## 通过JavaWeb监听(listener)存取Spring的配置文件

1. 在web.xml配置文件中先读取Spring的配置文件

|  |
| --- |
| <context-param>  <param-name>contextConfigLocation</param-name>  <param-value>classpath:applicationContext.xml</param-value>  </context-param> |

1. 配置监听器

|  |  |
| --- | --- |
| <listener>  <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>  </listener> | |
| //通过ServletContext的监听器把spring配置文件解析后的对象通过setAttribute()方法存储,并key为”***ROOT\_WEB\_APPLICATION\_CONTEXT\_ATTRIBUTE***”.所以在servlet中可以通过ServletContext对象的getAttribute()指定名为”***ROOT\_WEB\_APPLICATION\_CONTEXT\_ATTRIBUTE***”获取一个解析文件对象.  ServletContext sc =req.getServletContext();  ApplicationContext ac = (ApplicationContext) sc.getAttribute(WebApplicationContext.***ROOT\_WEB\_APPLICATION\_CONTEXT\_ATTRIBUTE***);  ServiceDaoInte sd = (ServiceDaoInte) ac.getBean("serviceDaoInte"); |

## 通过JavaWeb的过滤器(filter)设置编码集

|  |
| --- |
| <filter>  <filter-name>filterEncoding</filter-name>  <filter-class>org.springframework.web.filter.CharacterEncodingFilter</filter-class>  <init-param>  <param-name>encoding</param-name>  <param-value>utf-8</param-value>  </init-param>  </filter>  <filter-mapping>  <filter-name>filterEncoding</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping> |