

# Java Web 员工管理系统

## 项目报告

funhuman

2021/1/17

# MVC 模式下的 web 程序项目任务书

课题名称	员工管理系统	
设计目的	设计并实现一个员工管理系统，有用户注册、登录、退出、维护信息，员工签到，经理管理员工信息、签到信息。	
实验环境	操作系统：Windows 10 家庭中文版 集成开发环境：Eclipse Jee Mars 2 数据存储：MySQL 5.5.20 服务支持：Tomcat 7.0.86	
任务要求	模块分工  A：主页和登录  B：员工信息管理  C：员工考勤管理  D：员工签到  E：信息发布	
工作进度计划		
序号	起止日期	工 作 内 容
1	2021/01/13-2021/01/14	项目分工讨论，确定各个成员分工内容，并初步实施，项目试运行。
2	2021/01/14-2021/01/15	项目初步完成。
3	2021/01/15-2021/01/16	项目互查，优化各个模块。
4	2021/01/16-2021/01/16	项目总结。

指导教师（签章）：

\_\_\_\_\_年\_\_\_\_月\_\_\_\_日

# 目 录

1	前言.....	1
2	问题背景.....	1
3	系统需求分析.....	1
3.1	功能需求分析.....	1
3.2	可行性分析.....	2
3.3	系统性能需求分析.....	3
4	总体设计报告.....	4
4.1	系统功能总体设计.....	4
4.2	系统数据库总体设计.....	4
5	详细设计报告.....	4
5.1	系统数据库设计.....	4
5.2	登录设计.....	5
5.3	权限控制设计.....	5
6	系统实现与测试.....	5
6.1	测试原理与方法.....	5
6.2	登录模块测试.....	5
6.3	注册模块测试.....	7
7	总结.....	8
7.1	项目小结.....	8
7.2	项目反思.....	9
7.3	项目收获.....	9
7.4	展望.....	9
8	参考文献.....	10
9	主要代码.....	10

# 1 前言

随着科技的进步，时代的发展，各级各类公司、企业、学校等单位的规模越来越大，要求越来越高，单纯的人工管理，以及简单的计算机管理已经不能满足公司管理的基本需求。在公司的管理中，使用一款能包含庞大的信息，能有效快速管理信息，能满足公司管理的需求的管理系统必不可少的。

员工管理系统的应用，极大程度上的解决了对于员工管理的需求，解决了人工记录和操作不便，不仅能够精准快捷地容纳更大的数据量，也能随着需求的变化不断地更新。通过 MVC 模式下 J2EE 平台上的 Java Web 技术和 MySQL 的数据管理功能，能打造出一款满足用户需求的员工管理平台。

## 2 问题背景

某软件公司计划开发一款员工管理系统，新入职的员工可以通过系统注册，也可以由管理员注册。员工在登录程序后修改自己的信息。经理登录后可对员工信息进行维护，发布和结束签到，发布相关工作信息。员工登录后可进行签到和接收工作信息。

## 3 系统需求分析

### 3.1 功能需求分析

本系统是员工管理系统，根据员工管理的特点，可以将员工管理系统划分为用户信息管理、考勤管理、工作信息管理三大功能，其中用户信息管理模块用来维护系统的正常运行。每个模块之间在表面上虽然是相互独立的，但在对数据库访问上却是紧密相连的，各模块访问的是同一个数据库，只是其所访问的表不同。系统在三大块的基础上每一大模块又分为几个模块：

1. 具体系统功能模块图如图 3-1 所示。

系统在三大块的基础上每一大模块又分为几个模块：

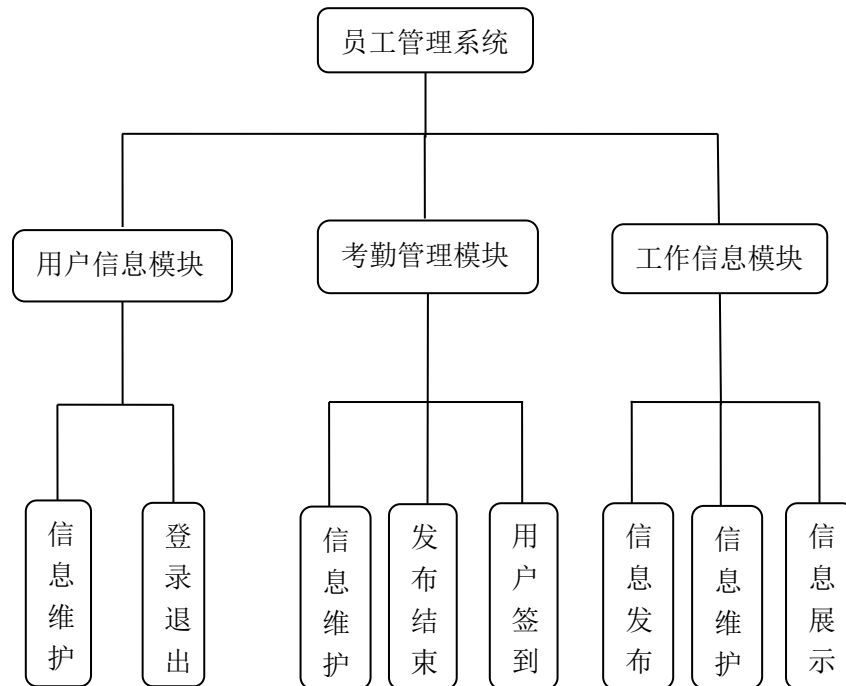


图 3-1 系统功能模块图

2. 各模块的具体功能如下：

(1) 用户模块：

信息维护：对用户信息进行查询、修改、新增和删除。

登录退出：登录程序和退出程序。

(2) 考勤管理模块：

信息维护：对考勤信息进行修改。

发布结束：对所有员工考勤信息进行修改。

员工签到：员工进行签到。

(3) 工作信息模块：

信息发布：对工作信息进行新增。

信息维护：对考勤信息进行修改和删除。

信息展示：对工作信息进行查询。

## 3.2 可行性分析

可行性分析是通过对项目的主要内容和配套条件，对项目建成以后可能取得的财务、经济效益及社会环境影响进行预测，从而提出该项目是否值得投资和如何进行建设的咨询意见，为项目决策提供依据的一种综合性的系统分析方法。可行性分析应具

有预见性、公正性、可靠性、科学性的特点。

(1) 经济可行性：软件经济可行性指的是软件所带来的经济上效益与开发其所要投资相比，是不是合适。与此同时，还需要注意软件能不能真正的为用户带来足够多的经济上效益。系统经济上的效益可从对管理员与用户两者的效益考虑。本系统成功的运行可以减少人力资源的使用，节省了开销，会带来更多的经济效益。

(2) 技术可行性：依照本网站目标衡量所需技术是不是具备，一般情况下，可从软、硬件的性能要求、环境条件、操作人员的水平和数量等方面去考虑分析。

在软件方面选择了 Eclipse 开发平台，后台使用 MySQL5 的数据库来存储数据。

在硬件方面选择 Windows 10 系统及以上，内存在 8G 以上、硬盘在 60GB 的计算机可以满足本系统的开发需要。

因此，通过以上分析，无论在经济上、性能上等方面都可以满足本网站的开发要求。

(3) 运行可行性：随着如今社会电子信息化的飞速发展，计算机应用已渗透至到我们的日常方面。计算机自身及其在生活中所发挥出的作用都意味着一种高效率、高科技、高水平技术的诞生。随着用户数量的不断增加，对这种快捷方便的员工管理系统的需求是很大的，所以运行是可行性的。

通过以上分析，可以得到：在技术、经济、运行的前提下，本系统的开发具有低投入、短周期等特点，以此，本系统的开发是完全可行的。

### 3.3 系统性能需求分析

系统具有特点如下：

(1) 时间经济性：通过优化逻辑设计与物理设计，使系统运行效率高，反应速度快。具有即时可见性，即对操作的处理将立即在页面的对应栏目显示出来，达到“及时修改、即时查询”的功能。

(2) 安全保密性：保证系统的物理安全，数据存储和存取的安全和保密，数据传输的安全和保密，做好使用人员的授权管理。

(3) 可靠性：在可靠性需求上，根据使用需要，在各项数据的输入、输出及传输过程中，可以满足各种精度的需求。对于数据精确度性能，在注册过程中，如果出现不符合数据规范化的成绩，则会提醒用户错误。

(4) 可维护性需求：要求系统具备简便易行的数据维护功能，可随时在退出系统后对数据库进行备份。

4 总体设计报告

4.1 系统功能总体设计

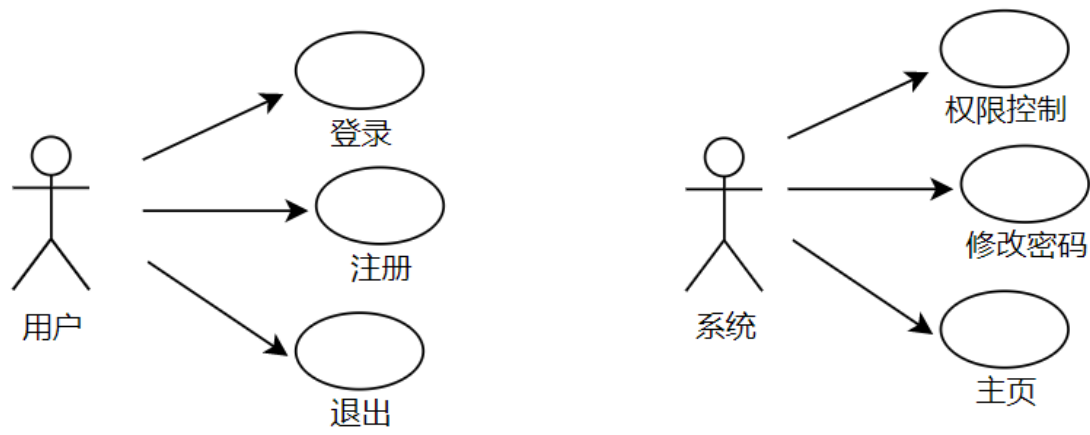


图 4-1 系统用例图

4.2 系统数据库总体设计

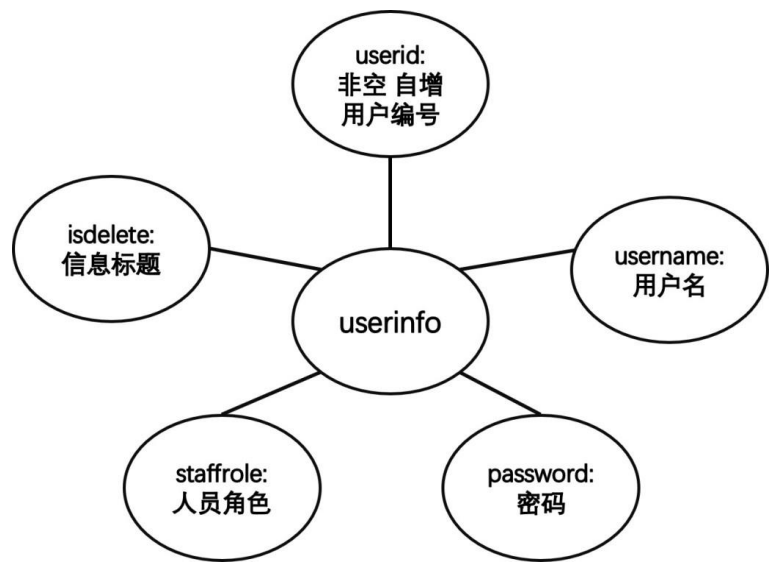


图 4-2 系统用例图

5 详细设计报告

5.1 系统数据库设计

我所负责的如表 5-1 所示，具体 SQL 代码见代码 9-1。

表 5-1 用户信息表

序号	列名	数据类型	列约束	说明
----	----	------	-----	----

1	userid	int(64)	primary key, auto_increment	用户编号
2	username	varchar(255)	not null	用户名
3	password	varchar(255)	not null	密码
4	staffrole	varchar(255)	not null	角色（经理,员工）
5	isdeleted	int(1)	not null	删除标记（0:正常,1:删除）

## 5.2 登录设计

首先是获取登录信息用户名与密码，然后分别判断是否为空并给予相应提示。登录信息都不为空，再根据用户名和密码去数据库查询，如果查询不到，说明用户名或密码有错误，给予用户相应提示。排除以上情况登录成功。成功登录后要写入 Session 暂存，并跳转到主页。登录的主要前端代码如代码 9-2 所示，主要后台代码如代码 9-4 所示。

如果用户需要注册，点击注册按钮进入注册页面。

## 5.3 权限控制设计

系统的权限控制尤其重要，本系统主要采用两个方法防止越权访问，一是“看不见”，二是“进不去”。系统设计时就考虑在前端页面对权限不足的员工用户隐藏管理员菜单，同时在后台涉及管理权限的操作都要验证用户的身份，确保万无一失。

# 6 系统实现与测试

## 6.1 测试原理与方法

测试是使用人工和自动手段来运行或检测某个系统的过程，其目的在于检验系统是否满足规定的需求，或弄清预期结果与实际结果之间的差别。测试过程的目标在于以最少的时间和人力，系统的找出软件中存在的错误和缺陷<sup>[1]</sup>。

本系统主要采用黑盒测试法。黑盒测试法以程序的功能为依据，把程序看成一个黑盒子，完全不考虑程序的内部结构和处理过程，它只检查程序功能是否能按照规格说明书的规定正常使用，程序是否能适当地接收输入数据，产生正确地输出信息。

## 6.2 登录模块测试

登录模块要实现用户的登录功能，如图 6-1 所示，登录测试如图 6-2 所示。

表 6-1 系统登录模块测试表

操作步骤	执行操作	测试数据	期望结果	实际结果	测试状态
------	------	------	------	------	------



1	输入正确操作员信息	用户名=admin 密码=Ad. 123	成功进入主页 用户名为 Admin	一致	正常
2	输入正确员工信息	用户名=12 密码=12	成功进入主页 用户名为 12	一致	正常
3	不输入信息	用户名= 密码=	提示输入不能为空	一致	正常
4	输入错误信息	用户名=ad 密码=123	提示登录信息错误错误	一致	正常

登录页如登录成功后主页如图 6-3 所示。

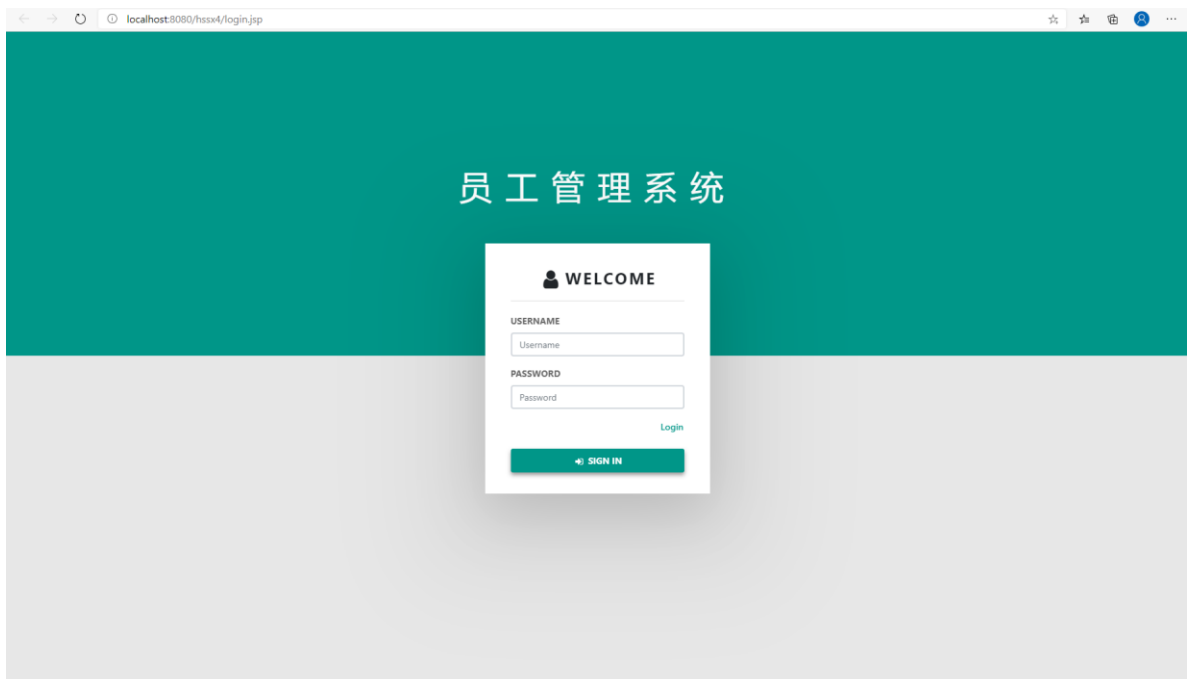


图 6-1 登录图

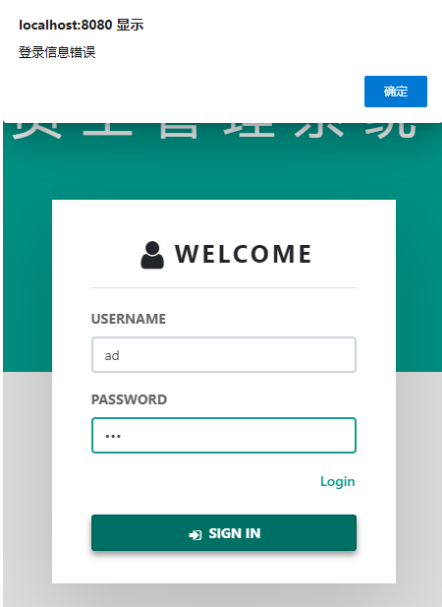


图 6-2 登录测试图

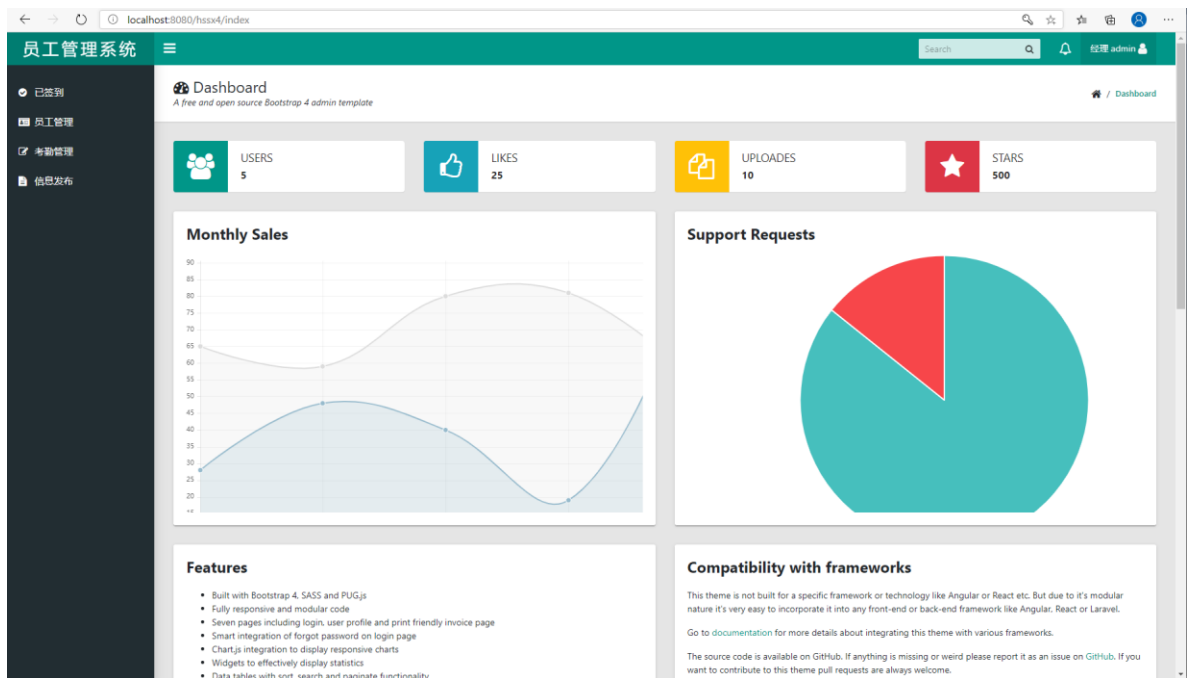


图 6-3 主页图

## 6.3 注册模块测试

注册一名新的用户。

表 6-2 系统注册模块测试表

操作步骤	执行操作	测试数据	期望结果	实际结果	测试状态
1	不输入信息	用户名= 密码= 姓名= 手机号= 性别= 年龄= 用户名=admin 密码=666	提示输入不能 为空	一致	正常
2	输入错误信息	姓名=管理员 手机号=12344556677 性别=男 年龄=20 用户名=leek 密码=lee123	提示用户名不 能重复	一致	正常
3	输入正确信息	姓名=力克 手机号=17851939999 性别=男 年龄=20	跳转到登录页 面	一致	正常



图 6-4 注册页图



图 6-5 注册页测试图

## 7 总结

### 7.1 项目小结

又经历了为期六天的线上实训,我在老师和同学们的帮助下了解了 Java Web 的基本知识,学习了 Eclipse J2EE 搭配 MySQL 项目的基本结构,掌握了 Java 语言和 SQL 语言的基本语法。在搜索引擎的帮助下,可以完成一定程度的 Java Web 项目。

在本次的线上实训中,一边被错误的运行结果和不知所云的错误提示折磨着,一边利用搜索引擎尝试解决问题,在这样不断循环的过程中,有时痛苦,有时想放弃,有时又忍不住打开 Eclipse 再改一个 bug。在总结的时候回过头了,感觉更多的是快乐,同学们互相询问,帮助解决问题,项目越来越好,错误越来越少,友谊越来越深。

通过团队合作,在实践中体会到“众人拾柴火焰高,集中力量干大事”的团结合作精神。在团队分工、讨论各个环节中,小组全体成员都认真参与,积极作为。在整

个程序的需求、设计、实现、测试各个过程中互相帮助、集体实现了本项目员工管理系统。

## 7.2 项目反思

本系统功能和设计方法上还有很多可以改进的地方，比如前后端对话使用基于 JavaScript 的 ajax 功能，代码对用户可见，容易遭到恶意访问甚至攻击。设计方法还停留在原始的想到什么写什么阶段，对软件开发没有清晰的认识，客观上容易产生一些不易排查的逻辑错误。开发过程中一味追求实现，对代码不理解，也导致程序容易出一些细小的拼写错误、空指针错误、异常处理错误等等。设计上代码注释少、不规范，客观上造成软件维护困难。在以后的软件开发过程中要加以改进，尽力避免。

在学习的过程中既有收获，也暴露出个人学习上的一些问题。在平时课程学习中，只顾着记录老师的代码，没有思考实现的方法，以至于在练习的时候常常犯一些像是代码拼写的低级错误。对课堂上学习到的一些新知识，一些遇到的错误，只简单的了解，没有记录下来。在遇到问题百思不得其解的时候，才想起来老师课堂上好像讲过，浪费时间。

在项目的设计实现过程中，互联网是我最重要的资源。设计程序过程中遇到的问题，都能通过搜索其他软件开发者的作品和结论发现问题、解决问题、总结问题的过程和经验。其次就是老师上课所教授的内容在我的开发过程中起到了引导我完成作品的重要作用，以及在我遇到瓶颈的过程中，同学和小组成员给了我很多帮助，在此非常感谢老师和同学。

## 7.3 项目收获

在项目中，我最大的收获就是两个文本文档，一个是 MVC 笔记，一个是 Java 和 JavaScript 笔记，笔记中记录了一些问题和错误，还有对应的解决方法和网络上的参考资料，以后开发的时候就可以避免这些错误节约时间。

另外一个收获就是，在上学期学过软件工程的基础上，通过小组合作的形式把软件工程的知识在项目团队开发实践中运用起来，更进一步的学习了软件过程知识。

还有一个收获就是这个项目了，做完一个项目并完成测试，暂时看不出有问题，自然是自豪的。在项目圆满完成的时候，回顾这几天的开发过程，只有克服困难的开心，还有解决问题的过程，给我增加了一点经验。

## 7.4 展望

在本次实训的锻炼下，我收获了很多知识，学到了很多，也暴露出了许多问题，这样的经验为将来的毕业设计打下了良好的基础，也为未来的学习提供了进步的

空间。在未来的设计实验中，我会更加注重对课堂上和书上内容的总结和分析，加强对程序逻辑和代码的理解；更多地吸收来自于互联网的资料和其他人设计的思路，让自己的能力更加完善，达到更好地程序开发水平。

## 8 参考文献

- [1] 李宗花, 朱林. 软件工程原理与实践[M]. 南京: 南京大学出版社, 2020. 8
- [2] 王珊, 萨师煊. 数据库系统概论: 第 5 版[M]. 北京: 高等教育出版社, 2016. 2

## 9 主要代码

代码 9-1 数据库 SQL 代码

```
CREATE TABLE `userinfo` (  
  `userid` int(64) unsigned NOT NULL AUTO_INCREMENT COMMENT '用户编号',  
  `username` varchar(255) DEFAULT NULL COMMENT '用户名',  
  `password` varchar(255) DEFAULT NULL COMMENT '密码',  
  `staffrole` varchar(255) DEFAULT NULL COMMENT '角色 (经理,员工) ',  
  `isdeleted` int(1) unsigned DEFAULT NULL COMMENT '删除标记 (0:正常,1:删除) ',  
  PRIMARY KEY (`userid`)  
) ENGINE=InnoDB AUTO_INCREMENT=106 DEFAULT CHARSET=utf8;  
  
insert into `userinfo`(`userid`,`username`,`password`,`staffrole`,`isdeleted`) values (1  
, 'admin', 'Ad.123', '经理', 0), (10, 'ad', '1', '员工', 0), (12, '12', '12', '员工', 0);
```

代码 9-2 前端公用判段函数代码

```
// 判空函数  
// https://blog.csdn.net/qq\_36150631/article/details/94553703  
function isEmpty(v) {  
  switch (typeof v) {  
    case 'undefined':  
      return true;  
    case 'string':  
      if (v.replace(/^[ \t\n\r]*|([ \t\n\r]*$)/g, '').length == 0) return true;  
      break;  
    case 'boolean':  
      if (!v) return true;  
      break;  
    case 'number':  
      if (0 === v || isNaN(v)) return true;  
      break;  
    case 'object':  
      if (null === v || v.length === 0) return true;  
      for (var i in v) {  
        return false;  
      }  
      return true;  
    }  
  }  
  return false;  
}  
  
// 数字和小写字母判断  
https://blog.csdn.net/LY9457/article/details/88857457  
function isnoabcnum(str) {  
  var pattern = new RegExp("[a-z0-9]+");  
  if(pattern.test(str)){  
    return false;  
  } else {  

```

```

        return true;
    }
}

```

代码 9-3 登录前端代码

```

<script type="text/javascript">
    function checkInfo() {
        // 获取对应元素的值
        var username=$("#username").val();
        var password=$("#password").val();
        // 简单判断
        if (isEmpty(username) || isEmpty(password)) {
            alert("输入不能为空");
            return;
        }
        // 调用 ajax 判断是否登录成功
        $.ajax({
            type: "post", // 方法类型
            url: "loginCheck", // url
            dataType: "text", // 数据类型    string json xml
            async: true, // 是否异步
            data: {"username": username, "password": password},
            // 回调方法
            success: function(data) {
                if (data=="ok") {
                    // alert("登录成功");
                    window.location.href="index"; // 跳转至主页
                } else {
                    alert("登录信息错误");
                }
            },
            error: function() {
                alert("异常请求! ");
            }
        });
    }
    function isEmpty(v) {...} // 判空函数
</script>

```

代码 9-4 登录后台代码

```

protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    // 获取前端信息
    String username = request.getParameter("username");
    String password = request.getParameter("password");
    // 到数据库查询
    UserService userService = new UserService();
    UserInfo user = userService.selectUser(username, password);
    // 判断查询是否为空, 为空登录失败
    if (user == null || user.getId() == null) {
        response.getWriter().write("no");
        return;
    }
    // 登录成功, 记录 Session
    HttpSession session = request.getSession();
    session.setAttribute("userInfo", user);
    response.getWriter().write("ok");
}

```

代码 9-5 登录数据库操作代码

```

public UserInfo selectUser(String username, String password) {

```

```

String sql = "select userid as id, staffrole as role, checkstatus from userinfo u, checkinfo c where isdeleted=0 and u.userid=c.staffid and username=? and password=?";
UserInfo user = new UserInfo();
try {
    user = qr.query(sql, new BeanHandler<>(UserInfo.class), username, password);
    if (user != null) {
        user.setUsername(username);
        user.setPassword(password);
    }
} catch (SQLException e) {
    e.printStackTrace();
    System.err.println("\n\n 错误: 数据库连接超时 (UserService.java:32)\n\n"); // 调试语句)
}
return user;
}

```

代码 9-6 修改密码前端代码

```

function updatePassword() {
    if ($("#oldpassword").val() != $("#oldPwd").val()) {
        alert("原密码错误! ");
    } else {
        $.ajax({
            type: "post", // 方法类型
            url: "changePassword.user", // url
            cache: false,
            data: {
                "id": $("#id").val(), // 增加没有 id
                "newpassword": $("#newpassword").val(),
            },
            dataType: "text", // 数据; 类型    string json xml
            // 回调方法
            success: function(data) {
                if (data=="no") {
                    alert("修改失败");
                } else {
                    alert("修改成功");
                    window.location.href='index';
                    //location.reload(); // 刷新
                }
            },
            error: function() {
                alert("请求失败! ");
            }
        });
    }
}
}

```

代码 9-7 修改密码后台代码

```

public void changePassword(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    // 获取到前端的信息
    String id = request.getParameter("id");
    String newpassword = request.getParameter("newpassword");
    // 调用 SQL 语句
    UserService stService = new UserService();
    int i = stService.changePassword(Integer.parseInt(id), newpassword);
    String rtn = (i > 0 ? "ok" : "no");
    response.getWriter().write(rtn);
}

```

代码 9-8 修改密码数据库操作代码

```
public int changePassword(int id, String newpassword) {
    String sql = "update userinfo set password=? where userid=?";
    int i = 0;
    try {
        i = qr.execute(sql, newpassword, id);
    } catch (SQLException e) {
        e.printStackTrace();
    }
    return i;
}
```

代码 9-5 注册前端代码

```
function addStaff() {
    // 获取对应元素的值
    // var id=$("#id").val() // 增加没有 id
    var username=$("#username").val().trim()
    var password=$("#password").val().trim() // 修改没有 password
    var name=$("#name").val().trim()
    var phone=$("#phone").val().trim()
    var age=$("#age").val().trim()
    var sex=$("#sex").val().trim()
    // var role=$("#role").val() // 增加没有 role
    // 输入判断
    if (isEmpty(username)||isEmpty(password)
        ||isEmpty(name)||isEmpty(phone)
        ||isEmpty(age)||isEmpty(sex)) {
        alert("输入不能为空");
        return;
    }
    if (isnoabnum(username)) {
        alert("请输入小写字母和数字的用户名");
        return;
    }
    if (!(sex == "男" || sex == "女")) {
        alert("请输入真实的性别 (特殊性别请联系管理员) ");
        return;
    }
    if (isNaN(age) || parseInt(age) < 0 || parseInt(age) > 127) {
        alert("请输入真实的年龄");
        return;
    }
    if (isNaN(phone) || phone.length < 11) {
        alert("请输入真实的 11 位手机号 (特殊手机号请联系管理员) ");
        return;
    }
    // 调用 ajax 判断是否登录成功
    $.ajax({
        type: "post",
        url: "insertStaff.st",
        dataType: "text",
        async: true,
        data: {
            "username": username,
            "password": password,
            "name": name,
            "phone": phone,
            "age": age,
            "sex": sex,
        },
        // 回调方法
        success: function(data) {
```



```

        if (data=="re") {
            alert("用户名不能重复");
        } else if (data=="ok") {
            if (isEmpty($("#userid").val())) {
                alert("注册成功");
                parent.quit();
            }
        } else {
            alert("注册失败");
        }
    },
    error: function() {
        alert("请求失败! ");
    }
});
}
}

```

代码 9-9 注册后台代码

```

public void insertStaff(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
    // 获取到前端的信息
    String username = request.getParameter("username");
    UserService userService = new UserService();
    if (userService.selectUserByName(username)) {
        //System.out.println("StaffServlet(96): " + "return \"re\""); // 调试语句
        response.getWriter().write("re");
        return;
    }
    String password = request.getParameter("password");
    String name = request.getParameter("name");
    String phone = request.getParameter("phone");
    String age = request.getParameter("age");
    String sex = request.getParameter("sex");
    // 存为 StaffInfo 类型
    StaffInfo st = new StaffInfo();
    st.setUsername(username);
    st.setPassword(password);
    st.setName(name);
    st.setPhone(phone);
    st.setAge(Integer.parseInt(age));
    st.setSex(sex);
    StaffService stService = new StaffService();
    int i = stService.insertStaffBySt(st);
    String rtn = (i > 0 ? "ok" : "no");
    response.getWriter().write(rtn);
}

```

代码 9-10 注册数据库代码

```

public int insertStaffBySt(StaffInfo st) {
    String sql = "insert into userinfo set username=?, password=?, staffrole='员工',
isdeleted=0;";
    String sql2 = "insert into staffinfo set staffid=(select userid from userinfo where
username=?), staffname=?, staffphone=?, staffsex=?, staffage=?;";
    String sql3 = "insert into checkinfo set staffid=(select userid from userinfo where
username=?), checkstatus='未签到';";
    int i = 0;
    try {
        i += qr.execute(sql, st.getUsername(), st.getPassword());
        i += qr.execute(sql3, st.getUsername());
        i += qr.execute(sql2, st.getUsername(), st.getName(), st.getPhone(), st.getSex(),
st.getAge());
    } catch (SQLException e) {
    }
}

```

```
        e.printStackTrace();  
    }  
    return (i == 3 ? 1 : 0);  
}
```