

XI`AN TECHNOLOGICAL UNIVERSITY

实验报告

实验课程名称 组件技术

专 业： 软件工程

班 级： 17060721

姓 名： 李正阳

学 号： 17060208112

实验学时： 2

指导教师： 白小军

成 绩：

2019 年 10 月 28 日

西安工业大学实验报告

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 专业 | 软件工程 | 班级 | 17060721 | 姓名 | 李正阳 | 学号 | 17060208112 |
| 实验课程 | 组件技术 | 指导教师 | 白小军 | 实验日期 | 2019/10/28 | 同实验者 | 无 |
| 实验项目 | Servlet技术 | | | | | | |
| 实验设备及器材 | PC机 | | | | | | |

1. 实验目的

1）熟悉Java Web开发环境和工具

2）掌握Servlet技术的基本原理，理解常用API

3）掌握使用Servlet进行请求处理和响应客户端的方法

4）掌握Servlet的开发、部署和运行调试方法

5）理解Web会话跟踪的原理，能熟练使用Session进行会话跟踪

二、实验原理

1、Servlet的基本原理和常用API；

基本原理：

Servlet是运行在服务端的小程序，就我看来是利用java 书写的一种后端逻辑处理功能，专门处理用户发送给服务端的数 据和处理后的数据和资源。Servlet最经典的就是JSP技术，JSP 同时也是一个Servlet。

常用API：

ServletRequest

ServletResponse

HttpServlet

HttpSession

还有就是包含Servlet的内置七个对象

1. Servlet的基本处理流程及页面跳转方法；

基本流程：

（1）浏览器发送http请求，到达tomcat的内置web服务器

（2） 将请求与web.xml中的servlet进行映射

（3） servlet收到请求后会加载servlet，然后产生实例

（4） 根据http的请求路径自动调用servlet的方法，然后根据方法返 回相应的数据或者页面

页面跳转的方法：

分为四种方法：

在Servlet下

1. redirect方式
2. forward方式

在JSP下

1. 调用response对象的sendRedirect方法
2. <jsp:forward page = “”/>

3、Web会话跟踪的原理和实现技术；

Cookie和Session机制

Cookie是将用户的一些信息保存在用户机器的硬盘上，然后用户下一次再访问，然后将Cookie中的信息发送回服务器。

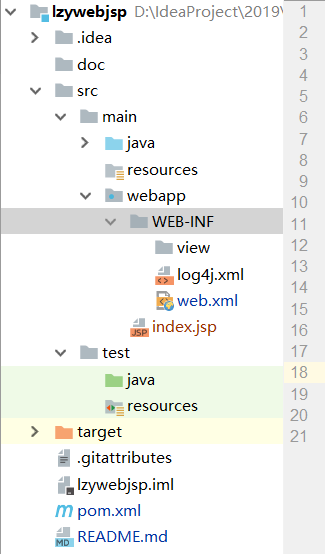
Session保存在服务端，而且是每一个浏览器再一开始就会创建，并且一般会保存在服务器的内存中。

URL重写，客户程序在每个URL的尾部添加一些额外数据。这些数据表示当前的会话，服务器将这个标识符与它存储的用户相关数据关联起来。

隐藏的表单域，表单中可以提交隐藏的信息域，但是每个页面都是由表单提交动态生成，主要应用与在线商店的结账过程。

1. 实验步骤、数据记录及处理
2. 开发Servlet版的“HelloWorld”程序，向客户端输出HTML内容；

搭建基本的WEB项目的结构，如图：



采用maven管理jar包，日志管理采用log4j。

创建index.jsp,配置tomcat，然后运行tomcat

在浏览器输入**<http://localhost:8080/index.jsp>**

出现如图实验结果：

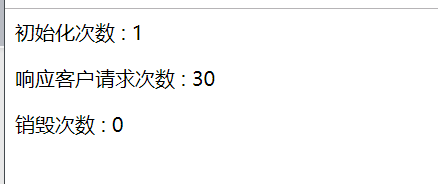
****

1. 开发简单Servlet，理解Servlet的生命周期及编程方法；

首先编写test.jsp用于测试，代码如下：

*<%--  
 Created by IntelliJ IDEA.  
 User: 李正阳 17060208112 17060721  
 Date: 2019/11/14  
 Time: 0:25  
 To change this template use File | Settings | File Templates.  
--%>*<%@ **page language**="**java**" **contentType**="**text/html; charset=UTF-8**"  
 **pageEncoding**="**UTF-8**"%>  
<**html**>  
<**head**>  
 <**title**>.测试Servelet的生命周期</**title**>  
</**head**>  
<**body**>  
  
**<%!  
 private int initVar**=0;  
 **private int serviceVar**=0;  
 **private int destroyVar**=0;  
**%>  
  
<%!  
 public void** jspInit(){  
 **initVar**++;  
 System.***out***.println(**"jspInit(): JSP被初始化了"**+**initVar**+**"次"**);  
 }  
 **public void** jspDestroy(){  
 **destroyVar**++;  
 System.***out***.println(**"jspDestroy(): JSP被销毁了"**+**destroyVar**+**"次"**);  
 }  
**%>  
  
<%  
 serviceVar**++;  
 System.***out***.println(**"\_jspService(): JSP共响应了"**+**serviceVar**+**"次请求"**);  
  
 String content1=**"初始化次数 : "**+**initVar**;  
 String content2=**"响应客户请求次数 : "**+**serviceVar**;  
 String content3=**"销毁次数 : "**+**destroyVar**;  
**%>**<**p**>**<%=**content1 **%>**</**p**>  
<**p**>**<%=**content2 **%>**</**p**>  
<**p**>**<%=**content3 **%>**</**p**>  
  
</**body**>  
</**html**>

浏览器访问结果如下：

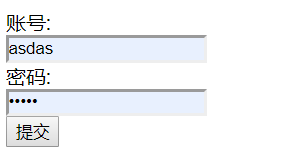


1. 开发登陆系统，含登录页面、成功页面、失败页面以及后台验证的Servlet，并将该系统部署到TomCat容器中运行；

登陆页面（index.jsp文件)：

*<%--  
 Created by IntelliJ IDEA.  
 User: 李正阳  
 Date: 2019/11/13  
 Time: 13:24  
 To change this template use File | Settings | File Templates.  
--%>*<%@ **page contentType**="**text/html;charset=UTF-8**" **language**="**java**" %>  
<**html**>  
<**head**>  
 <**title**>登陆界面</**title**>  
</**head**>  
<**body**>  
 **${**ERROR\_INFO**}** <**form action="login" method="post"**>  
 账号:<**br**/>  
 <**input name="name" id="name" type="text"**><**br**/>  
 密码:<**br**/>  
 <**input name="password" id="password" type="password"**><**br**/>  
 <**button type="submit"**>提交</**button**>  
 </**form**>  
</**body**>  
</**html**>

登陆页面如图：



成功页面（success.jsp）

*<%--  
 Created by IntelliJ IDEA.  
 User: 李正阳  
 Date: 2019/11/14  
 Time: 0:57  
 To change this template use File | Settings | File Templates.  
--%>*<%@ **page contentType**="**text/html;charset=UTF-8**" **language**="**java**" %>  
<**html**>  
<**head**>  
 <**title**>Title</**title**>  
</**head**>  
<**body**>  
 登陆成功  
</**body**>  
</**html**>

失败页面（失败页面）

*<%--  
 Created by IntelliJ IDEA.  
 User: 李正阳  
 Date: 2019/11/14  
 Time: 15:12  
 To change this template use File | Settings | File Templates.  
--%>*<%@ **page contentType**="**text/html;charset=UTF-8**" **language**="**java**" %>  
<**html**>  
<**head**>  
 <**title**>错误页面</**title**>  
</**head**>  
<**body**>  
 发生了未知的错误  
</**body**>  
</**html**>

验证系统（LoginServerlet.java）

**package** com.darksouls;  
  
**import** com.alibaba.druid.util.StringUtils;  
  
**import** javax.servlet.ServletException;  
**import** javax.servlet.http.HttpServlet;  
**import** javax.servlet.http.HttpServletRequest;  
**import** javax.servlet.http.HttpServletResponse;  
**import** java.io.IOException;  
  
*/\*\*  
 \** ***@author*** *李正阳 17060208112  
 \*/***public class** LoginServerlet **extends** HttpServlet {  
 @Override  
 **public void** doPost(HttpServletRequest req, HttpServletResponse resp) **throws** ServletException, IOException {  
 req.setCharacterEncoding(**"utf-8"**);  
 resp.setCharacterEncoding(**"utf-8"**);  
 String userName = req.getParameter(**"name"**);  
 String userPassword = req.getParameter(**"password"**);  
 **if**(!StringUtils.*isEmpty*(userName)){  
 **if**(StringUtils.*equals*(**"123456"**,userPassword)){  
 req.getRequestDispatcher(**"WEB-INF/view/success.jsp"**).forward(req,resp);  
 }**else** {  
 req.setAttribute(**"ERROR\_INFO"**,**"<span style='color:red;'>用户名密码错误！</span>"**);  
 req.getRequestDispatcher(**"index.jsp"**).forward(req,resp);  
 }  
 }**else** {  
 req.setAttribute(**"ERROR\_INFO"**,**"<span style='color:red;'>用户名密码错误！</span>"**);  
 req.getRequestDispatcher(**"index.jsp"**).forward(req,resp);  
 }  
 req.getRequestDispatcher(**"WEB-INF/view/error.jsp"**).forward(req,resp);  
  
 }  
}



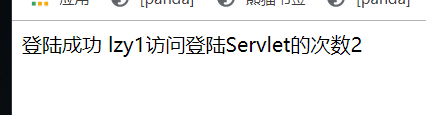
对账号和密码的检验，采用了EL表达式来去读取发送回来的Request对象中的值

1. 开发会话跟踪的Servlet，能统计并显示同一个会话中用户访问该Servlet的次数。(选作）

为了完成这个功能我在LoginServerlet添加Session的值来保存用户对Servelt的访问次数。利用HashMap来区分不同用户的登陆的次数。

修改LoginServerlet文件如下：

登陆lzy1账号访问3次如图：



切换成lzy账号再登陆两次访问的记录不一样



再次登陆lzy1可见是3次

K4O{)GG[PM0$(P3T}PY@PTP