

## 1. EasyMall项目

### a. 环境搭建

- i. 创建一个web应用--EasyMall。
- ii. 配置www.easymall.com的虚拟主机，并配置为缺省主机。将EasyMall部署到www.easymall.com的虚拟主机管理的目录中，并配置成缺省web应用，配置主页，实现在地址栏输入www.easymall.com直接访问主页。
  - 1) 配置虚拟主机，并设置为缺省虚拟主机：
    - a) 在[tomcat]/conf/server.xml中的<Engine>标签中配置defaultHost="www.easymall.com"
    - b) 在<Engine>标签下，书写一对<Host> </Host>标签，内容如下：

```
<Host name="www.easymall.com" appBase="E:\www.easymall.com">
```

**注意：**书写完成之后启动tomcat则会自动创建此目录，这个目录作为虚拟主机使用。
  - 2) 修改hosts文件，配置ip地址映射和主机名。

在其中添加：127.0.0.1 [www.easymall.com](http://www.easymall.com)
  - 3) 将EasyMall部署到虚拟主机中，并配置为缺省web应用，配置主页：
    - 1) 在Myeclipse中发布EasyMall到www.easymall.com的虚拟主机目录下，需要使用自定义发布的方式来发布：
      - i) 点击add添加web应用。
      - ii) 在项目发布的下拉框中选择：<Custom Location>，并修改下方路径为：  
E:\www.easymall.com\ROOT

**注意：**此处将EasyMall工程配置为缺省web应用。
    - 2) 配置主页，EasyMall中的WEB-INF/web.xml中配置

### b. 导入页面

在EasyMall/WebRoot中添加课前资料中的页面、样式等资源。

- i. 导入主页---index.jsp

主页中包含三部分：头、身体、脚，分别对应三个html页面，可从课前资料中引入。引入后将.html文件修改为.jsp文件：

  - 1) 引入index.html文件
  - 2) 删除index.jsp中第一行外所有内容，复制index.html中所有内容，粘贴至index.jsp中
  - 3) 删除index.html。
- ii. 修改\_head.jsp、\_foot.jsp
- iii. 在index.jsp中引入\_head.jsp和\_foot.jsp

利用包含技术将头页面和脚页面引入：

通过<%@include file="\_head.jsp"%>引入头页面。

通过<%@include file="\_foot.jsp"%>引入脚页面。
- iv. 引入regist.jsp页面
  - 1) 直接将regist.jsp页面相关的内容引入到EasyMall中即可。
- v. 修改\_head.jsp中的注册url地址

```
<a href="<%=request.getContextPath()%>/regist.jsp" />
```

### c. 实现注册功能

- i. 修改jsp页面中form表单action的url地址为：<%=request.getContextPath()%>/RegistServlet
- ii. 创建RegistServlet.java文件，在其中添加如下内容：

//1.解决乱码问题 **RegistServlet.java中编写代码:**

```
request.setCharacterEncoding("utf-8");
response.setContentType("text/html;charset=utf-8");
```

//2.获取请求参数 **RegistServlet.java中编写代码:**

```
String username = request.getParameter("username");
String password = request.getParameter("password");
```

...(依次获取其他输入框信息)

//3.校验数据 **RegistServlet.java中编写代码:**

```
if(WebUtils.isNull(username)){//校验失败
    //设置提示信息
    request.setAttribute("msg", "用户名不能为空");
    //转发回注册页面进行提示
    request.getRequestDispatcher("/regist.jsp").forward(request, response);
    return;
}
```

...(依次校验其他输入框的信息)

//4.注册失败消息提示 **regist.jsp中编写代码:**

在注册表单所在的table中添加一行,利用java代码取出request域中保存的提示信息,进行提示!

```
<tr>
    <td class="tds" colspan="2" style="color:red; text-align: center;">
        <%=request.getAttribute("msg") == null ? ""
        : request.getAttribute("msg") %>
    </td>
</tr>
```

//5.注册失败时,输入框数据回显 **regist.jsp中编写代码:**

```
<input type="text" name="username" value="<%=request.getParameter("username") ==
null ? "" : request.getParameter("username") %>" />
```

...(修改页面中其他输入框中的信息)

**注意:** 导入mysql的数据驱动包和c3p0连接池的包,并提供配置文件。

**注意:** 创建easymall数据库, user表, 插入数据。

//6.用户名是否已存在 **RegistServlet.java中编写代码:**

```
String sql = "select * from user where username=?";
ps = conn.prepareStatement(sql);
ps.setString(1, username);
rs = ps.executeQuery();
if(rs.next()){//用户名存在
    request.setAttribute("msg", "用户名已存在");
    request.getRequestDispatcher("/regist.jsp").forward(request, response);
    return;
}
```

//7.将数据保存在数据库中 **RegistServlet.java中编写代码:**

```
sql = "insert into user values(null, ?, ?, ?, ?)";
ps = conn.prepareStatement(sql);
ps.setString(1, username);
ps.setString(2, password);
ps.setString(3, nickname);
ps.setString(4, email);
ps.executeUpdate();
```

//8.注册成功给出提示信息, 3秒之后跳转到首页 **regist.jsp**中编写代码:

```
response.getWriter().write("<h1 style='text-align:center; color:red;'>恭喜您, 注册成功! 3秒之后跳转到首页...</h1>");  
response.setHeader("refresh", "3;url="+request.getContextPath()+"/index.jsp");
```

d. jQuery校验注册表单

i. 导入jQuery类库

参考regist.jsp页面

e. 2.5.利用java程序开发验证码图片 --不需要练习, 重点掌握禁止验证码缓存

//1.将VerifyCode工具类直接拷贝到cn.tedu.utils包下

//2.创建ValiImageServlet, 代码实现如下:

```
// --- 控制浏览器不要缓存验证码图片 ---  
response.setDateHeader("Expires", -1);  
response.setHeader("Cache-Control", "no-cache");  
VerifyCode vc = new VerifyCode();  
vc.drawImage(response.getOutputStream());  
  
//获取图片验证码  
String valistr = vc.getCode();  
  
//TODO--将验证码保存在session中
```

f. 2.6.禁止缓存验证码图片

编辑ValiImageServlet, 在最前面添加控制浏览器不缓存图片的代码:

```
response.setDateHeader("Expires", -1);  
response.setHeader("Cache-Control", "no-cache");
```

g. 2.7.点击图片换一张

//1.编辑image标签, 添加点击事件

```

```

//2.js代码实现换一张图片

```
<script>  
    function changeImage(thisObj){  
        thisObj.src = "<%= request.getContextPath() %>/ValiImageServlet?time="+new  
            Date().getTime();  
    }  
</script>
```