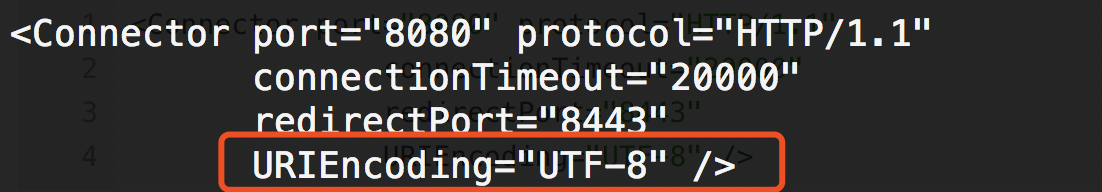
**[二。实战docker，编写Dockerfile定制tomcat镜像，实现web应用在线部署](https://blog.csdn.net/boling_cavalry/article/details/70184605" \t "_blank)**

参考博客源码地址：[https://github.com/zq2599/blog\_demos.git](https://github.com/zq2599/blog_demos.git" \t "_blank) 项目名：loadbalancedemo

Step1：创建空文件夹：mkdir image\_tomcat

Step2：拷备apache-tomcat-7.0.86/conf目录下的文件server.xml和tomcat-users.xml到image\_tomcat目录下面

Step3：修改server.xml



<Connector executor="tomcatThreadPool"

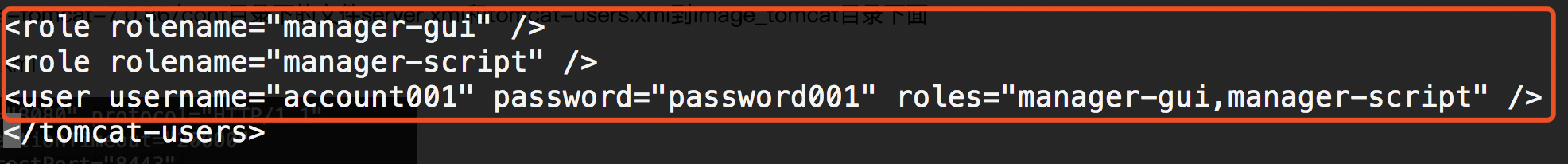
port="8080" protocol="HTTP/1.1"

connectionTimeout="20000"

redirectPort="8443" />

增加：URIEncoding="UTF-8" 目的：处理中文问题

Step4：修改tomcat-users.xml,增加如下内容：



<role rolename="manager-gui" />

<role rolename="manager-script" />

<user username="account001" password="password001" roles="manager-gui,manager-script" />

以上配置完成后，即可用account001账号在对应的tomcat上进行在线部署操作了.

Step5：定制Docker镜像

[复制代码](javascript:void(0);)

cd image\_tomcat

touch Dockerfile //创建Dockerfile文件

vim Dockerfile //编辑Dockerfile文件

//###start 添加如下内容

#First docker file from bolingcavalry  
#version 0.0.1  
#Author:shiyufeng  
#基础镜像  
FROM tomcat:7.0.77-jre8

#作者  
MAINTAINER shiyufeng

#定义工作目录  
ENV WORK\_PATH /usr/local/tomcat/conf

#定义要替换的文件名  
ENV USER\_CONF\_FILE\_NAME tomcat-users.xml

#定义要替换的server.xml文件名  
ENV SERVER\_CONF\_FILE\_NAME server.xml

#删除原文件tomcat-users.xml  
RUN rm $WORK\_PATH/$USER\_CONF\_FILE\_NAME

#复制文件tomcat-users.xml  
COPY ./$USER\_CONF\_FILE\_NAME $WORK\_PATH/

#删除原文件server.xml  
RUN rm $WORK\_PATH/$SERVER\_CONF\_FILE\_NAME

#复制文件server.xml  
COPY ./$SERVER\_CONF\_FILE\_NAME $WORK\_PATH/

//###end

[复制代码](javascript:void(0);)

以上：image\_tomcat目录结构：

ZBMAC-C02VX5Y7H:image\_tomcat shiyufeng$ tree

.

├── Dockerfile

├── server.xml

└── tomcat-users.xml

Step6:构建docker镜像

//在当前目录构建镜像。注意后面的点为当前目录

docker build -t tomcat001:0.0.1 .

//查看镜像

docker image ls tomcat001

Step7:镜像构建完毕，启动tomcat

[复制代码](javascript:void(0);)

//启动tomcat容器

docker run --name=tomcat001 -p 8080:8080 -e TOMCAT\_SERVER\_ID=tomcat\_server\_001 -idt tomcat001:0.0.1

/\*

--name：为容器起的别名

-p:映射端口，宿主机与容器

\*/

[复制代码](javascript:void(0);)

Step8:启动浏览器，访问localhost:8080,显示tomat图标即为启动成功.

Step9:修改mavenl配置文件setting.xml，增加如下配置

<server>

<id>tomcat7</id>

<username>account001</username>

<password>password001</password>

</server>

以上是tomcat的tomcat-users.xml文件中配置的用户和密码，这样执行maven插件的时候就能从此处取得对应的用户名和密码去tomcat上做操作了。

Step10：进入web工程，pom.xml目录执行如下命令

注：刚定制的docker镜像必须启动状态，才可以成功构建，否则会报“构建失败”

mvn clean package -U -Dmaven.test.skip=true tomcat7:redeploy

执行结果如下所示，即代表tomcat7-maven-plugin插件已经帮我们把war部署到docker容器上去了

[复制代码](javascript:void(0);)

[INFO] tomcatManager status code:200, ReasonPhrase:OK

[INFO] OK - Deployed application at context path /loadbalancedemo

[INFO] ------------------------------------------------------------------------

[INFO] BUILD SUCCESS

[INFO] ------------------------------------------------------------------------

[INFO] Total time: 6.561 s

[INFO] Finished at: 2018-12-15T17:08:27+08:00

[INFO] Final Memory: 21M/216M

[INFO] ------------------------------------------------------------------------

[复制代码](javascript:void(0);)

Step11：浏览器输入http://localhost:8080/loadbalancedemo/hello?name=开心玉凤

显示结果：

您好开心玉凤，这是来自tomcat\_server\_001的问候！2018-12-16 01:51:44

效果如下所示：



以上：项目名称修改为ssm-docker，修改点：

（1）右击项目名-》Refactor-》重新命名

（2）pom.xml中修改<finalname>

（3）project structure-》相应修改项目名称即可。

地址栏输入：http://localhost:8080/ssm-docker/hello?name=开心玉凤

