

Spring 面试题分析

1. 开发中主要使用 Spring 的什么技术？

- ①. IOC 容器管理各层的组件
- ②. 使用 AOP 配置声明式事务
- ③. 整合其他框架.

2. 简述 AOP 和 IOC 概念

AOP: Aspect Oriented Program, 面向(方面)切面的编程; Filter(过滤器)也是一种 AOP. AOP 是一种新的方法论, 是对传统 OOP(Object-Oriented Programming, 面向对象编程)的补充. AOP 的主要编程对象是切面(aspect), 而切面模块化横切关注点. 可以举例通过事务说明.

IOC: Invert Of Control, 控制反转. 也成为 DI(依赖注入)其思想是反转资源获取的方向. 传统的资源查找方式要求组件向容器发起请求查找资源. 作为回应, 容器适时的返回资源. 而应用了 IOC 之后, 则是容器主动地将资源推送给它所管理的组件, 组件所要做的仅是选择一种合适的方式来接受资源. 这种行为也被称为查找的被动形式

3. 在 Spring 中如何配置 Bean？

Bean 的配置方式: 通过全类名(反射)、通过工厂方法(静态工厂方法 & 实例工厂方法)、FactoryBean

4. IOC 容器对 Bean 的生命周期:

- ①. 通过构造器或工厂方法创建 Bean 实例
- ②. 为 Bean 的属性设置值和对其他 Bean 的引用
- ③. 将 Bean 实例传递给 Bean 后置处理器的 postProcessBeforeInitialization 方法
- ④. 调用 Bean 的初始化方法(init-method)
- ⑤. 将 Bean 实例传递给 Bean 后置处理器的 postProcessAfterInitialization 方法
- ⑦. Bean 可以使用了
- ⑧. 当容器关闭时, 调用 Bean 的销毁方法(destroy-method)

5. Spring 如何整合 Struts2？

整合 Struts2, 即由 IOC 容器管理 Struts2 的 Action:

- ＞ 安装 Spring 插件: 把 `struts2-spring-plugin-2.2.1.jar` 复制到当前 WEB 应用的 WEB-INF/lib 目录下
- ＞ 在 Spring 的配置文件中配置 Struts2 的 Action 实例
- ＞ 在 Struts 配置文件中配置 action, 但其 class 属性不再指向该 Action 的实现类, 而是指向 Spring 容器中 Action 实例的 ID

6. Spring 如何整合 Hibernate

整合 Hibernate, 即由 IOC 容器生成 SessionFactory 对象, 并使用 Spring 的声明式事务

- ＞ 利用 LocalSessionFactoryBean 工厂 Bean, 声明一个使用 XML 映射文件的 SessionFactory 实例.
- ＞ 利用 HibernateTransactionManager 配置 Hibernate 的事务管理器

7. Spring MVC 比较 Struts2

- ①. Spring MVC 的入口是 Servlet, 而 Struts2 是 Filter
- ②. Spring MVC 会稍微比 Struts2 快些. Spring MVC 是基于方法设计, 而 Struts2 是基于类, 每次发一次请求都会实例一个 Action.
- ③. Spring MVC 使用更加简洁, 开发效率 Spring MVC 确实比 struts2 高: 支持 JSR303, 处理 ajax 的请求更方便
- ④. Struts2 的 OGNL 表达式使页面的开发效率相比 Spring MVC 更高些.

8. Spring MVC 的运行流程

- ①. 在整个 Spring MVC 框架中, DispatcherServlet 处于核心位置, 负责协调和组织不同组件以完成请求处理并返回响应的工作
- ②. SpringMVC 处理请求过程:
 - ＞ 若一个请求匹配 DispatcherServlet 的请求映射路径(在 web.xml 中指定), WEB 容器将该请求转交给 DispatcherServlet 处理
 - ＞ DispatcherServlet 接收到请求后, 将根据请求信息(包括 URL、HTTP 方法、请求头、请求参数、Cookie 等)及 HandlerMapping 的配置找到处理请求的处理器(Handler). 可将 HandlerMapping 看成路由控制器, 将 Handler 看成目标主机.
 - ＞ 当 DispatcherServlet 根据 HandlerMapping 得到对应当前请求的 Handler 后, 通过 HandlerAdapter 对 Handler 进行封装, 再以统一的适配器接口调用 Handler.
 - ＞ 处理器完成业务逻辑的处理后将返回一个 ModelAndView 给



> 得到真实视图对象 View 后，DispatcherServlet 使用这个 View 对 ModelAndView 中的模型数据进行视图渲染

9. 说出 Spring MVC 常用的 5 个注解:

@RequestMapping 、 @PathVariable 、 @RequestParam 、 @RequestBody 、 @ResponseBody

10. 如何使用 SpringMVC 完成 JSON 操作:

- ①. 配置 MappingJacksonHttpMessageConverter
- ②. 使用 @RequestBody 注解或 ResponseEntity 作为返回值