## springMVC的强大之处

1. Spring MVC 实现了即用的 MVC 的核心概念。它为控制器和处理程序提供了大量与此模式相关的功能。并且当向 MVC 添加反转控制（Inversion of Control，IoC）时，它使应用程序高度解耦，提供了通过简单的配置更改即可动态更改组件的灵活性。Spring MVC 为您提供了完全控制应用程序的各个方面的力量。
2. Spring 的 Web MVC 模块是围绕 DispatcherServlet 而设计的。DispatcherServlet 给处理程序分派请求，执行视图解析，并且处理语言环境和主题解析，此外还为上传文件提供支持。
3. DispatcherServlet 通过使用处理程序映射来决定哪一个处理程序应当处理传入的请求。处理程序映射只是用于标识使用哪一个处理程序来处理特定 URL 模式的映射。处理程序是只有一种方法 ModelAndView handleRequest(request,response) 的控制器接口的实现。Spring 还有一些可用的高级处理程序实现；其中一个重要的高级处理程序实现是 SimpleFormController，它提供了将命令对象绑定到表单、对其执行验证等功能。

## springMVC优势

1. 清晰的角色划分：
2. 前端控制器（DispatcherServlet）
3. 请求到处理器映射（HandlerMapping）
4. 处理器适配器（HandlerAdapter）
5. 视图解析器（ViewResolver）
6. 后端控制器（Handler）
7. 验证器（ Validator）
8. 命令对象（Command  请求参数绑定到的对象就叫命令对象）
9. 表单对象（Form Object 提供给表单展示和提交到的对象就叫表单对象）

2、分工明确，而且扩展点相当灵活，可以很容易扩展，虽然几乎不需要；

3、由于命令对象就是一个POJO，无需继承框架特定API，可以使用命令对象直接作为业务对象；

4、和Spring 其他框架无缝集成，是其它Web框架所不具备的；

5、可适配，通过HandlerAdapter可以支持任意的类作为处理器；

6、可定制性，HandlerMapping、ViewResolver等能够非常简单的定制；

7、功能强大的数据验证、格式化、绑定机制；

8、利用Spring提供的Mock对象能够非常简单的进行Web层单元测试；

9、本地化、主题的解析的支持，使我们更容易进行国际化和主题的切换。

10、强大的JSP标签库，使JSP编写更容易。

……还有比如RESTful风格的支持、简单的文件上传、约定大于配置的契约式编程支持、基于注解的零配置支持等等。