# 过滤器、拦截器、监听器

过滤器（Filter）、拦截器（Interceptor）和监听器（Listener）是Java Web开发中常用的组件，它们在功能和使用场景上有一些区别。以下是它们的主要区别：

**1. 过滤器（Filter）**

* **定义**：过滤器是Servlet规范中定义的一种组件，用于在请求到达Servlet之前或响应返回客户端之前对请求和响应进行预处理和后处理。
* **作用范围**：作用于Servlet容器级别，可以对所有请求和响应进行处理。
* **生命周期**：
  + init()：初始化过滤器，通常在过滤器被创建时调用一次。
  + doFilter()：处理请求和响应，每次请求都会调用。
  + destroy()：销毁过滤器，通常在Web应用被卸载时调用。
* **配置方式**：通过web.xml文件或注解@WebFilter进行配置。
* **使用场景**：常用于日志记录、权限校验、请求编码设置、响应压缩等。

**2. 拦截器（Interceptor）**

* **定义**：拦截器是Spring框架中定义的一种组件，用于在方法调用前后或异常抛出时进行处理。
* **作用范围**：作用于Spring容器级别，只能对Spring管理的Bean的方法调用进行拦截。
* **生命周期**：
  + preHandle()：在方法调用之前执行。
  + postHandle()：在方法调用之后、视图返回之前执行。
  + afterCompletion()：在视图返回之后执行。
* **配置方式**：通过实现HandlerInterceptor接口或使用@Aspect注解进行配置。
* **使用场景**：常用于日志记录、权限校验、性能监控、事务管理等。

**3. 监听器（Listener）**

* **定义**：监听器是Servlet规范中定义的一种组件，用于监听Web应用中的各种事件，如请求、会话、应用的创建和销毁等。
* **作用范围**：作用于Servlet容器级别，可以监听整个Web应用的生命周期事件。
* **生命周期**：
  + contextInitialized()：在应用初始化时调用。
  + contextDestroyed()：在应用销毁时调用。
  + sessionCreated()：在会话创建时调用。
  + sessionDestroyed()：在会话销毁时调用。
  + requestInitialized()：在请求初始化时调用。
  + requestDestroyed()：在请求销毁时调用。
* **配置方式**：通过web.xml文件或注解@WebListener进行配置。
* **使用场景**：常用于初始化资源、统计在线人数、清理资源等。

**总结**

* **过滤器**：作用于Servlet容器级别，用于处理请求和响应，常用于日志记录、权限校验等。
* **拦截器**：作用于Spring容器级别，用于处理方法调用，常用于日志记录、权限校验、事务管理等。
* **监听器**：作用于Servlet容器级别，用于监听Web应用的生命周期事件，常用于初始化资源、统计在线人数等。