**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06班 |
| 学 号 姓 名 | 04171419 刘银升 |
| 指 导 教 师 | 郭晓燕 |
| 完 成 时 间 | 2019年5月- |

# 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

根据题目要求，计划设计几个文件来完成实验，每个文件的名字、作用。如果是类，要求写出类的变量和方法

# 关键源码（部分）

贴上这道题中最关键的部分源码。也是你实验练习的主要知识点相关的源码。并在代码旁附上详细的解释

**package** cn.mvcproject.model;

**import** java.util.Date;

/\*\*

\* 用戶类

\* **@author** 35140

\*

\*/

**public** **class** User {

**private** **int** id;

**private** String username;

**private** String password;

**private** String phoneNo;

**private** String address;

**private** Date regDate;

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

**public** String getPhoneNo() {

**return** phoneNo;

}

**public** **void** setPhoneNo(String phoneNo) {

**this**.phoneNo = phoneNo;

}

**public** String getAddress() {

**return** address;

}

**public** **void** setAddress(String address) {

**this**.address = address;

}

**public** Date getRegDate() {

**return** regDate;

}

**public** **void** setRegDate(Date date) {

**this**.regDate = date;

}

**public** User(**int** id, String username, String password, String phoneNo, String address, Date regDate) {

**super**();

**this**.id = id;

**this**.username = username;

**this**.password = password;

**this**.phoneNo = phoneNo;

**this**.address = address;

**this**.regDate = regDate;

}

**public** User() {

**super**();

// **TODO** Auto-generated constructor stub

}

@Override

**public** String toString() {

**return** "User [id=" + id + ", username=" + username + ", password=" + password + ", phoneNo=" + phoneNo

+ ", address=" + address + ", regDate=" + regDate + "]";

}

}

package cn.mvcproject.service;

import java.util.List;

import cn.mvcproject.model.User;

public interface UserService {

/\*\*

\* 实现插入一条新用户信息数据

\* @param user

\* @return

\*/

public int save(User user) ;

/\*\*

\* 根据用户编号删除对应的用户数据

\* @param id

\* @return

\*/

public int deleteUserById(int id);

/\*\*

\* 根据用户ID修改对应用户信息数据

\* @param id

\* @return

\*/

public int updateUserById(User user);

/\*\*

\* 根据用户编号获取一条用户数据，封装成类User的一个对象

\* @param conn

\* @param id

\* @return

\*/

public User get(int id);

/\*\*

\* @throws SQLException

\* 支持事物的

\* @param conn

\* @param id

\* @return

\* @throws

\*/

public User getTransation(int id);

/\*\*

\* 获取所以用户数据

\* @return

\*/

public List<User> getListAll();

/\*\*

\* 查询指定用户名的用户有多少条

\* @param username

\* @return

\*/

public long getCountByName(String username);

/\*\*

\* 这是用户模糊查询的方法

\* @param username

\* @param address

\* @param phoneNo

\* @return

\*/

public List<User> query(String username, String address, String phoneNo);

}

package cn.mvcproject.service;

import java.sql.Connection;

import java.sql.SQLException;

import java.util.List;

import cn.mvcproject.dao.FactoryDao;

import cn.mvcproject.dao.UserDao;

import cn.mvcproject.dao.UserDaoImpl;

import cn.mvcproject.model.User;

import cn.mvcproject.utils.JdbcUtils;

public class UserServiceImpl implements UserService {

UserDao userDao=FactoryDao.getUserDao();

public int save(User user) {

return userDao.save(user);

}

@Override

public int deleteUserById(int id) {

return userDao.deleteUserById(id);

}

@Override

public int updateUserById(User user) {

return userDao.updateUserById(user);

}

@Override

public User get(int id) {

return userDao.get(id);

}

@Override

public User getTransation(int id){

Connection conn=null;

User user=null;

try {

conn=JdbcUtils.getConnection();

conn.setAutoCommit(false);//开启事务

user=userDao.get(conn, id);

conn.commit();

} catch (Exception e) {

JdbcUtils.rollbackTransation(conn);//回滾事物

}

return user;

}

@Override

public List<User> getListAll() {

return userDao.getListAll();

}

@Override

public long getCountByName(String username) {

return userDao.getCountByName(username);

}

@Override

public List<User> query(String username, String address, String phoneNo) {

return userDao.query(username,address,phoneNo);

}

}

package cn.mvcproject.controller;

import java.io.IOException;

import java.lang.reflect.Method;

import java.util.Date;

import java.util.List;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import com.sun.xml.internal.bind.v2.model.core.ID;

import cn.mvcproject.model.User;

import cn.mvcproject.service.FactoryService;

import cn.mvcproject.service.UserService;

import cn.mvcproject.service.UserServiceImpl;

public class UserController extends HttpServlet{

private static final long serialVersionUID = 1L;

UserService userService=FactoryService.getUserService();

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

doPost(req, resp);

}

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

req.setCharacterEncoding("UTF-8");

resp.setCharacterEncoding("UTF-8");

String mn=req.getServletPath();

mn=mn.substring(1);

mn=mn.substring(0, mn.length()-4);

System.out.println(mn);

try {

Method method=this.getClass().getDeclaredMethod(mn,HttpServletRequest.class,HttpServletResponse.class );

method.invoke(this, req,resp);

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

@SuppressWarnings("unused")

private void add(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

User user =new User();

user.setUsername(req.getParameter("username"));

user.setPassword(req.getParameter("password"));

user.setPhoneNo(req.getParameter("phoneNo"));

user.setAddress(req.getParameter("address"));

user.setRegDate(new Date());

int rows=userService.save(user);

if(rows>0) {

resp.sendRedirect(req.getContextPath()+"/index.jsp");

}else {

System.out.println(rows);

resp.sendRedirect(req.getContextPath()+"/error.jsp");

}

}

/\*\*

\* 实现首页的模糊查询

\* @param req

\* @param resp

\* @throws ServletException

\* @throws IOException

\*/

@SuppressWarnings("unused")

private void query(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

String username=req.getParameter("username");

String address=req.getParameter("address");

String phoneNo=req.getParameter("phoneNo");

phoneNo=phoneNo.replaceAll("[`~!@#$%^&\*()+|{}:;',\\[\\].<>/?~!@#￥......&\*（）&mdash;-|{}【】‘；：“”’。，、？", "");

List<User> list =userService.query(username,address,phoneNo);

req.setAttribute("userList", list);

req.getRequestDispatcher("/index.jsp").forward(req, resp);

}

@SuppressWarnings("unused")

private void delete(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

int id=Integer.parseInt(req.getParameter("id"));

int rows=userService.deleteUserById(id);

if(rows>0) {

resp.sendRedirect(req.getContextPath()+"/index.jsp");

}else {

System.out.println(rows);

resp.sendRedirect(req.getContextPath()+"/error.jsp");

}

}

@SuppressWarnings("unused")

private void update(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

}

}

package cn.mvcproject.controller;

import java.io.IOException;

import java.lang.reflect.Method;

import java.util.Date;

import java.util.List;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import com.sun.xml.internal.bind.v2.model.core.ID;

import cn.mvcproject.model.User;

import cn.mvcproject.service.FactoryService;

import cn.mvcproject.service.UserService;

import cn.mvcproject.service.UserServiceImpl;

public class UserController extends HttpServlet{

private static final long serialVersionUID = 1L;

UserService userService=FactoryService.getUserService();

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

doPost(req, resp);

}

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

req.setCharacterEncoding("UTF-8");

resp.setCharacterEncoding("UTF-8");

String mn=req.getServletPath();

mn=mn.substring(1);

mn=mn.substring(0, mn.length()-4);

System.out.println(mn);

try {

Method method=this.getClass().getDeclaredMethod(mn,HttpServletRequest.class,HttpServletResponse.class );

method.invoke(this, req,resp);

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

@SuppressWarnings("unused")

private void add(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

User user =new User();

user.setUsername(req.getParameter("username"));

user.setPassword(req.getParameter("password"));

user.setPhoneNo(req.getParameter("phoneNo"));

user.setAddress(req.getParameter("address"));

user.setRegDate(new Date());

int rows=userService.save(user);

if(rows>0) {

resp.sendRedirect(req.getContextPath()+"/index.jsp");

}else {

System.out.println(rows);

resp.sendRedirect(req.getContextPath()+"/error.jsp");

}

}

/\*\*

\* 实现首页的模糊查询

\* @param req

\* @param resp

\* @throws ServletException

\* @throws IOException

\*/

@SuppressWarnings("unused")

private void query(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

String username=req.getParameter("username");

String address=req.getParameter("address");

String phoneNo=req.getParameter("phoneNo");

// username=username.replaceAll("[`~!@#$%^&\*()+|{}:;',\\[\\].<>/?~!@#￥......&\*（）&mdash;-|{}【】‘；：“”’。，、？", "");

// address=address.replaceAll("[`~!@#$%^&\*()+|{}:;',\\[\\].<>/?~!@#￥......&\*（）&mdash;-|{}【】‘；：“”’。，、？", "");

// phoneNo=phoneNo.replaceAll("[`~!@#$%^&\*()+|{}:;',\\[\\].<>/?~!@#￥......&\*（）&mdash;-|{}【】‘；：“”’。，、？", "");

List<User> list =userService.query(username,address,phoneNo);

req.setAttribute("userList", list);

req.getRequestDispatcher("/index.jsp").forward(req, resp);

}

@SuppressWarnings("unused")

private void delete(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

int id=Integer.parseInt(req.getParameter("id"));

int rows=userService.deleteUserById(id);

if(rows>0) {

resp.sendRedirect(req.getContextPath()+"/index.jsp");

}else {

System.out.println(rows);

resp.sendRedirect(req.getContextPath()+"/error.jsp");

}

}

@SuppressWarnings("unused")

private void update(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

}

}

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

pageEncoding=*"UTF-8"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

</head>

<style>

**tr**{height:*40px*}

**td**{padding: *10px*}

</style>

<body>

<form action=*"*<%=request.getContextPath() %>*/add.udo"* method=*"post"*>

<table style="margin-left: *100px*; padding: *60px*; border:*1px #ccc solid*;">

<tr>

<td style="text-align: *right* ">用户名：</td>

<td style="text-align: *left*"> <input type=*"text"* name=*"username"*></td>

</tr>

<tr>

<td style="text-align: *right* ">用户密码：</td>

<td style="text-align: *left*"> <input type=*"text"* name=*"password"*></td>

</tr>

<tr>

<td style="text-align: *right*">用户地址：</td>

<td style="text-align: *left*"> <input type=*"text"* name=*"address"*></td>

</tr>

<tr>

<td style="text-align: *right*">用户电话：</td>

<td style="text-align: *left*"> <input type=*"text"* name=*"phoneNo"*></td>

</tr>

<tr>

<td style="text-align: *center*" colspan=*"2"*><input type=*"submit"* value=*"注册用户"*>

</td>

</tr>

</table>

</form>

</body>

</html>

# 运行效果图

# 

# 收获与问题

谈一谈你在本实验中所收获的内容及问题。