

Java认证:Struts开发的最佳实践Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/559/2021_2022_Java_E8_AE_A4_E8_AF_81_c104_559571.htm 本文向您讲解了如何使用灵活的 Struts 框架调节 Web 应用程序开发。在这里，本文作者向您展示了一些最佳实践，您可以遵循这些实践来优化这个成熟的开源框架。此外，您还可以学习使用一些十分有价值的标准 Struts 组件，包括 ActionForm、Action 类和 ActionErrors。

Struts：简介 Struts 是一种开源框架，用来构建 Web 应用程序，它基于流行的 Model-View-Controller (MVC2) 设计范型。该框架构建在一些标准的技术之上，比如 Java Servlets、JavaBeans、ResourceBundles 和 XML，并且可提供灵活和可扩展的组件。Struts 以 ActionServlet 的形式实现了 Controller 层，并建议使用 JSP 标记库构建 View 层。Struts 通过 Action 类提供了围绕 Model 层的包装器。图 1 展示了基于 Model-View-Controller 设计的 Struts 框架。

图 1. Struts 和 MVC

Struts 组件概览 首先，我们在最佳实践上下文中解释 Struts 组件，以及它们在 Web 应用程序开发中所起的作用。Action 应用程序的每个 Action 都会扩展 Struts 的 `org.apache.struts.action.Action` 类。这些 Action 类为应用程序的 Model 层提供了一个接口，充当围绕业务逻辑的包装器。每个 Action 类都必须向 `perform()` 方法提供其特定于用例的实现。`perform()` 方法经常返回类型 `ActionForward` 的一个值。ActionForm 应用程序的 ActionForm 扩展了 Struts 的 `org.apache.struts.action.ActionForm` 类。ActionForm 是一些封装和验证请求参数的简单 JavaBean。要验证请求数据

，ActionForm 的 validate () 方法必须给出一个特定于该情况的实现。ActionForm 作为运载工具，向 Action 类提供请求数据。一个 JSP 对象与各自的 ActionForm 对象相结合，构成应用程序的 View 层。在该层，几乎 JSP 对象的每个表单字段都映射到相应的 ActionForm 的属性。JSP 定制标记库 JSP 定制标记库是用标记表示的一组行为的集合。这是 JSP Specification 1.1 的一个强大特性；它将其他应用程序层的表示区别了开来。这些库易于使用，而且可以以一种类似 XML 的方式来读取。只要尽量少地在其中使用 Java scriptlet，就可以轻松维护 JSP 组件。Struts 提供的 JSP 标记包括 HTML、逻辑和 bean 标记。ActionErrors 可以使用 ActionError 来支持异常处理。ActionError 捕捉应用程序异常，并将其传送给 View 层。每个异常都是一个 ActionError 实例的集合。ActionError 可以封装错误消息，而 Presentation 层中的 `gt` 可以呈现 ActionError 集合内的所有错误消息。

最佳实践 1. 跨多个 ActionForm 重用数据

熟悉了 Struts 组件之后，就可以继续学习如何充分利用这一框架。首先，Struts 建议将每个 JSP 对象与一个 ActionForm 相关联，后者可以封装屏幕上显示的数据。可以通过 ActionForm 内的附加方法来访问 JSP 对象内的表单数据。清单 1 展示了 ActionForm 标记在 View 层中的传统方法。

清单 1. 使用 ActionForm `gt` `gt` `gt`

这个 ActionForm 被称为“BP1AForm”，它包括属性 `attrib1` 及其 `getter` 和 `setter` 方法。在配置文件 `struts-config.xml` 中，行为“/bp1”通过 `name` 属性映射到 `bp1AForm`。这有助于在 JSP 中显示数据。

- 1、要实现这一最佳实践，Struts 建议您进行以下两个操作：
- 2、创建一个 `JavaBean` (BP1BForm)，且其属性是 BP1AForm 属性的子

集，还要创建这些属性的 getter 和 setter 方法。通过将这个 bean 与 BP1AForm 关联，用 bean BP1BForm 的属性替代 BP1AForm 中的属性。现在就可以通过 BP1BForm 访问 BP1AForm 中的属性子集了。清单 2 展示了访问的方式。清单 2. 访问 JSP 中的表单属性 gt. gt. gt. gt. 要点 这种实践的主要优势是可用于多个 ActionForm 访问一个属性集。在遵循这一最佳实践的同时，需要记住以下几点：

- Struts 实现 gt. 标记。

当代码 gt. 指向 struts-bean.tld 时，gt. 标记开始在 JSP 组件内工作。由 ActionForm 扩展而来的 BP1AForm 验证框架必须验证 BP1BForm 的数据。当在应用程序中创建 Action 类时，不需要直接扩展 org.apache.struts.action.Action，可以通过扩展 org.apache.struts.action.Action 创建一个 Action 类（IntermediateAction），用于处理应用程序中的常见事务。所有其他的 Action 类都扩展 IntermediateAction 类。

最佳实践 2. 使用 Action 类处理请求

通常，在使用这个 Struts 框架时，对于 JSP 组件请求应用程序执行的每个动作，应用程序都必须扩展 Struts 的 org.apache.struts.action.Action 以创建 Action 类。在处理请求时，单个的 Action 类与应用程序的 Model 层连接。

要实现这一最佳实践，Struts 建议您遵循以下步骤：

- 1、通过扩展 org.apache.struts.action.Action 创建一个 Action 类，比如 BP2Action.
- 2、通过扩展 BP2Action 在 Web 应用程序中创建所有其他 Action 类。
- 3、在 BP2Action 类中创建一个方法 performTask ()，就像在公共抽象类 ActionForward performTask (ActionMapping mapping , ActionForm form , HttpServletRequest request , HttpServletResponse response) throws IOException , ServletException 中一样。
- 4、在

BP2Action 类中向应用程序添加一个或多个泛型方法，比如 `serverSideValidate()`。考虑以下因素后决定方法的访问修饰符：如果所有 Action 类都必须实现此方法，则让其为抽象。如果某些 Action 类提供一个特定的实现，则将此方法声明为受保护，并给它一个默认实现。

5、在 BP2Action 类中，将方法 `perform()` 声明为 `final`。调用上述的泛型方法（通常在处理请求前调用该方法）。现在调用步骤 3 中创建的方法 `performTask()`。

6、在每个扩展 BP2Action 的 Action 类，添加具有特定实现的方法 `performTask()`。优势这一实践有两个主要优势。首先，它避免了 Web 应用程序中每个 Action 类的冗余代码。其次，通过将 Action 类的行为集中在一起，使应用程序能够更多地控制通用的任务。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com