

JEECMS部署War包文档手册

基于JEECMS X1.1版编写 更新时间 2019 年  12月 2日

JEECMS部署War包文档

目录

[JEECMS部署War包文档 1](#_Toc25917020)

[目录 1](#_Toc25917021)

[一． Eclipse创建War包 1](#_Toc25917022)

[二． IDEA创建War包 5](#_Toc25917023)

[三． Tomcat 启动War (Windows版本) 9](#_Toc25917024)

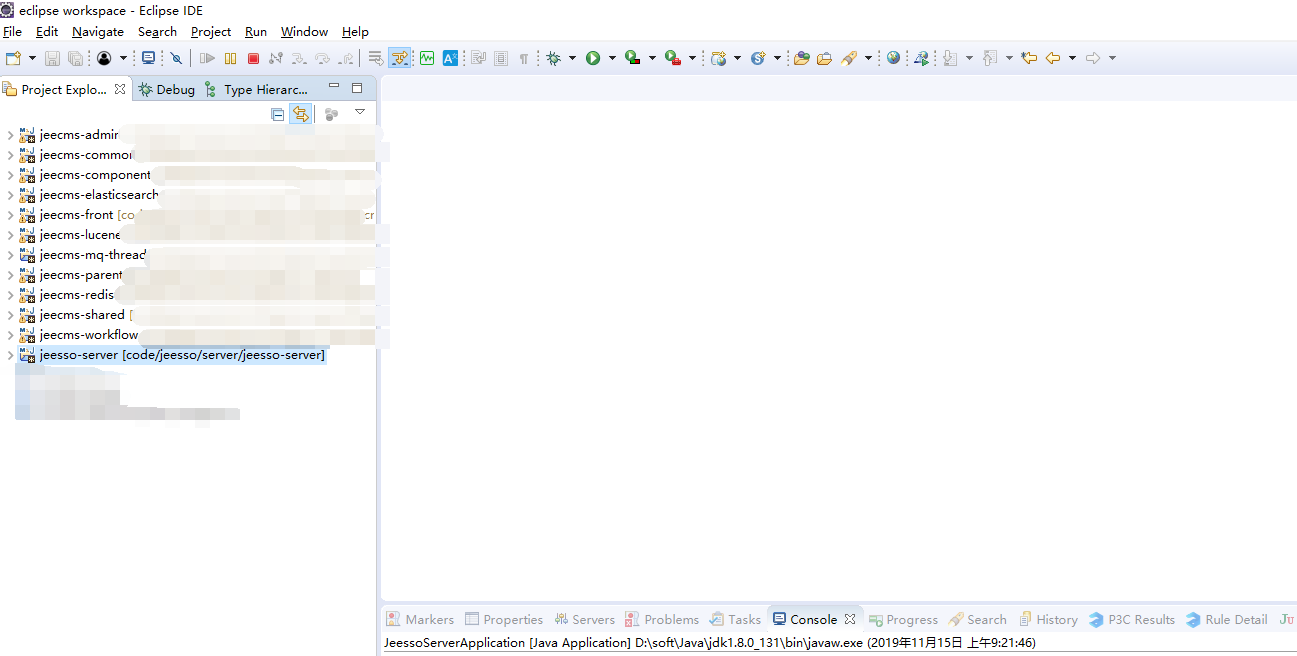
[四． Tomcat 启动War (Linux CentOS版本) 13](#_Toc25917025)

[五． Nginx 配置 15](#_Toc25917026)

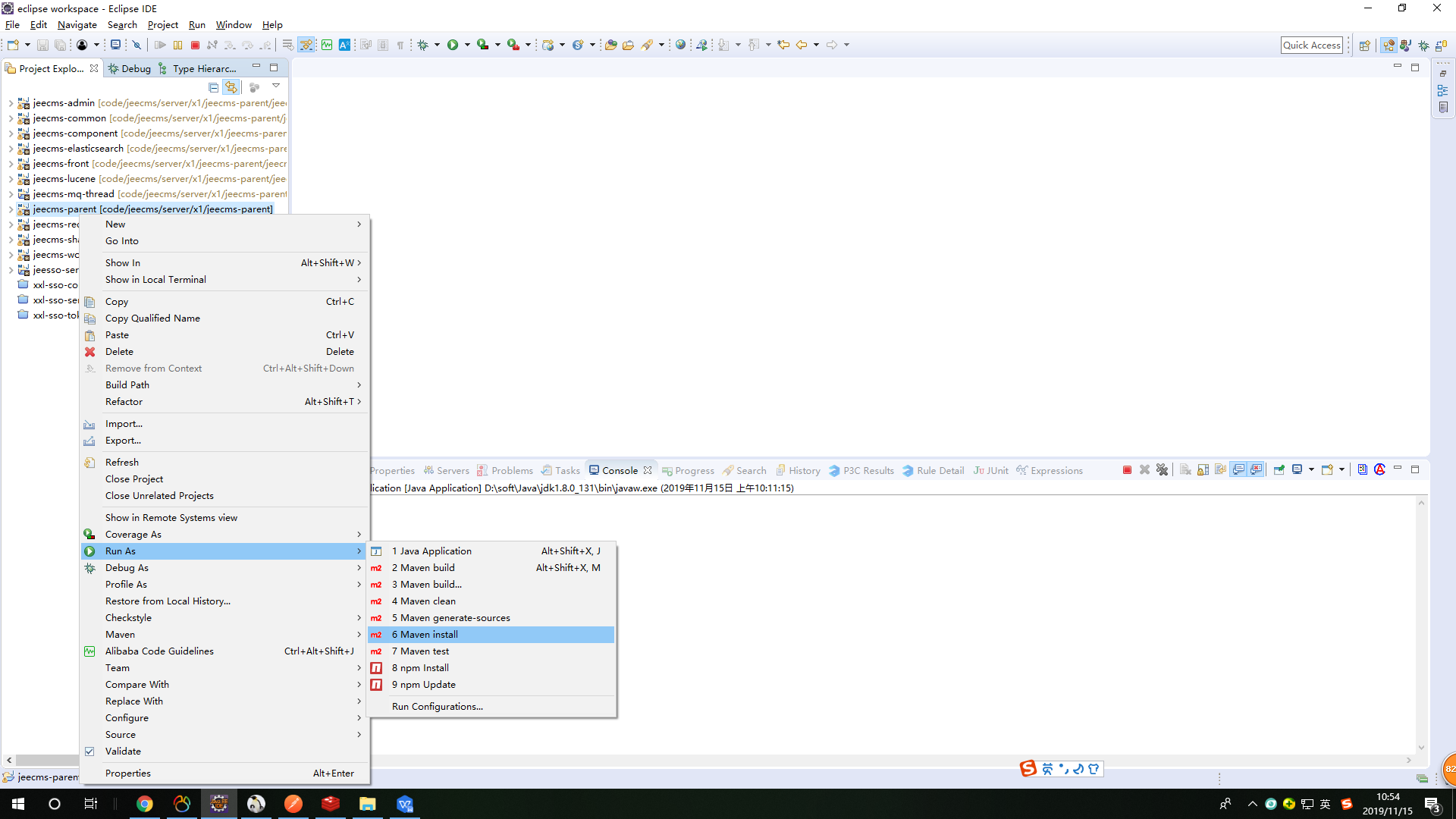
1. Eclipse创建War包

**方式一：Eclipse打包部署**

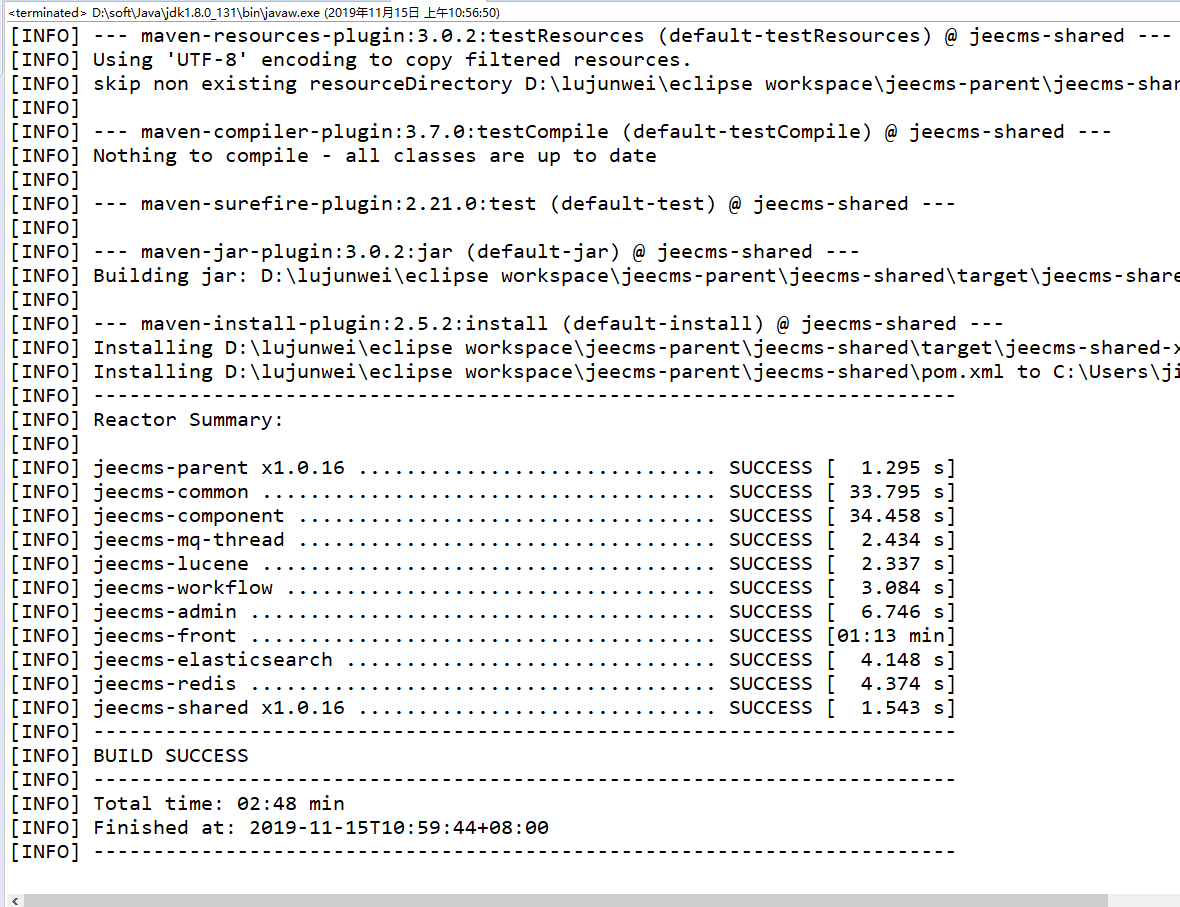
1. 打开Eclipse 界面（前提是已经导入源码）



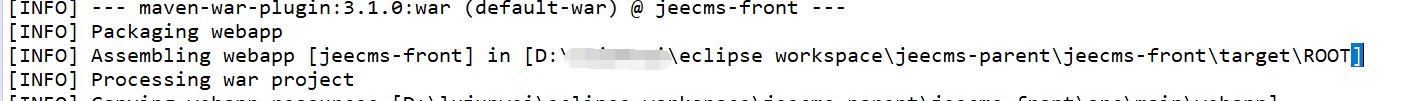
1. 选择parent工程，右键选择Run As - Maven install;

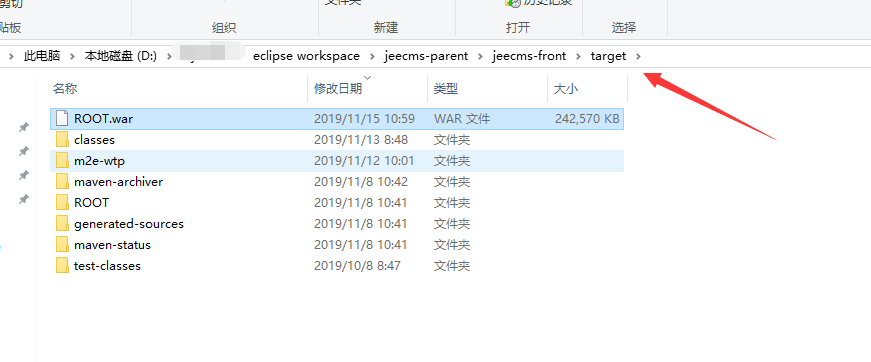


3. 如图所示，打包成功了



1. 查找路径，打包为Front工程下的target目录下，如下图所示

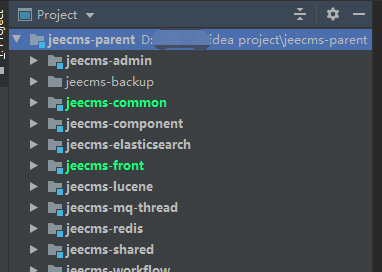




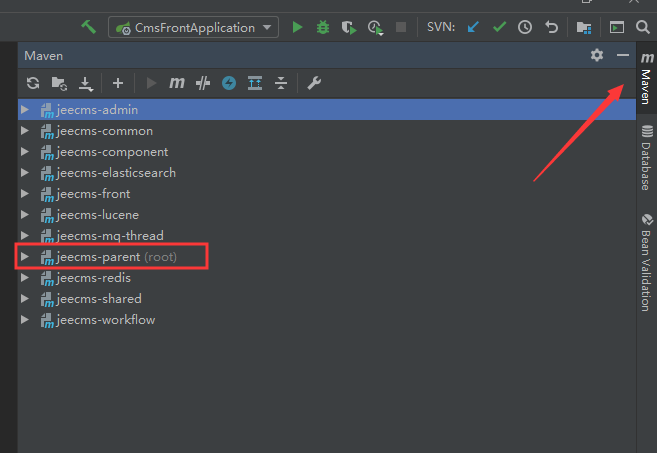
1. 打包成功如上图所示
2. IDEA创建War包

**方式一：直接运行Maven Package 打包命令**

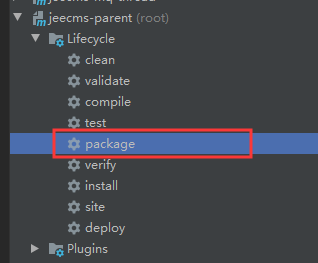
1. 打开IDEA编辑器，如图所示，此时已导入项目



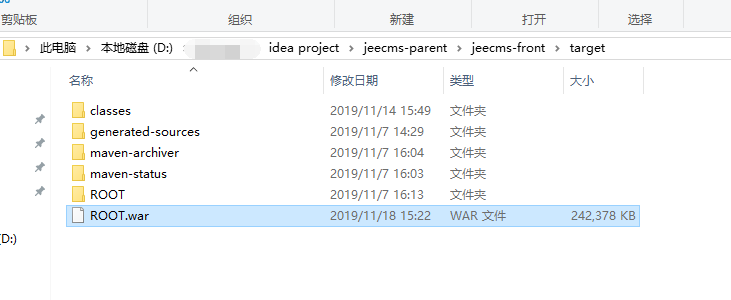
1. 选择编辑器最右上脚的Maven选项，选择jeecms-parent工程



1. 选择package 打包命令，直接双击运行

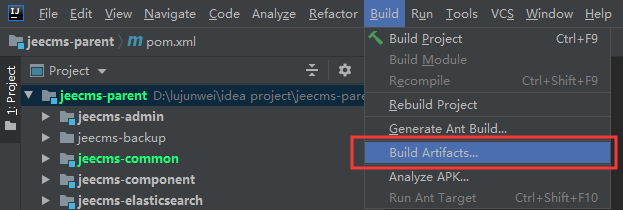


1. 打包成功，如下图所示

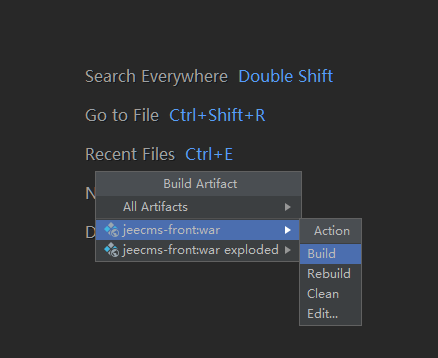


**方式二：使用IDEA本地打包命令**

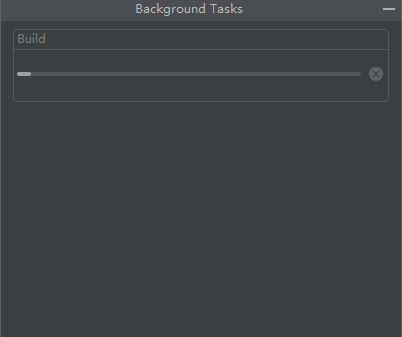
1. 选择Build - Build Artifacts



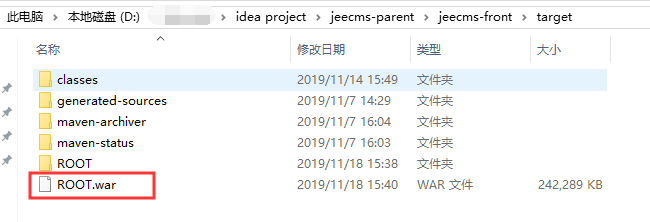
1. 出现如下界面，选择jeecms-front:war ---选择Build



1. 打包进行中...

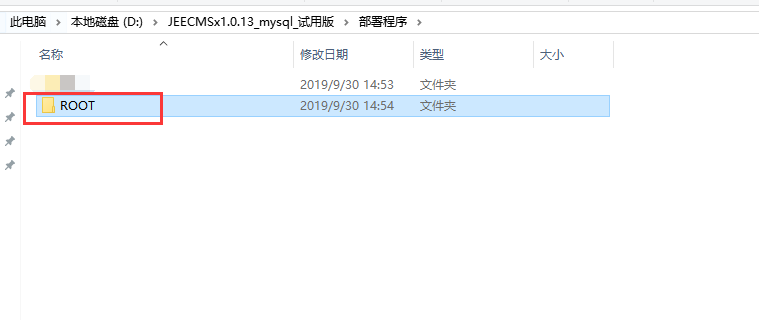


1. 打包成功，如下图所示

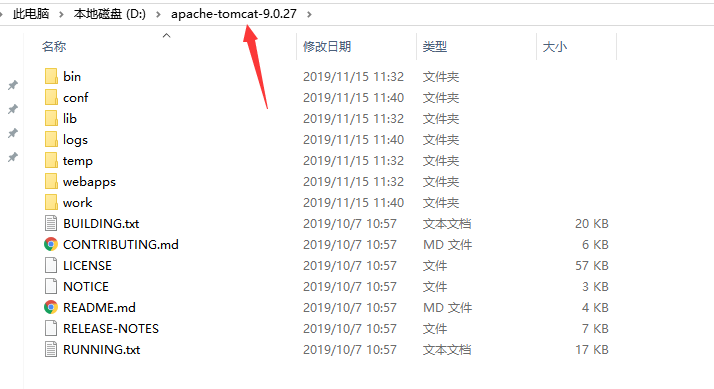


1. Tomcat 启动War (Windows版本)
2. 从官网下载程序，得到ROOT包，

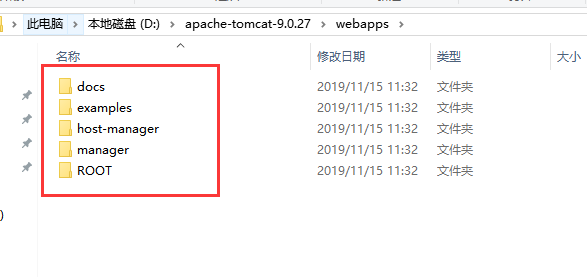
官网下载程序地址：<http://bbs.jeecms.com/fabu/80838.jhtml>



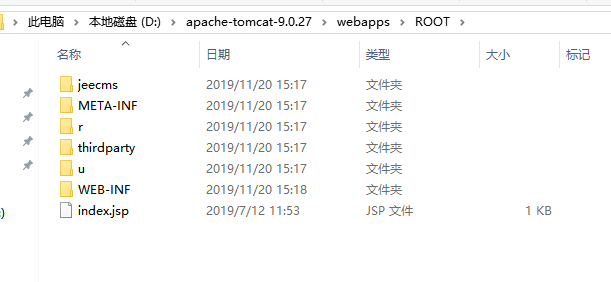
1. 打开之前下载的Tomcat服务器数据包，如下图所示；



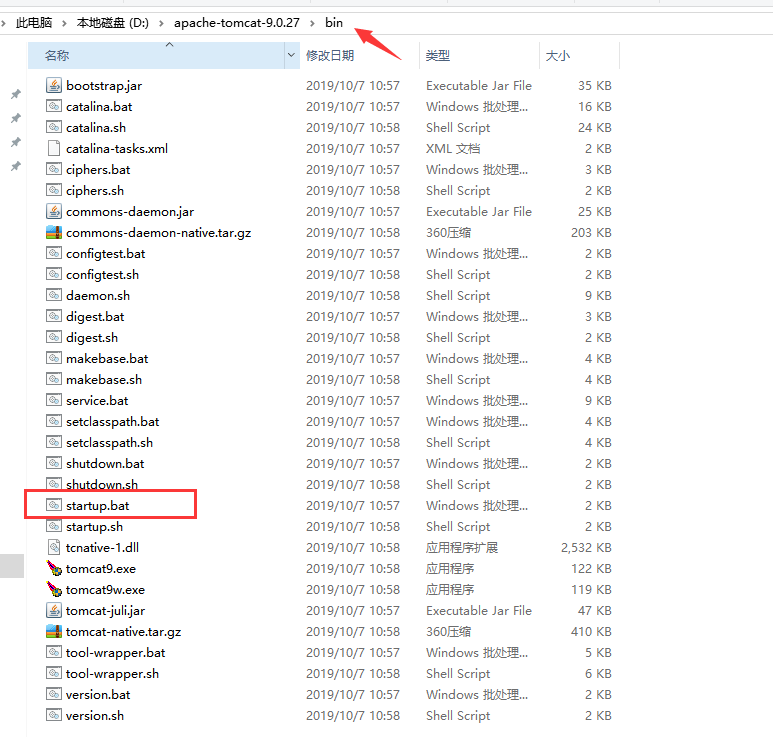
1. 将之前的ROOT包放到webapps目录文件下，覆盖ROOT文件夹下的文件；



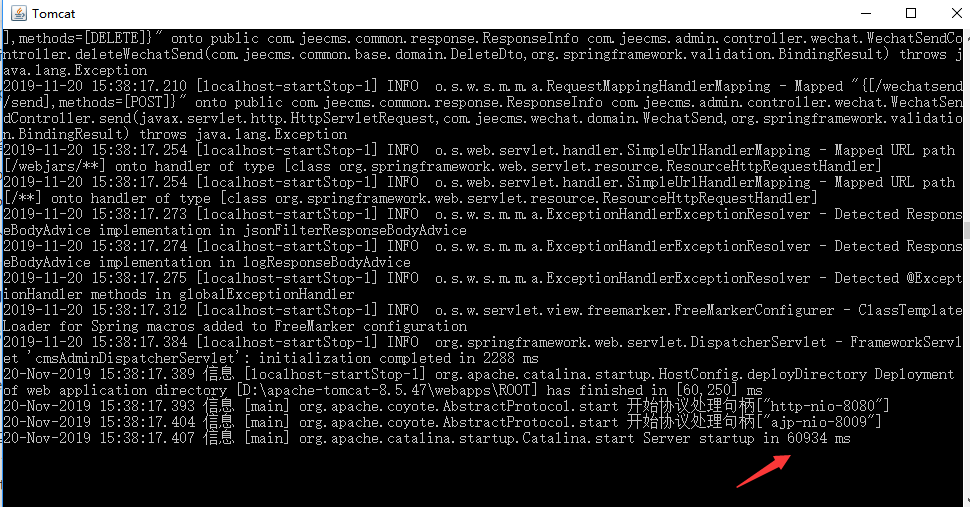
1. 得到如下文件所示



1. 打开Tomcat文件bin目录，双击startup.bat运行

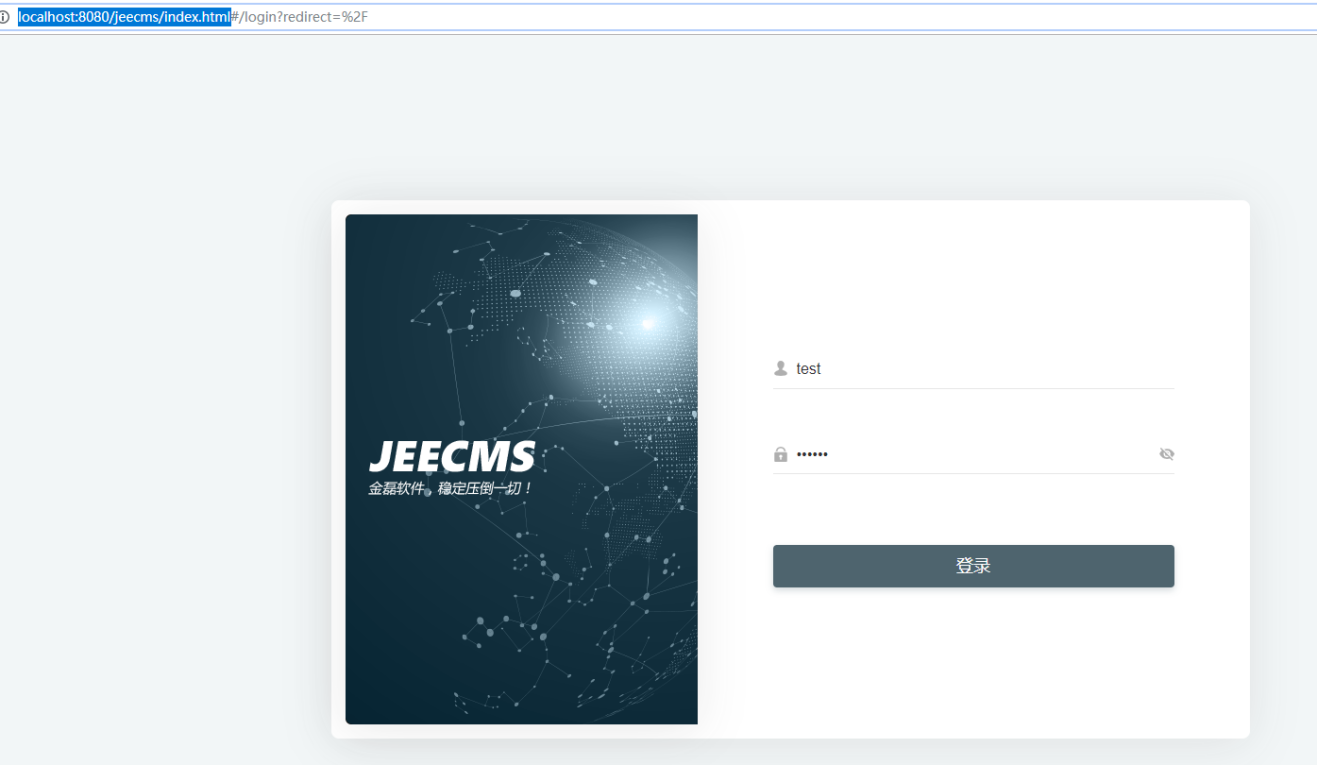


1. 启动成功

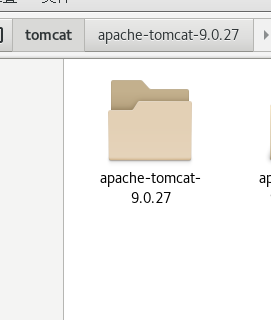


1. 浏览器输入地址：<http://localhost:8080/jeecms/index.html>

得到如下登录界面，部署成功



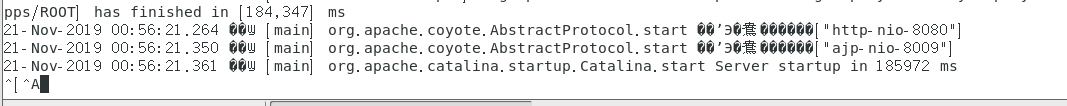
1. Tomcat 启动War (Linux CentOS版本)
2. 找到之前在Linux 下载好的Tomcat 数据包



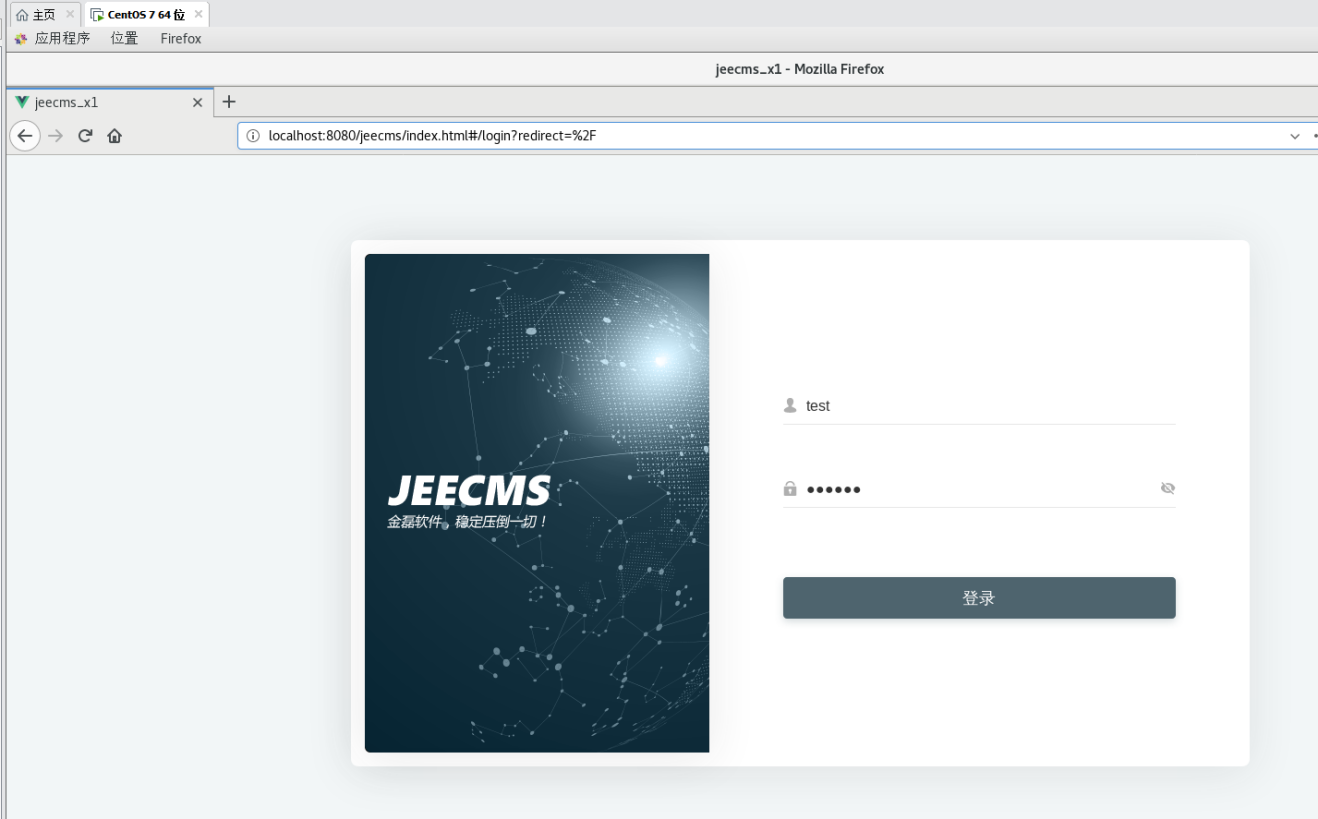
1. 还是将之前的ROOT包放到Webapps下面，覆盖之前的ROOT包，这里可以使用Xshell工具或者使用Xftp工具传输，笔者这里使用XFtp传输过去，得到如下数据，如图所示；



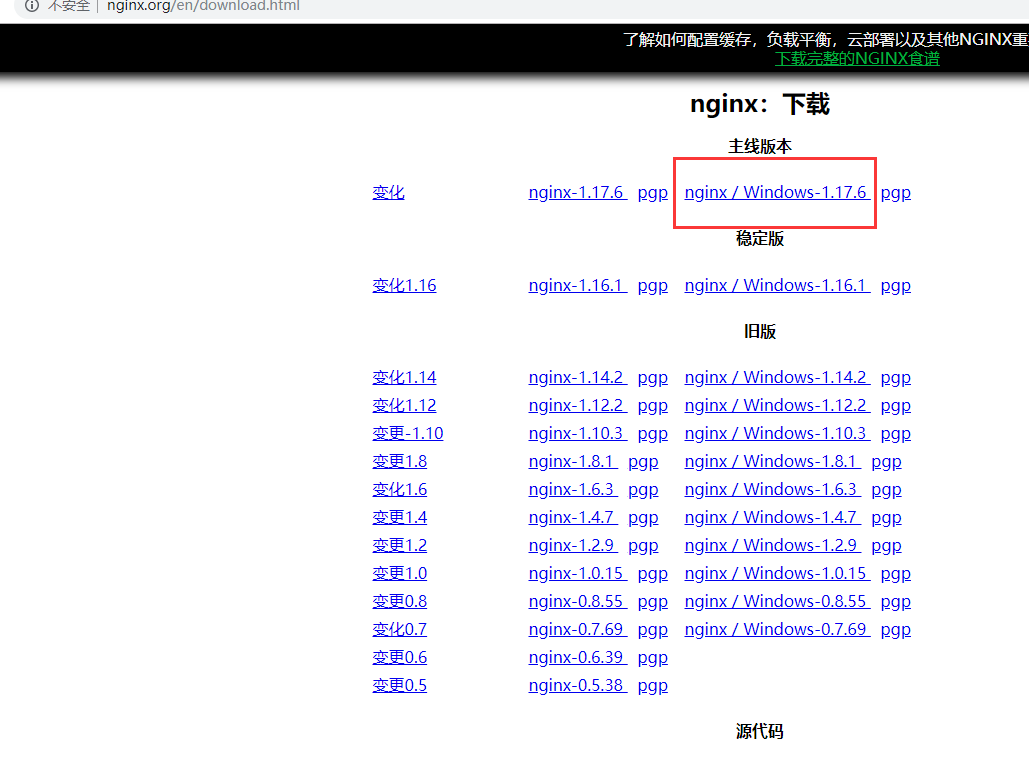
1. 进入bin 目录，启动命令 startup.sh 文件，启动成功，如下图所示，部署在8080端口；



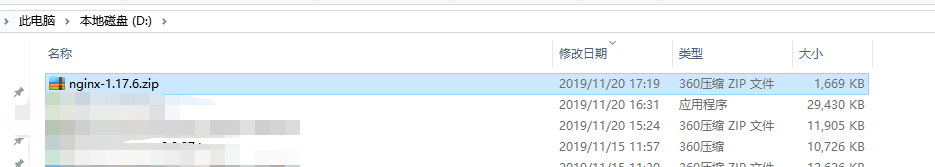
1. 验证是否启动成功，输入如下地址：<http://localhost:8080/jeecms/index.html>



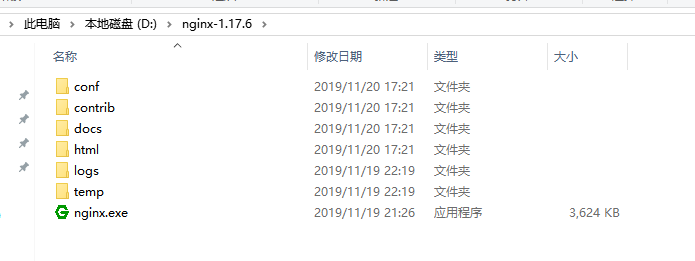
1. Nginx 配置
2. 安装Nginx，安装地址：<http://nginx.org/en/download.html>



1. 选择Windows 版本，单击下载，得到如下图所示；



1. 解压到当前文件夹，得到如下文件目录；



1. 找到 conf 目录里的 nginx.conf 文件，配置Nginx，核心配置如下：

#user nobody;

#指定nginx进程数

worker\_processes 1;

events {

# 连接数上限

worker\_connections 1024;

}

#设定http服务器，利用它的反向代理功能提供负载均衡支持

http {

#设定mime类型,类型由mime.type文件定义

include mime.types;

default\_type application/octet-stream;

#开启高效文件传输模式，sendfile指令指定nginx是否调用sendfile函数来输出文件，对于 #普通应用设为 on，如果用来进行下载等应用磁盘IO重负载应用，可设置为off，以平衡磁 #盘与网络I/O处理速度，降低系统的负载。注意：如果图片显示不正常把这个改成off。

sendfile on;

#连接超时时间

#keepalive\_timeout 0;

#开启gzip压缩 ，压缩html

#gzip on;

#服务器的集群， #服务器集群名字

upstream  netitcast.com {

#服务器配置   weight是权重的意思，权重越大，分配的 概率越大。

  server    127.0.0.1:18080  weight=1;

server    127.0.0.1:28080  weight=2;

    }

#配置代理服务器的地址，即Nginx安装的服务器地址、监听端口、默认地址

#一个server相当于一个代理服务器，可以配置多个server

server {

#1.侦听80端口代表当前代理服务器的访问端口号，默认是80端口。如果要配置多个 # server，这里的默认端口需要改变，要不然系统不知道进入哪个代理服务。

listen 80;

#对于server\_name,如果需要将多个域名的请求进行反向代理，可以配置多个 #server\_name来满足要，多个域名用空格隔开

server\_name localhost;

# 字符集

#charset koi8-r;

#表示匹配客户端发送请求的路径，这里“/”代表所有请求的路径都能匹配。

location / {

# 默认主页目录在nginx安装目录的html子目录。

root html;

#如果代理没有指定主页，将默认进入index配置下寻找主页，可以配置多个，第 # 一个主页找不到，访问第二个，以此类推。

index index.html index.htm;

#这里配置的proxy\_pass属性表示代理路径，nginx可以直接转发到该链接。

proxy\_pass http://127.0.0.1:10001;

proxy\_set\_header Host $http\_host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

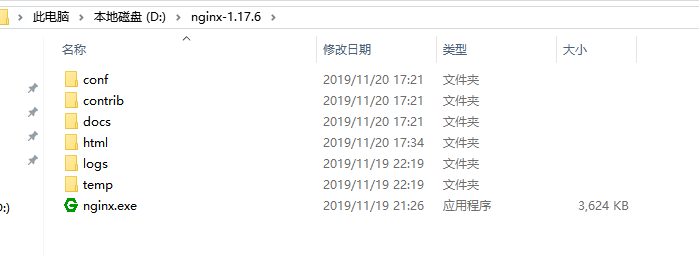
}

}

}

1. 启动Nginx

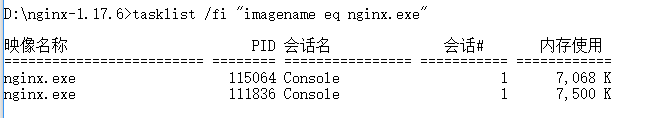
注意不要直接双击nginx.exe，这样会导致修改配置后重启、停止nginx无效，需要手动关闭任务管理器内的所有nginx进程



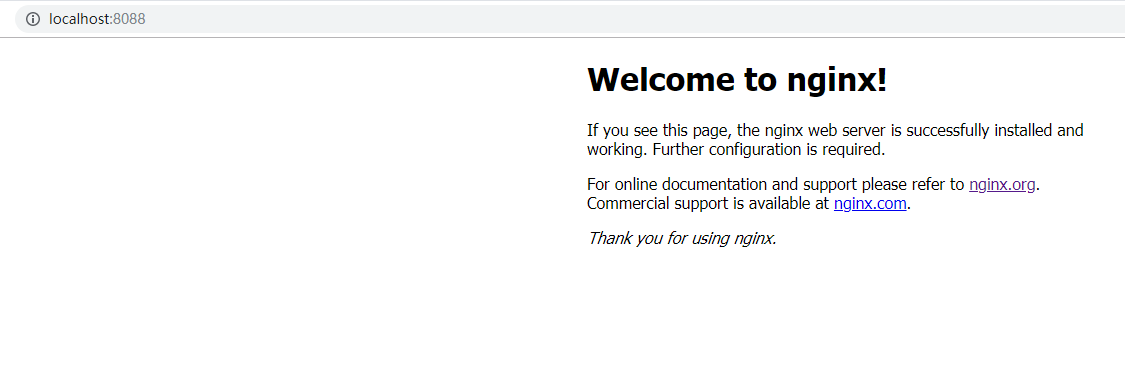
进入cmd 命令窗口，输入start nginx.exe，有一个一闪而过的效果！

查看任务进程是否存在，dos或打开任务管理器都行

tasklist /fi "imagename eq nginx.exe"



1. 访问 [http://localhost:80](http://localhost/)88，笔者改变了监听的端口，启动成功如图：



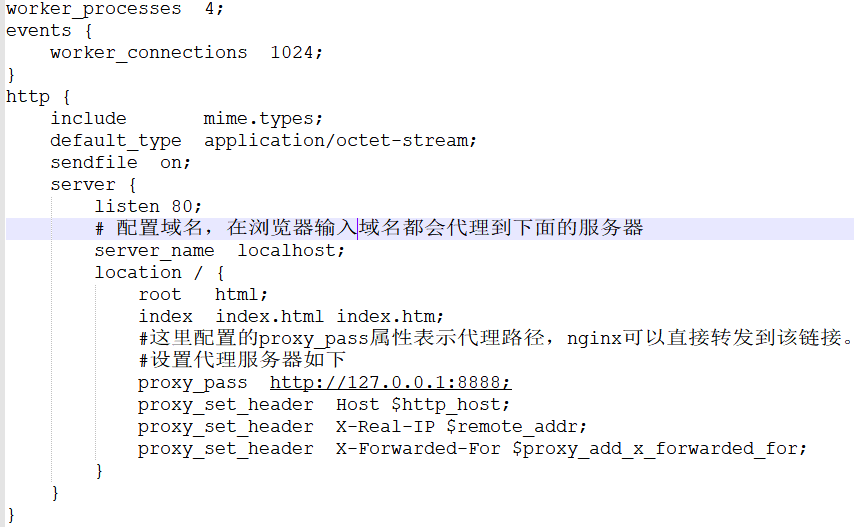
1. **Nginx + Tomcat 实现反向代理配置**

这里使用Nginx解析域名，做出域名到Tomcat的映射；

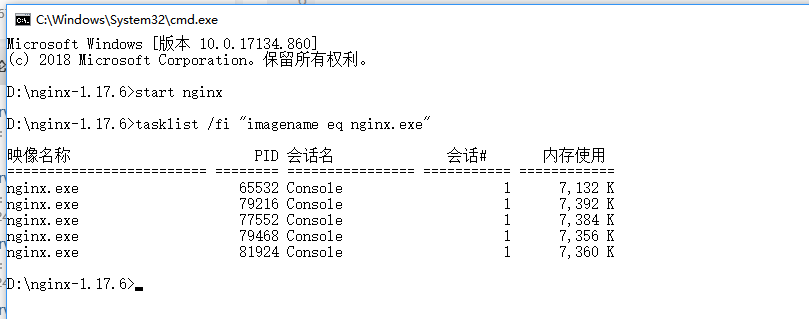
* 1. 首先我们启动一个Tomcat 服务器，部署在8888端口



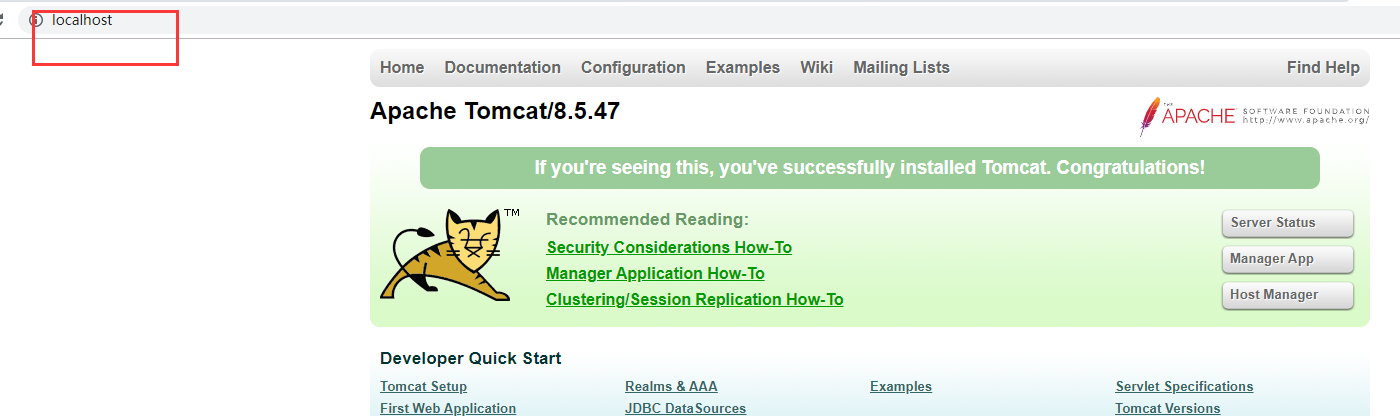
* 1. 配置Nginx.conf文件，将代理服务器域名写入进去，默认localhost,如下图所示；



* 1. 启动nginx.exe，按照之前的文档说明的方式启动；如下图所示，启动成功；

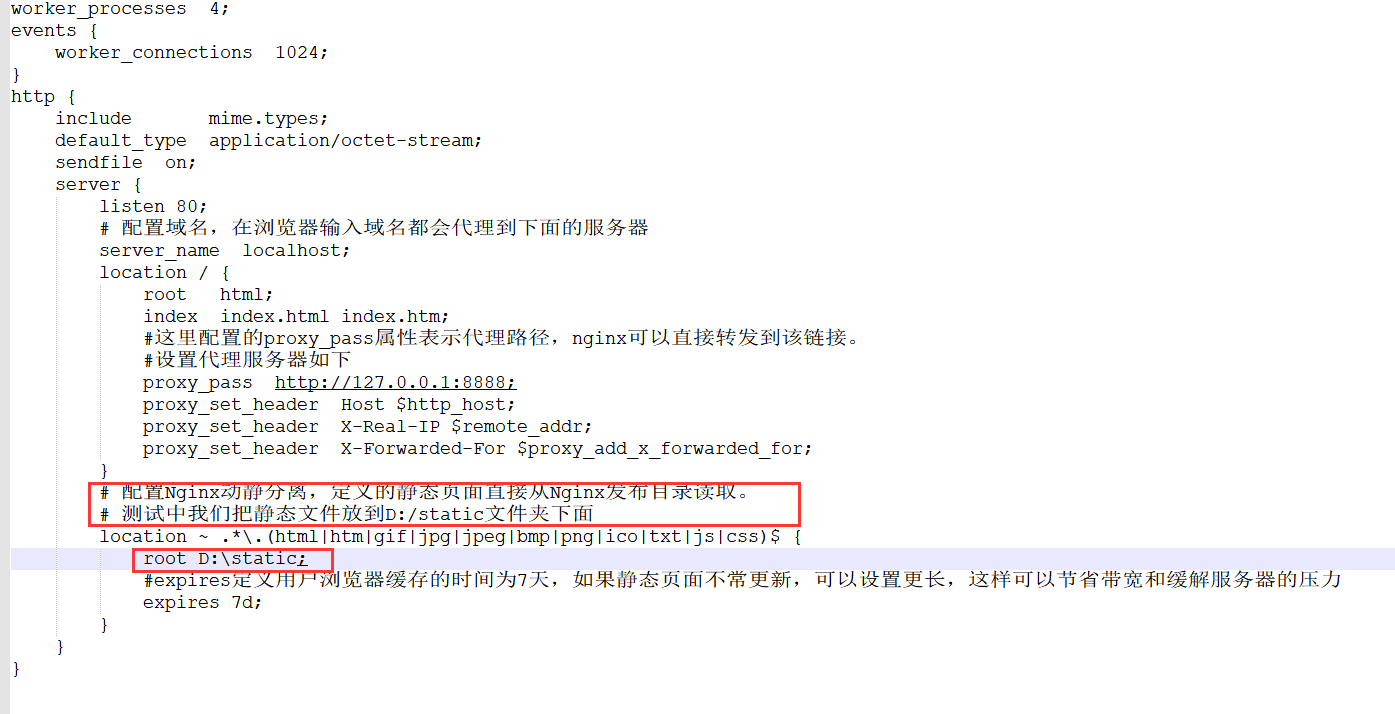


* 1. 在浏览器输入域名，就会代理到我们之前的Tomcat了

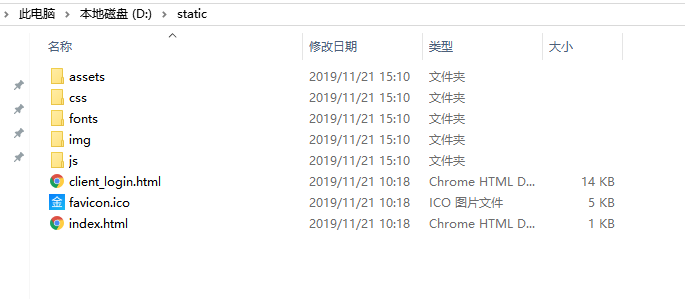


1. **Nginx + Tomcat 实现动静分离**
   1. 首先对于静态文件的请求，我们也新建一个location，将常见图片、css、js等请求匹配到该location中，在之前的基础上，如下图所示，注意看注释；
   2. 记得重启nginx的配置，nginx支持热部署，输入如下命令即可；



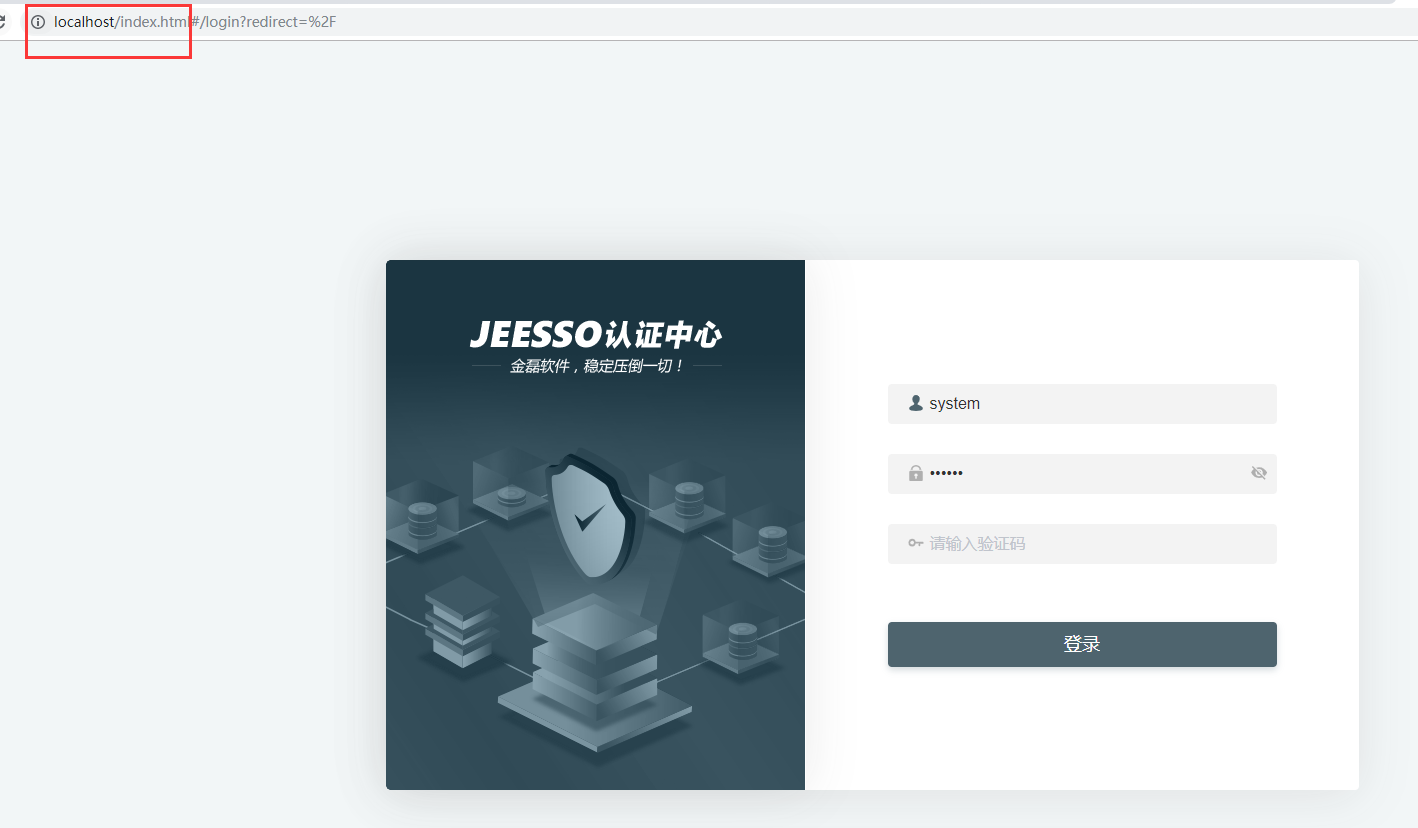


8.2 存放静态资源的路径如下

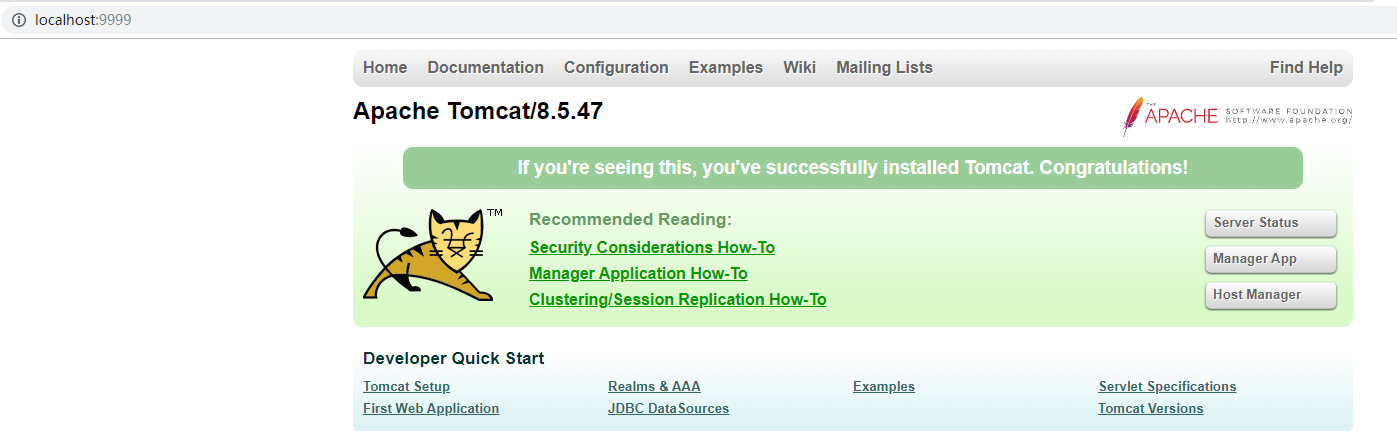


* 1. 校验结果，在浏览器输入：
  2. [http://localhost/index.html。](http://localhost/index.html。出现如下所示配置成功；)

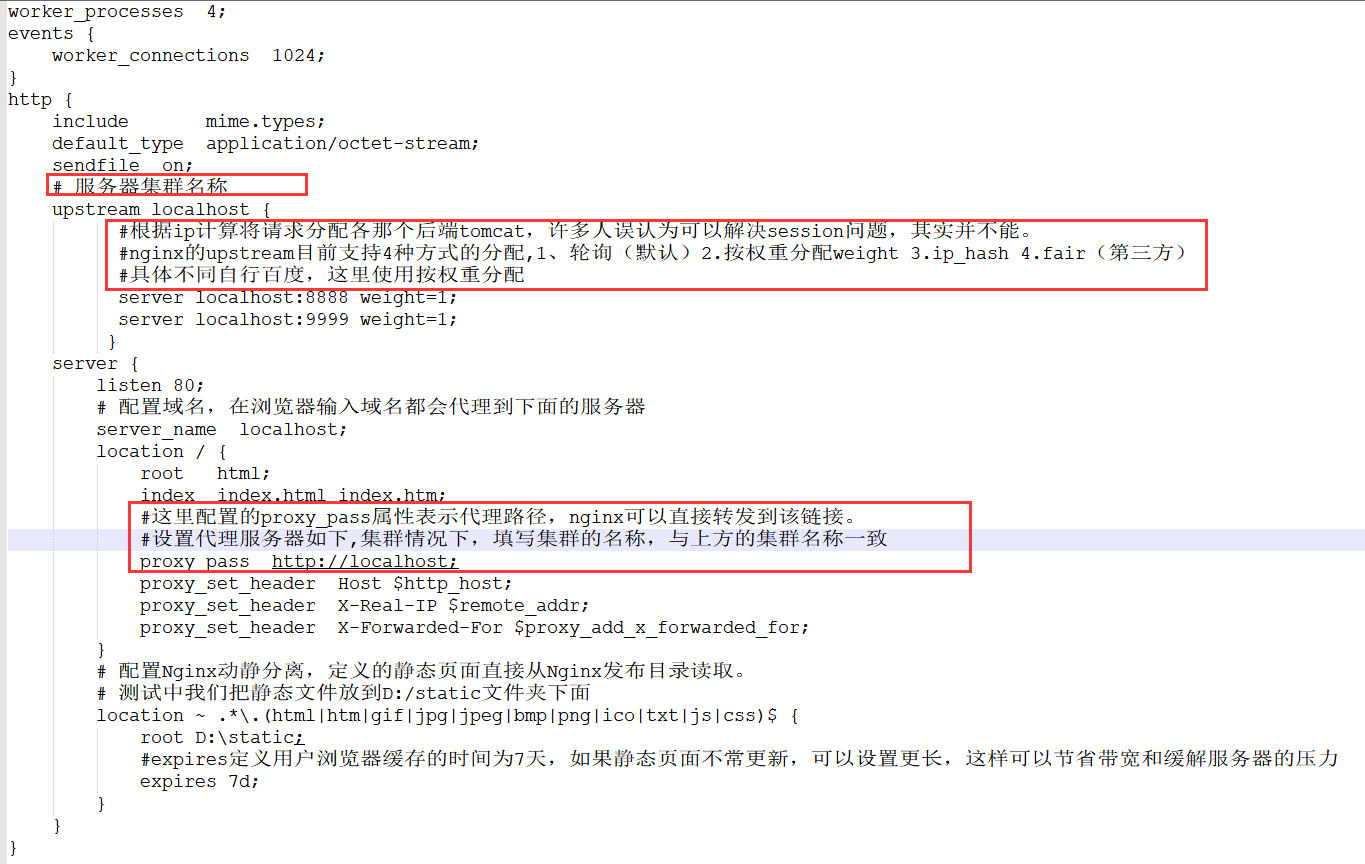
[出现如下所示配置成功；](http://localhost/index.html。出现如下所示配置成功；)



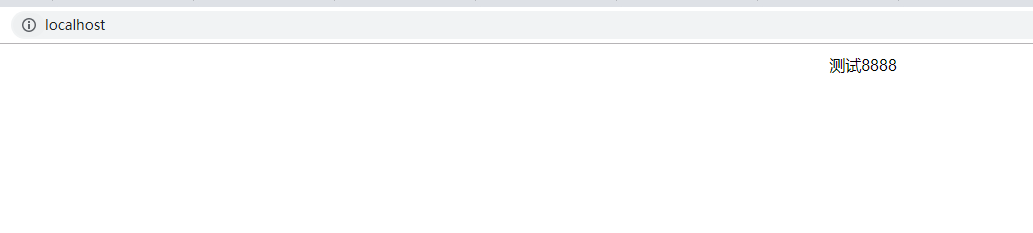
1. **Nginx + Tomcat 实现Tomcat集群配置负载均衡**
   1. 在之前的基础上，我们在开启一个Tomcat，端口部署在9999

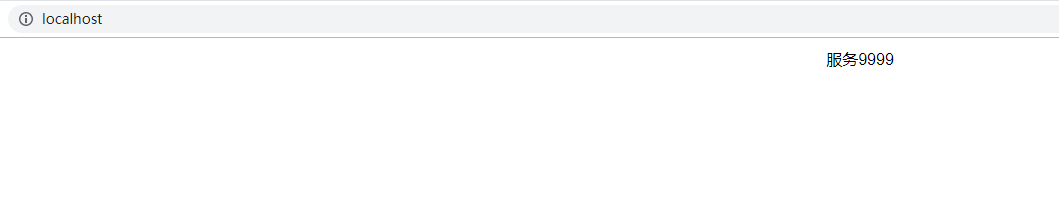


* 1. 配置nginx tomcat集群,如下图所示；注意看注释



* 1. 测试如下，同一个请求会分发到不同tomcat服务器





9.4 测试成功；