**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07班 |
| 学 号 姓 名 | 04171503刘建超 |
| 指 导 教 师 | 郭晓燕 |
| 完 成 时 间 | 2018年4月-25 |

# 实验题目

实验内容：

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；

# 分析与设计

设计一个JSP作为登陆界面，设置一个JavaBean作为计算个人应交的所得随，设置第二个JSP作为成功输出结果的界面，完成mvc模式。

# 关键源码（部分）

index.jsp

<%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<title>javabean基本应用</title>

</head>

<body>

<form action=*"shouru"* method=*"post"*>

<table align=*"center"* border=*"1"* weight=*"200"* height=*"200"*>

<tr align=*"center"*><td>请填写本人收入信息</td></tr>

<tr><td>收入：<input type=*"text"* name=*"shouru"*></td></tr>

<tr align=*"center"*><td><input type=*"submit"* value=*"提交信息"*></td></tr>

</table>

</form>

</body>

</html>

Index2.jsp

<%@ 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 'ex8success.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>

<br>

您应缴纳的个人所得税为：${user.dr}

</body>

</html>

Shouru.java

package test7;

package test7;

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;

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 javax.servlet.http.HttpSession;

/\*\*

\* Servlet implementation class shouru

\*/

@WebServlet("/shouru")

public class shouru extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public shouru() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

\*/

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

response.setContentType("text/html");

response.setCharacterEncoding("UTF-8");

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

double dr=Integer.parseInt(name);

test7 user=new test7();

user.setDr(dr);

HttpSession session=request.getSession();

session.setAttribute("user", user);

RequestDispatcher dispatcher = request.getRequestDispatcher("index2.jsp");

dispatcher.forward(request, response);

}

/\*\*

\* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

\*/

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

// TODO Auto-generated method stub

doGet(request, response);

}

}

Test7.java

**package** test7;

**public** **class** test7

{

**public** **double** dr;

**public** test7()

{

}

**public** **double** getDr() {

**return** dr;

}

**public** **void** setDr(**double** dr)

{

**if**(dr<=1600)

**this**.dr=0;

**if**(dr>1600&&dr<=3000)

**this**.dr=(dr-1600)\*0.1-25;

**if**(dr>3000&&dr<=5000)

**this**.dr=(dr-1600)\*0.15-125;

**if**(dr>5000&&dr<=20000)

**this**.dr=(dr-1600)\*0.2-375;

**if**(dr>20000&&dr<=40000)

**this**.dr=(dr-1600)\*0.25-1375;

**if**(dr>40000&&dr<=60000)

**this**.dr=(dr-1600)\*0.3-3375;

**if**(dr>60000&&dr<=80000)

**this**.dr=(dr-1600)\*0.35-6375;

**if**(dr>80000&&dr<=100000)

**this**.dr=(dr-1600)\*0.4-10375;

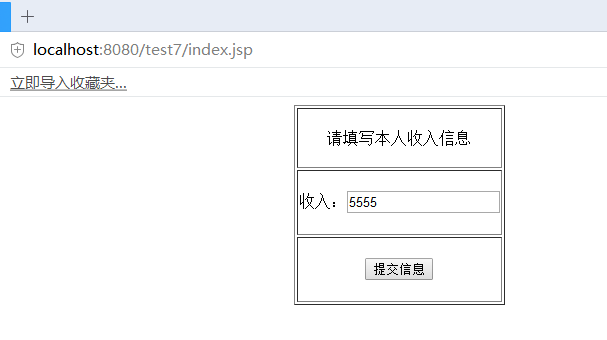
**if**(dr>100000)

**this**.dr=(dr-1600)\*0.45-15375;

}

}

# 运行效果图





# 收获与问题

理解MVC模式的基本思想；

理解两种模型：模型1及模型

熟练掌握MVC模式的编程。