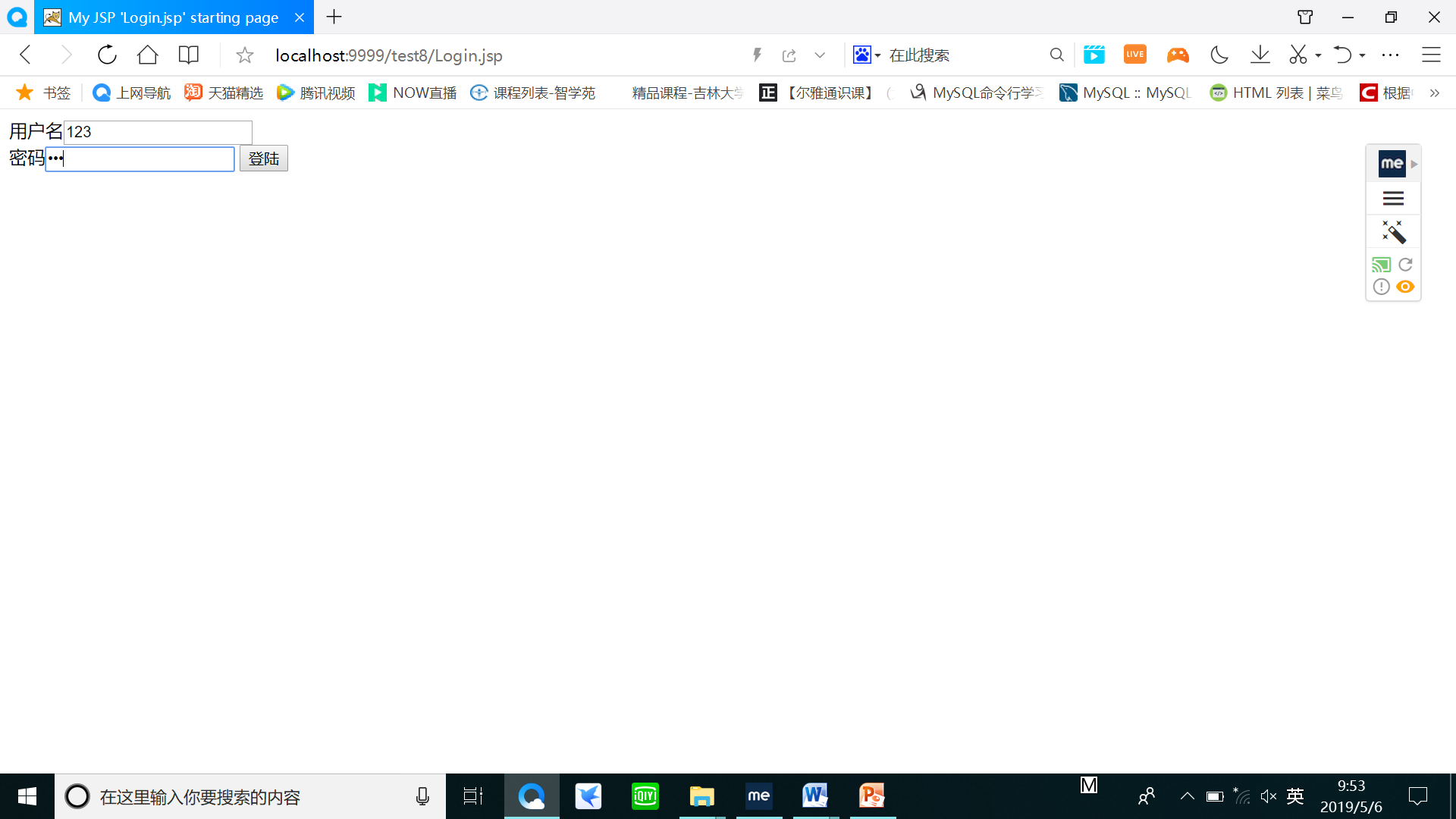
第四个是success.jsp的登陆成功页面。提示用户登陆成功。

<body>

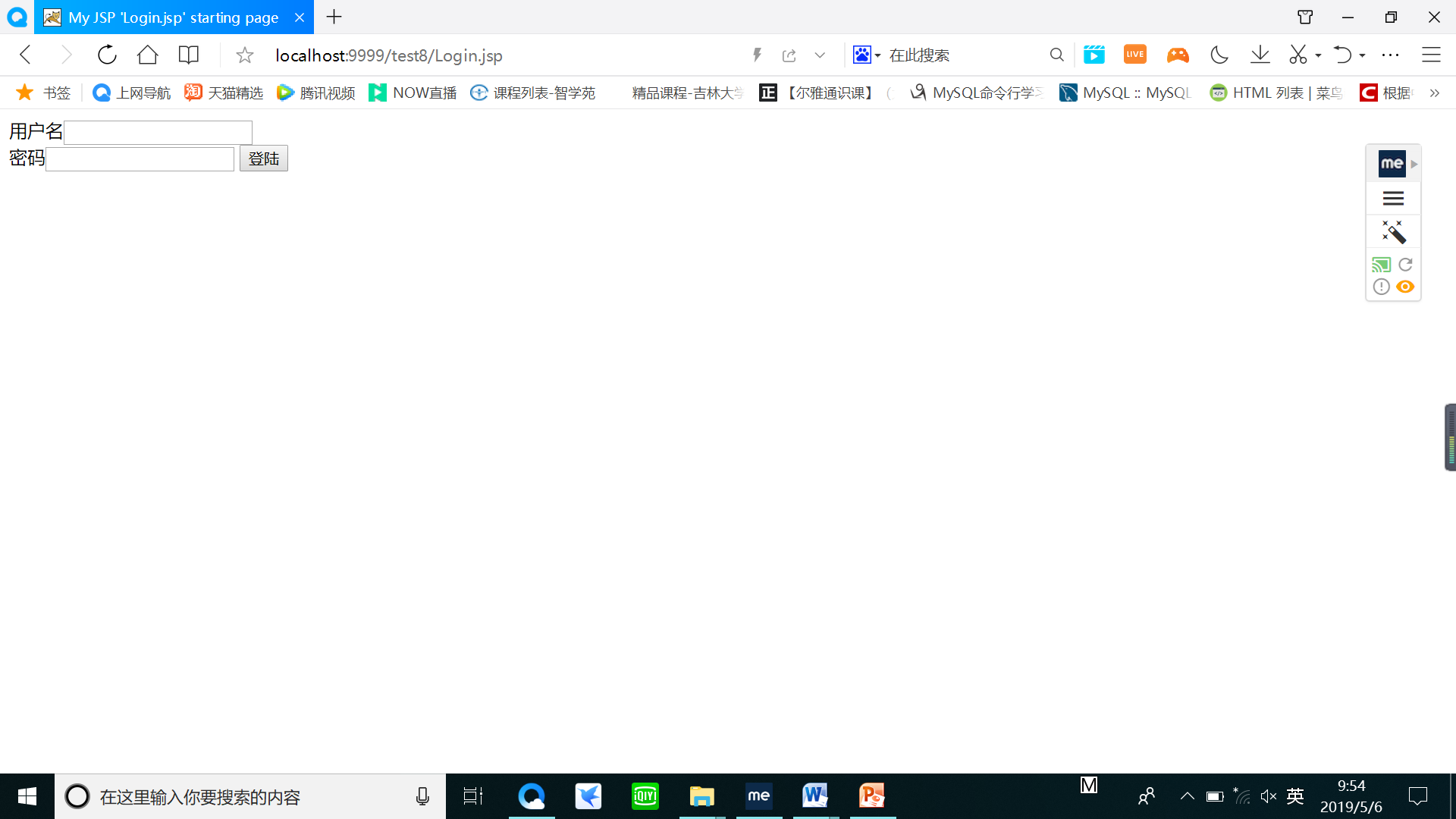
登陆成功<br>

</body>

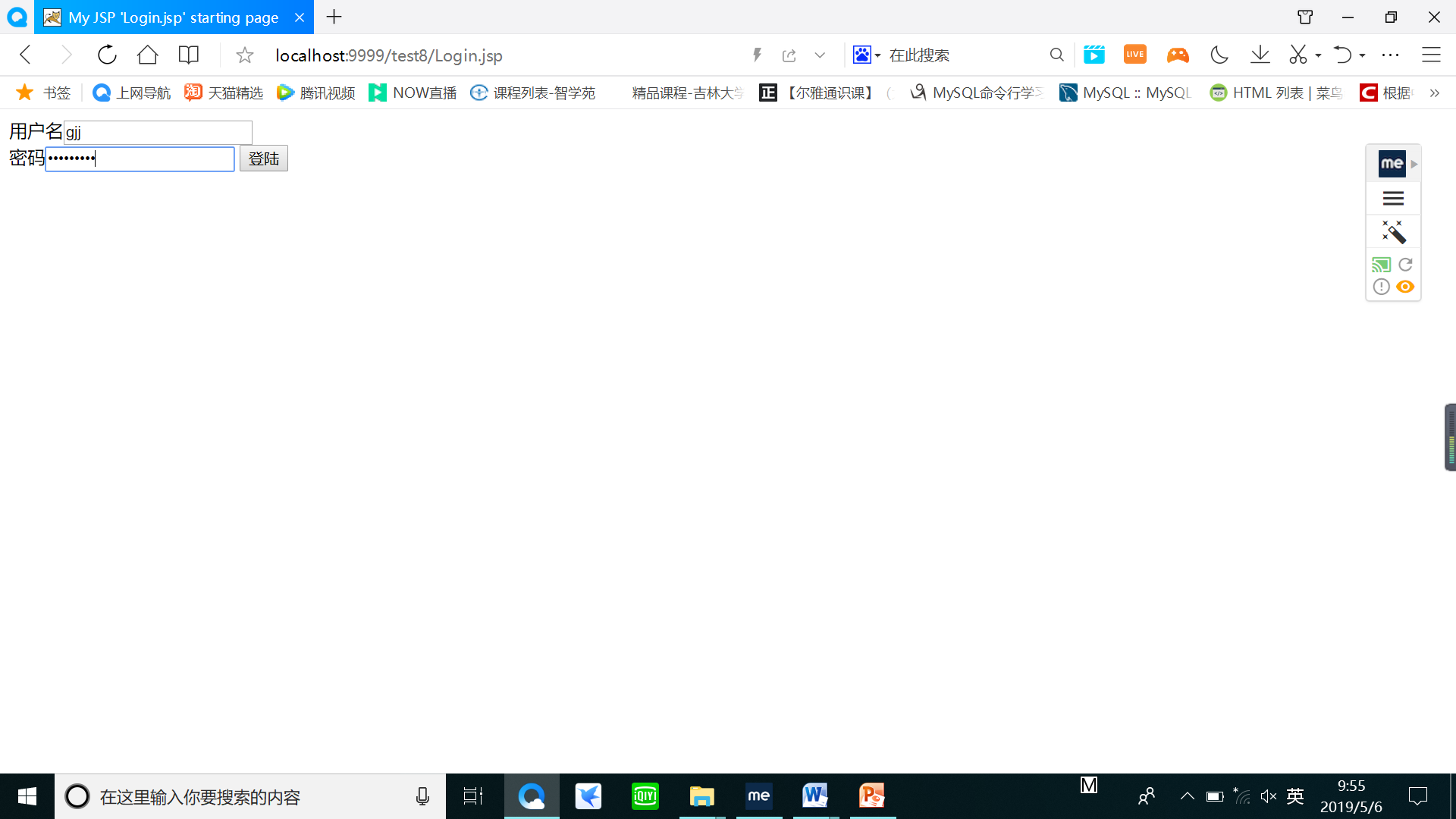
# 运行效果图



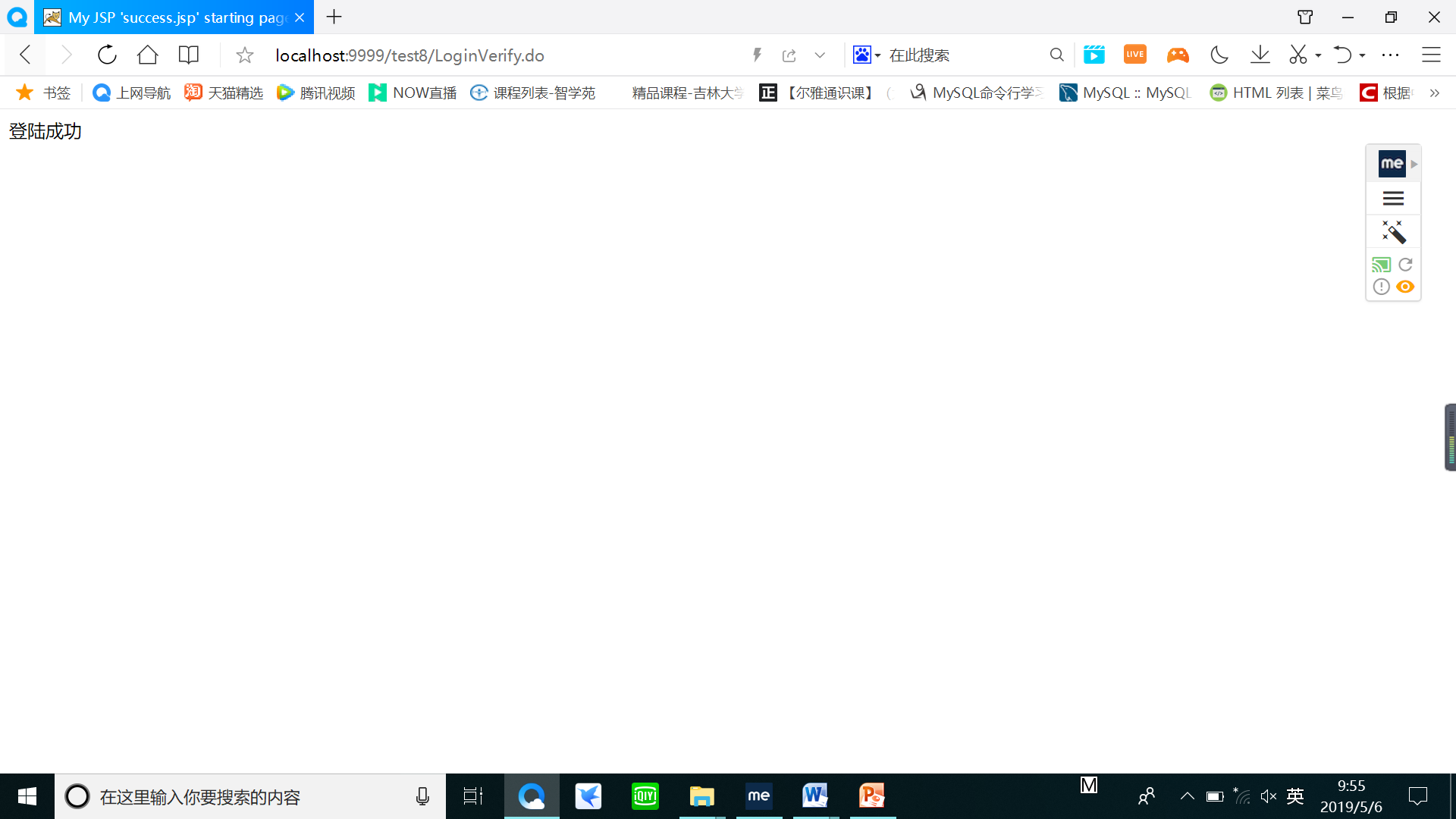
上图为Login.jsp页码负责输入用户的用户名和密码。此处我先输入了错误的用户名。



点击登陆后，因为是错误的用户名，所以需要重新登陆。



此次输入正确的用户名和密码并点击登陆。



提示用户登陆成功。

# 收获与问题

掌握JavaBean方法的使用，了解JavaBean在Java中的两个类。封装业务逻辑的JavaBean和封装数据的JavaBean。还需熟练掌握MVC模式的书写方法，将各种功能分开实现。