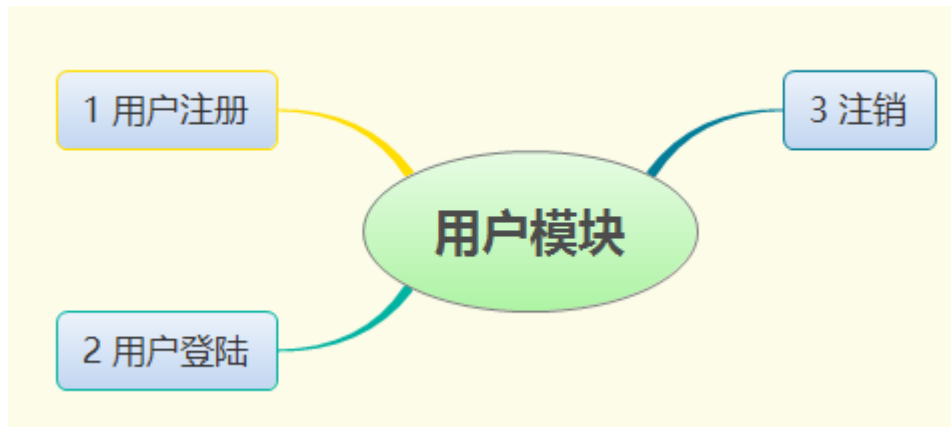


# 用户模块

## 一 学习目标



## 二 注册

### 1 需求分析

用户名，必填项

若未输入值，则失去焦点时，提示用户，用户名不能为空；

若已输入值，则失去焦点时，会验证用户名是否已存在；如果已存在，则提示用户；

用户登录密码，必填项；

用户输入的确认密码，需要与登录密码保持一致，必填项；

单选按钮，默认选择男；

日期框，可点击选择日期，非必填项；

用户的身份证号，非必填项；

用户的邮箱，非必填项；

用户的手机号，必填项；

用户的收货地址，必填项；

点击提交注册按钮

1) 判断必填字段不能为空（带"\*"都是必填字段）

2) 判断登录密码是否和确认密码相等

3) 验证登录，则提示注册成功，跳转到首页

## 2 页面原型

易买网  
yimai.com

购物车(0件商品) a 注销 注册 留言

首页

服饰

数码

美食

家电

音乐 | 影视 | 少儿 | 运动鞋 | 美容护肤 | 家纺用品 | 婴幼儿奶粉 | 饰品 | 测试标签 | 测试2

搜索

欢迎注册易买网

填写注册信息

注册成功

用户名(\*):

a

登录密码(\*):

•

确认密码(\*):

性别(\*):

☒男 ☐女

出生日期:

身份证号:

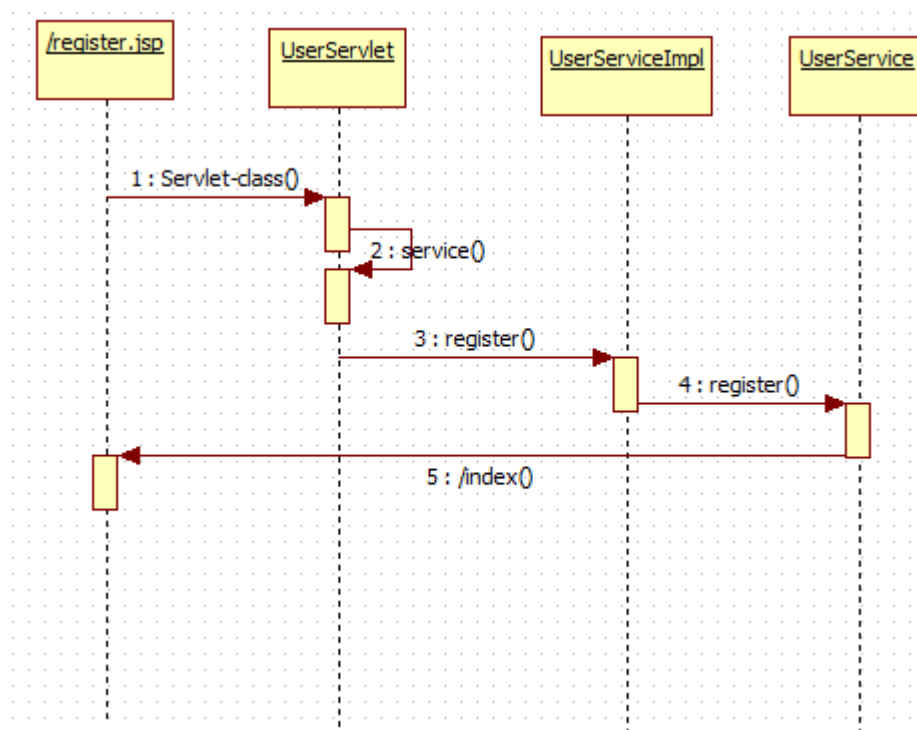
Email:

手机号码(\*):

收获地址(\*):

提交注册

## 3 流程图



## 4 数据库信息

(t\_user)

	列名	数据类型（精度范围）	允许空	约束条件
主键列	id	int	否	主键，自动增长
收货地址	address	varchar(200)	是	
出生年月	birthday	datetime	是	
身份证号	dentityCode	varchar(20)	是	
用户邮箱	email	varchar(20)	是	
用户手机	mobile	varchar(20)	是	
用户密码	password	varchar(20)	是	
用户性别	sex	varchar(5)	是	
用户状态	status	varchar(5)	否	1=普通用户 2=管理员账户
真实名称	trueName	varchar(20)	是	
用户名称	userName	varchar(20)	是	

## 5 编写Servlet

```

@WebServlet("/userServlet")
public class UserServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private UserService userService = new UserServiceImpl();

    protected void service(HttpServletRequest req, HttpServletResponse resp) throws
ServletException, IOException {
        //获取请求的oper
        String oper = req.getParameter("oper");
        if(!"".equals(oper)&&oper!=null&&"register".equals(oper)){
            try {
                register(req,resp);
            } catch (Exception e) {

```

```

        e.printStackTrace();
    }
} else if(!"".equals(oper)&&oper!=null&&"checkName".equals(oper)) {
    checkName(req, resp);
}

}
//注册
private void register(HttpServletRequest req, HttpServletResponse resp) throws Exception
{
    //获取请求参数
    String userName = req.getParameter("userName");
    String password = req.getParameter("password");
    String sex = req.getParameter("sex");
    Date birthday = DateUtil.formatString(req.getParameter("birthday"), "yyyy-MM-dd");
    String dentityCode = req.getParameter("dentityCode");
    String email = req.getParameter("email");
    String mobile = req.getParameter("mobile");
    String address = req.getParameter("address");
    //构建对象
    User user = new User(userName, password, sex, birthday, dentityCode, email, mobile,
address);
    //添加
    userService.register(user);
    //跳转
    req.getRequestDispatcher("/reg-result.jsp").forward(req, resp);
}

}
//检查用户名是否唯一
private void checkName(HttpServletRequest req, HttpServletResponse resp) throws
IOException {
    String uname = req.getParameter("userName");
    boolean flag=userService.checkName(uname);
    PrintWriter out = resp.getWriter();
    out.print(flag);
}
}

```

## 6 编写Service

```

public interface UserService {
    void register(User user); //注册
    boolean checkName(String uName); //检查用户名是否唯一
}

```

## 7 编写ServiceImpl

```

public class UserServiceImpl implements UserService{
    //注册
    @Override
    public void register(User user) {
        //声明连接
        Connection conn = null;
        try{
            //获取连接通道
            conn = DBUtil.getConn();
            String sql = "INSERT INTO T_USER (ADDRESS,BIRTHDAY,DENTITYCODE,"
                + "EMAIL,MOBILE,PASSWORD,SEX,STATUS,USERNAME) VALUES ("
                + "?, ?, ?, ?, ?, ?, ?, ?, ?)";

```

```

        //设置绑定变量
        Object[] pars=new Object[]{
            user.getAddress(),user.getBirthday(),
            user.getIdentityCode(),user.getEmail(),
            user.getMobile(),user.getPassword(),
            user.getSex(),user.getStatus(),
            user.getUserName()
        };
        QueryRunner qr=new QueryRunner();
        qr.update(conn, sql, pars);
    }catch(Exception e){
        System.out.println("register ERROR");
        e.printStackTrace();
    }finally{
        DBUtil.closeConn(conn);
    }
}
//检查用户名是否唯一
@Override
public boolean checkName(String uName) {
    boolean flag = false;
    //声明连接
    Connection conn = null;
    try {
        //获取
        conn = DBUtil.getConn();
        String sql = "SELECT * FROM T_USER WHERE USERNAME = ?";

        //设置绑定变量
        Object[] pars = new Object[] {uName};

        QueryRunner qr = new QueryRunner();

        User ls = (User) qr.query(conn, sql, new BeanHandler(User.class), pars);//查询一个
        结果 new BeanHandler(实体类.class)

        if(ls==null){
            flag = true;
        }
    }catch(SQLException e) {
        System.out.println("checkName error");
        e.printStackTrace();
    }finally {
        DBUtil.closeConn(conn);
    }
    return flag;
}
}

```

## 8 对应实体类

```

public class User {
    private int id;//id
    private String trueName;//真实名称
    private String userName;//昵称
    private String password;//密码
    private String sex;//性别
    private Date birthday;//出生日志
    private String dentityCode;//身份证号
}

```

```

private String email;//邮件
private String mobile;//手机
private String address;//地址
private int status=1;//状态 1普通用户 2管理员账户
public User() {
}
public User(String userName, String password, String sex, Date birthday, String
dentityCode, String email,
        String mobile, String address) {
    super();
    this.userName = userName;
    this.password = password;
    this.sex = sex;
    this.birthday = birthday;
    this.dentityCode = dentityCode;
    this.email = email;
    this.mobile = mobile;
    this.address = address;
}
public User(String uname, String pwd) {
    this.userName = uname;
    this.password = pwd;
}
public int getId() {
    return id;
}
public void setId(int id) {
    this.id = id;
}
public String getTrueName() {
    return trueName;
}
public void setTrueName(String trueName) {
    this.trueName = trueName;
}
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 getSex() {
    return sex;
}
public void setSex(String sex) {
    this.sex = sex;
}
public Date getBirthday() {
    return birthday;
}
public void setBirthday(Date birthday) {
    this.birthday = birthday;
}
public String getDentityCode() {

```

```

        return dentityCode;
    }
    public void setDentityCode(String dentityCode) {
        this.dentityCode = dentityCode;
    }
    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }
    public String getMobile() {
        return mobile;
    }
    public void setMobile(String mobile) {
        this.mobile = mobile;
    }
    public String getAddress() {
        return address;
    }
    public void setAddress(String address) {
        this.address = address;
    }
    public int getStatus() {
        return status;
    }
    public void setStatus(int status) {
        this.status = status;
    }
}

```

## 9 前台

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Insert title here</title>
<script type="text/javascript">

function checkForm(){
    var userName=$("#userName").val();
    var password=$("#password").val();
    var rePassword=$("#rePassWord").val();
    var mobile=$("#mobile").val();
    var address=$("#address").val();

    if(userName==""){
        $("#error").html("用户名不能为空! ");
        return false;
    }
    if(password==""){
        $("#error").html("密码不能为空! ");
        return false;
    }
    if(rePassword==""){

```

```

        $("#error").html("确认密码不能为空! ");
        return false;
    }
    if(password!=rePassWord){
        $("#error").html("确认密码和密码不一致, 请重新输入! ");
        return false;
    }
    if(mobile==""){
        $("#error").html("手机号码不能为空! ");
        return false;
    }
    if(address==""){
        $("#error").html("收货地址不能为空! ");
        return false;
    }
    return true;
}

function checkUserName(){
    var userName=$("#userName").val();
    if(userName==""){
        $("#userErrorInfo").html("用户名不能为空! ");
        $("#userName").focus();
        return;
    }
    $.post("userService?oper=checkName",{userName:userName},function(data){
        /* var result=eval('(' +result+')'); */
        if(data=='true'){
            $("#userErrorInfo").html("");
        }else{
            $("#userErrorInfo").html("用户名已存在, 请重新输入! ");
            $("#userName").focus();
        }
    });
}

</script>
</head>
<body>
<div id="header" class="wrap">
    <jsp:include page="common/top.jsp"/>
</div>

<div id="register" class="wrap">
    <div class="shadow">
        <em class="corner lb"></em>
        <em class="corner rt"></em>
        <div class="box">
            <h1>欢迎注册易买网</h1>
            <ul class="steps clearfix">
                <li class="current"><em></em>填写注册信息</li>
                <li class="last"><em></em>注册成功</li>
            </ul>
            <form id="regForm" method="post" action="userService?oper=register"
onsubmit="return checkForm()">
                <table>
                    <tr>
                        <td class="field">用户名(*): </td>
                        <td><input class="text" type="text" id="userName" name="userName"
onblur="checkUserName(this.value)" />&nbsp;  <font id="userErrorInfo" color="red"></font></td>

```



```

        </tr>
        <tr>
            <td class="field">登录密码(*): </td>
            <td><input class="text" type="password" id="password"
name="password" /></td>
        </tr>
        <tr>
            <td class="field">确认密码(*): </td>
            <td><input class="text" type="password" id="rePassword"
name="rePassword" /></td>
        </tr>

        <tr>
            <td class="field">性别(*): </td>
            <td>
                <input type="radio" name="sex" value="男" checked="checked"/>男
                <input type="radio" name="sex" value="女"/>女
            </td>
        </tr>
        <tr>
            <td class="field">出生日期: </td>
            <td>
                <input type="text" id="birthday" name="birthday"
class="wdate" onClick="wdatePicker()"/>
            </td>
        </tr>
        <tr>
            <td class="field">身份证号: </td>
            <td><input class="text" type="text" id="identityCode"
name="identityCode" /></td>
        </tr>
        <tr>
            <td class="field">Email: </td>
            <td><input class="text" type="text" id="email" name="email" /></td>
        </tr>
        <tr>
            <td class="field">手机号码(*): </td>
            <td><input class="text" type="text" id="mobile" name="mobile" /></td>
        </tr>
        <tr>
            <td class="field">收获地址(*): </td>
            <td><input class="text" type="text" id="address" name="address" />
        </td>
        </tr>
        <tr>
            <td></td>
            <td><label class="ui-green"><input type="submit" name="submit"
value="提交注册" /></label></td>
        </tr>
        <tr>
            <td>&nbsp;</td>
            <td><font id="error" color="red"></font> </td>
        </tr>
    </table>
</form>
</div>
<div class="clear"></div>

```

```
</div>

<div id="footer">
  <jsp:include page="common/footer.jsp"/>
</div>
</body>
</html>
```

## 三 登陆

### 1 需求分析

用户点击顶部登录按钮跳转到登录界面，同时输入用户名、密码、验证码进行登录。

### 2 页面原型

## 欢迎回到易买网

用户名:

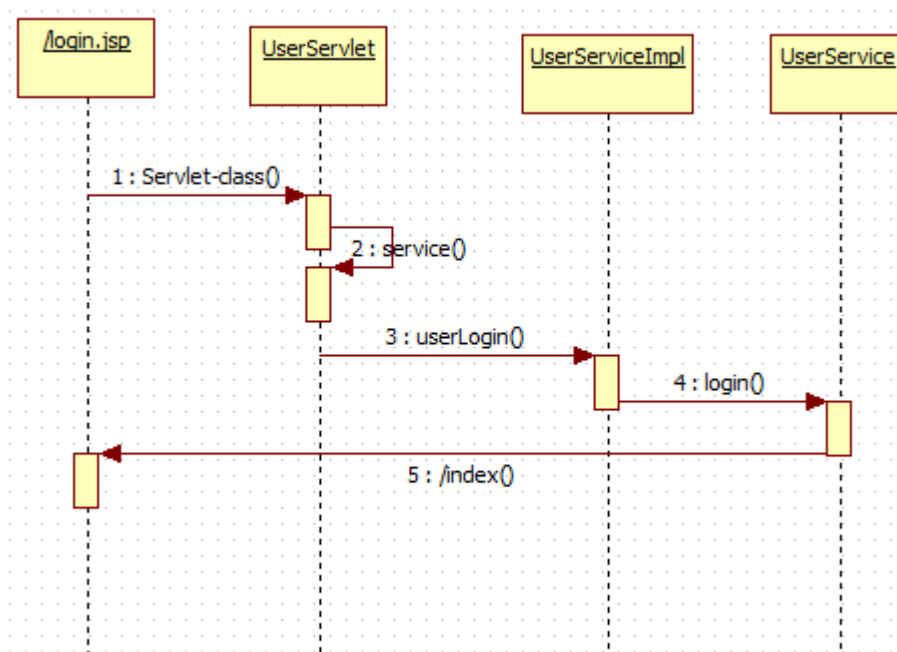
登录密码:

验证码:

0609

立即登录

### 3 流程图



## 4 数据库信息

(t\_user)

	列名	数据类型（精度范围）	允许空	约束条件
主键列	id	int	否	主键，自动增长
收货地址	address	varchar(200)	是	
出生年月	birthday	datetime	是	
身份证号	identityCode	varchar(20)	是	
用户邮箱	email	varchar(20)	是	
用户手机	mobile	varchar(20)	是	
用户密码	password	varchar(20)	是	
用户性别	sex	varchar(5)	是	
用户状态	status	varchar(5)	否	1=普通用户 2=管理员账户
真实名称	trueName	varchar(20)	是	
用户名称	userName	varchar(20)	是	

## 5 编写Servlet

```

@WebServlet("/userServlet")
public class UserServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private String error = "";
    private UserService userService = new UserServiceImpl();

    protected void service(HttpServletRequest req, HttpServletResponse resp) throws
ServletException, IOException {
        //获取请求的oper
        String oper = req.getParameter("oper");
        //请求分发
        if(!"".equals(oper)&&oper!=null&&"register".equals(oper)){
            try {
                register(req, resp);
            } catch (Exception e) {

```

```

        e.printStackTrace();
    }
} else if(!"".equals(oper)&&oper!=null&&"login".equals(oper)) {
    userLogin(req, resp);
}
}

//登陆
private void userLogin(HttpServletRequest req, HttpServletResponse resp) throws
ServletException, IOException {
    //获取请求参数 准备登陆
    String uname = req.getParameter("userName");
    String pwd = req.getParameter("password");
    String imageCodeInput = req.getParameter("imageCode");
    //获取Session对象 获取存储的验证码
    String imageCode = (String) req.getSession().getAttribute("sRand");
    User user = new User(uname, pwd);
    if(imageCodeInput.equals(imageCode)){
        //判定用户是否注册
        User currentUser= userService.login(user);
        if(currentUser!=null) {
            //存储作用域
            req.getSession().setAttribute("currentUser", currentUser);
            //跳转页面
            req.getRequestDispatcher("/index").forward(req, resp);
            return;
        } else {
            error = "用户名或者密码错误";
        }
    } else {
        //跳转页面
        error = "验证码错误";
    }
    req.setAttribute("error", error);
    req.getRequestDispatcher("/login.jsp").forward(req, resp);
}
}

```

## 6 编写Service

```

public interface UserService {
    void register(User user);
    User login(User user);
}

```

## 7 编写ServiceImpl

```

//登陆
@Override
public User login(User user) {
    User u = null;
    //声明连接
    Connection conn = null;
    try {
        //获取
        conn = DBUtil.getConn();
        String sql = "SELECT * FROM T_USER WHERE USERNAME = ? AND PASSWORD = ?";

        //设置绑定变量

```

```

        Object[] pars = new Object[] {user.getUserName(),user.getPassword()};

        QueryRunner qr = new QueryRunner();

        u = (User) qr.query(conn, sql, new BeanHandler(User.class), pars);

    }catch(SQLException e) {
        e.printStackTrace();
    }finally {
        DBUtil.closeConn(conn);
    }
    return u;
}

```

## 8 对应实体类

注册的时候已经写好了User实体类

## 9 前台

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Insert title here</title>
<link type="text/css" rel="stylesheet" href="css/style.css" />
<script type="text/javascript" src="${pageContext.request.contextPath}/js/jquery-1.11.1.js">
</script>
<script type="text/javascript">
    function loadImage(){
        document.getElementById("randImage").src = "image.jsp?" + Math.random();
    }

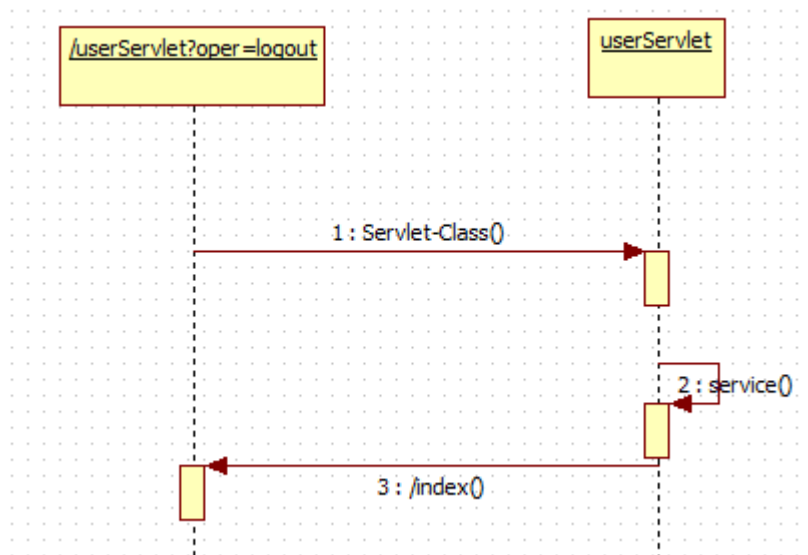
    function checkForm(){
        var userName=$("#userName").val();
        var password=$("#password").val();
        var imageCode=$("#imageCode").val();
        if(userName==""){
            $("#error").html("用户名不能为空! ");
            return false;
        }
        if(password==""){
            $("#error").html("密码不能为空! ");
            return false;
        }
        if(imageCode==""){
            $("#error").html("验证码不能为空! ");
            return false;
        }
        return true;
    }
</script>

</head>
<body>
<div id="header" class="wrap">

```



### 3 流程图



### 4 数据库信息

无

### 5 编写Servlet

```
if(!"".equals(oper)&&oper!=null&&"logout".equals(oper)) {
    logout(req, resp);
}
private void logout(HttpServletRequest req, HttpServletResponse resp) throws IOException {
    //清空session
    req.getSession().invalidate();
    //跳转页面
    resp.sendRedirect(req.getContextPath()+"/index.jsp");
}
```

### 6 编写Service

无

### 7 编写ServiceImpl

无

### 8 对应实体类

无

## 9 前台

```
function logout() {  
    if (confirm('您确定要退出系统吗? ')) {  
        window.location.href = "userServlet?oper=logout";  
    }  
}
```

## 五 总结

