

2011年计算机二级考试JAVA知识点整理 (68) PDF转换可能丢失图片或格式, 建议阅读原文

https://www.100test.com/kao_ti2020/647/2021_2022_2011_E5_B9_B4_E8_AE_A1_c97_647210.htm 2.4 Java Web应用中的任务调度

为何需要任务调度? 在web应用中, 大多数任务是以一种"防止用户长时间等待"的方式完成的。在Google搜索这样的例子中, 减少等待时间对用户体验来说至关重要。异步任务的一种解决方案是在用户提交后生成一个线程(来处理异步任务), 但这也不能解决那些需要以一定时间间隔重复运行任务、或在每天的指定时间运行任务的情况。让我们从一个数据库报表的例子来看看任务调度能如何帮助改善系统设计。报表可能是错综复杂的, 这取决于用户所需数据的种类, 以及是否需要从一个或多个数据库收集大量数据。用户可能需要很长时间来运行这样的"按需"报表。因此, 我们向这个报表示例中添加任务调度机制, 以使用户可以安排在任何他们需要的时间生成报表, 并以PDF或其他格式在email中发送。用户可以让报表在每天的凌晨2:22, 系统正处于低负荷时运行.也可以选择只在特定时间运行一次。通过在报表应用中加入任务调度, 我们可以为产品添加一项有用的功能, 并改善用户体验。幸运的是, 有一个强大的开源解决方案可以让我们以标准的方式在web应用(或任何Java应用)中实施任务调度。以下示例展示了在web应用中, 如何使用Quartz来创建一个任务调度框架。这个示例还使用了Struts Action framework 插件, 以便在web应用启动时初始化任务调度机制。Struts是最常见的MVC框架, 为大多数开发人员所熟悉。当然除此之外还有许多框架可以协助在web应用中实现MVC模式。启动时初始

化任务调度器 我们首先要做的是建立一个Struts插件，让它在容器启动时创建我们的任务调度器。在以下例子中，我们选择Tomcat作为web应用容器，不过这些示例在其他容器中也应当可以运行。我们要创建一个Struts插件类，并在struts-config.xml中加入几行代码以使之可以工作。这个插件有两个可配置的初始化参数：startOnLoad指定是否要在容器启动时立即启动任务调度器，而startupDelay指定启动任务调度器之前的等待时间。启动延时很有用，因为我们可能需要首先执行一些更重要的初始化步骤。此外还可以使用listener机制，以更复杂的方式来通知SchedulerPlugIn何时启动Quartz Scheduler。我们要创建的是一个实现Struts插件接口org.apache.struts.action.PlugIn的单子类SchedulerPlugIn。Struts会按照配置文件中出现的顺序初始化各个插件。要特别注意的是init()方法中的代码，在此我们初始化了所需的Quartz对象，并得到Scheduler。我们的任务信息就要提交到此org.quartz.Scheduler对象，后者将在随后讨论。Scheduler对象由Quartz servlet根据其配置初始化，就像Struts初始化它的ActionServlet类一样。让我们来看init()方法：

```
public void
init(ActionServlet actionServlet, ModuleConfig moduleConfig) {
System.out.println("Initializing Scheduler PlugIn for Jobs!"). //
Retrieve the ServletContext // 获取ServletContext ServletContext
ctx = actionServlet.getServletContext(). // The Quartz Scheduler //
Quartz Scheduler对象 Scheduler scheduler = null. // Retrieve the
factory from the ServletContext. // It will be put there by the Quartz
Servlet // 从ServletContext取得由Quartz Servlet放置在此
的factory对象。 StdSchedulerFactory factory =
```

(StdSchedulerFactory)

```
ctx.getAttribute(QuartzInitializerServlet.QUARTZ_FACTORY_KEY).try{ // Retrieve the scheduler from the factory // 从factory取得scheduler scheduler = factory.getScheduler(). // Start the scheduler in case, it isnt started yet // 如果scheduler尚未启动，则启动它 if (m_startOnLoad != null 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
```