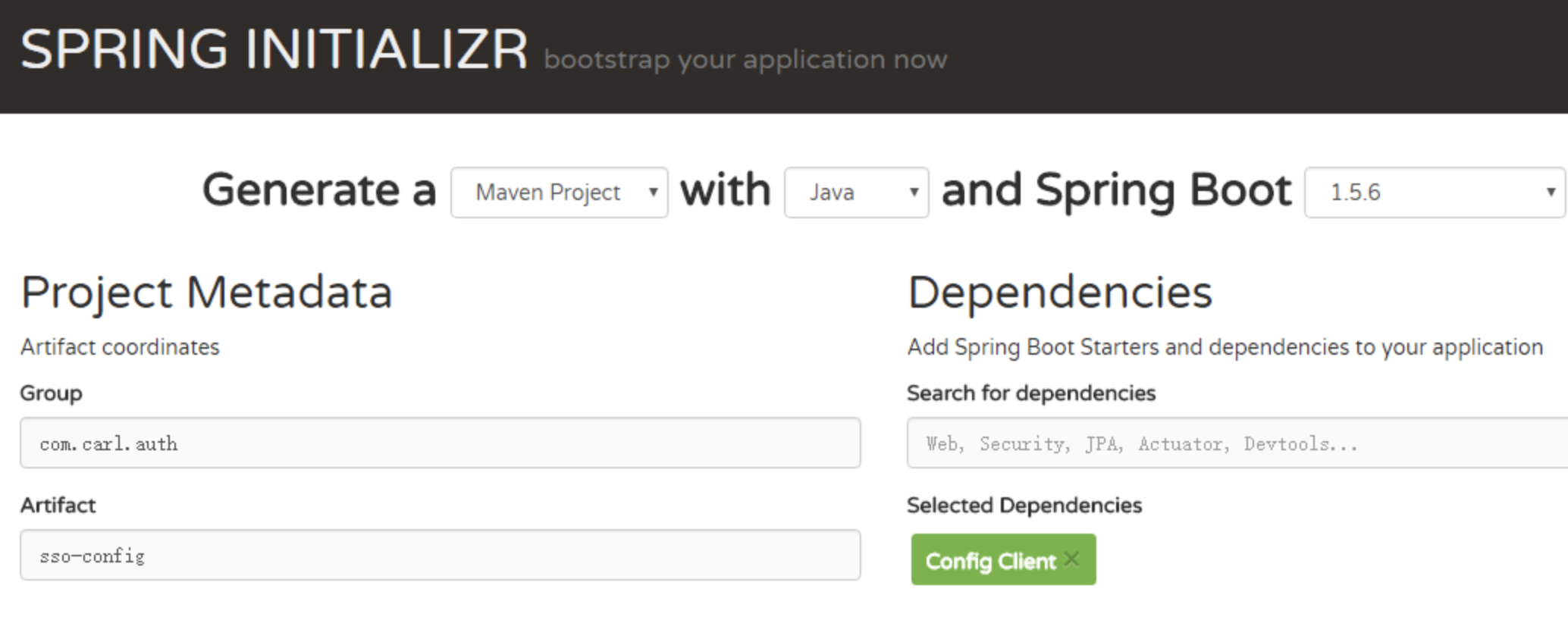
# 1、创建springboot项目 (不过依赖应该选择的是config server)



# 2、pom文件如下

*<?*xml version="1.0" encoding="UTF-8"*?>*<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
  
 <groupId>com.carl.auth</groupId>  
 <artifactId>sso-config</artifactId>  
 <version>0.0.1-SNAPSHOT</version>  
 <packaging>jar</packaging>  
  
 <name>sso-config</name>  
 <description>Demo project for Spring Boot</description>  
  
 <parent>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-parent</artifactId>  
 <version>1.5.10.RELEASE</version>  
 <relativePath/> <!-- lookup parent from repository -->  
 </parent>  
  
 <properties>  
 <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
 <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>  
 <java.version>1.8</java.version>  
 <spring-cloud.version>Edgware.SR2</spring-cloud.version>  
 </properties>  
  
 <dependencies>  
 <dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-config-server</artifactId>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-test</artifactId>  
 <scope>test</scope>  
 </dependency>  
 </dependencies>  
  
 <dependencyManagement>  
 <dependencies>  
 <dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-dependencies</artifactId>  
 <version>${spring-cloud.version}</version>  
 <type>pom</type>  
 <scope>import</scope>  
 </dependency>  
 </dependencies>  
 </dependencyManagement>  
  
 <build>  
 <plugins>  
 <plugin>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-maven-plugin</artifactId>  
 </plugin>  
 </plugins>  
 </build>  
  
  
</project>

# 3、添加配置管理器支持注解

@EnableConfigServer  
@SpringBootApplication  
public class SsoConfigApplication {  
  
 public static void main(String[] args) {  
 SpringApplication.run(SsoConfigApplication.class, args);  
 }  
}

# 4、编辑 application配置文件

#指定日志输出文件  
logging:  
 file: "logs/sso-config.log"  
info:  
 name : "配置中心"  
  
---  
server:  
 #服务端口  
 port: 8888  
 #访问路径  
 context-path: /config  
spring:  
 profiles:  
 #本地配置文件  
 active:  
 #配置文件本地化  
 - native  
 application:  
 #指定应用名称  
 name: sso-config

# 5、运行这个spring boot项目，浏览器访问 http://localhost:8888/config/ 说明下面的启动成功

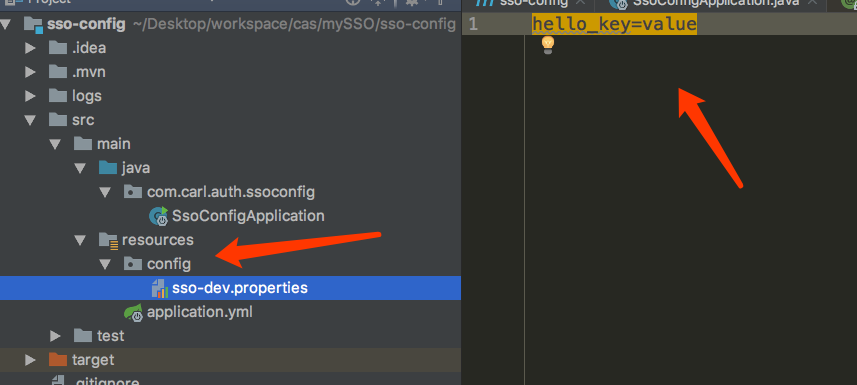


# 6、配置中心的位置，创建resource 下创建config文件夹，创建sso-dev.propetries 编辑配置文件

## 1、上面我们提到配置是放置于配置中心，而不放在服务内

## 2、由于配置是放在与本地，那么spring的默认配置在目录为resources/config

hello\_key=value



# 7、重新启动spring boot项目，访问 http://localhost:8888/config/sso/dev



#### name:应用名称 sso -dev.properties配置文件的名称

#### label 表示分支名称 master

#### version 表示git上对应的版本号

{

"name": "sso",

"profiles": ["dev"],

"label": null,

"version": null,

"state": null,

"propertySources": [{

"name": "classpath:/config/sso-dev.properties",

"source": {

"hello\_key": "value"

}

}]

}