1. 拦截器和过滤器的区别?
2. **过滤器(Filter)：**它依赖于servlet容器。在实现上，基于函数回调，它可以对几乎所有请求进行过滤，但是缺点是一个过滤器实例只能在容器初始化时调用一次。使用过滤器的目的，是用来做一些过滤操作，获取我们想要获取的数据，比如：在Javaweb中，对传入的request、response提前过滤掉一些信息，或者提前设置一些参数，然后再传入servlet或者Controller进行业务逻辑操作。通常用的场景是：在过滤器中修改字符编码（CharacterEncodingFilter）、在过滤器中修改HttpServletRequest的一些参数（XSSFilter(自定义过滤器)），如：过滤低俗文字、危险字符等。
3. **拦截器（Interceptor）：**它依赖于web框架，在SpringMVC中就是依赖于SpringMVC框架。在实现上,基于Java的反射机制，属于面向切面编程（AOP）的一种运用，**就是在service或者一个方法前，调用一个方法，或者在方法后，调用一个方法，比如动态代理就是拦截器的简单实现，在调用方法前打印出字符串（或者做其它业务逻辑的操作），也可以在调用方法后打印出字符串，甚至在抛出异常的时候做业务逻辑的操作。**由于拦截器是基于web框架的调用，因此可以使用Spring的依赖注入（DI）进行一些业务操作，同时一个拦截器实例在一个controller生命周期之内可以多次调用。**但是缺点是只能对controller请求进行拦截**，对其他的一些比如直接访问静态资源的请求则没办法进行拦截处理。
4. 重定向和转发的区别?

1：重定向是浏览器向服务器发送一个请求并收到响应后再次向一个新地址发出请求，转发是服务器收到请求后为了完成响应跳转到一个新的地址。

2：重定向有两次请求，不共享数据，转发是有一次请求且共享数据。

3：重定向后地址栏会发生变化，转发不会。

4：重定向的地址可以是任意地址，转发的地址只能是当前应用类的某一个地址。

1. Action和Servlet的区别?

Servlet是单实例多线程的，而Action是多实例多线程的。因为多实例，所有每个请求会产生不同的Action和ActionContext对象，因此线程是安全的。同时因为Action是多实例的，所以Action中可以有成员变量，因为多实例不会公用。而Servlet是单实例多线程，所以一般情况下不会定义成员变量，因为共用同一个成员变量可能出现线程安全的问题。

1. ServletContext,Request,Session三种方式?
2. 重写和重载的区别?
3. 