山东大学 计算机科学与技术 学院

现代软件开发技术 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：2020001300 | 姓名： | | 班级： |
| 实验题目：前后端JSON数据交互、多文件上传，Spring Boot基础 | | | |
| 实验学时：2学时 | | 实验日期： 2023-10-14 | |
| 实验目的：  1. 会使用JSON进行前后端数据交互，了解jQuery Ajax技术  2. 掌握SpringMVC项目的配置  3. 练习多文件上传应用  4. 练习SpringBoot项目 | | | |
| 硬件环境： | | | |
| 软件环境：  操作系统：windows 10  开发平台与工具：IDEA开发平台、jdk-11.0.1、apache-tomcat-9.0.37 | | | |
| 实验内容与问题回答：  1. 前、后端JSON数据交互  打开项目ch2\_5，分析代码，解决以下问题：   1. **项目依赖的JSON解析转换包有哪些？**     **（2）JSON格式数据转换的注解有哪些？如何使用？**  @RequestBody 和 @ResponseBody。@RequestBody用于将HTTP请求的JSON数据转换为Java对象。  @ResponseBody用于将方法的返回对象转换为JSON格式的响应数据。  （3）**index.jsp中使用的前端技术有哪些？**  在上述代码中，以下前端技术和概念被使用：  1. HTML：页面是用HTML编写的  2. JavaScript：JavaScript用于处理页面的动态请求。  3. jQuery： jQuery是一个封装的JavaScript库。  5. JSON：JSON是一种前后端规定数据格式，用于在前后端之间交换数据。  **jQuery库是如何引入的？**    **$<"#page"> 的含义？**  获取页面中具有 "page" id 的HTML元素，然后，可以使用其他jQuery方法来操作或获取这个元素的内容。  **$.ajax({ ... })中，data:JSON.stringify( ...) 产生了什么JSON数据对象？**  **success:function(data){ 对应什么功能？接收的是什么格式的数据？**  **请求路径url:中的/testJson 对应的是谁？本文件中function testJson()由谁来调用？**  JSON.stringify(...) 用于将JavaScript对象转换为JSON格式的字符串。  success: function(data)当服务器成功响应请求时，它将被执行。data 参数用于接收从服务器返回的数据。  接收格式：JSON格式  /testJson对应的是TestController类里面的testJson（函数）  ${pageContext.request.contextPath} 用于获取应用程序的上下文路径，然后附加 /testJson 作为相对路径，以构建完整的URL。  function testJson()    **Q：JSON数据是怎样在网页是Java程序间交互的？**  客户端向服务器发送JSON数据，服务器端接收和处理JSON数据后再像客户端返回JSON结果。  **（4）为什么在配置文件中，要对/js/文件夹进行设置？**  为了确保资源可以被浏览器直接请求和加载，而不需要通过Java控制器来处理。不需要在每次请求时都调用Java代码来提供静态资源，提高了性能和减少了服务器的负担。  **（5）pojo包中的实体类Person对数据传递有什么作用？**  Person 类用于封装有关个人信息的数据，如姓名 (pname)、密码 (password) 和年龄 (page)。这些属性是应用程序需要处理和传递的数据。Person 类的实例可用于在应用程序中的不同部分之间传递数据。  **（6）controller包中的控制器类TestController中，@RequestBody和@ResponseBody注解的作用。**  @RequestBody 和 @ResponseBody。@RequestBody用于将HTTP请求的JSON数据转换为Java对象。  @ResponseBody用于将方法的返回对象转换为JSON格式的响应数据  2. 多文件上传项目：ch2\_6。分析其中的代码，解决以下问题。  **（1）项目依赖的包有哪些？**    **（2）分析multiFiles.jsp，与其他表单相比，其独特的标签属性是？**  enctype="multipart/form-data"  **（3）pojo包中，MultiFileDomain类中, 属性myfile的类型应该来自哪儿？**  org.springframework.web.multipart.MultipartFile  （4）控制器类MultiFilesController中，  **文件参数是用什么方式传递的？**  在控制器类 MultiFilesController 中，文件参数是通过@ModelAttribute注解和MultiFileDomain对象进行传递的。  **本地文件是如何建立的？**  本地文件的建立是通过以下代码实现的：    **哪个函数方法具体执行了文件的上传？**  文件上传的具体过程是在 multiFileUpload 方法中执行的。在这个方法中，通过 multiFileDomain.getMyfile() 获取包含上传文件的列表，然后遍历每个文件，将文件保存到指定的目录。  **（5）配置类中MultipartResolver配置了什么参数？**  maxUploadSize：通过 multipartResolver.setMaxUploadSize(5400000) 配置了上传文件的最大大小。  defaultEncoding：通过 multipartResolver.setDefaultEncoding("UTF-8") 配置了请求的编码格式。  **（6）showMulti.jsp页面代码，用什么方式获取上传文件的信息？**  在 showMulti.jsp 页面中，上传文件的信息是通过JSTL（JavaServer Pages Standard Tag Library）标签库中的 <c:forEach> 标签和EL（Expression Language）表达式获取的。  **（7）为何ch2\_6项目run on server，会显示404错误？**  因为URL是错误的  （8）上传的文件存到哪儿了？  在上述项目中，上传的文件存储在服务器的指定目录中。具体存储位置是在 MutiFilesController 控制器中的以下代码指定的：    3. 使用Eclipse创建一个简单的SpringBoot应用  参照第三章PPT，使用Eclipse创建一个Maven项目，自己命名，如T3\_1。  完成构建后，右击项目，快捷菜单“Maven/add dependency",在中间对话框中输入“web”搜索，给项目添加web-starter依赖。  再次打开pom.xml，查看依赖是否已添加。    4. 打开课本例题源码项目：ch3\_1，  分析pom.xml配置文件的内容。  了解Maven项目结构：（与ch2中的项目结构有什么不同？）  <groupId> 和 <artifactId>：这些元素定义了项目的唯一标识符。<groupId> 定义了项目的组织或包名，而 <artifactId> 定义了项目的名称。这些信息通常用于生成项目的输出文件名称。  <packaging>：<packaging> 指定项目的打包方式。在这个项目中，它的值是 "war"，表示这是一个Web应用程序（WAR文件）。  <version>：<version> 定义了项目的版本号。在这个示例中，版本号是 "0.0.1-SNAPSHOT"。  <name> 和 <url>：这些元素提供了项目的名称和URL信息。  <parent>：这个元素定义了项目的父POM。它指向了 Spring Boot 的父POM，这意味着项目继承了 Spring Boot 的配置和依赖。这有助于简化项目的配置。  <dependencies>：<dependencies> 元素列出了项目的依赖项。在这个项目中，它依赖于 Spring Boot 的 spring-boot-starter-web，以及 JUnit。  <build>：<build> 元素定义了项目的构建配置。在这里，<finalName> 指定了构建后的WAR文件的名称。  **关于Maven项目结构的其他问题：**  src/main/java 目录包含了Java源代码。  src/main/resources 目录包含了项目资源文件。  src/test 目录包含了测试相关的代码和资源。  src/main/webapp 目录包含了Web应用程序的资源，包括WEB-INF目录。与普通的Web项目结构类似，但Maven项目结构通常更标准化，更易于管理和构建。  **Q: WEB-INF现在的位置在哪儿？其中还有lib目录吗？**  WEB-INF在 src/main/webapp 下，有lib目录  **Q: 查看Java Resources/Libraries/Maven Dependencies目录中的依赖包，其对应的真实路径是？**  真实路径是本地仓库的路径。  **ch3\_1项目在eclipse中不同的运行方式，显示有什么不同？**  **（1）运行方式一：右击ch3\_1项目名称，Run as/Run on Server: http://localhost:8080/ch3\_1/ 显示的结果？**      **（2）运行方式二：右击Ch3\_1Application.java, Run as/Java Application, 在浏览器中发以下请求，显示的结果？**  http://localhost:8080/ch3\_1/  error  http://localhost:8080/hello    **通过控制台或输出窗口信息，观察以上两种运行方式的tomcat、JDK版本号，说明什么?**  第一种运行方式相应的版本号：    第二种运行方式对应的版本号：    第一种运行方式的Tomcat版本和我们配置的Tomcat版本一致，说明项目是通过Tomcat服务器来运行的，Tomcat的版本号告诉你所使用的Tomcat版本。。  第二种运行方式下，不会显示Tomcat版本信息，因为项目直接通过Spring Boot内嵌的Tomcat来运行，而不是外部Tomcat服务器。 | | | |
| 思考题解答：  **1.JSON标准的数据格式有哪些？**  对象（Object）：使用花括号（{}）表示，包含一组无序的键值对。每个键值对由一个键和一个值组成，键和值之间使用冒号（:）分隔，多个键值对之间使用逗号（,）分隔。  示例：  json  {  "name": "John",  "age": 30,  "city": "New York"  }  数组（Array）：使用方括号（[]）表示，包含一组有序的值。每个值之间使用逗号（,）分隔。  示例：  json  ["apple", "banana", "orange"]  字符串（String）：使用双引号（""）或单引号（''）括起来的字符序列。  示例：  json  "Hello, World!"  数字（Number）：表示数字，可以是整数或浮点数。  示例：  json  42  3.14  布尔值（Boolean）：表示真（true）或假（false）。  示例：  true  false  **2.ajax中如何接收传回的数据？**  在使用.ajax进行请求时，可以通过回调函数来接收传回的数据。在$.ajax()函数中，有一个success属性，它接受一个回调函数作为参数。这个回调函数会在请求成功时被调用，并且可以接收到服务器返回的数据作为参数。  **3.ch2\_6中多文件上传比传统技术的便利之处。**  1.简化的配置：通过Spring MVC和Spring Boot，你可以使用少量的配置实现多文件上传，而无需编写繁琐的Servlet代码。  2.内嵌Tomcat：Spring Boot内嵌了Tomcat，这意味着你不需要手动配置和启动Tomcat服务器。项  3.Maven构建：项目使用Maven作为构建工具，这使得依赖管理和项目构建更加简单。  **4.总结SpringBoot的特点**  （1）快速开发  （2）内嵌tomcat和jetty容器，不需要单独安装容器  （3）基于注解的零配置思想 | | | |