Spring boot 入门

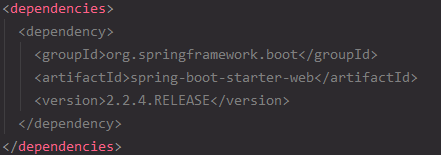
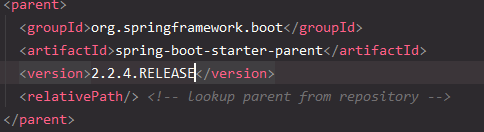
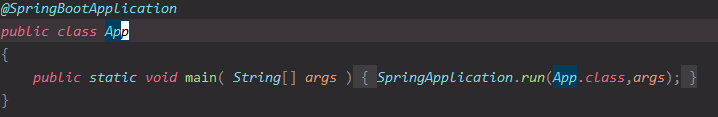
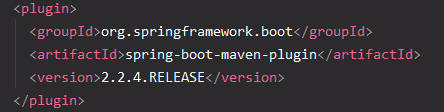
# Spring boot简介

1. 快速创建独立运行的spring项目以及主流框架继承
2. 使用嵌入式的Servlet容器，应用无需打包成war包
3. Starters自动依赖与版本控制
4. 大量的自动配置，简化开发，也可修改默认值
5. 无需配置XML，无需代码生成，开箱即用
6. 准生产环境的运行时应用监控
7. 与云计算的天然集成

# 微服务

1. 2014年martin fowler
2. 架构风格
3. 一个应用应该是一组小型服务，可以通过HTTP方式进行互通
4. 每一个功能元素最终都是一个可独立替换和独立升级的软件单元

# Hello World

1. 创建一个maven项目
2. 导入spring boot相关依赖
3. 编写controller
4. 编写主类
5. 运行主程序
   1. 自带tomcat
6. 打包
   1. 导入插件
      1. 打包成jar包
   2. mvn package
   3. 运行jar包 

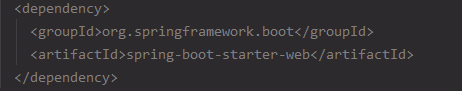
# Hello World探究

## Pom.xml

### 父项目

1. Springboot父pom管理所有依赖版本
2. 以后导入依赖无需写版本（没有在dependencies里面的依赖需要声明版本号）

### 启动器

1. Spring-boot-starter-web
   1. Spring-boot-starter
      1. Spring-boot场景启动器
      2. 帮我们导入了web模块模块正常运行所依赖的组件
2. Spring-boot将所有功能场景都抽取出来，做成一个个的starter启动器，只需在项目里面引入这些starter相关场景的所有依赖都会导入进来。

## 主程序类

1. @SpringBootApplication
   1. 标注这个类是SpringBoot的主配置类，springboot就应该运行这个类的main方法来启动springboot应用
   2. **只有主配置类所在包及下面所有子包里面的所有组件扫描到spring容器**
   3. SpringBoot在启动的时候从类路径下的META-INF/spring.factories中获取EnableAutoCondiguration指定的值，将这些值作为自动配置类导入到容器中，自动配置类就生效，帮我们自动配置工作；

# 快速创建

1. 使用Spring Initializer快速创建项目
   1. 选择spring web
   2. 默认生成Spring Boot项目：
      1. 主程序已经生成好
      2. resource文件夹中的目录结构
         1. static：保存所有的静态资源js,css,images；
         2. template：保存所有的模板页面（Spring Boot默认jar包使用嵌入的Tomcat，默认不支持JSP）
         3. application.properties：spring boot应用的默认配置文件；可以修改一些默认设置；