

浅谈如何事半功倍的学习Struts框架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_ti2020/644/2021_2022__E6_B5_85_E8_B0_88_E5_A6_82_E4_c104_644487.htm 由于Struts框架在全球的广泛使用，学习它的人也非常之多，但是大部分人花费了太多不必要的时间和精力在一些不太重要的地方，导致学习代价高昂，成效也不是很好。我这里根据多年培训经验的总结，精心设计了一套学习Struts框架的曲线，让大伙学习Struts框架既快且准，节省大伙宝贵的时间，一定会对各位学习者大有帮助。大家知道，Struts是Apache基金会Jakarta项目组的一个开源项目，它采用MVC设计模式，是一个基于Sun J2EE平台的WEB表现层框架。这里有几个概念值得大家关注：开源，MVC，J2EE，WEB表现层，框架。我来逐一解释。开源：不用多说，源代码可以让你一行一行仔细分析，深入了解其内部实现机制。MVC：它强制性的使应用程序的输入、处理和输出分开。使用MVC应用程序被分成三个核心部件：模型、视图、控制器。它们各自处理自己的任务。MVC享有“全球第一设计模式”的美誉，是当前很流行的一种设计理念和思想，你要学好Struts，必要要让这种思想深入骨髓。（我在另一篇文章中对MVC进行过阐述，关于MVC的流程及例子请参阅拙作：<http://blog.csdn.net/lenotang/archive/2008/06/18/2562348.aspx>）J2EE：J2EE是使用Java技术开发企业级应用的一种事实上的工业标准。J2EE将组成一个完整企业级应用的不同部分纳入不同的容器(Container)，每个容器中都包含若干组件（这些组件是需要部署在相应容器中的），同时各种组件都能使用各种J2EE Service/API。其中最常用的容器就

是WEB容器(就是我们常说的WEB服务器), 里面最核心的两个组件就是JSP和SERVLET。而Struts就是基于这两种技术, 依赖于WEB容器。 WEB表现层: 我们常说的J2EE四层体系架构: WEB表现层, 业务逻辑层, 持久层, 数据库层。这是一种逻辑划分。WEB表现层的特点就是依靠WEB服务器, 直接跟客户端(一般是WEB浏览器)打交道。Struts处于WEB表现层, 主要就是接受并处理用户请求, 并将处理结果响应给用户。

框架: 通俗地讲, 框架就是一个“半成品”, 它做了一些通用的东西让你去配置和扩展。框架的强大之处不是说它能让你做什么, 而是它不能让你做什么! 它使混乱的东西变得结构化。我们来打个比方。框架就像是一间有很多屋梁的房子, 当你需要扩建房子的时候, 譬如增加新的房间、窗户和过道或者在卧室增加一个壁炉, 由于屋梁的限制, 你并没有什么其他的选择。虽然较少的屋梁会让你有更多的选择, 但是当台风来袭或者发生地震的时候, 你让家人住在这样一间只有屋顶的房子里, 恐怕不会觉得安全吧。总之, 框架是在结构和创造力之间的一个精确的天平。好了。这里假设大家已经理解了上述概念, 并会用JSP/SERVLET做一些普通的WEB应用程序。现在就让我们来看看如何学习Struts框架吧。这里我以Struts 1.2版本为例。(Struts 2.0的实现跟以前的版本有很大不同, 我以后会讲到。) 让我们出发吧! www.Examda.CoM 考试就到百考试题 首先, 你要照书本或者网上的例子跑一个最简单的Struts应用, 就是在客户端显示一个HELLOWORLD的那种, 这叫先睹为快。在这里你开始接触到做Struts应用需要的东西: Struts的JAR包, 核心控制器ActionServlet在web.xml上的配置, 针对具体用户请求 URL的Action的编写以及Struts应用

的核心配置文件Struts-config.xml。做完后停下来思考一下。然后，你开始跑一个稍微大点的用户登陆的应用，成功就跳转到我们的主页面，失败就继续回到登陆页面。我们引入ActionForm组件和对后台数据库的操作。这个时候MVC发挥作用了。ActionServlet和自己编写的Action分别充当前，后端控制器组件，DAO/DTO充当了模型组件，JSP充当视图组件。Struts-config.xml也变的饱满起来，上面开始多了些配置datasources，form-beans,action-mappings。做完后到网上找一些好的参考资料或者利用自己经典的Struts的案头书(最好是有图解)，开始认真琢磨 Struts的运行原理和执行流程，特别要好生体会ActionForm组件的作用和Struts-config.xml这个核心配置文件每一处用到的配置。这个时候，你应该开始有些领悟：我们自己写的Action还有ActionForm类一定要继承框架已有的Action和ActionForm，写完了之后还必须得将它们配置到Struts-config.xml中去。接着，你已经对Struts框架有了大致的了解，可能开始对Struts的其他主题感兴趣了，OK,现在扩充你的登陆应用吧。先加上验证，不然你的系统会不堪一击。验证分为客户端验证和服务器端验证。服务器端的验证我们可以在ActionForm里面的validate()方法里做，也可以在随后的Action类的execute()方法里做，甚至可以使用验证框架。这里大家要把重点方在验证框架的学习上。做好后再加上资源国际化I18N，针对不同区域不同语言的客户提供不同语言版本的页面显示。我们开始写资源属性文件，开始在Struts-config.xml里面配置,开始使用Struts自带的标签,等。这个里面涉及的知识点都很多，需要大家耐心细致地学习。大家在跑这个应用的时候会发现各种各样的异常和问题。不要

害怕，这些错误都是我们的好朋友，我们要把它们通通都记录下来，并坚韧地排除掉它们。这里要记住：尽量避免重复犯错，不要在同一地方跌倒两次。来源：www.examda.com

做完后，认真总结和思考再接着，你把上面的都基本消化了，但是总隐隐觉得不够系统和完整，你就开始做几个CRUD的应用，完成普通的增删查改的工作。在做这些应用的过程中，你要认真总结核心配置文件Struts-config.xml里面的8个重要节点元素的配置以及它们的顺序，每个节点的重要属性，特别是里的子节点的属性是重中之重，一定要好好理解并灵活运用。编辑特别推荐: 指点一下：到底该不该去考JAVA认证? Java认证权威问答精华集 你也要开始系统地学习Struts里面的标签。但这个时候千万不要追求大而全，只学习几种常用的标签就可以了，重点放在html,bean,还有logic标签上。这时候还有些小知识点你也许要用到，譬如：文件上传控件，防止信息重复提交，自定义标签，cookie的使用等等。这就是个相对比较耗时间和精力阶段了，你要做的就是不断地参考，不断地思考，不断地总结，最好还能和志同道合的人不断地争论。慢慢地，你对Struts框架有了一个全局性的宏观把握，用它来完成一些小模块的开发已经没有太大问题，你已经开始志得意满了：Struts不过如此嘛!如果你到了这一步，我首先要恭喜你，你接近成功了。但我也要小心地提醒你：Struts里面还有很多东西值得挖掘!你不妨多问一下自己，软件有一个人开发的吗?页面我使用了良好的布局吗?页面导航都是那么简单吗?我的所有做法就不能有更简单的实现了吗?我对学习Struts框架的好处了如指掌了吗?好的，如果你想通了，带着这些问题，你就会重新披挂上阵，开始学习Struts的多模块开

发，开始学习tiles框架，开始认真研读Struts的源代码，开始学习一些更简单有效的Action组件类和ActionForm组件类.....，真应了一句古语：吾生也有涯，知也无涯!最后呢，大家可能会悟出一些道理：譬如Struts框架基本的好处就是用配置代替编码，它是一个半成品，是让我们扩展和配置的等等。上面我粗略地勾勒了一下学习Struts的曲线。归纳一下：学习框架需要循序渐进，实例驱动，思考总结。实际上，我在做培训的过程中发现，如果我们在学习Struts之前，自己动手写过一个简易的MVC框架，学习Struts将会非常轻松，大概只需要五天的时间，而且可以学的非常好。大家不要认为框架就是很庞大很复杂的怪兽，自己绝对无法写出来。其实，无非就是将我们所理解的朴素思想用代码实现出来罢了。好了，时间关系，今天就写到这里，百考试题祝愿大家学的开心!

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