Atitit maven打包war流程总结

目录

[1. 编译错误不能打包的问题 1](#_Toc3765)

[1.1. 生成war文件夹 1](#_Toc1821)

[1.2. 复制web inf目录 1](#_Toc6441)

[1.3. 复制class文件夹，，貌似默认就可以了 2](#_Toc30380)

[1.4. 复制libs 2](#_Toc21212)

[1.5. Code--- 3](#_Toc4373)

# 编译错误不能打包的问题

使用一个空项目 ，war打包lib即可。。哈哈，class文件复制过去即可。。

## 生成war文件夹

设置 <build>

<!-- war filename -->

<finalName>admin</finalName>

## 复制web inf目录

<resources>

<!-- -->

<resource>

<!-- 指定resources插件处理哪个目录下的资源文件 -->

<directory>src/main/webapp</directory>

<!--注意此次必须要放在此目录下才能被访问到 rlt dir is rlt classdir -->

<targetPath>${basedir}/target/admin</targetPath>

<includes>

<include>\*\*/\*\*</include>

</includes>

</resource>

## 复制class文件夹，，貌似默认就可以了

<resource>

<!-- 指定resources插件处理哪个目录下的资源文件 -->

<directory>${basedir}/target/classes</directory>

<!--注意此次必须要放在此目录下才能被访问到 -->

<targetPath>${basedir}/target/admin/WEB-INF/classes</targetPath>

<includes>

<include>\*\*/\*\*</include>

</includes>

</resource>

## 复制libs

<resources>

<resource>

<!-- 指定resources插件处理哪个目录下的资源文件 -->

<directory>lib</directory>

<!--注意此次必须要放在此目录下才能被访问到 -->

<targetPath>${basedir}/target/admin/WEB-INF/lib</targetPath>

<includes>

<include>\*\*/\*\*</include>

</includes>

## Code---

<?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>ttadmingroupId</groupId>

<artifactId>tt-admin</artifactId>

<name>tt-admin</name>

<version>1.1</version>

<packaging>war</packaging>

<!-- <parent> <groupId>org.springframework.boot</groupId> <artifactId>spring-boot-starter-parent</artifactId>

<version>2.0.1.RELEASE</version> </parent> -->

<!--</profile></profiles> -->

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>

<java.version>1.8</java.version>

<shiro.version>1.4.0</shiro.version>

<kaptcha.version>2.3.2</kaptcha.version>

<ehcache.version>3.3.1</ehcache.version>

<beetl.version>2.9.3</beetl.version>

<swagger.version>2.9.1</swagger.version>

<ehcache.core.version>2.6.11</ehcache.core.version>

<mysql-connector-java.version>8.0.11</mysql-connector-java.version>

<jwt.version>0.9.0</jwt.version>

</properties>

<dependencies>

</dependencies>

<build>

<!-- war filename -->

<finalName>admin</finalName>

<!-- war position def positon is : <directory>0warout\adminWarOut</directory> -->

<!-- eclipse out class dir is tt-admin/target/classes -->

<sourceDirectory>srcnone</sourceDirectory>

<plugins>

<plugin>

<artifactId>maven-war-plugin</artifactId>

<version>2.6</version>

<configuration>

<!--如果想在没有web.xml文件的情况下构建WAR，请设置为false。 -->

<failOnMissingWebXml>false</failOnMissingWebXml>

</configuration>

</plugin>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.1</version>

<configuration>

<source>${java.version}</source>

<target>${java.version}</target>

</configuration>

</plugin>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

<configuration>

<fork>true</fork><!-- 如果没有该项配置，肯呢个devtools不会起作用，即应用不会restart -->

</configuration>

</plugin>

</plugins>

<resources>

<!-- -->

<resource>

<!-- 指定resources插件处理哪个目录下的资源文件 -->

<directory>src/main/webapp</directory>

<!--注意此次必须要放在此目录下才能被访问到 rlt dir is rlt classdir -->

<targetPath>${basedir}/target/admin</targetPath>

<includes>

<include>\*\*/\*\*</include>

</includes>

</resource>

<resource>

<!-- 指定resources插件处理哪个目录下的资源文件 -->

<directory>${basedir}/target/classes</directory>

<!--注意此次必须要放在此目录下才能被访问到 -->

<targetPath>${basedir}/target/admin/WEB-INF/classes</targetPath>

<includes>

<include>\*\*/\*\*</include>

</includes>

</resource>

<resource>

<!-- 指定resources插件处理哪个目录下的资源文件 -->

<directory>lib</directory>

<!--注意此次必须要放在此目录下才能被访问到 -->

<targetPath>${basedir}/target/admin/WEB-INF/lib</targetPath>

<includes>

<include>\*\*/\*\*</include>

</includes>

</resource>

<!-- jeig only for pom compiler..if eclise the res is ide def copy -->

<resource>

<directory>src/main/resources</directory>

<filtering>true</filtering>

</resource>

<resource>

<directory>src/main/java</directory>

<includes>

<include>\*\*/\*.xml</include>

</includes>

</resource>

</resources>

</build>

<profiles>

<profile>

<id>local</id>

<properties>

<spring.active>local</spring.active>

</properties>

<activation>

<activeByDefault>true</activeByDefault>

</activation>

</profile>

<profile>

<id>dev</id>

<properties>

<spring.active>dev</spring.active>

</properties>

</profile>

<profile>

<id>test</id>

<properties>

<spring.active>test</spring.active>

</properties>

</profile>

<profile>

<id>produce</id>

<properties>

<spring.active>produce</spring.active>

</properties>

</profile>

</profiles>

</project>