**[Dreamer CMS（梦想家CMS内容管理系统）](https://gitee.com/isoftforce/dreamer_cms" \t "_blank)**

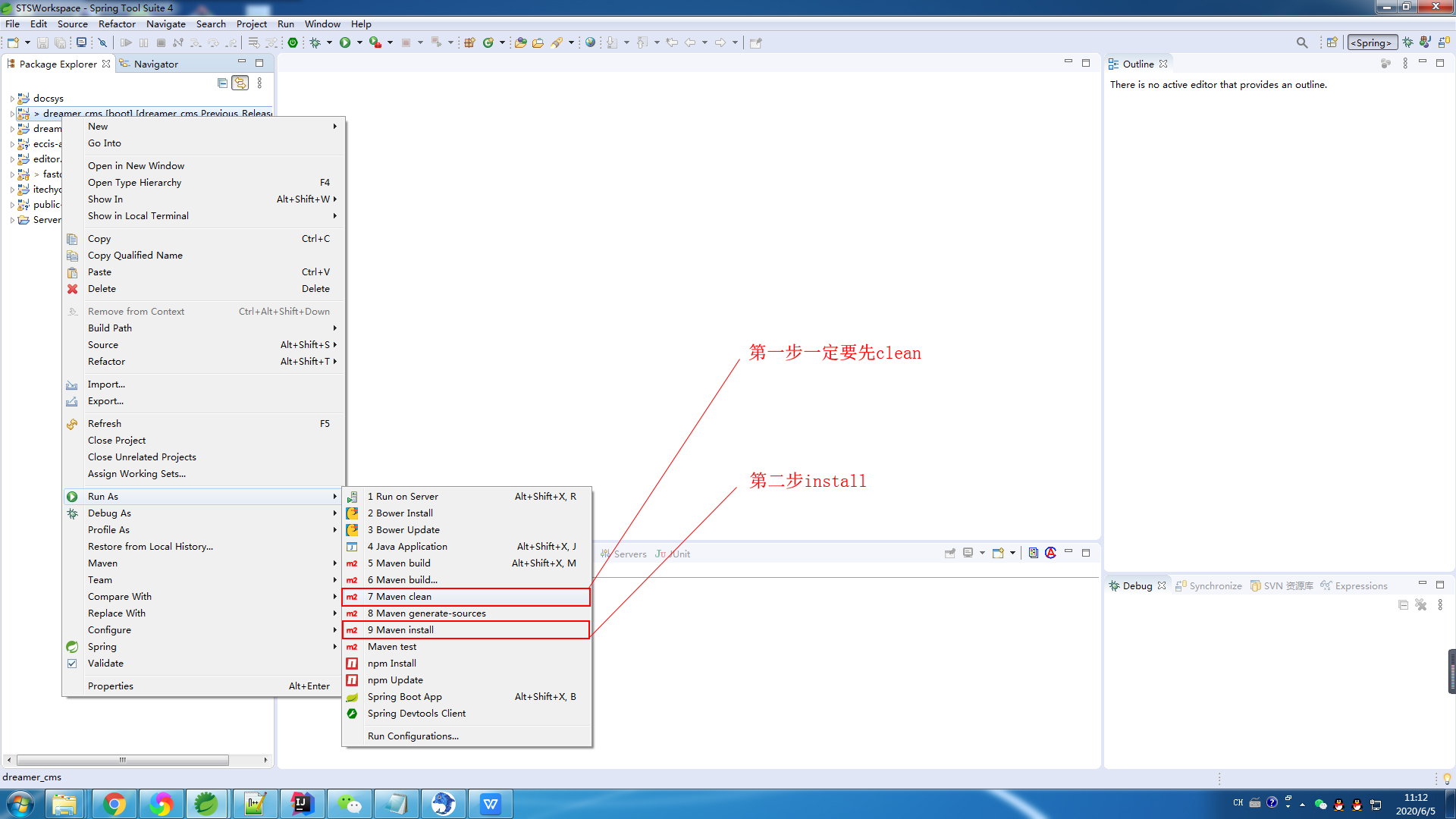
**部署手册**

## **环境准备**

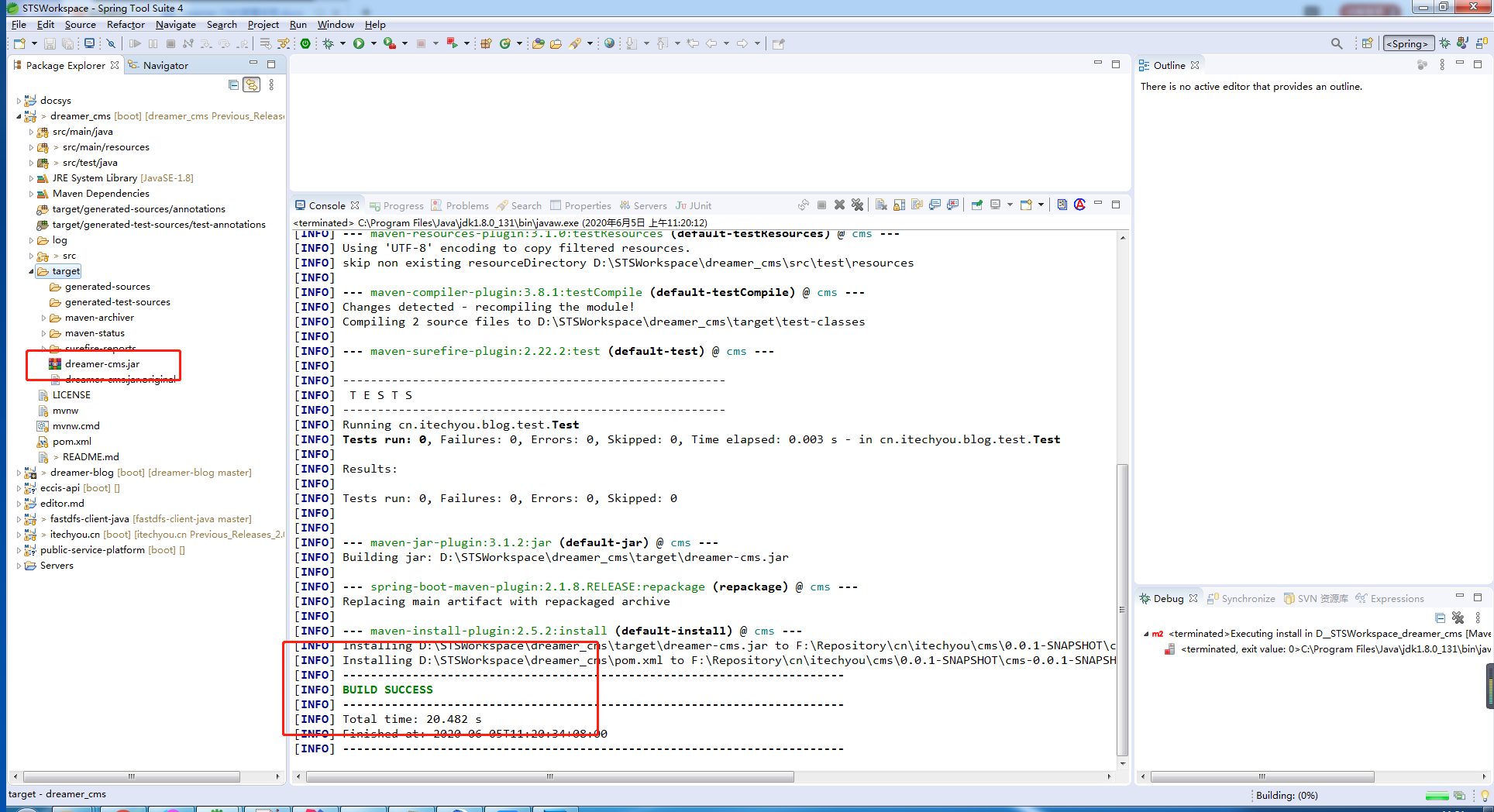
1. Redis
2. Tomcat8+
3. Jdk1.8+
4. Mysql5.5+(这里需要注意一下，Mysql5.7以上版本需要注意timestamp格式的字段设置为非必填)
5. IDE(Sprint tools suite4 或 Idea)

## **Jar包部署**

1. 在IDE中将项目中的配置文件都修改好，然后执行maven打包命令，打包前强烈建议先clean。在Eclipse或Sprint tools suite4中打包，Console输出Success表示成功。如下图：

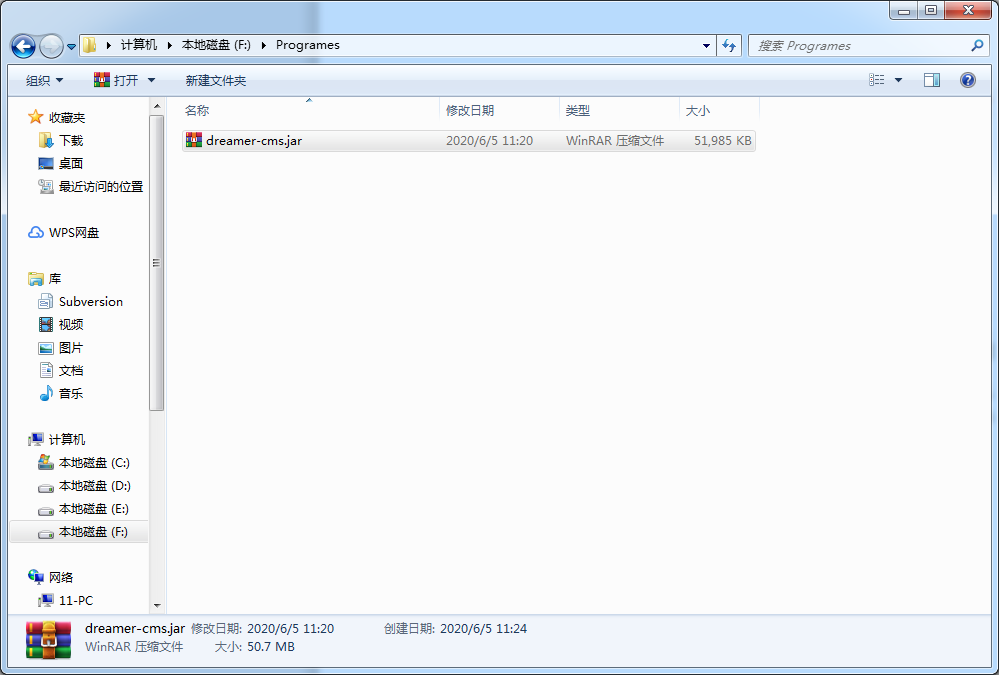


1. 打包完成后在项目目录下的target目录下会生成jar包，如下图

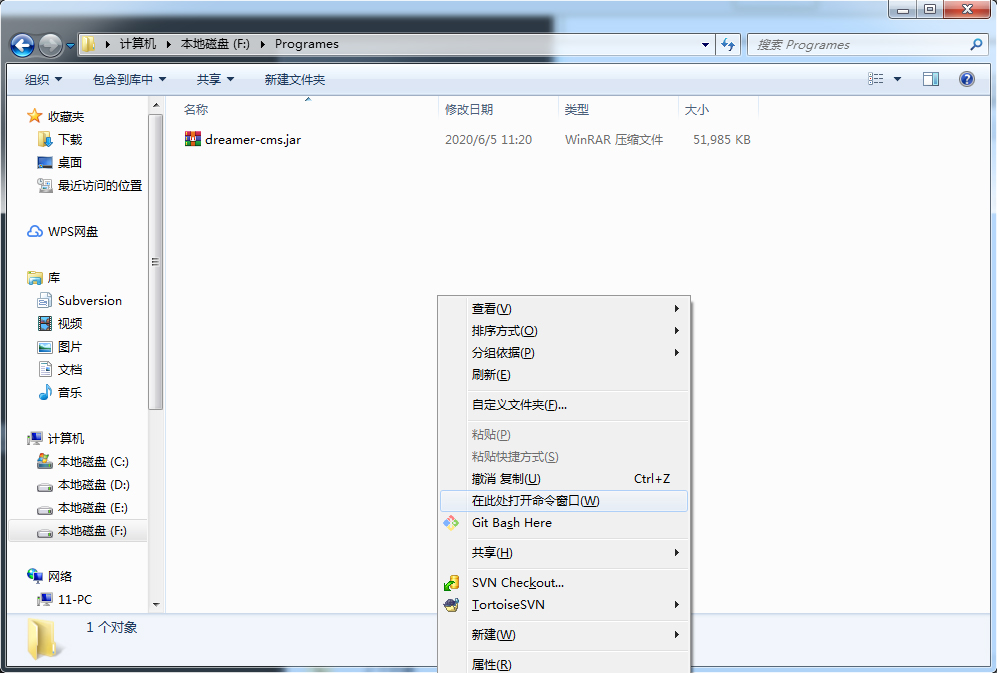


### Windows部署

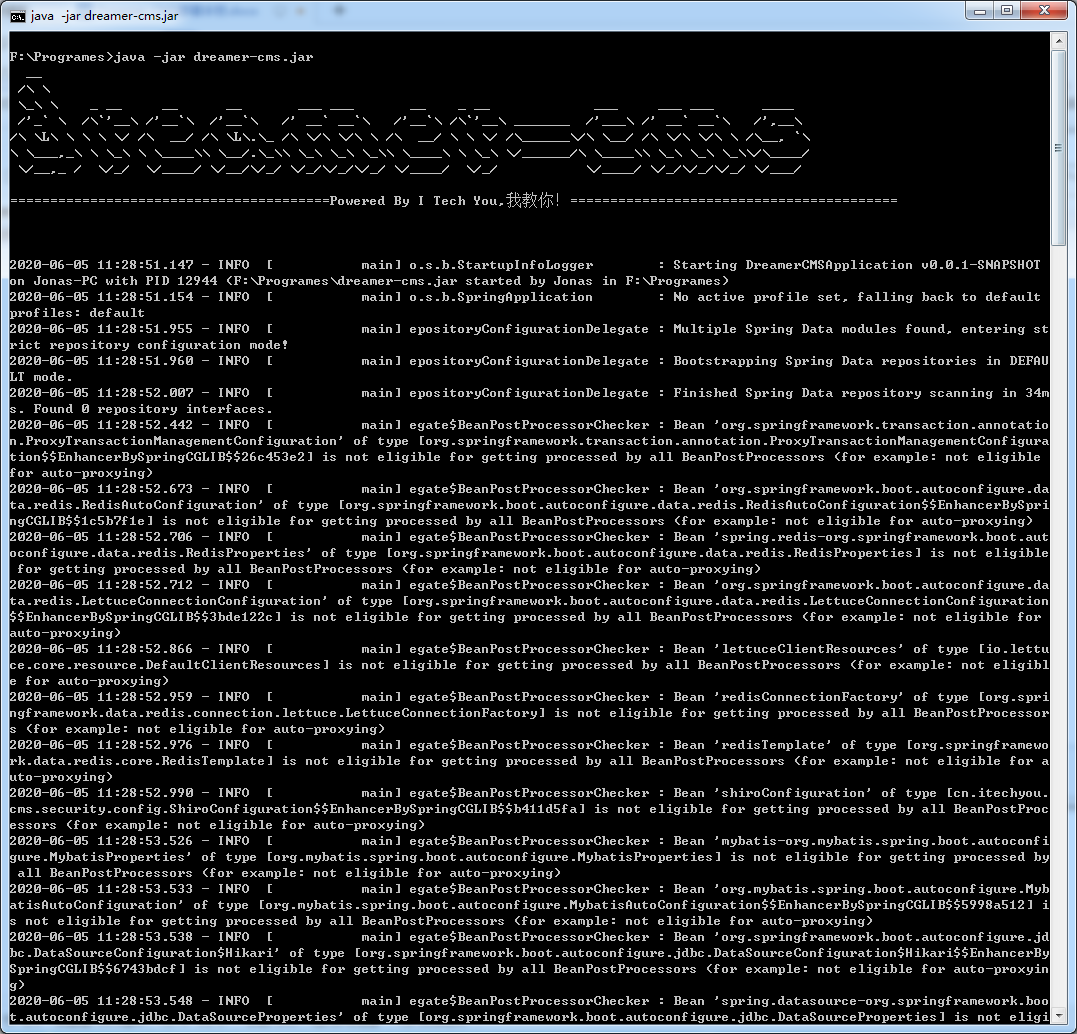
1. 将Jar文件复制到任意目录，建议不要在C盘，这里放在F:\Programes\目录下，如下图



1. 在该目录按住Shift键点击鼠标右键，点击在此处打开命令窗口，如下图



1. 在打开的命令行窗口中执行java -jar dreamer-cms.jar（如更换包名则对应修改此命令），启动如下图



1. 运行过程中，不要关闭该命令行窗口，到此Windows部署完成，这里需要注意的是资源路径一定要正确。

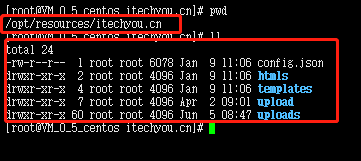
### Linux部署

1. 登录XShell或CRT到Linux系统，建议在/opt下进行部署操作，在/opt目录下新建如下目录：

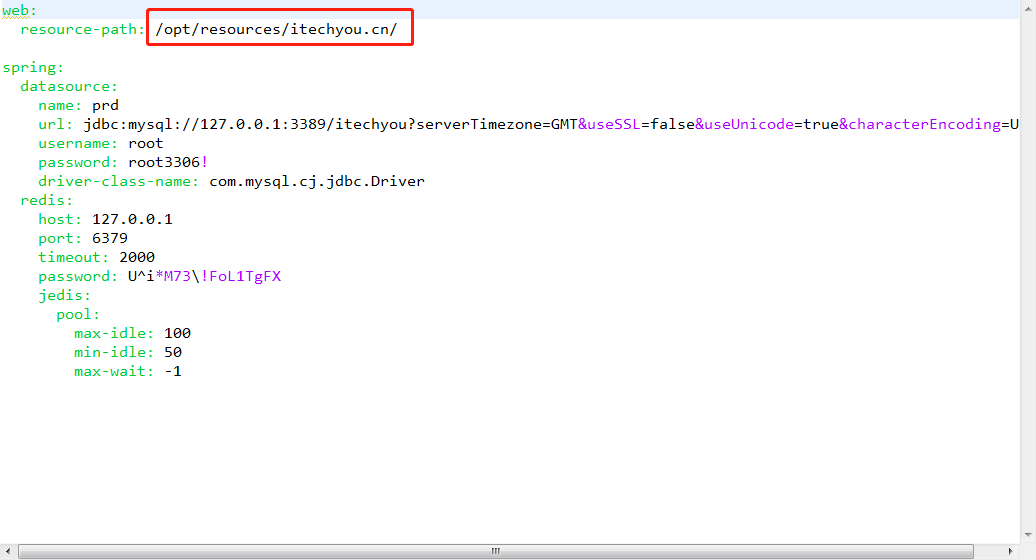
mkdir programs

mkdir resources

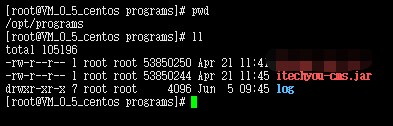
1. 将资源文件上传到resources/站点名/目录下，如下图



1. 打包时将application-对应环境.yml中的web.resource-path修改为/opt/resources/itechyou.cn/，如图



1. 将打好的Jar包上传到/opt/programs/目录下，如图



1. Linux运行Jar一定要在后台运行，否则关闭XShell或CRT程序会停止，这里可以使用Nohup或Screen。Nohup一般的Linux都自带，如没有请参考：

<https://blog.csdn.net/weixin_43926604/article/details/96001694>

Screen使用教程请参考：

<https://blog.csdn.net/han0373/article/details/81352663>

1. 本教程使用Screen运行，安装Screen，命令如下：

yum install screen

1. 运行Jar文件

**首次运行：**需要使用Screen新建一个任务，命令如下

screen -S 任务名（这里建议站点名）

在此命令成功后，会启一个新的命令行界面，然后执行java -jar dreamer-cms.jar（如更换包名则对应修改此命令）命令即可，启动成功后退出Screen，退出一定要Ctrl + a + d

**如果非首次运行：**则需要先使用命令screen -r 任务名（填写新建任务时指定的名字）

screen -r itechyou.cn

## War包部署

war打包，需要把Spring Boot自带的Tomcat排除掉，具体操作步骤如下

1. 添加servlet-api依赖

<!--添加servlet-api的依赖,用来打war包 -->

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<scope>provided</scope>

</dependency>

1. 排除SpringBoot内置的Tomcat

<!--最终打成war包，排除内置的tomcat-->

<dependency>

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

<artifactId>spring-boot-starter-web</artifactId>

<exclusions>

<exclusion>

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

<artifactId>spring-boot-starter-tomcat</artifactId>

</exclusion>

</exclusions>

</dependency>

1. 改造启动类

修改前：

@EnableTransactionManagement

@SpringBootApplication

@MapperScan("cn.itechyou.cms.dao")

public class DreamerCMSApplication{

public static void main(String[] args) {

SpringApplication.run(DreamerCMSApplication.class,args);

}

}

修改后：

@EnableTransactionManagement

@SpringBootApplication

@MapperScan("cn.itechyou.cms.dao")

public class DreamerCMSApplication extends SpringBootServletInitializer {

public static void main(String[] args) {

SpringApplication.run(DreamerCMSApplication.class,args);

}

@Override

protected SpringApplicationBuilder configure(SpringApplicationBuilder builder) {

return builder.sources(DreamerCMSApplication.class);

}

}

1. 在pom.xml中添加maven编译插件

<build>

<plugins>

<plugin>

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

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

</plugin>

</plugins>

</build>

1. 修改pom.xml打包方式为war

<packaging>war</packaging>

1. 然后执行打包命令（还是先clean再install）打包成功后在项目目录下的target目录下会生成war包。

### Windows部署

下载Tomcat解压，将war包复制到webapps目录下，启动tomcat即可。

### Linux部署

下载Tomcat解压，将war包上传到webapps目录下，启动tomcat即可。