

1.springboot-helloworld

springboot-helloworld

下载地址: <https://github.com/QiangBoCai/springbootDemo/>

1.新建一个maven springboot web工程;

1.1 编写maven pom.xml配置

1.1.1 配置项目继承spring-boot-starter-parent

1.1.2 pom配置 Spring Boot web工程依赖

1.1.3 pom配置war 打包

1.1.4 配置 兼容非springboot内置tomcat中运行

1.1.5 maven update project 更新配置中的lib到本地

1.2.配置application.yml

1.3 配置日志打印, 默认logback日志配置

1.4. springboot程序入口 继承SpringBootServletInitializer 类

1.4.1 页面动态访问

1.5 编译运行

1.5.1 Run as maven install

1.5.2 Run as Java Application

1.6.访问web静态资源

1.7. 访问web动态网页

1.新建一个maven springboot web工程;

create a simple project

Artifact	
Group Id:	com.lance
Artifact Id:	springboot-helloworld
Version:	0.0.1-SNAPSHOT
Packaging:	war
Compiler Level:	1.8
Name:	springboot-helloworld
Description:	springboot-helloworld
Parent Project	
Group Id:	org.springframework.boot
Artifact Id:	spring-boot-starter-parent
Version:	2.0.1.RELEASE
Advanced	

继承parent的一些默认配置

当前springboot版本号

1.1 编写maven pom.xml配置

参考：

```

<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>
  <!-- 1.spring-boot-starter-parent -->
  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.0.1.RELEASE</version>
  </parent>

  <groupId>com.lance</groupId>
  <artifactId>springboot-helloworld</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <!-- war 打包 配置1-->
  <packaging>war</packaging>

  <name>springboot-helloworld</name>
  <description>springboot-helloworld</description>

  <dependencies>
    <!--2. springboot web工程 -->
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>

    <!--4. 兼容在外部tomcat中运行 -->
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-tomcat</artifactId>
      <scope>provided</scope>
    </dependency>
  </dependencies>

  <build>
    <plugins>
      <!-- 5.配置默认的JDK版本 -->
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <configuration>
          <source>1.8</source>
          <target>1.8</target>
        </configuration>
      </plugin>
      <!-- 3.war 打包 配置2 -->
      <plugin>
        <artifactId>maven-war-plugin</artifactId>
        <configuration>
          <version>3.1</version>
          <!-- 忽略找不到web.xml的错误 -->
          <failOnMissingWebXml>false</failOnMissingWebXml>
        </configuration>
      </plugin>

    </plugins>
  </build>

</project>

```

1.1.1 配置项目继承spring-boot-starter-parent

Maven的用户可以通过继承spring-boot-starter-parent项目来获得一些合理的默认配置。这个parent提供了以下特性：

- 默认使用Java 8
- 使用UTF-8编码
- 一个引用管理的功能，在dependencies里的部分配置可以不用填写version信息，这些version信息会从spring-boot-dependencies里得到继承。
- 识别过来资源过滤 (Sensible resource filtering.)
- 识别插件的配置 (Sensible plugin configuration (exec plugin, surefire, Git commit ID, shade).)
- 能够识别application.properties和application.yml类型的文件，同时也能支持profile-specific类型的文件（如： application-foo.properties and application-foo.yml，这个功能可以更好的配置不同生产环境下的配置文件）。

<!-- 继承默认配置 -->

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>2.0.1.RELEASE</version>
</parent>
```

1.1.2 pom配置 Spring Boot web工程依赖

```
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-web</artifactId>
</dependency>
```

1.1.3 pom配置war打包

```
<packaging>war</packaging>
<plugin>
  <artifactId>maven-war-plugin</artifactId>
  <configuration>
    <version>3.1</version>
    <!-- 忽略找不到web.xml的错误 -->
    <failOnMissingWebXml>false</failOnMissingWebXml>
  </configuration>
</plugin>
```

1.1.4 配置 兼容非springboot内置tomcat中运行

<!--spring boot tomcat（默认可以不用配置，但当需要把当前web应用布置到外部servlet容器时就需要配置，并将scope配置为provided）-->

```
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-tomcat</artifactId>
<scope>provided</scope>
</dependency>
```

1.1.5 配置默认的JDK版本

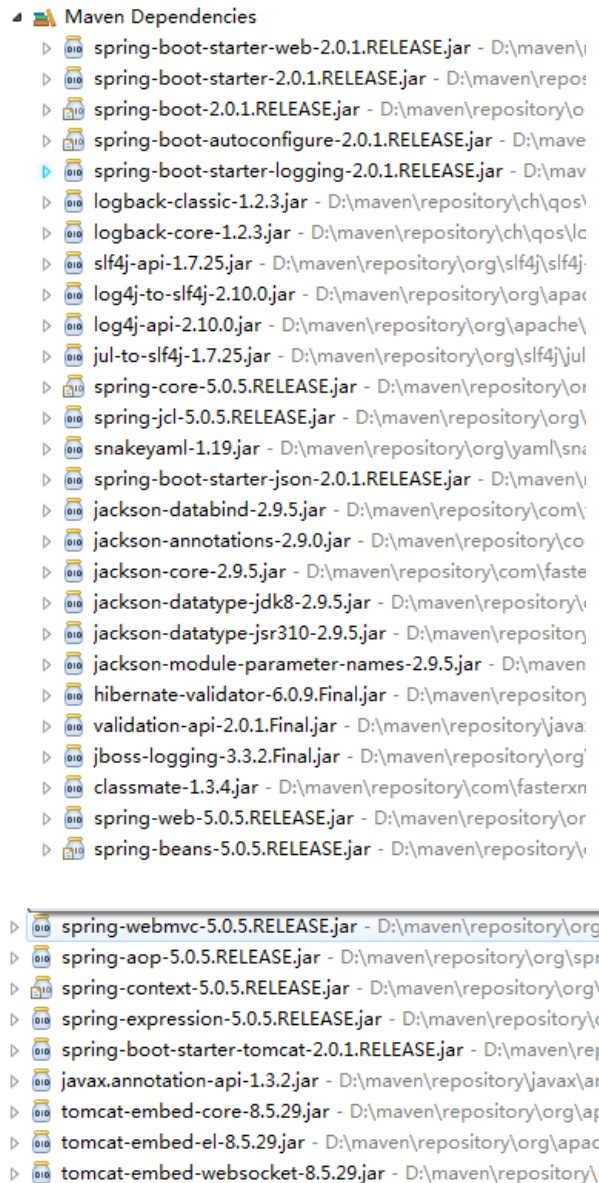
```
<plugin>
```

```

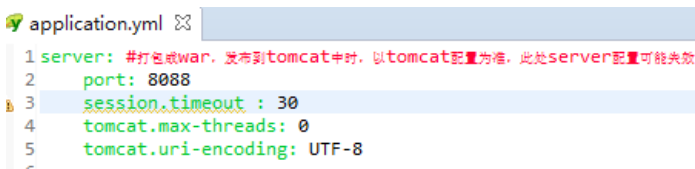
<artifactId>maven-compiler-plugin</artifactId>
<configuration>
  <source>1.8</source>
  <target>1.8</target>
</configuration>
</plugin>

```

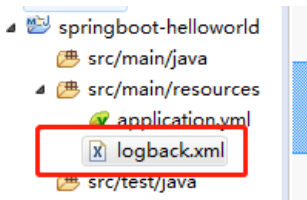
1.1.6 maven update project 更新配置中的lib到本地



1.2.配置 application.yml



1.3 配置日志打印，默认logback日志配置



1.4. springboot程序入口 继承SpringBootServletInitializer 类

1.6. 访问web静态资源

默认静态资源放置在/src/main/resources/static目录下

<http://localhost:8080/>

1.7. 访问web动态网页

<http://localhost:8080/hello>
