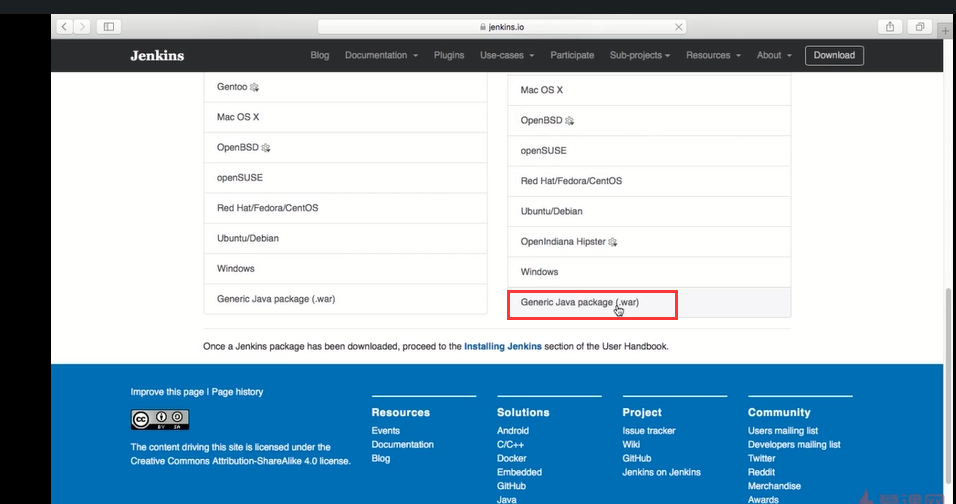
# Jenkins基础

## 2-1安装

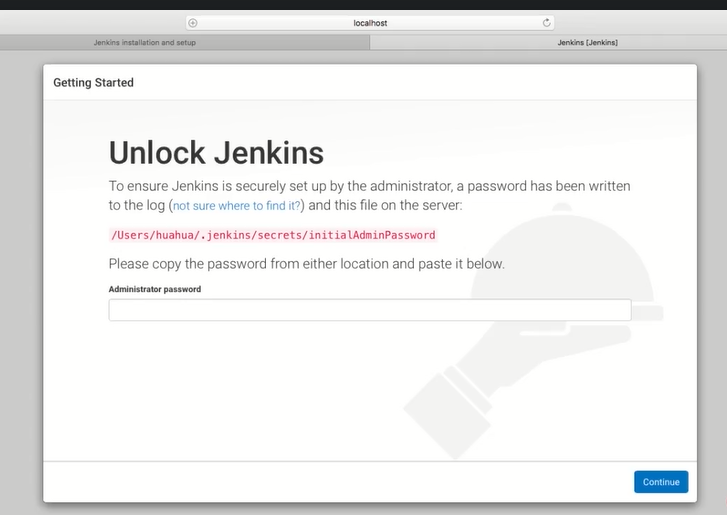
Jenkins.io下载地址



启动jenkins：

1在war包目录下命令行输入： java -jar jenkin.war

2输入localhost：8080



3进入到上图中红色路径中，查看密码，复制到页面

4自动安装插件



## 2-2Jenkins插件

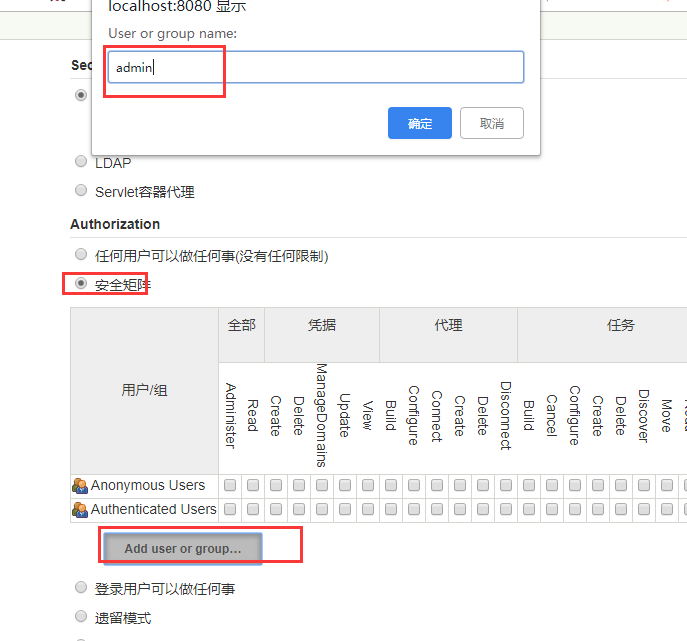
在插件管理中搜索

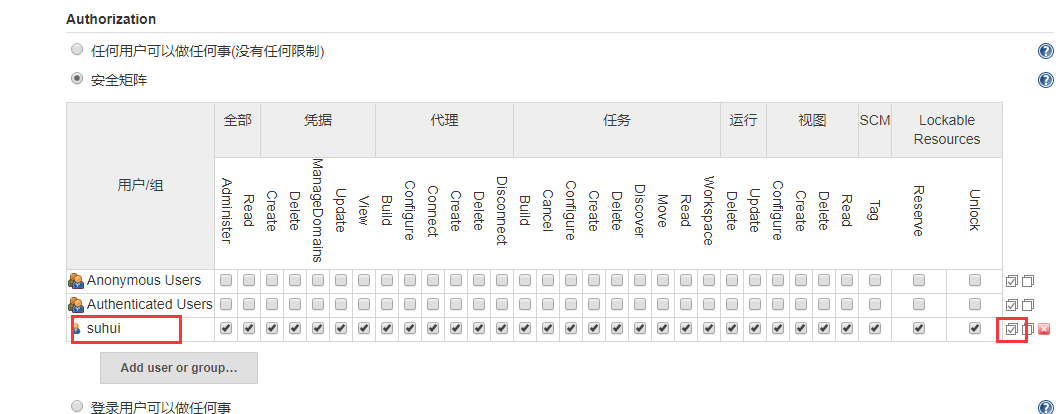
1 Rebuilder 2 safe restart 搜索并安装这两个插件

## 2-3jenkins基础配置

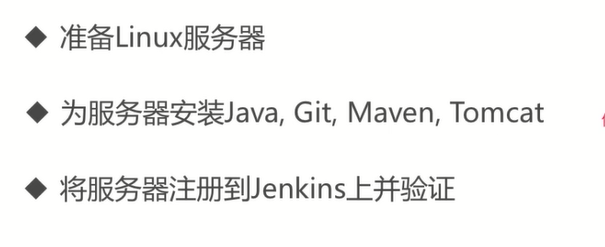
### 全局安全性配置及添加用户







# 3本章介绍



# 3-1linux系统准备

应用部署服务器准备

1准备IP地址 命令ifconfig

2确定用户名密码

3确定可以远程ssh连接 检查是否启动ssh命令行netstat -anp|grep :22

4确定防火墙已关闭 命令行 service iptables stop

# 3-3安装并配置git

## 运行安装git命令 并检验结果

在linux中执行 yum install git

验证 git version

## 将git初始化并生成授信证书

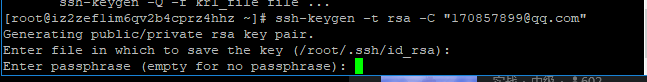
设置用户名

git config --global user.name “xxxx”

git config --global user.email “1@1.com”

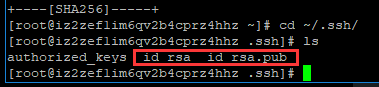
生成授信证书

Ssh-keygen -t rsa -C [“1@1.com”](mailto:\“1@1.com\”)



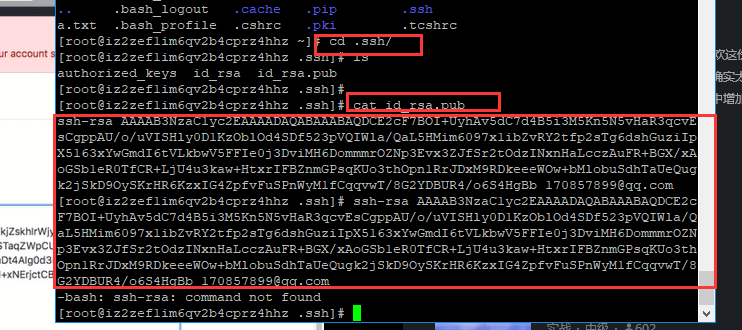
密码不输入，直接回车

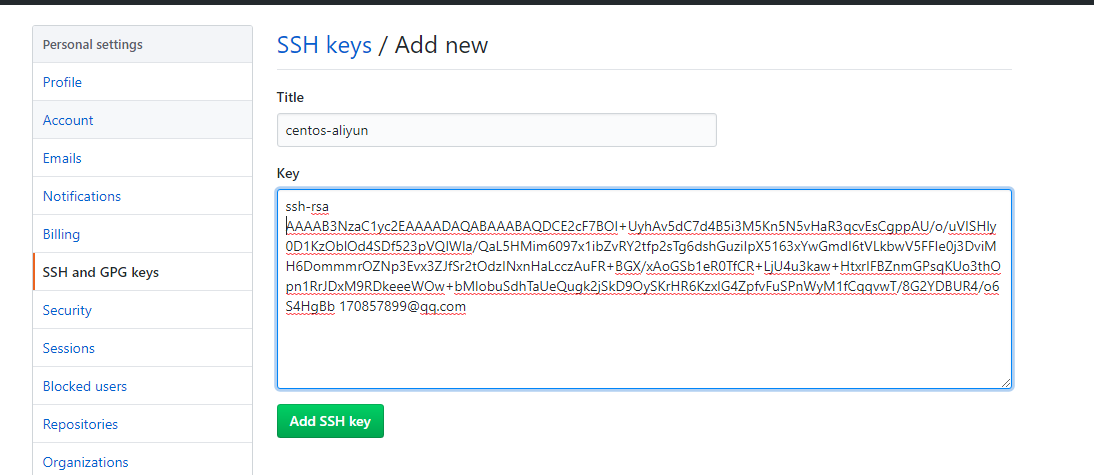
验证是否生成：



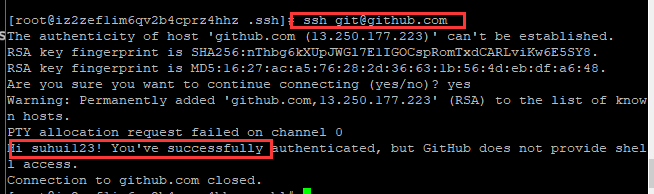
## 将证书git配置到github上，保证linux服务器与github之间可联通

找到公钥





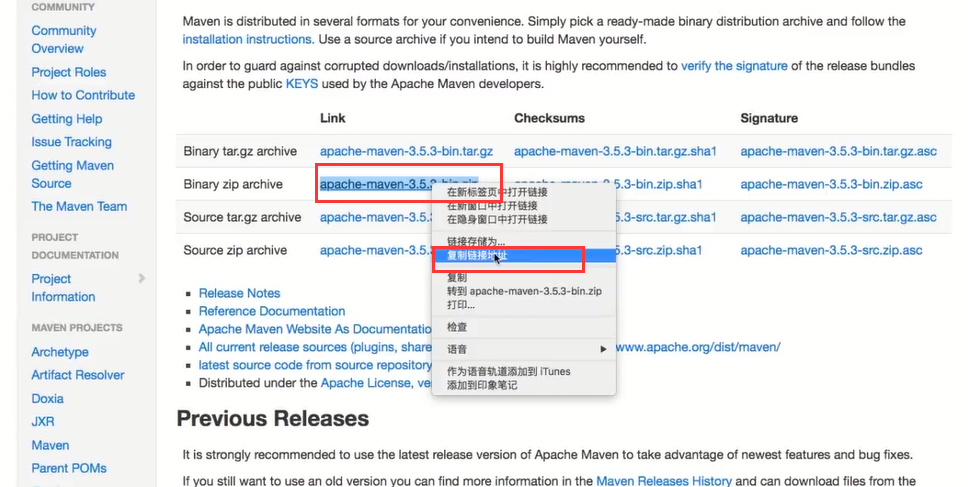
验证公钥的添加是否正确



# 3-4安装并配置maven

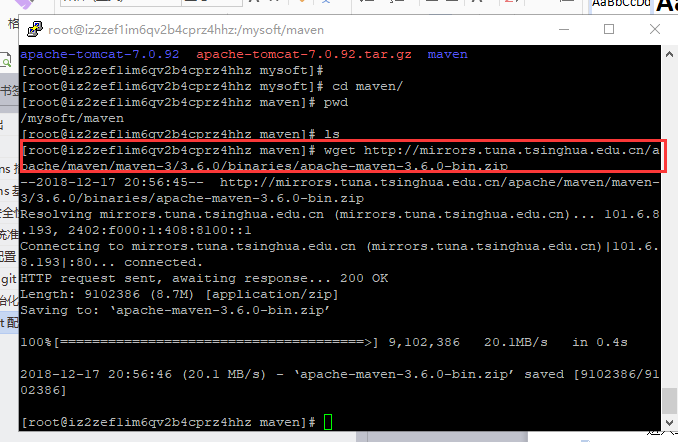
## 下载maven包

http://maven.apache.org/



进入到要下载到的目录下

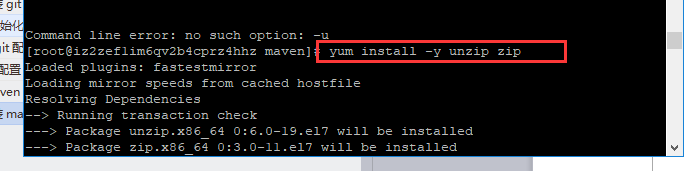
<http://mirrors.tuna.tsinghua.edu.cn/apache/maven/maven-3/3.6.0/binaries/apache-maven-3.6.0-bin.zip>



## 解压安装maven包



如果有unzip命令，需要先安装解压工具如下图



## 配置Maven系统参数，验证maven是否安正确

Vi /etc/profile

结尾处添加

export MAVEN\_HOME=/mysoft/maven/apache-maven-3.6.0

export PATHE=$MAVEN\_HOME/bin:$PATH

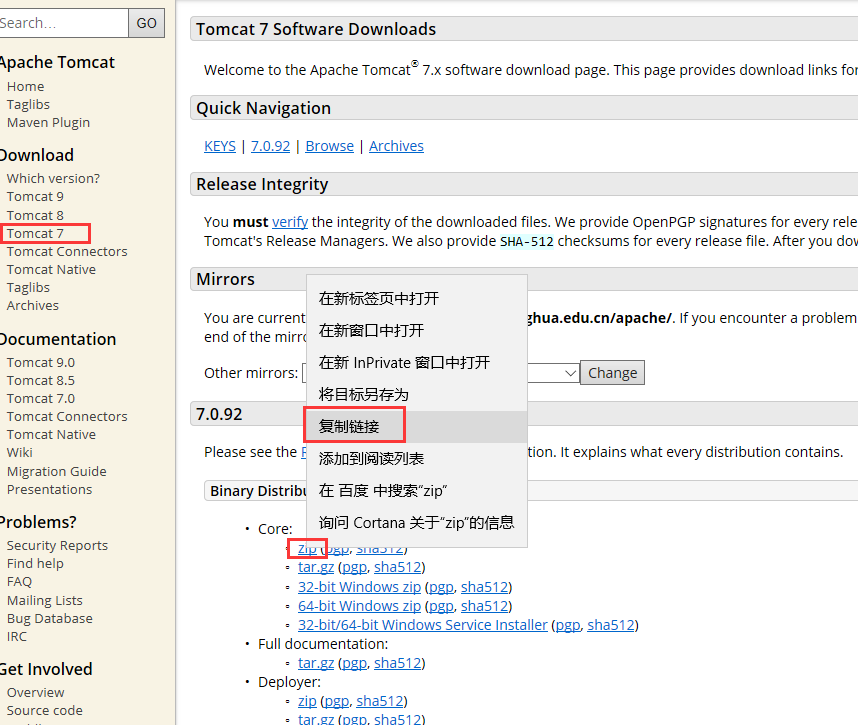
执行 . /etc/profile 让配置文件生效。. /之间有一个空格

验证maven是否配置生效输入 mvn -version

# 3-5tomcat下载

下载tomcat

<http://tomcat.apache.org/>



使用wget 下载

# 3-6解压安装tomcat

使用unzip解压

# 3-7配置tomcat

## 在tomcat根目录下，给所有文件可执行权限

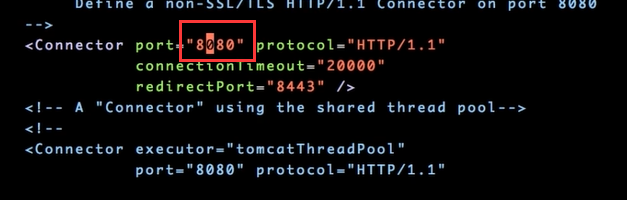
Chmod a+x -R \*

a+x:给多所有用户可执行权限

R：当前路径和子路径

\*：所有文件

## 配置tomcat端口



# 3-8验证tomcat

执行bin目录下的 startup.sh，启动tomcat

进程验证：输入 ps -ef|grep tomcat

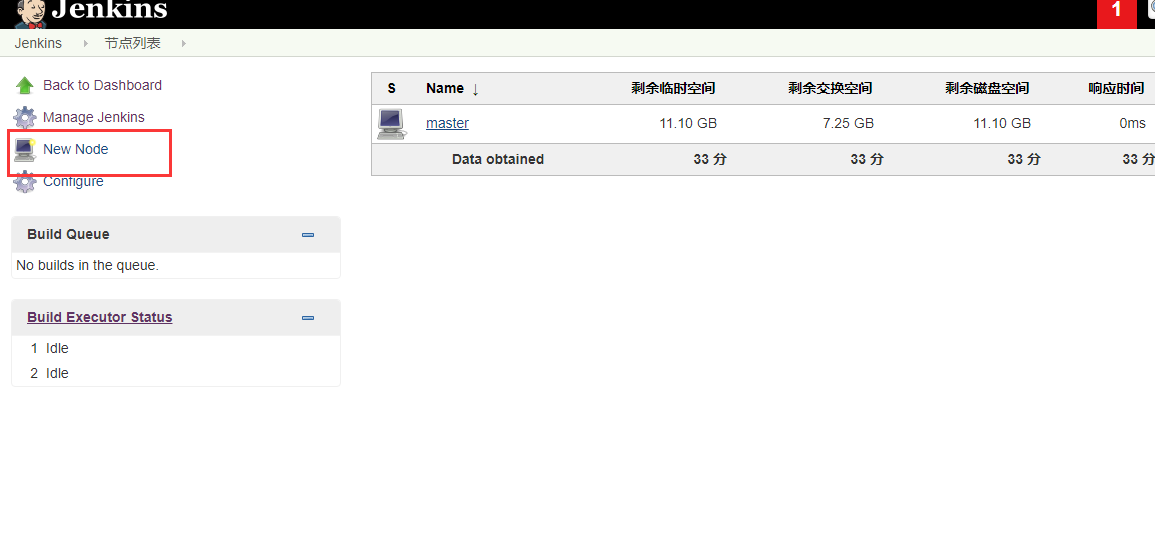
页面验证：localhost：8080

# 3-9将linux注册到jenkins上

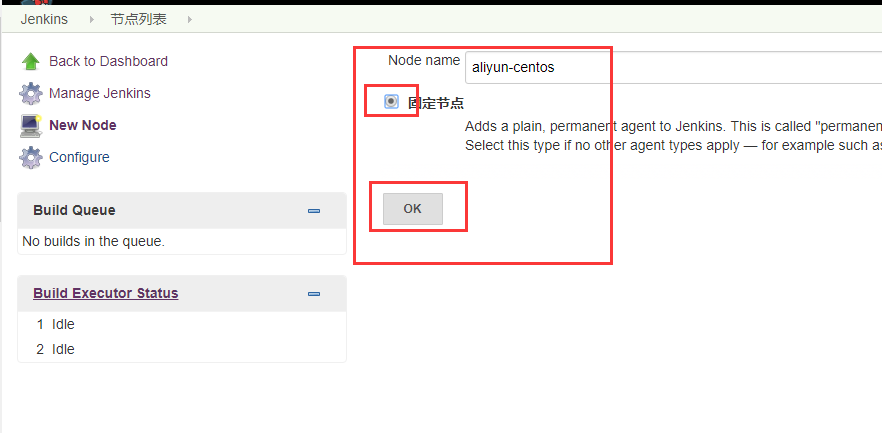
## 1新建节点



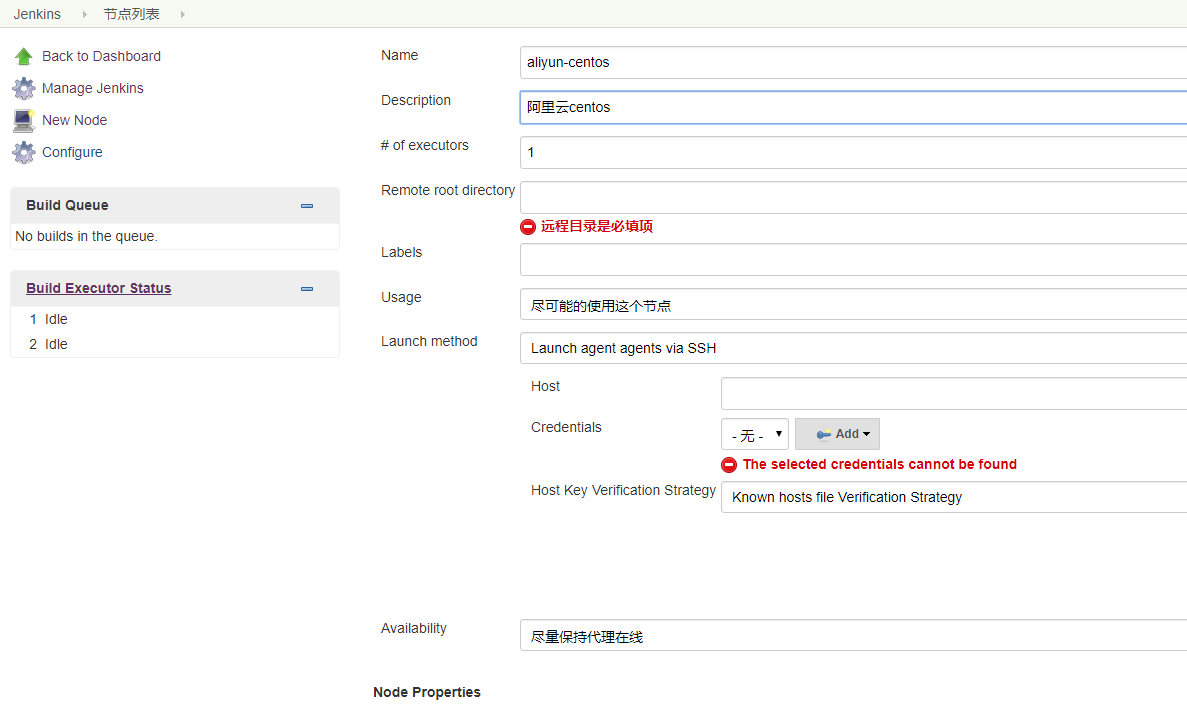
2

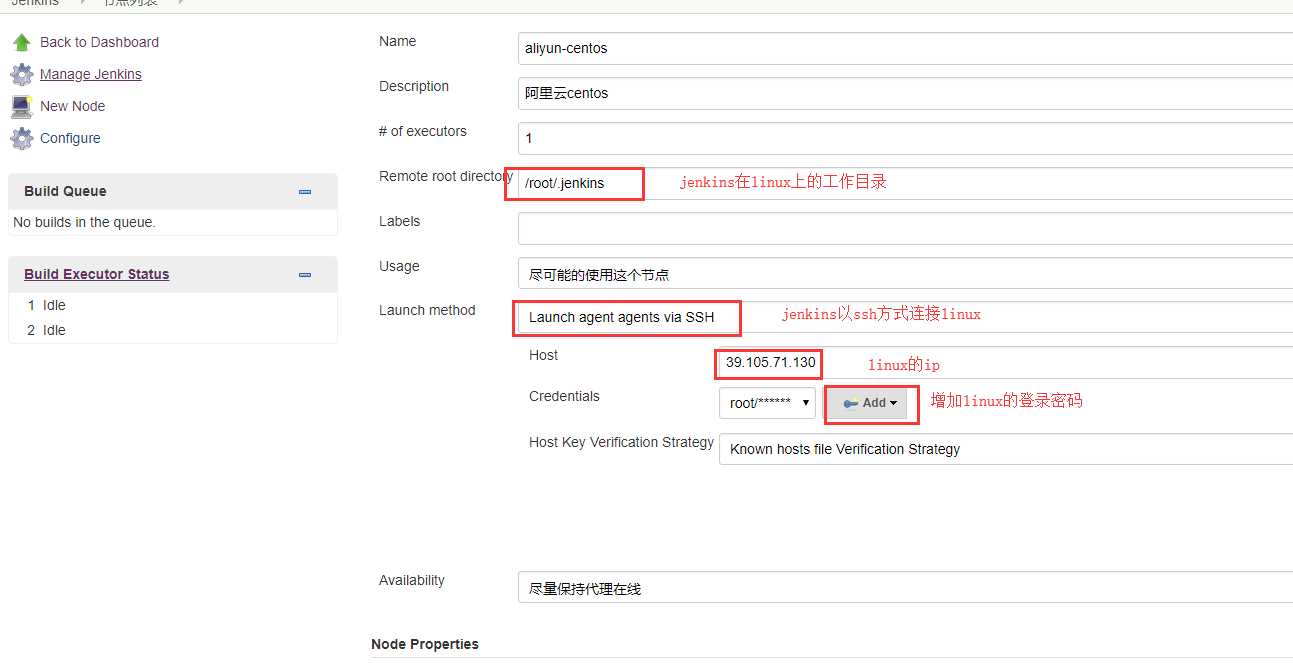


3



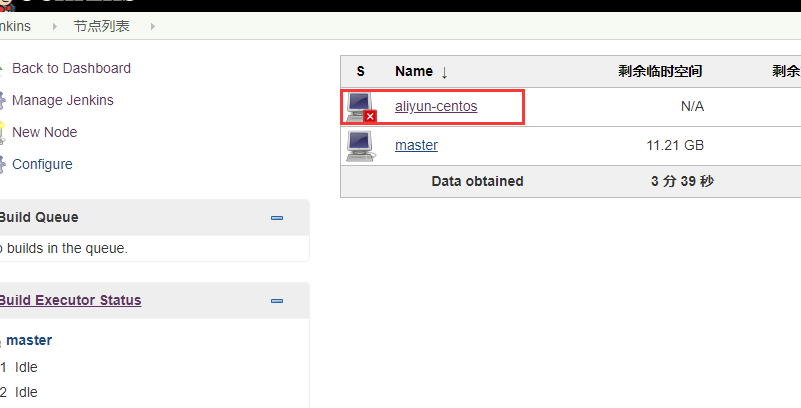
## 2配置路径、ip、用户名、密码





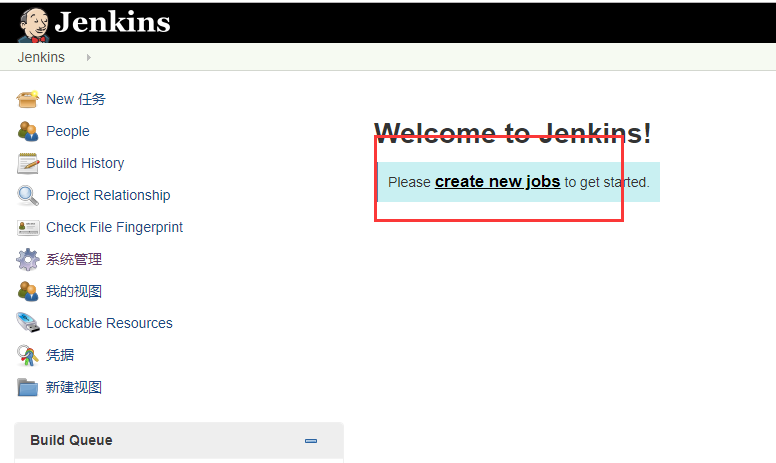


## 3启动从节点验证linxu服务器已经成功注册到jenkins上



## 4新建job验证jenkins任务可以远程运行在linu服务上

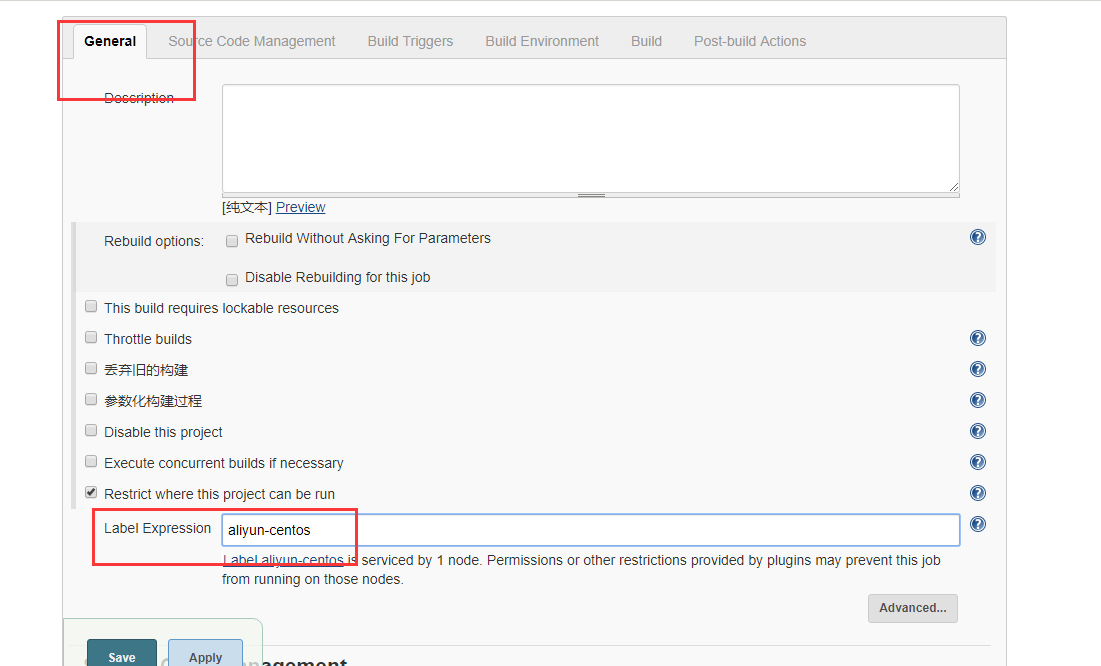
1



2

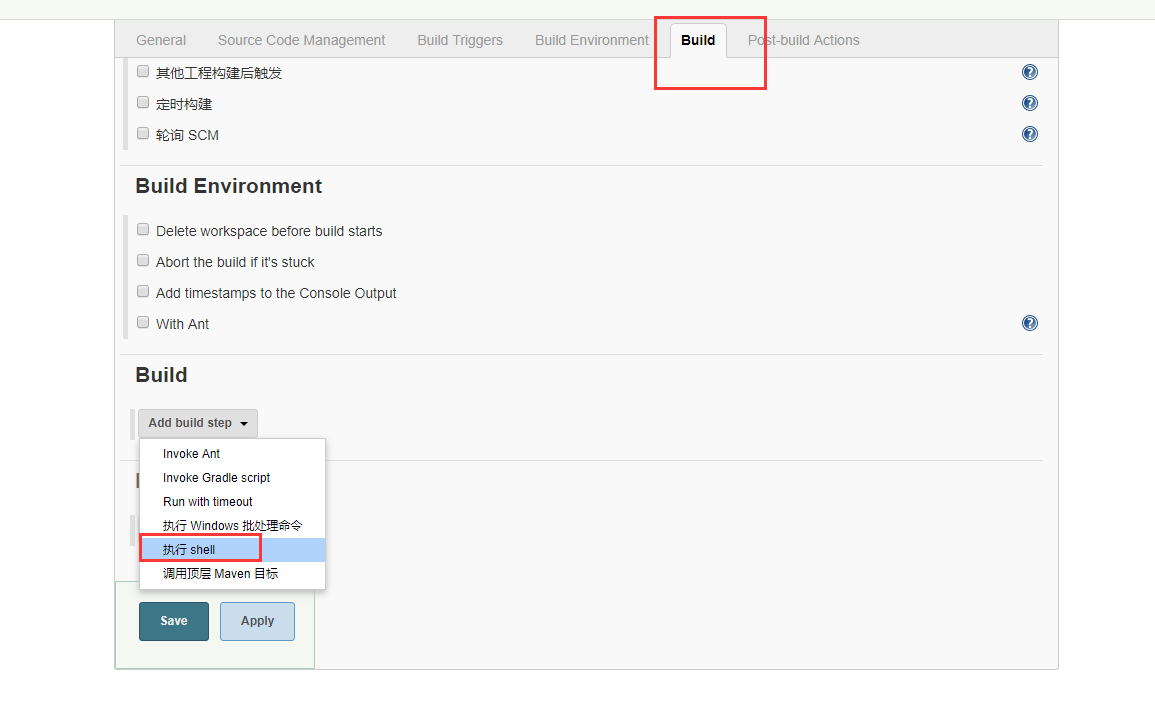


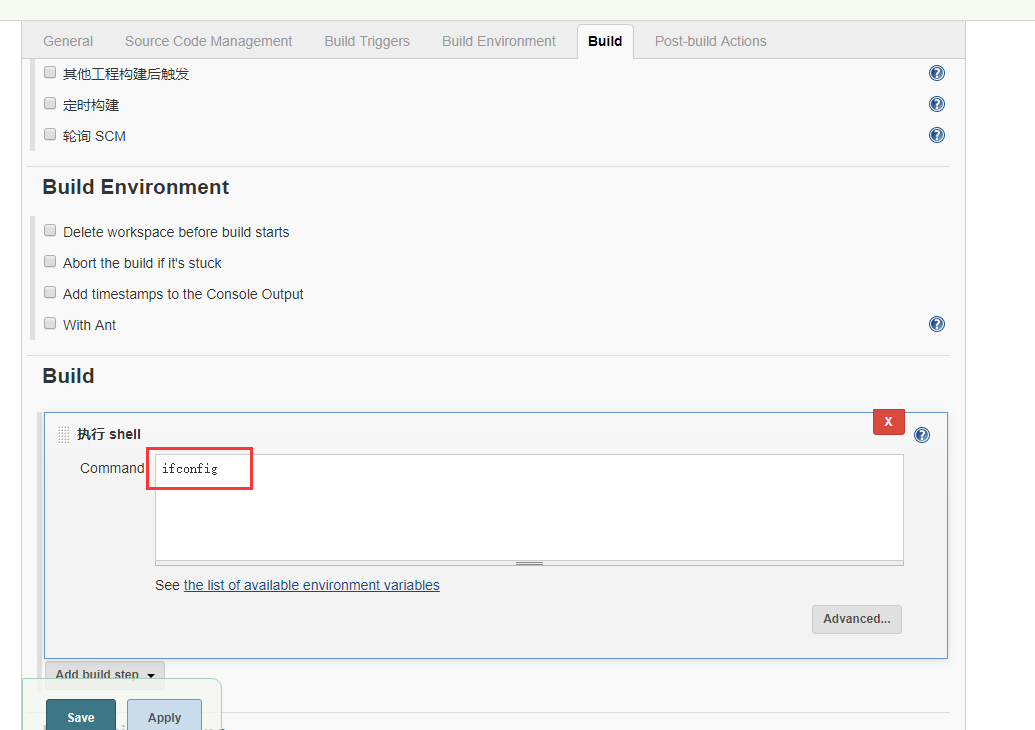
3



表明这个任务只允许在这个环境下运行

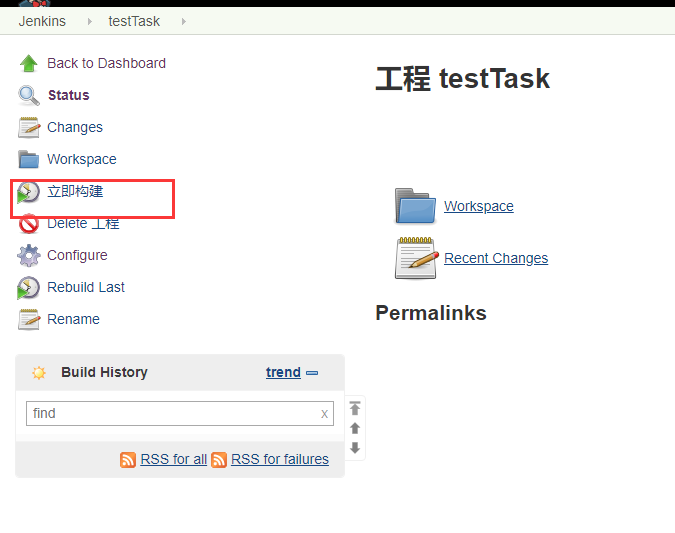
4

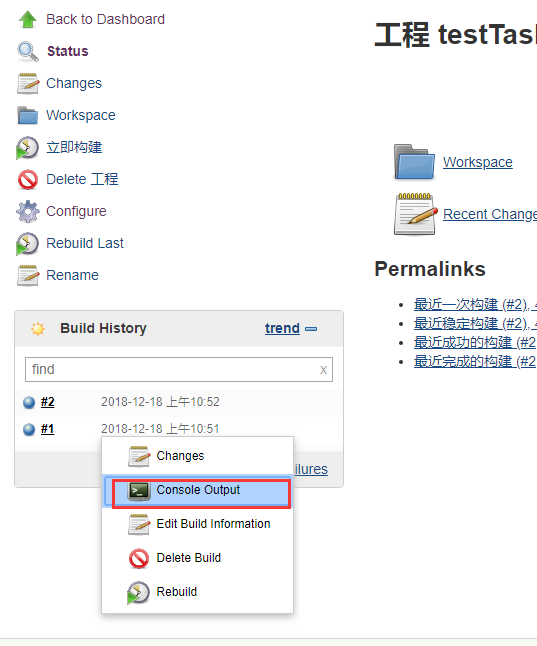


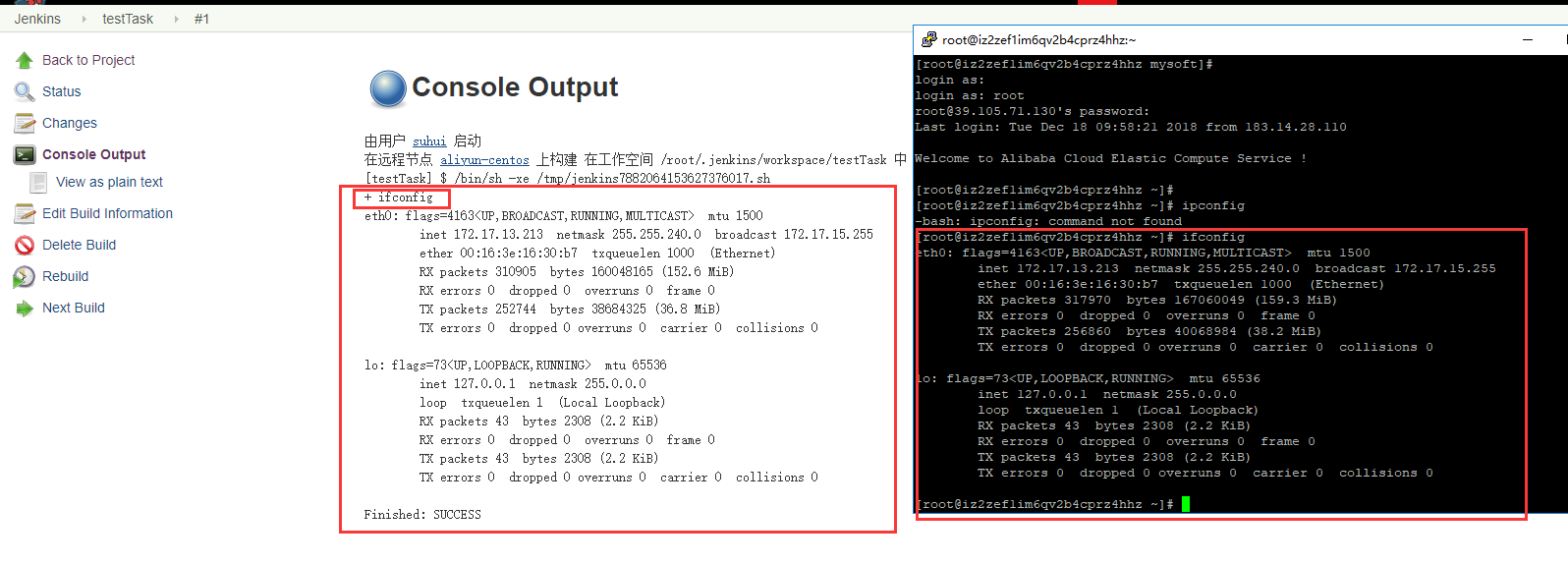


这里输入ifconfig，在执行构建任务是，会在远程linux上执行该命令

5

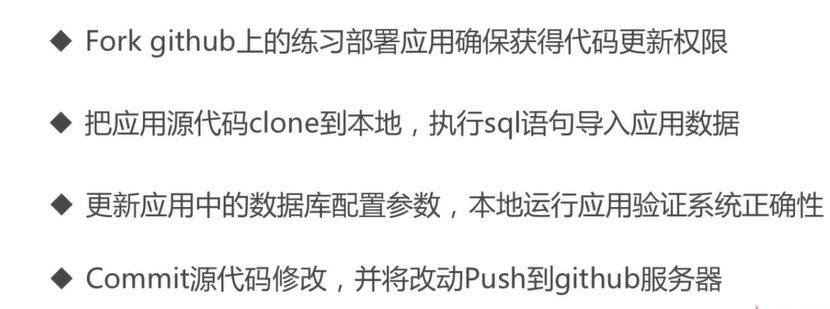






至此表明jenkins可以在远程linxu上执行命令了

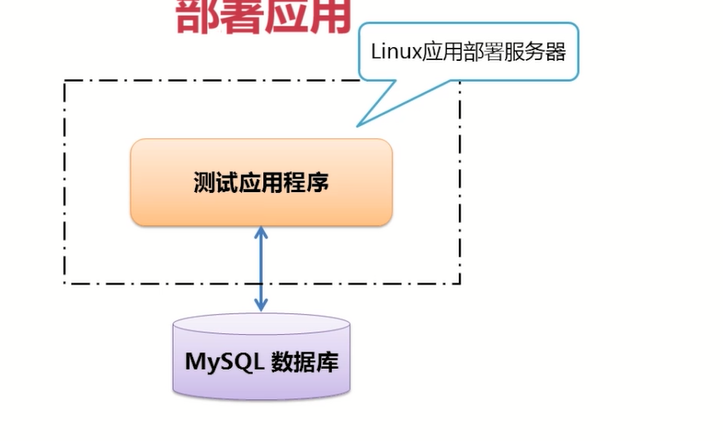
# 4本章介绍



# Fork部署应用

## 4-1github介绍

## 4-2部署应用程序介绍

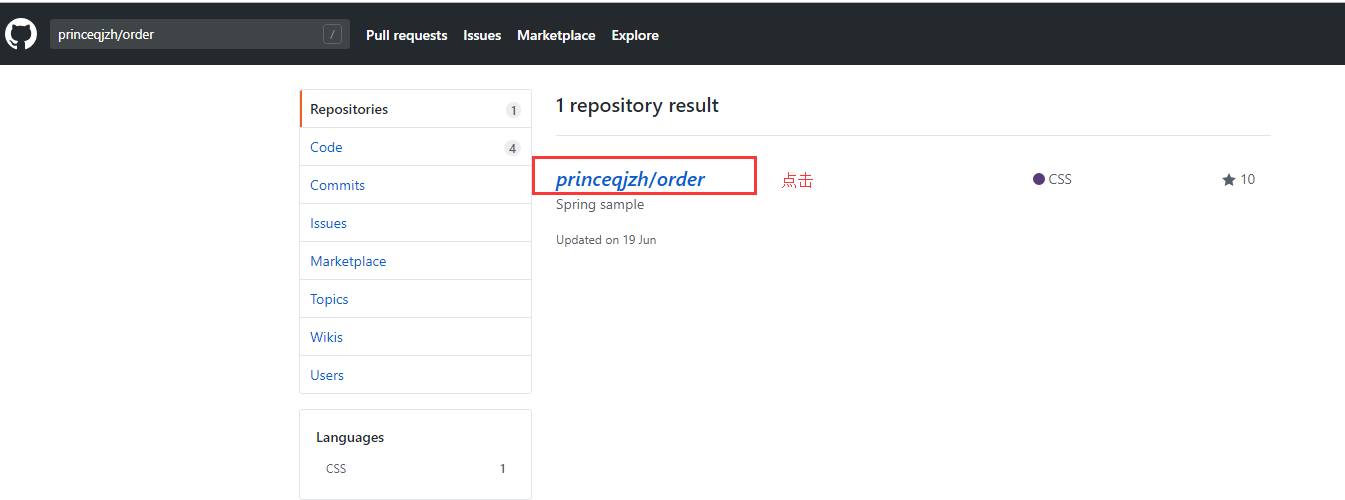


Web程序部署在linux中

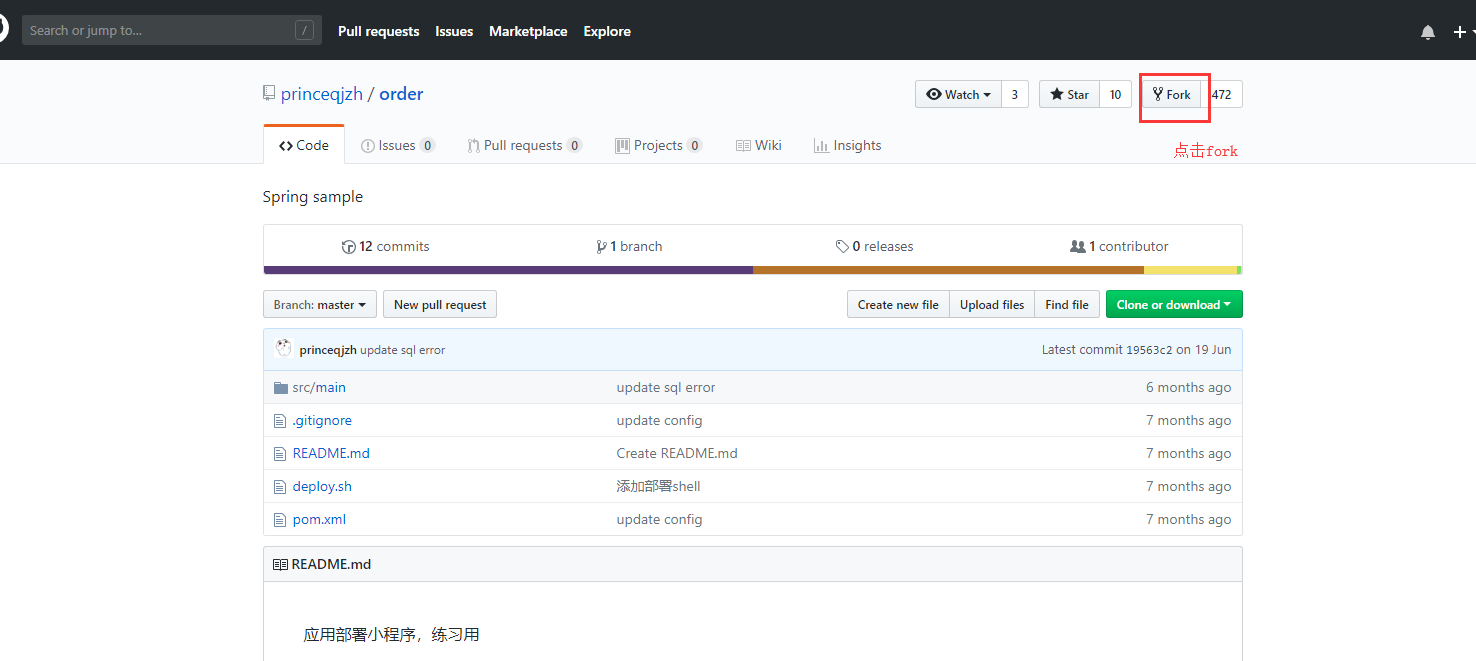
Mysql部署在另一台服务器中

## 4-3github上fork部署应用程序

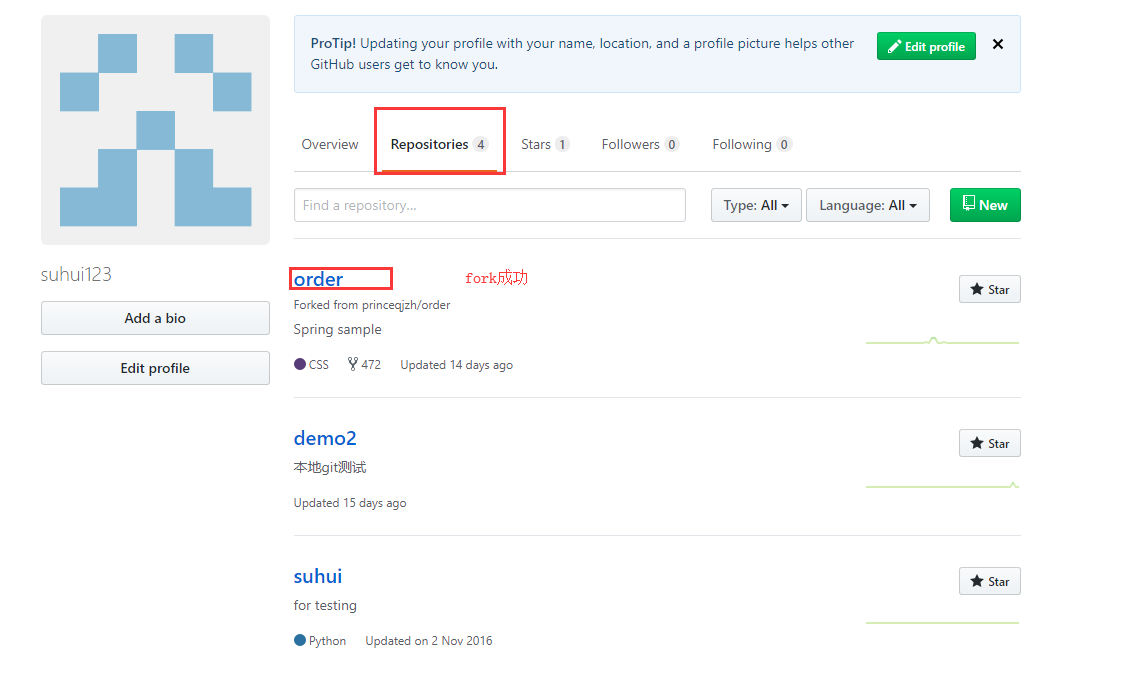
1搜索项目



2fork该项目



3在自己的repositories中查看该项目

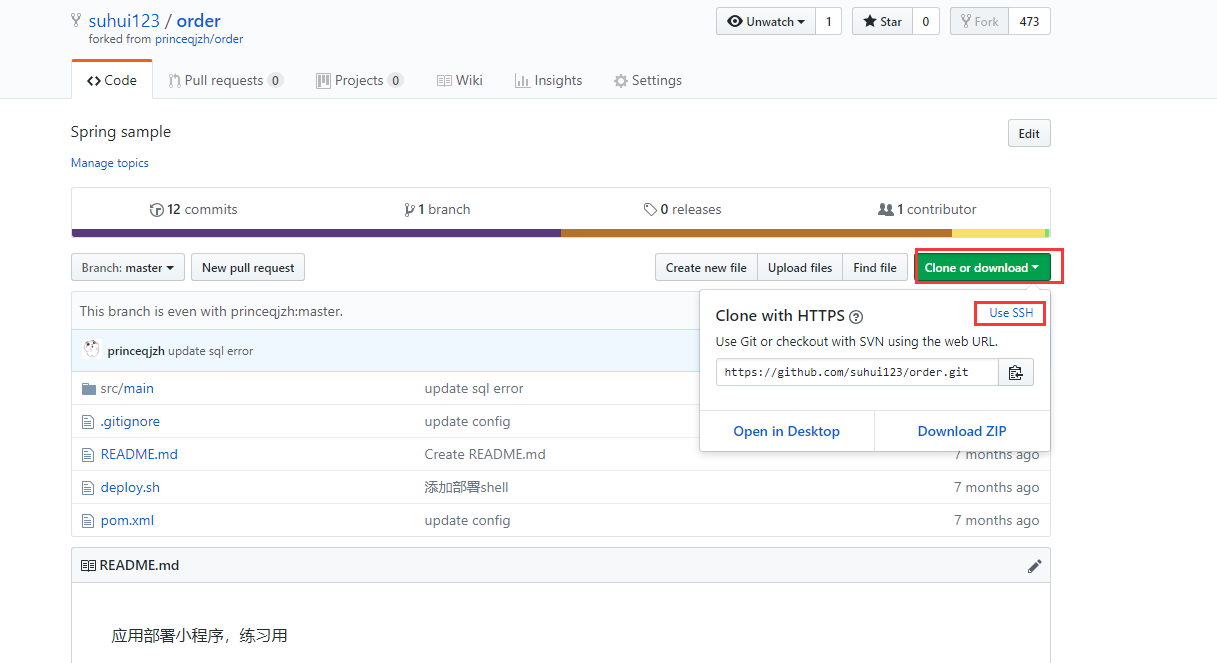


至此，应用项目fork到了自己的github上

# 将应用源代码同步到本地

## 1在github上打开刚刚fork的应用（4-4）

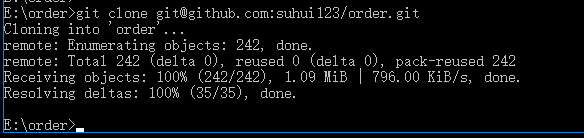
## 2单机clone or download按钮，复制克隆链接（4-5）



Ssh格式：

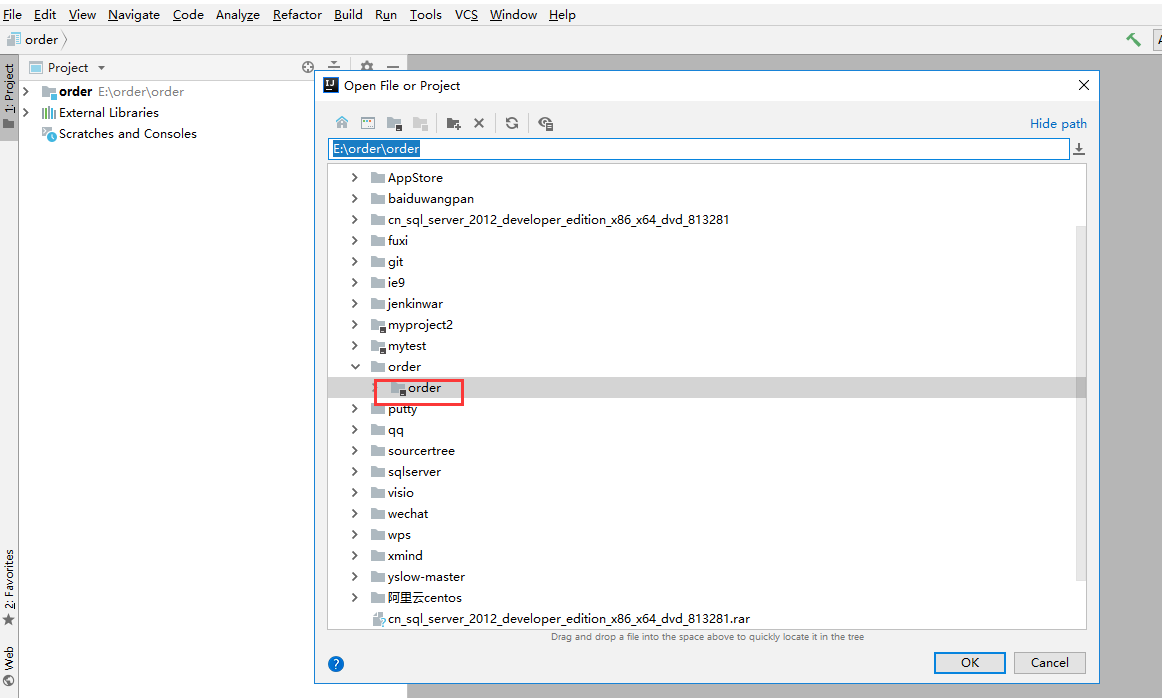
[git@github.com:suhui123/order.git](mailto:git@github.com:suhui123/order.git)

## 3进入本地需要存储应用源程序的路径，将代码clone到本地（4-6）



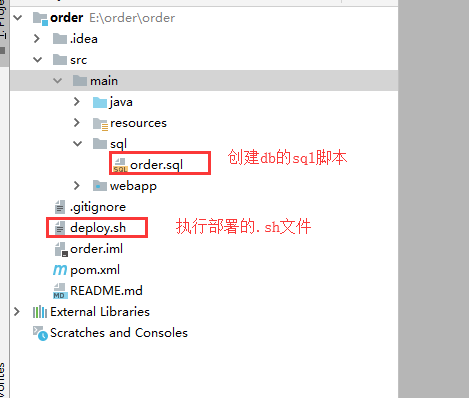
# 简要分析部署应用

## 4-7用idea打开克隆好的项目



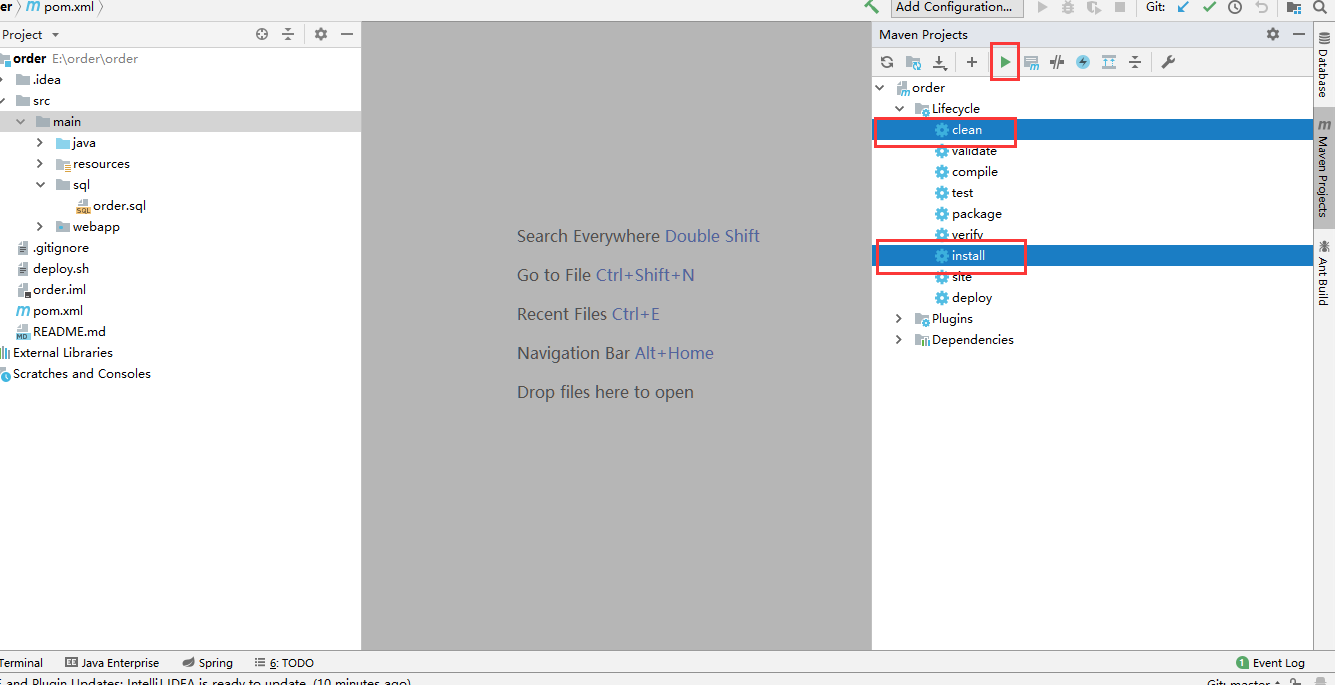
## 4-8简要说明代码源程序

该项目是建立在maven架构之上的，所有根目录下有pom.xml



## 4-9在idea中使用maven构建该项目

选中clean和 install 点击运行按钮，进行构建





出现Bulild sucess表示构建成功

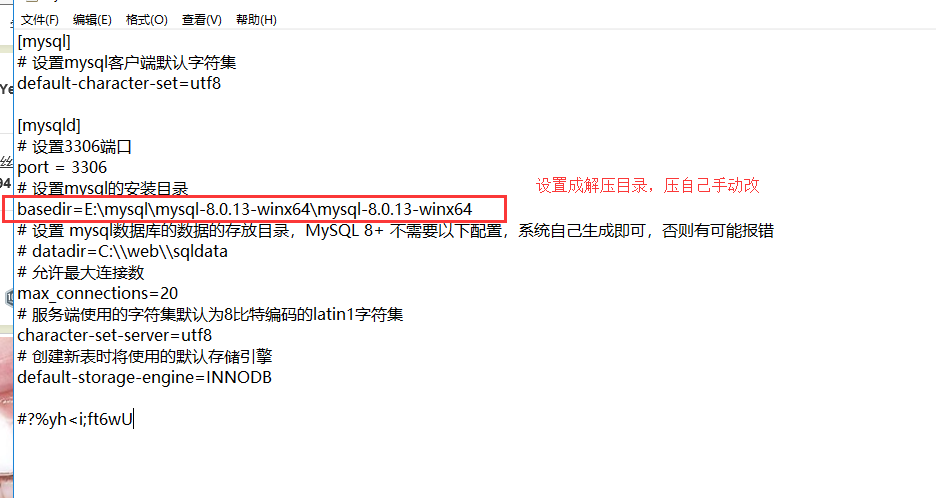
表示整个工程的源代码是正确的，可以完成整个工程的构建工作

# 数据库建表

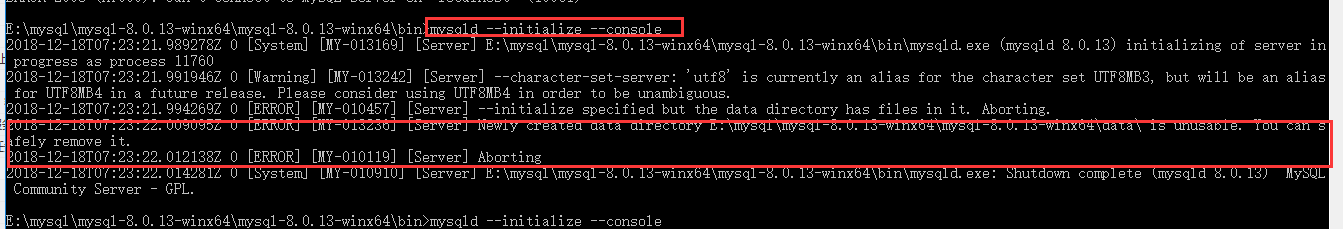
Mysql安装参照http://www.runoob.com/mysql/mysql-install.html

安装mysql时候的几个坑：

1

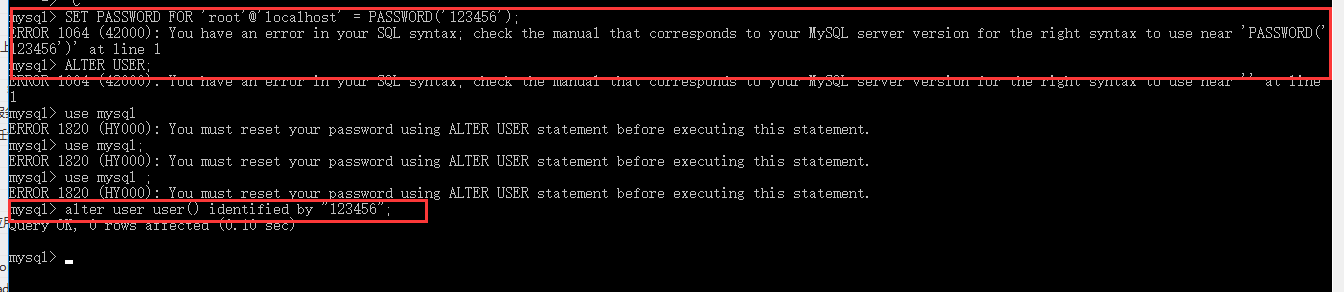


2



上图：初始化的时候可能报错，需要手动删除安装目录下的data文件夹，重新初始化

3



上图：修改初始密码时可能报错，需要执行语句

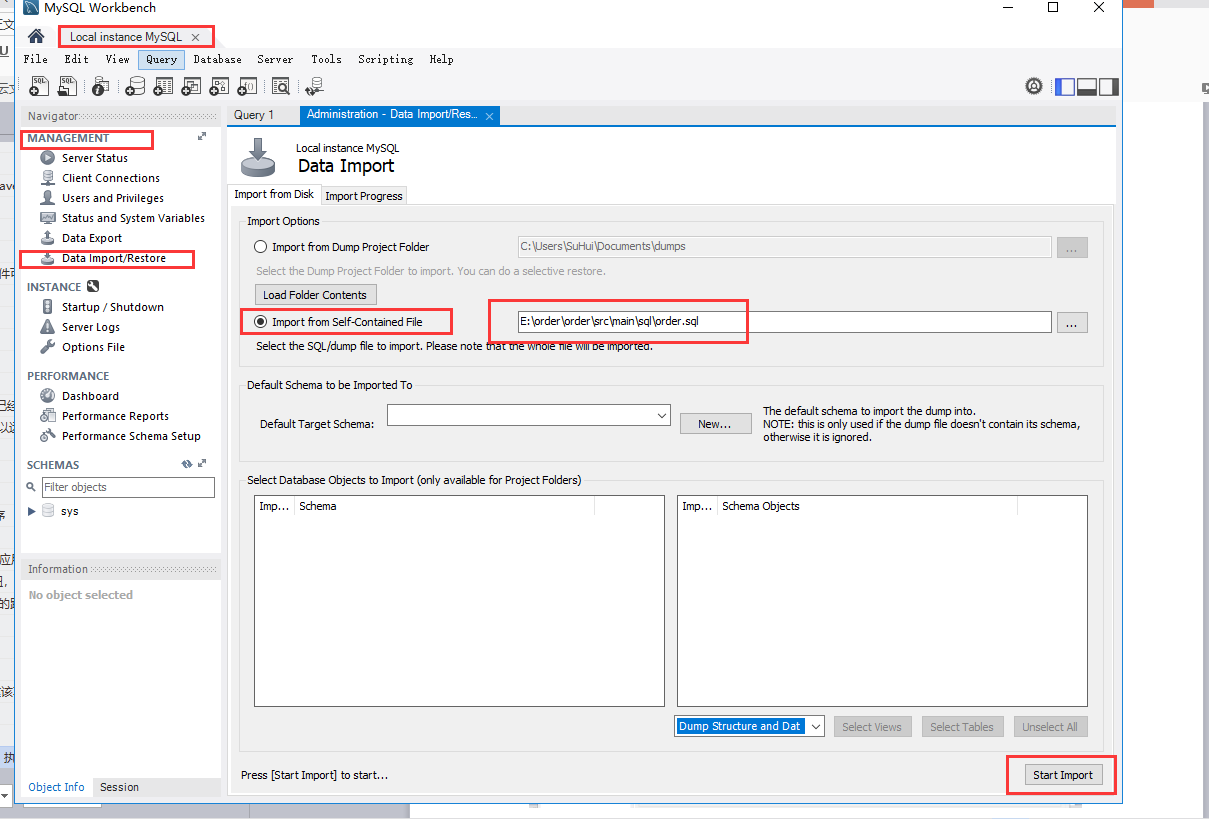
alter user user() identified by "123456";

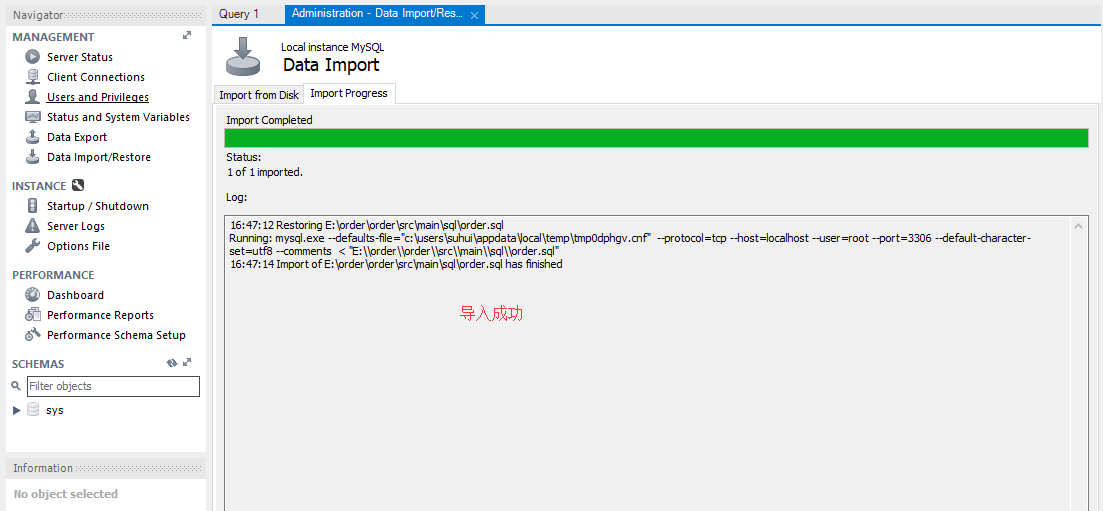
## 4-10打开mysql客户端工具

工具Mysqlworkbench

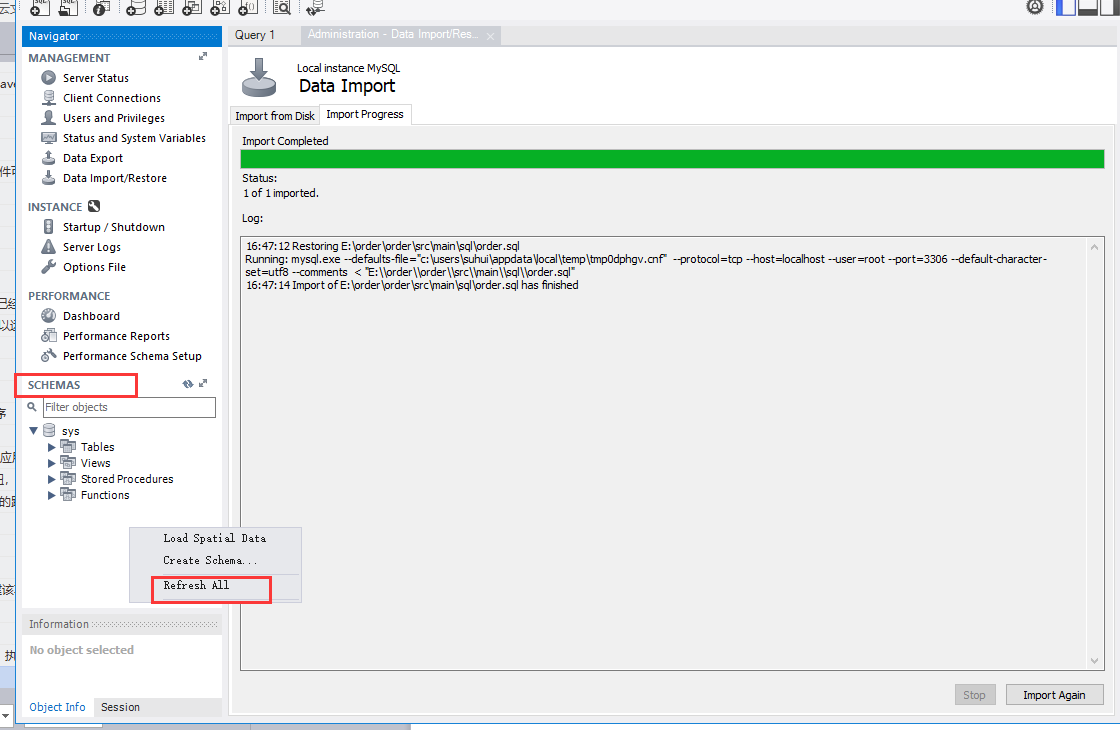
<https://dev.mysql.com/downloads/workbench/>

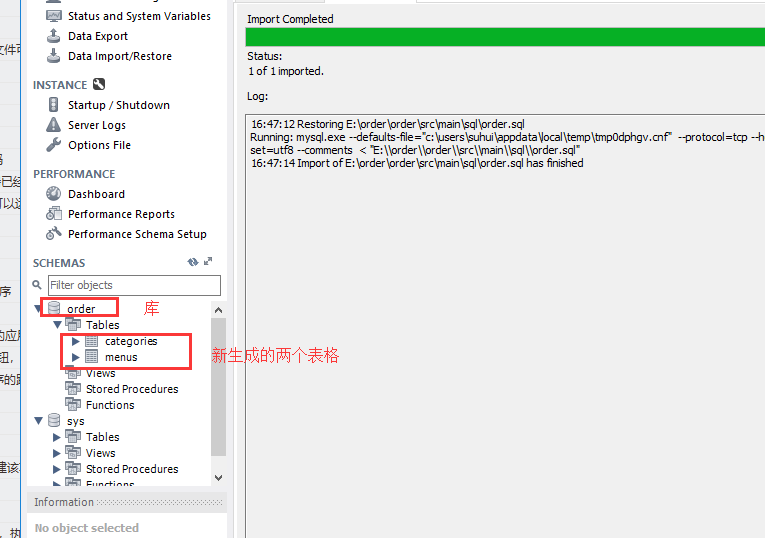
## 4-11导入源代码中的.sql文件，执行SQL语句

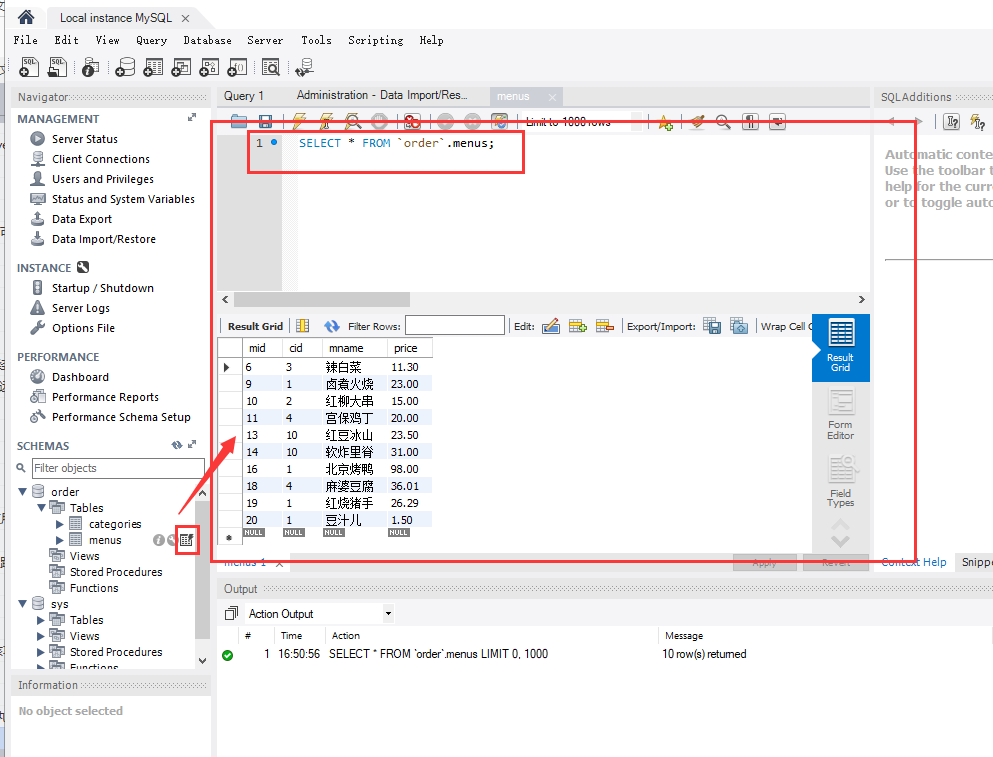




## 4-12执行sql后 检查结果



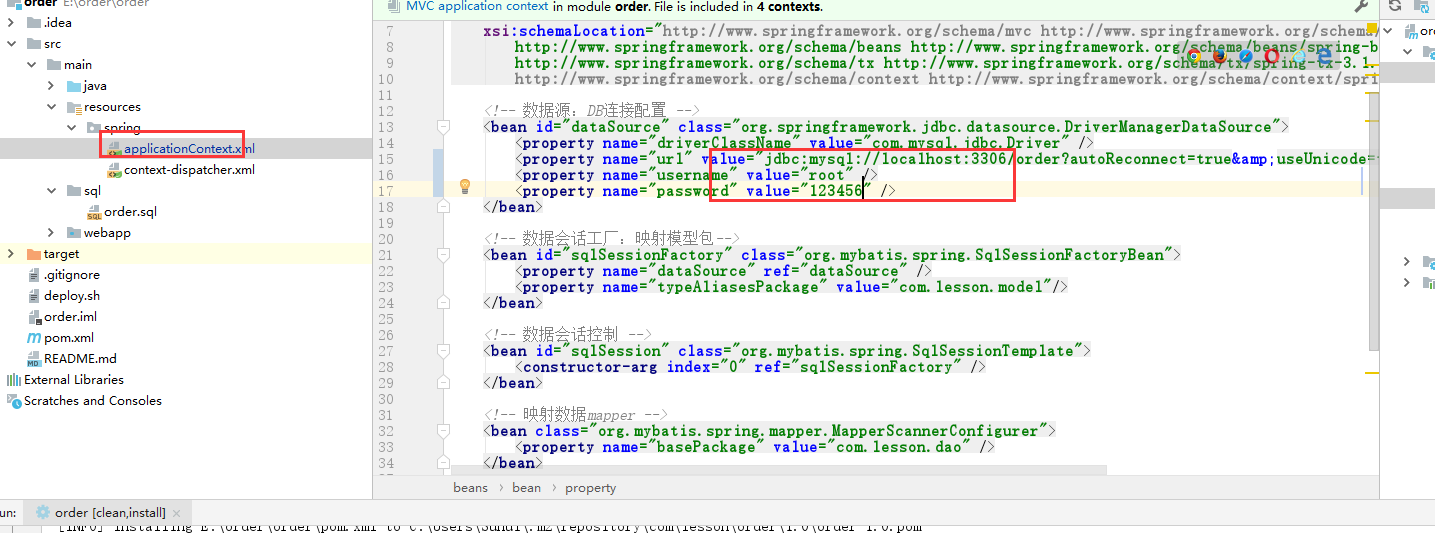




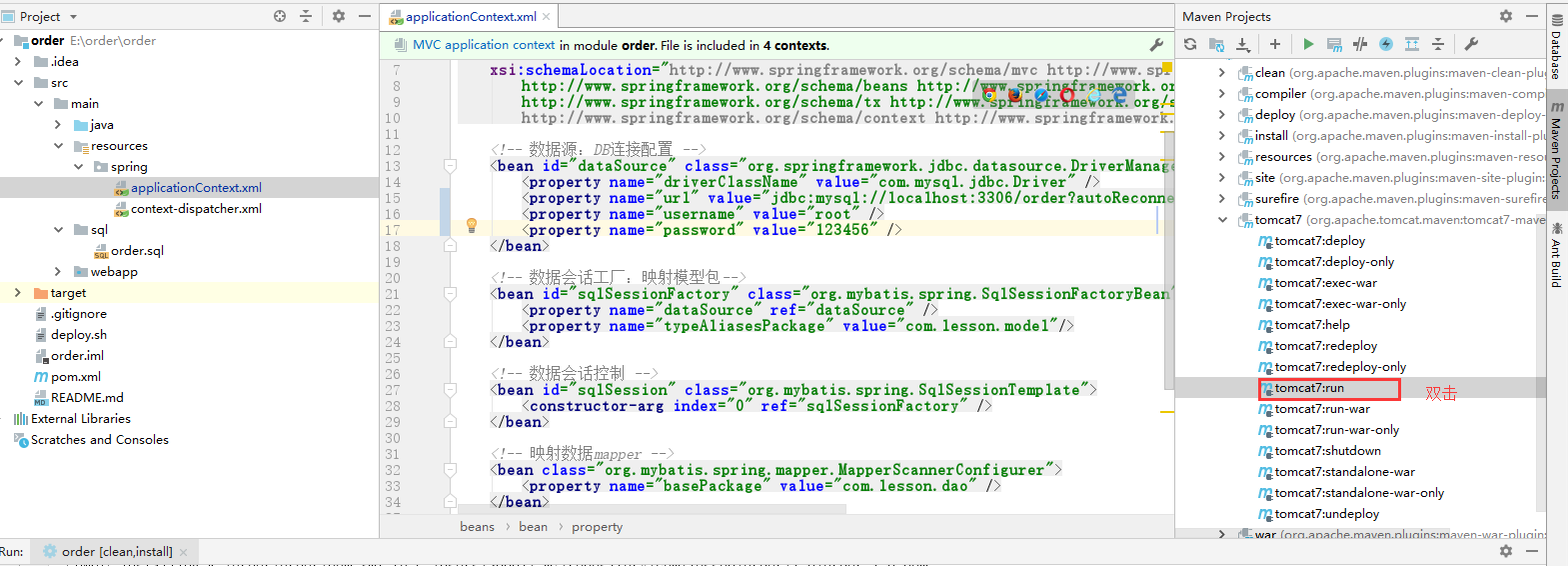
上图说明sql执行成功

# 修改应用配置参数然后本地运行

## 4-13更新应用中的mysql服务器host地址、端口、用户名、密码



## 4-14验证参数是否配置正确



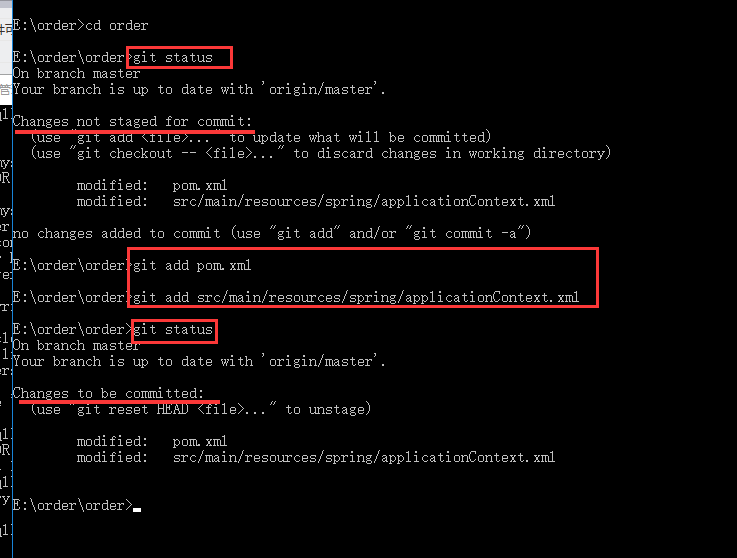


访问localhost：8071配置成功

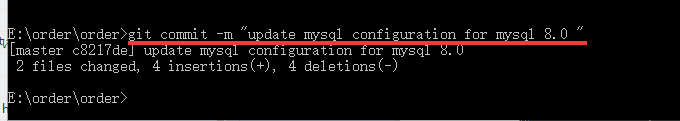
# 提价修改后的代码，推送到github上

## 4-15commit代码更新

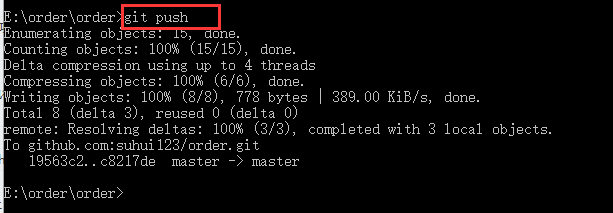
Git status 检查更新状态



下图：执行commit，提交代码



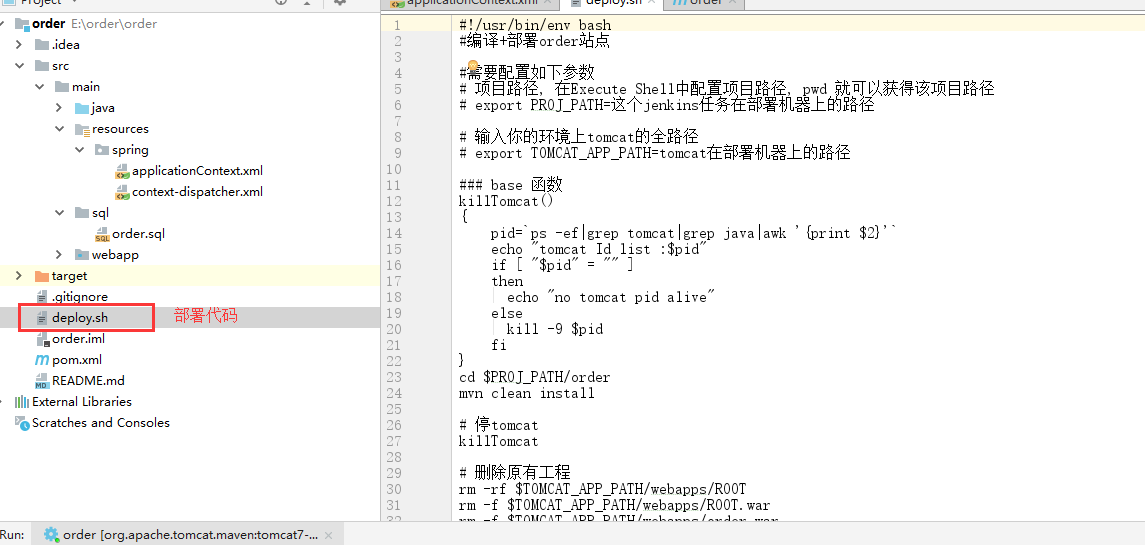
## 4-16将更新后的代码push到gihub上



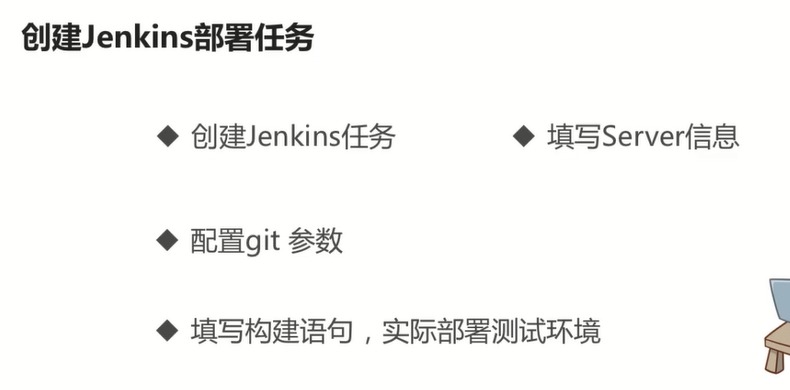
# 5-1部署过程介绍



# 5-2创建jenkins自动化部署任务

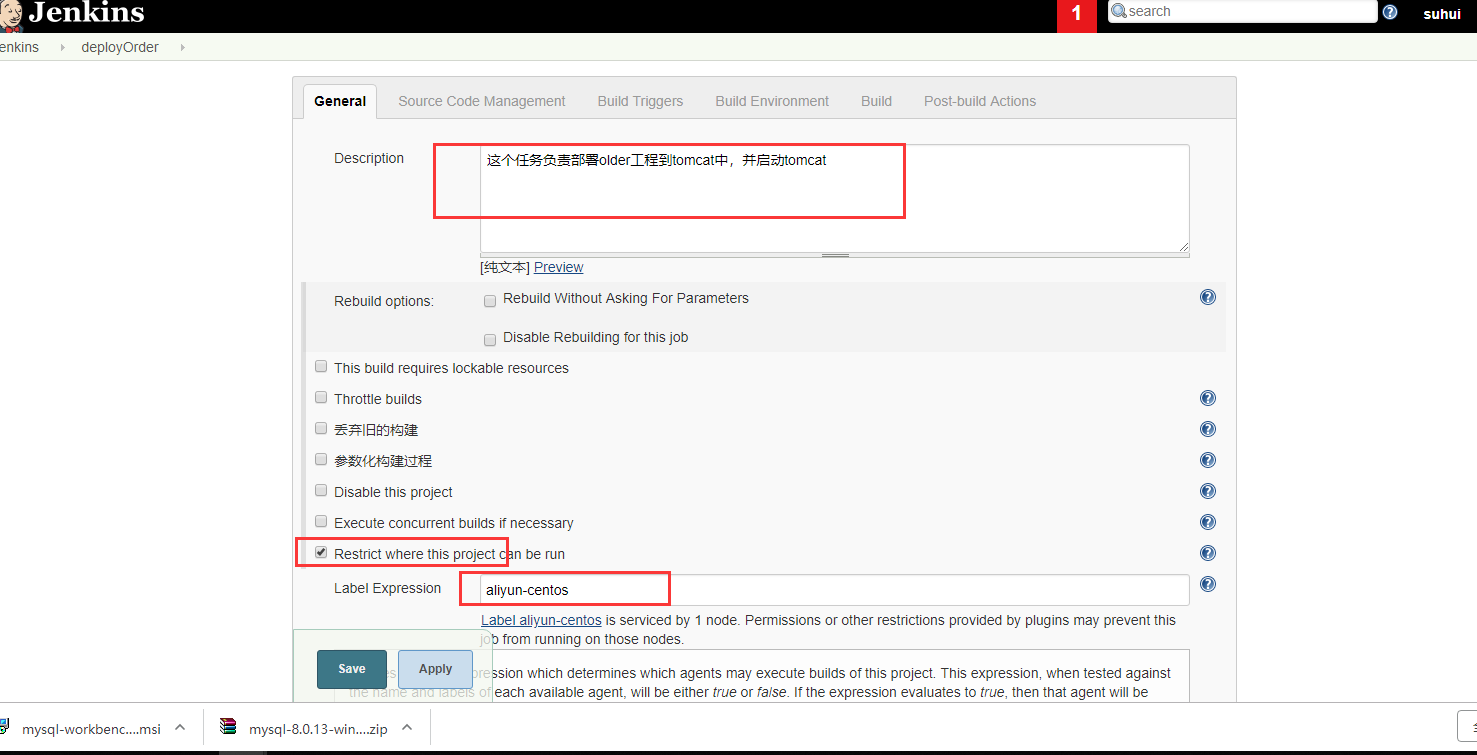


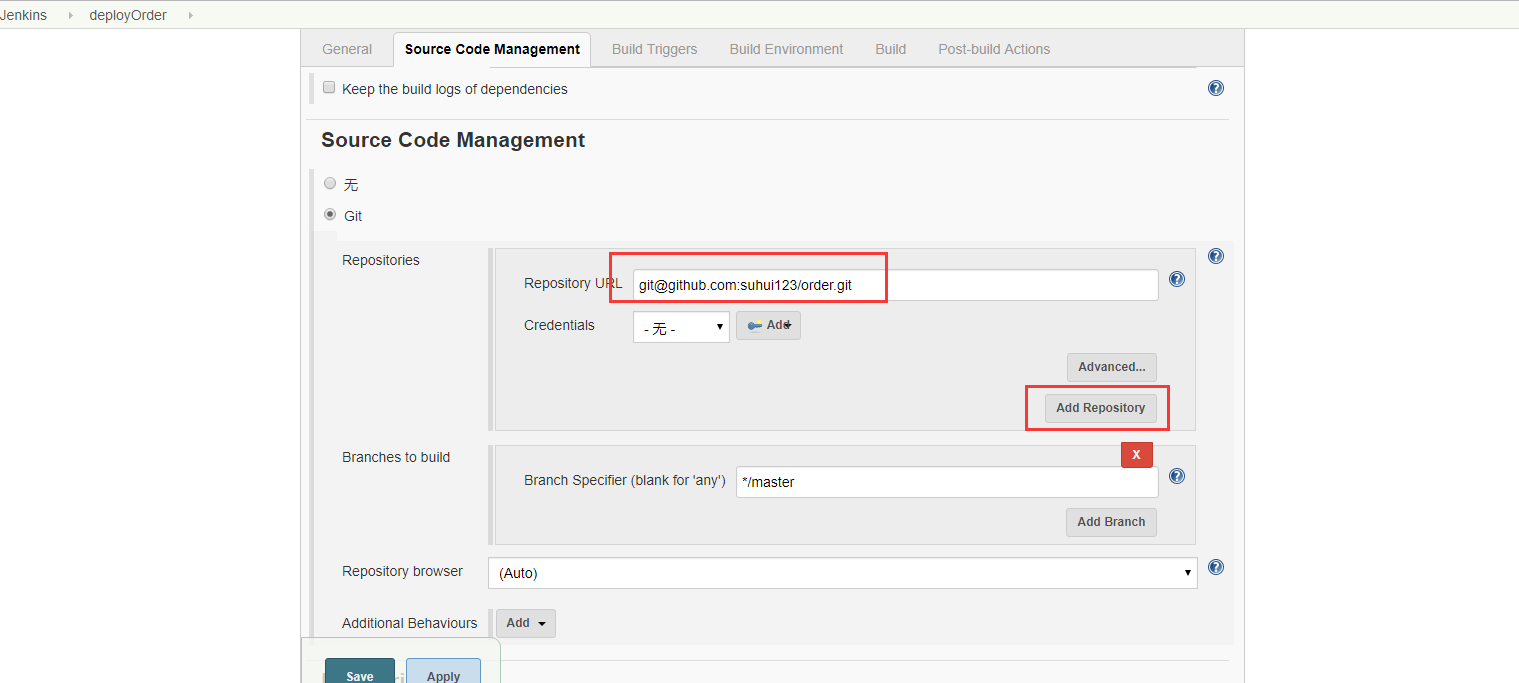
这个脚本有的语言是shell

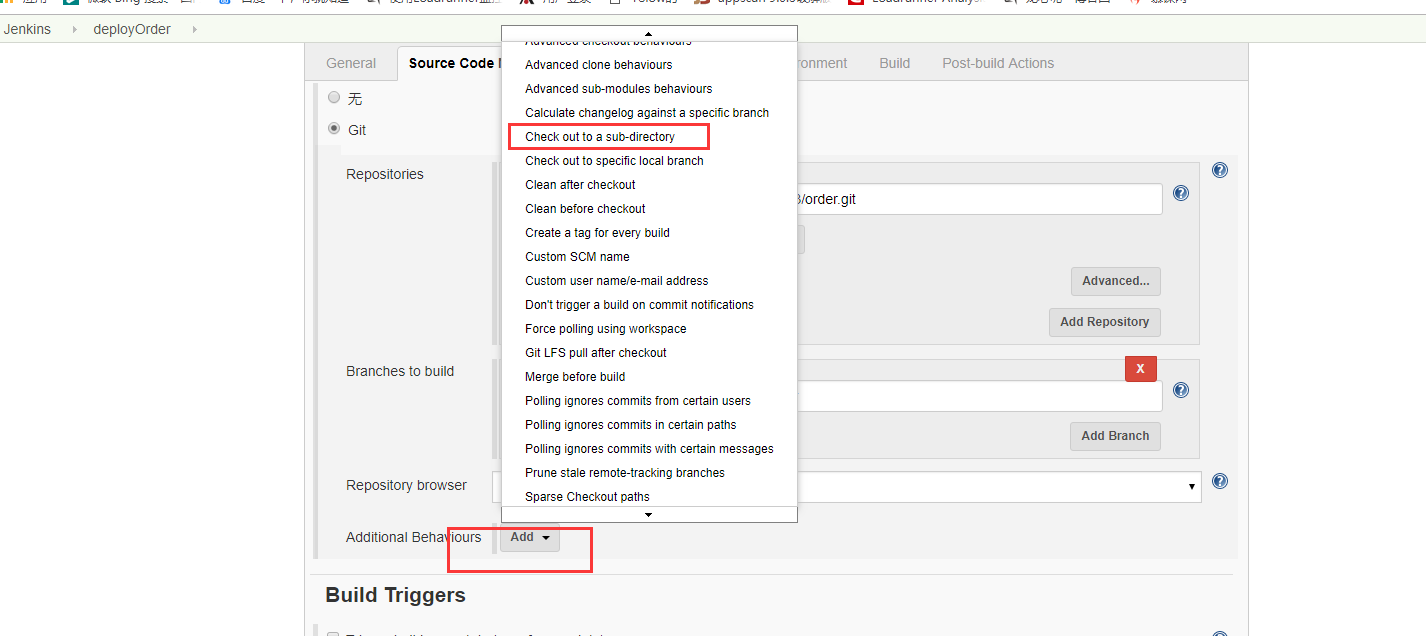


创建jenkins任务

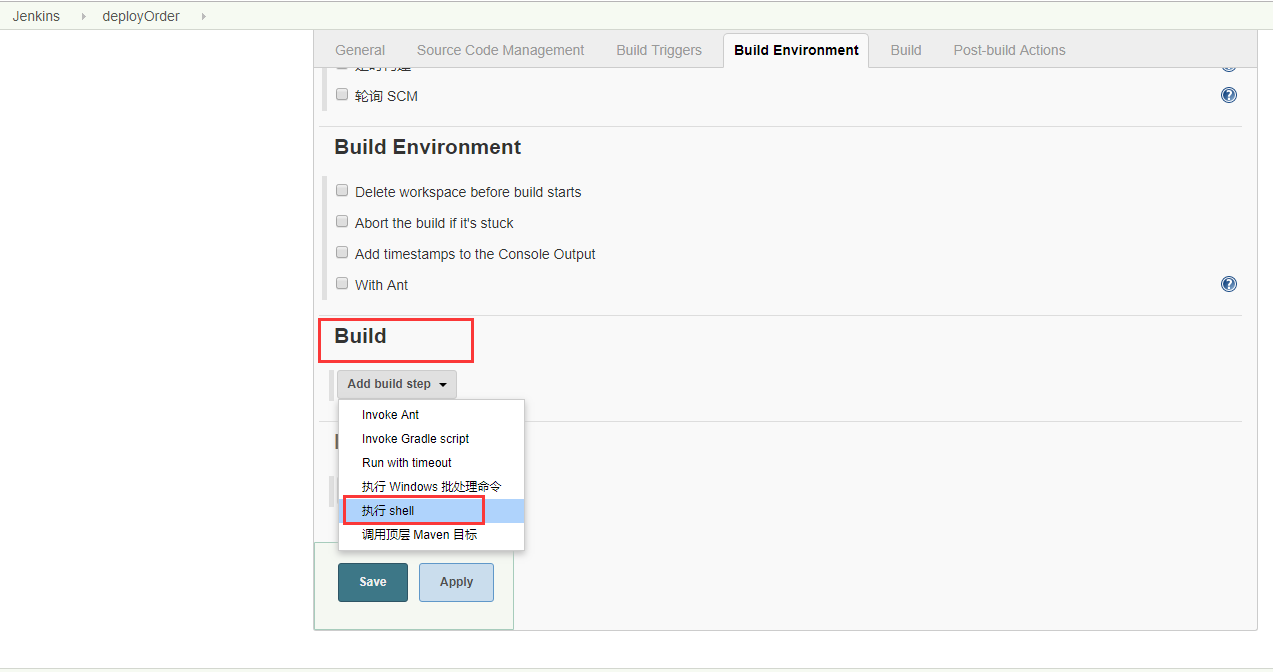


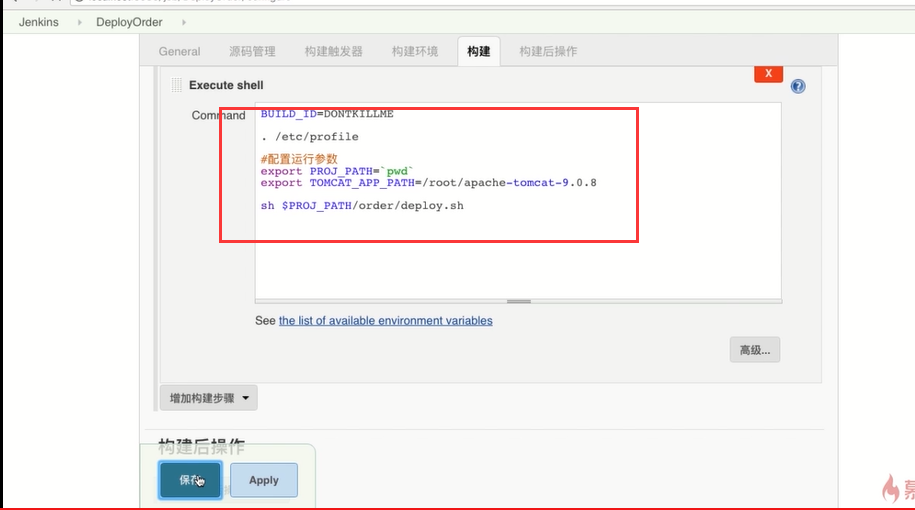












BUILD\_ID=DONTKILLME

这句话表示：当jenkins部署完server任务之后，server的线程不要被杀掉，

如果不加这句话，当jenkins执行完任务之后jenkins的进程会被自动杀掉，那么server也会被杀掉，

加上这句话jenkins的进程会被自动杀掉但是server不会被杀掉，这样才能实现jenkin执行完部署后jenkins进程关闭，而server进程一直存在的效果

. /etc/profile

表示：加载配置参数

# 5-3运行部署任务

执行jenkins部署任务

确定执行结果成功

打开浏览器访问部署应用程序

确定发布结果的正确性