# 实验5 Servlet

**一、实验目的**

1、理解Servlet的生命周期。

2、熟悉web.xml文件的作用。

3、熟悉ServletConfig接口的使用

4、熟悉ServletContext接口的使用。

**二、实验报告要求**

1、实验报告提交word文档，命名为“学号+姓名+实验5”。

2、本次实验共3题。每题应提供**完整代码**、以及**运行截图**。

（1）第1题应至少提供三部分的运行截图：随机输出一位学生的学号和姓名；通过**指定下标输出本人的学号和姓名**；出错信息的截图。参考图5-1~图5-3。

（2）第2题应至少提供两部分的运行截图：使用**本人的姓名全拼**作为用户名、学号作为密码登录的界面；以及登录成功的提示信息。参考图5-4~图5-5。

（3）第3题应至少提供两部分的运行截图：使用**本人姓名**作为星球名的输入页面；显示网站访问次数的页面。参考图5-6~图5-7。

3、附上调试程序的过程中，出现的问题、以及解决方法。【可选】

**三、实验内容**

1、创建一个Servlet，在浏览器中显示一位学生的信息。具体要求：

（1）该Servlet在第一次被调用时，读取 “student.txt”中的数据，初始化一个存储学生的集合。

（2）当用户通过浏览器访问Servlet时，如果URL中不带参数num，则从学生集合中随机选择一个学生的信息输出，如图5-1所示；如果URL中带参数num，则在浏览器中显示学生集合中下标为num的学生，如图5-2所示；图5-3所示为num的值非法时的出错提示。

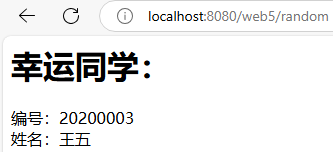


图5-1 随机输出学生

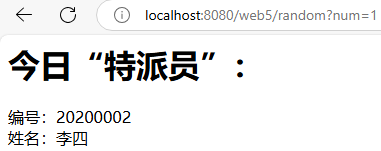


图5-2 输出指定学生

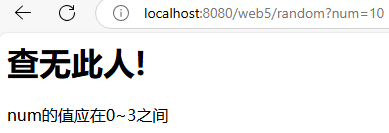


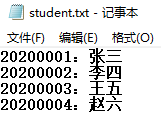
图5-3 非法提示

【实验步骤】

（1）新建一个Web应用程序，项目名称为“web5”。

（2）将“student.txt”文件拷贝到“web5”项目的“WEB-INF”节点下。

student.txt的编码格式为utf-8，数据格式如下所示：



**要求：每位同学自行编辑student.txt文件，文件中应至少包含三位学生的数据，其中一行必须为本人的学号和姓名。**

（3）在项目中创建“it.entity”包，并在包中创建Student类。

**package** it.entity;  
  
**public class** Student {  
 **private** String **sno**;  
 **private** String **sname**;  
 **//【代码1】 带2个参数的构造函数；以及成员变量的getter和setter**}

（4）在项目中创建“it.servlet”包，并在包中创建RandomeServlet类。

**package** it.servlet;  
  
**import** it.entity.Student;  
**import** javax.servlet.\*;  
**import** javax.servlet.annotation.WebServlet;  
**import** java.io.\*;  
**import** java.util.ArrayList;  
  
@WebServlet(**"/random"**)  
**public class** RandomServlet **implements** Servlet {  
 ArrayList<Student> **studentList** = **new** ArrayList<>();  
 @Override  
 **public void** init(ServletConfig servletConfig) **throws** ServletException {  
 BufferedReader reader=**null**;  
 **try** {  
 *//获取student.txt的实际路径* String filePath = servletConfig.getServletContext().getRealPath(**"/WEB-INF/student.txt"**);  
 System.***out***.println(**"student.txt的实际路径："** + filePath);  
 reader = **new** BufferedReader(**new** InputStreamReader(**new** FileInputStream(filePath), **"utf-8"**));  
 *//使用转换流，使用utf-8对读取的字节进行编码* String line = **null**;  
 **while** ((line = reader.readLine()) != **null**) {  
 **//【代码2】解析line字符串，往studentList中添加Student对象**  
 }  
 System.***out***.println(**"student data initialization completed!"**);  
 } **catch** (Exception ex) {  
 ex.printStackTrace();  
 }**finally** {  
 **if** (reader!=**null**){  
 **try** {  
 reader.close();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 }  
  
 @Override  
 **public** ServletConfig getServletConfig() {  
 **return null**;  
 }  
  
 @Override  
 **public void** service(ServletRequest servletRequest, ServletResponse servletResponse) **throws** ServletException, IOException {  
 String numString = servletRequest.getParameter(**"num"**);  
 servletResponse.setContentType(**"text/html;charset=UTF-8"**);  
 *//当没有传num参数时，随机显示一位学生的姓名* **if** (**/\*【代码3】\*/**) {  
 **int** index = (**int**) (Math.*random*() \* **studentList**.size());  
 servletResponse.getWriter().println(**"<h1>幸运同学：</h1><br>"** + **"编号："** + **studentList**.get(index).getSno() + **"<br>"** + **"姓名："** + **studentList**.get(index).getSname());  
 **return**;  
 }  
 *//判断num的值是否在学生集合的下标范围内，  
 //如果是，则显示集合中下标为num的学生信息  
 //否则，输出出错提示* **int** num = **/\*【代码4】\*/**;  
 **if** (num >= 0 && num < **studentList**.size()) {  
 servletResponse.getWriter().println(**"<h1>今日“特派员”：</h1><br>"** + **"编号："** + **studentList**.get(num).getSno() + **"<br>"** + **"姓名："** + **studentList**.get(num).getSname());  
 } **else** {  
  **//【代码5】输出出错提示**}  
 }  
  
 @Override  
 **public** String getServletInfo() {  
 **return null**;  
 }  
  
 @Override  
 **public void** destroy() {  
  
 }  
}

（5）设置项目的Application Context为“/web5”。启动tomcat，部署项目，查看结果。

2、实现用户的登录功能。要求：

（1）项目上线时，实施人员可以方便地修改登录的用户名和密码。

（2）登录页面如图5-4所示。当用户输入的用户名与密码正确，显示“\*\*\*，欢迎您”，如图5-5所示；否则，显示“用户名或密码错误”。

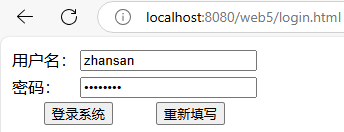


图5-4 登录页面

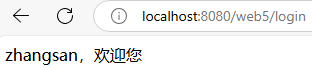


图5-5 登录成功

【实验步骤】

（1）在“web5”项目的“web”节点中，新建“HTML”文件，命名为“login”。

<!DOCTYPE **html**>  
<**html lang="en"**>  
<**head**>  
 <**meta charset="UTF-8"**>  
 <**title**>Title</**title**>  
</**head**>  
<**body**>  
<**form action="login" method="post"**>  
 <**table**>  
 <**tr**>  
 <**td**>用户名：</**td**>  
 <**td**><**input type="text" name="username"**></**td**>  
 </**tr**>  
 <**tr**>  
 <**td**>密码：</**td**>  
 <**td**><**input type="password" name="userpass"**></**td**>  
 </**tr**>  
 <**tr**>  
 <**td colspan="2" align="center"**><**input type="submit" value="登录系统"**>  
 **&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;** <**input type="reset" value="重新填写"**>  
 </**td**>  
 </**tr**>  
 </**table**>  
</**form**>  
</**body**>  
</**html**>

（2）在“it.servlet”包中新建java类“LoginServlet”。LoginServlet类实现Servlet接口，并判断用户输入的用户名与密码是否与web.xml文件中的用户名与密码一致，如果一致，显示“\*\*\*，欢迎您”，否则输出“用户名或密码错误”。

**package** it.servlet;  
  
**import** javax.servlet.\*;  
**import** java.io.IOException;  
**import** java.util.Objects;  
  
**public class** LoginServlet **implements** Servlet {

**// 【代码1】**  
}

（3）在web.xml文件中，实现LoginServlet与/login的绑定，并指定登录用户名为本人姓名的全拼、登录密码为本人的学号。



3、实现网站访问次数的统计。要求：

当用户在如图5-6所示的页面中，输入星球名，点击“开始遨游”按钮，将显示图5-7所示的页面。

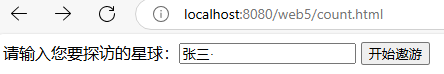


图5-6 输入“星球名”的页面



图5-7 网站访问次数统计

【实验步骤】

（1）在“web5”项目的“web”节点中，新建“HTML”文件，命名为“count”。

<!DOCTYPE **html**>  
<**html lang="en"**>  
<**head**>  
 <**meta charset="UTF-8"**>  
 <**title**>Title</**title**>  
</**head**>  
<**body**>  
 <**form action="count" method="get"**>  
 请输入您要探访的星球：<**input type="text" name="planet"**>  
 <**input type="submit" value="开始遨游"**>  
 </**form**>  
</**body**>  
</**html**>

（2）在“it.servlet”包中新建java类“CountServlet”。当浏览器通过“/count”访问该类时，能在浏览器中输出网站的访问次数。

**package** it.servlet;  
  
**import** javax.servlet.\*;  
**import** javax.servlet.annotation.WebServlet;  
**import** java.io.IOException;  
@WebServlet(**"/count"**)  
**public class** CountServlet **implements** Servlet {  
 @Override  
 **public void** init(ServletConfig servletConfig) **throws** ServletException {  
  
 }  
  
 @Override  
 **public** ServletConfig getServletConfig() {  
 **return null**;  
 }  
  
 @Override  
 **public void** service(ServletRequest servletRequest, ServletResponse servletResponse) **throws** ServletException, IOException {  
 **//【代码1】**

servletResponse.getWriter().println(**"欢迎来到<span style='color: blue'>"**+planet+**"</span>的星球！"**); *//输出星球名* servletResponse.getWriter().println(**"所有星球的总访问次数为：<span style='color: red'>"**+count+**"</span>"**); *//输出总访问次数*

}  
  
 @Override  
 **public** String getServletInfo() {  
 **return null**;  
 }  
  
 @Override  
 **public void** destroy() {  
  
 }  
}