spring bean 的scope有5种：

singleton：当一个bean的作用域为singleton, 那么Spring IoC容器中只会存在一个共享的bean实例，并且所有对bean的请求，只要id与该bean定义相匹配，则只会返回bean的同一实例。

换言之，当把一个bean定义设置为singleton作用域时，Spring IoC容器只会创建该bean定义的唯一实例。这个单一实例会被存储到单例缓存（singleton cache）中，并且所有针对该bean的后续请求和引用都将返回被缓存的对象实例。scope的默认值。

prototype：Prototype作用域的bean会导致在每次对该bean请求（将其注入到另一个bean中，或者以程序的方式调用容器的getBean()方法）时都会创建一个新的bean实例。根据经验，对有状态的bean应该使用prototype作用域，而对无状态的bean则应该使用singleton作用域。

request ：request表示针对每一次HTTP请求都会产生一个新的bean，同时该bean仅在当前HTTP request内有效。

配置实例： request、session、global session使用的时候首先要在初始化web的web.xml中做如下配置：

如果你使用的是Servlet 2.4及以上的web容器，那么你仅需要在web应用的XML声明文件web.xml中增加下述ContextListener即可:   
‍<listener>   
   <listener-class>org.springframework.web.context.request.RequestContextListener</listener-class>   
</listener>

如果是Servlet2.4以前的web容器,那么你要使用一个javax.servlet.Filter的实现：

<filter>   
    <filter-name>requestContextFilter</filter-name>   
    <filter-class>org.springframework.web.filter.RequestContextFilter</filter-class>  
 </filter>   
 <filter-mapping>   
    <filter-name>requestContextFilter</filter-name>   
    <url-pattern>/\*</url-pattern>  
 </filter-mapping>

session:session作用域表示该针对每一次HTTP请求都会产生一个新的bean，同时该bean仅在当前HTTP session内有效。

global session：global session作用域类似于标准的HTTP Session作用域，不过它仅仅在基于portlet的web应用中才有意义。Portlet规范定义了全局Session的概念，它被所有构成某个 portlet web应用的各种不同的portlet所共享。在global session作用域中定义的bean被限定于全局portlet Session的生命周期范围内。如果你在web中使用global session作用域来标识bean，那么web会自动当成session类型来使用。

额外一句：关于spring bean的scope使用最多的两个是singleton、prototype，关于request、session、global session需要结合web容器使用，一般用的不是太广泛，如果使用时，千万不要忘记在web.xml中配置ContextListener。