

Java:Web应用下实现定时任务的简便方法 PDF转换可能丢失  
图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/244/2021\\_2022\\_Java\\_Web\\_E5\\_BA\\_c104\\_244663.htm](https://www.100test.com/kao_ti2020/244/2021_2022_Java_Web_E5_BA_c104_244663.htm) 在WEB应用下实现定时任务的简便方法

在web方式下，如果我们要实现定期执行某些任务的话，除了用quartz等第三方开源工具外，我们可以使用Timer

和TimeTask来完成指定的定时任务：第一步：创建一个任务管理类，实现ServletContextListener 接口以下是引用片段

```
： public class TaskManager implements ServletContextListener {  
/** * 每天的毫秒数 */ public static final long PERIOD_DAY =  
DateUtils.MILLIS_IN_DAY. /** * 一周内的毫秒数 */ public static  
final long PERIOD_WEEK = PERIOD_DAY * 7. /** * 无延迟 */  
public static final long NO_DELAY = 0. /** * 定时器 */ private  
Timer timer. /** * 在Web应用启动时初始化任务 */ public void  
contextInitialized(ServletContextEvent event) { //定义定时器 timer  
= new Timer("数据库表备份",true). //启动备份任务,每月(4个星期)  
执行一次 timer.schedule(new  
BackUpTableTask(),NO_DELAY, PERIOD_WEEK * 4).//  
timer.schedule(new BackUpTableTask(),NO_DELAY, 30000). } /**  
* 在Web应用结束时停止任务 */ public void  
contextDestroyed(ServletContextEvent event) { timer.cancel(). // 定  
时器销毁 }} 第二步：创建一个时间任务类 以下是引用片段
```

```
： public class BackUpTableTask extends TimerTask { private static  
Log log = LogFactory.getLog(BackUpTableTask.class). private static  
boolean isRunning = false. public void run() { if (!isRunning) {  
isRunning = true. log.debug("开始执行任务..."). //开始任务
```

```
//working add what you want to do log.debug("执行任务完成...").  
//任务完成 isRunning = false. } else { log.debug("上一次任务执行  
还未结束..."). //上一次任务执行还未结束 } } } 第三步：在web  
中添加一个监听器以下是引用片段： ***.TaskManager 要定期  
执行的任务 那么在web服务器启动的时候,任务也被启动,并且  
会周期性的执行 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
```