

# 什么是Spring Boot

- Spring Boot是由Pivotal团队提供的全新框架，其设计目的是用来简化Spring应用的初始搭建以及开发过程。该框架使用了特定的方式来进行配置，从而使开发人员不再需要定义样板化的配置。通俗点说，就是spring boot其实不是什么新的框架，它默认配置了很多框架的使用方式，就像maven整合了所有的jar包，spring boot整合了所有的框架

# Spring Boot的好处

其实就是简单、快速、方便！平时如果我们需要搭建一个spring web项目的时候需要怎么做呢？

- 1) 配置web.xml，加载spring和spring mvc
- 2) 配置数据库连接、配置spring事务
- 3) 配置加载配置文件的读取，开启注解
- 4) 配置日志文件

...

配置完成之后部署tomcat 调试

但是如果使用spring boot呢？

很简单，我仅仅只需要非常少的几个配置就可以迅速方便的搭建起来一套web项目或者是构建一个微服务！



# Spring Boot简史

- Spring 1.x 时代

在 Spring1.x 时代，都是通过 xml 文件配置 bean，随着项目的不断扩大，需要将 xml 配置分放到不同的配置文件中，需要频繁的在 java 类和 xml 配置文件中切换。

把普通pojo实例化到Spring容器中

```
1 | @Component("userDao")
2 | public class UserDao{
3 |
4 | }
```

与在XML配置文件中等效

```
1 | <bean id="userDao" class="cn.my.UserDao"/>
```

POJO

```
CLASS User {
    string name;
    int age;
}
```

bean

```
@table
CLASS User {
    string name;
    int age; }
```



# Spring Boot简史

- Spring 2.x 时代

随着 JDK 1.5 带来的注解支持，Spring2.x 可以使用注解对 Bean 进行申明和注入，大大的减少了 xml 配置文件，同时也大大简化了项目的开发。

- Spring 3.x 时代

从 Spring3.x 开始提供了 Java 配置方式，使用 Java 配置方式可以更好的理解你配置的 Bean，并且 Spring4.x 和 Spring boot 都推荐使用 java 配置的方式。



# Spring Boot简史

- Spring 5.x时代

Spring 5.x是Java界首个支持响应式的Web框架，是Spring的一个重要版本。在此期间，大多数增强都是在Spring Boot项目中完成的，其最大的亮点就是提供了完整的端到端响应式编程的支持(新增Spring WebFlux模块)。

Spring webFlux同时支持使用旧的Spring MVC注解声明Reactive Controller。和传统的 MVC controller不同，Reactive Controller 操作的是非阻塞的 `serverHttpRequest`和`serverHttpResponse`，而不再是Spring MVC里的 `HttpServletRequest`和`HttpServletResponse`。

至此也代表着Java正式迎来了响应式异步编程的时代。

## hello spring boot实例

