# CentOS下安装JDK

# JDK简介

Java Development Kit（JDK）是太阳微系统针对Java开发人员发布的免费软件开发工具包（SDK,Software development kit）。自从Java推出以来,JDK已经成为使用最广泛的Java SDK。由于JDK的一部分特性采用商业许可证,而非开源[1]。因此,2006年太阳微系统宣布将发布基于GPL的开源JDK,使JDK成为自由软件。在去掉了少量闭源特性之后,太阳微系统最终促成了GPL的OpenJDK的发布。

作为Java语言的SDK,普通用户并不需要安装JDK来运行Java程序,而只需要安装JRE（Java Runtime Environment）。而程序开发者必须安装JDK来编译、调试程序。(下图,维基百科截图)

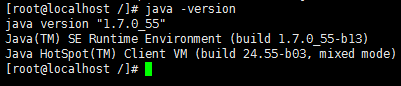


# 卸载JDK

查看系统是否已安装JDK,一般的linux都默认使用了开源的openJDK

**使用命令行查看JDK版本**

java -version



此表明有JDK,需要进行卸载

如果出现下图,说明服务器并没有JDK或者没有配置环境变量

graphic

**再使用以下命令查看是否存在java 和 jdk**

此方法较为保险,因为有些安装了JDK,却并没有配置环境变量

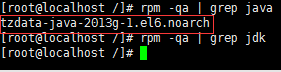
命令行:

rpm -qa|grep java

rpm -qa|grep jdk

**特殊情况**

很少出现这种情况,出现了用下方法解决,没出先忽略



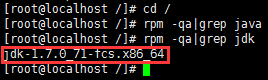
这种情况多出现在卸载系统默认的JDK时没有卸载干净

使用以下命令卸载:

rpm -e --nodeps tzdata-java-2013g-1.el6.noarch

卸载掉后使用上面命令就不会出现这个了

**正常情况**



上图表面查看到了有JDK,表示需要卸载

## rpm格式JDK或系统默认JDK卸载方式

**程序卸载指令,命令行:**

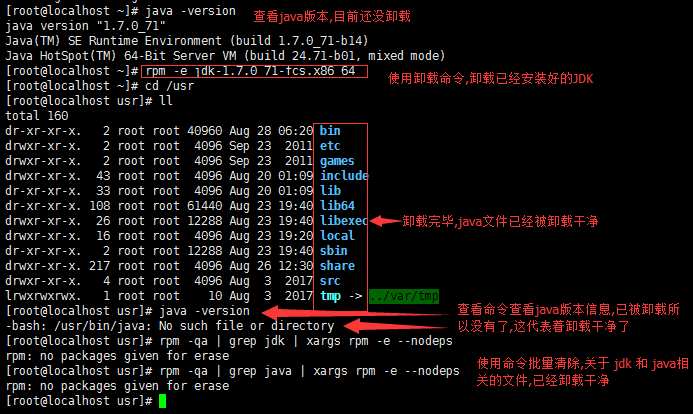
rpm -e jdk-1.7.0\_71-fcs.x86\_64

**批量卸载所有名字包含 jdk 的已安装程序**

命令行: rpm -qa | grep jdk | xargs rpm -e --nodeps

**批量卸载所有名字包含 java 的已安装程序**

命令行: rpm -qa | grep java | xargs rpm -e --nodeps



查看版本信息,是否卸载成功

java -version

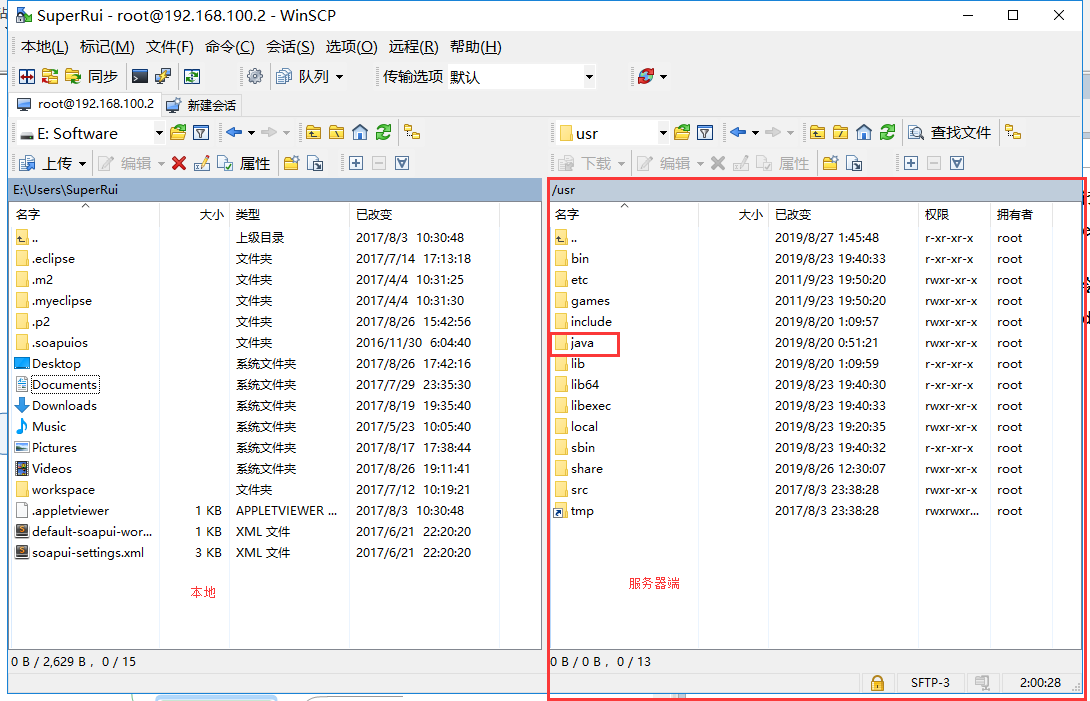
graphic

No such file or directory 表示卸载成功！

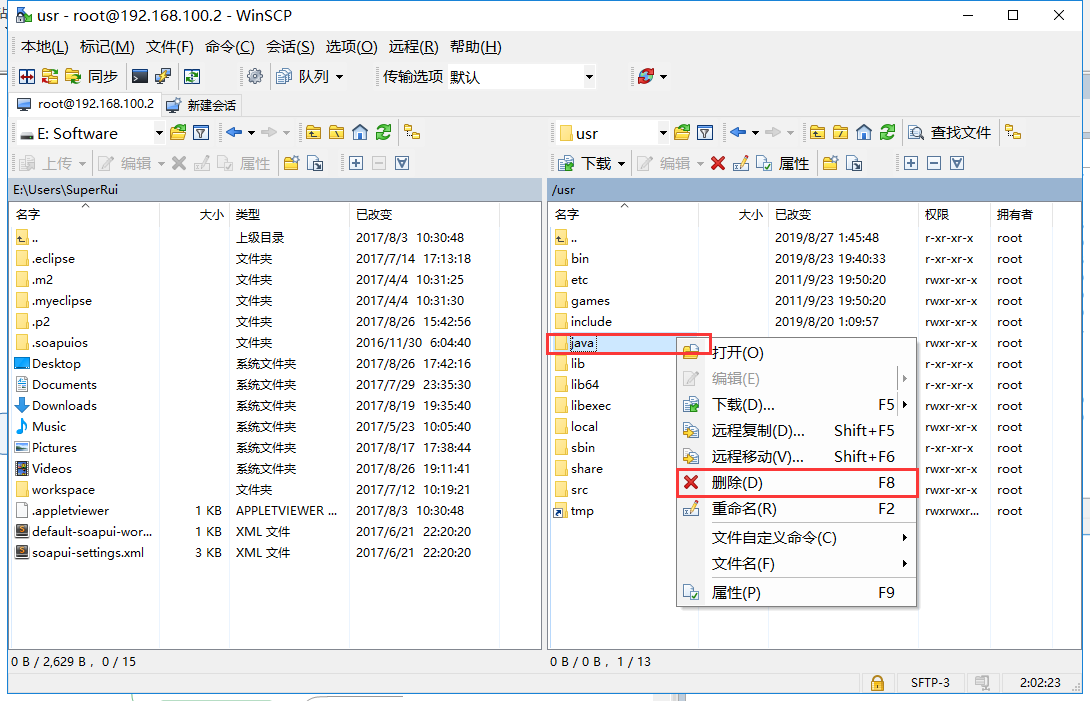
## tar.gz压缩格式的JDK卸载方式

**第一种方式,在Windows环境下可以使用WinSCP工具图形化卸载**

tar.gz普遍安装在/usr 目录下,所以进入/usr找到java的文件



**选中文件,右键选中删除即可,删除后刷新文件不复存在,表示删除成功**



**第二种方式,使用Xshell工具用命令卸载**

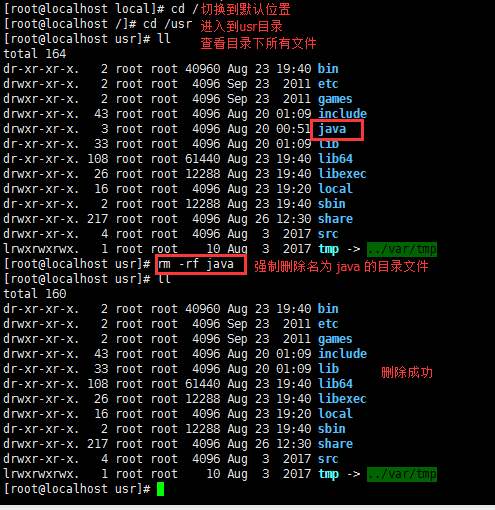
命令行:rm -rf java

rm:删除指令

-r:递归处理,将指定目录下的所有文件与子目录一并处理

-f:强制删除

java:目录下的文件



成功强制删除/usr下java文件成功

再次使用以下命令清理可能出现的相关的沉淀文件

**批量卸载所有名字包含 java 的已安装程序**

rpm -qa | grep java | xargs rpm -e --nodeps

graphic

**批量卸载所有名字包含 jdk 的已安装程序**

rpm -qa | grep jdk | xargs rpm -e --nodeps

graphic

查看版本信息,是否卸载成功

java -sersion

graphic

No such file or directory 表示卸载成功！

# 下载JDK

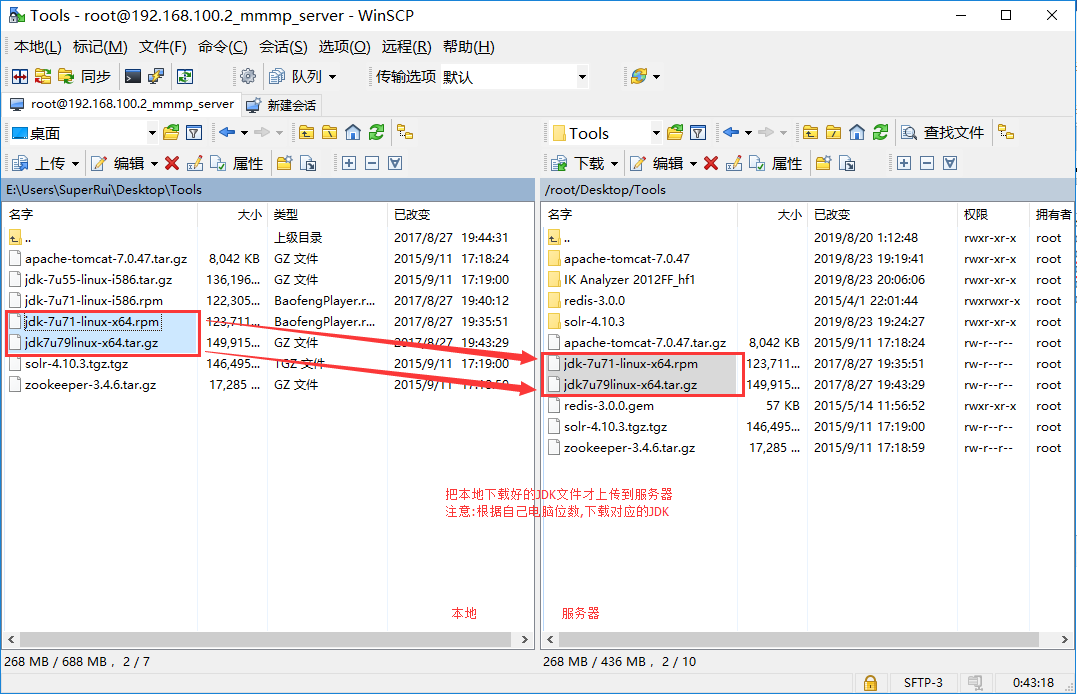
## 下载准备JDK文件,用WinSCP复制到服务器

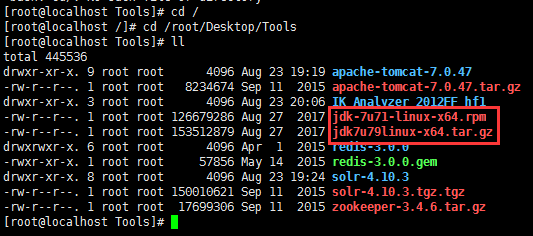
以下网址是官网现在JDK的url,最新的是1.8,如果下载1.7可以通过网上其他渠道找到下载

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

服务器系统:CentOS\_6.5\_x64,所以这里下载的64位的JDK

## 使用WinSCP工具将下载好的JDK文件复制上传到服务器上





JDK文件上传完毕,成功

# 安装JDK

## Windows环境下需要的工具

WinsSCP and Xshell 5

graphic

## .rpm格式JDK安装

**RPM简介**

RMP 是 LINUX 下的一种软件的可执行程序,你只要安装它就可以了。这种软件安装包通常是一个RPM包（Redhat Linux Packet Manager,就是Redhat的包管理器）,后缀是.rpm。

RPM是Red Hat公司随Redhat Linux推出了一个软件包管理器,通过它能够更加轻松容易地实现软件的安装。

1.安装软件：执行rpm -ivh rpm包名,如：

#rpm -ivh jk=7u71-linux-x64.rpm

2.升级软件：执行rpm -Uvh rpm包名。

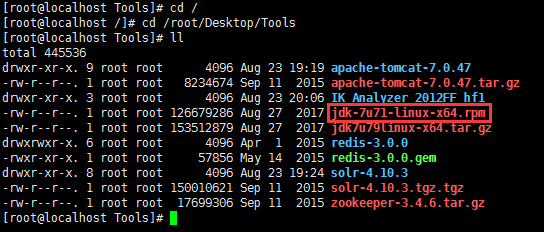
3.反安装：执行rpm -e rpm包名。

4.查询软件包的详细信息：执行rpm -qpi rpm包名

5.查询某个文件是属于那个rpm包的：执行rpm -qf rpm包名

6.查该软件包会向系统里面写入哪些文件：执行 rpm -qpl rpm包名

**使用Xshell工具通过命令,查看rmp格式JDK是否存在**



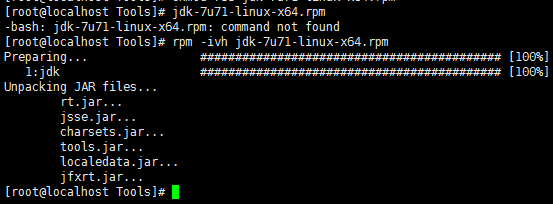
**使用命令给该将被操作的rpm格式的JDK文件设置权限,以免安装过程中出现权限文件**

命名行:chmod 755 jdk-7u71-linux-x64.rpm

graphic

使用rpm安装命令安装,执行rpm -ivh rpm包名

命名行: rpm -ivh jdk-7u71-linux-x64.rpm



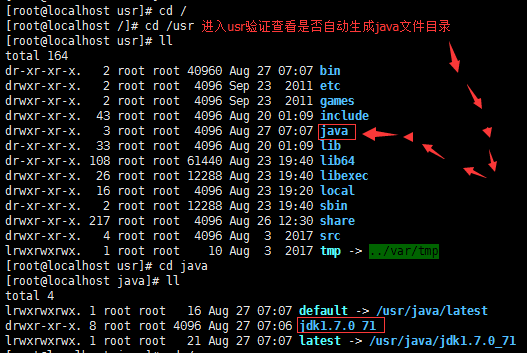
安装.rpm格式后缀的JDK不需要配置java环境变量,会自动配置,会在/usr/bin生成java、javac等JDK的快捷方式图标,这些图标会链接指向到/usr/java/jdk1.7.0\_71中

用echo $PATH查看环境变量,可以看到/usr/bin是加入了PATH环境变量中的

命令行： echo $PATH

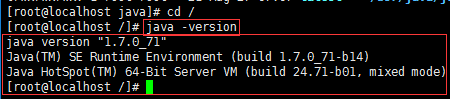
graphic

验证是否自动安装生成文件成功,下图表示自动安装成功



**使用基础的查看JDK版本命令才看环境变量是否配置成功**

命令行:java -version



rpm安装完毕,成功！

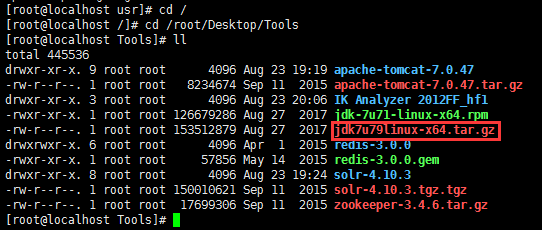
## tar.gz格式JDK安装

**tar.gz简介**

tar.gz为后缀的文件是一种压缩文件,在Linux和macOS下常见,Linux和macOS都可以直接解压使用这种压缩文件

.tar.gz一般情况下都是源代码的安装包,需要先解压再经过编译、安装.才能执行。总而言之它是一个压缩文件

**使用Xshell工具通过命令,查看tar.gz格式JDK是否存在**



使用Xshell工具链接服务器在/usr目录下创建名称为java的目录,并将上传好的JDK文件复制过去,/usr/java/jdk7u79linux-x64.tar.gz

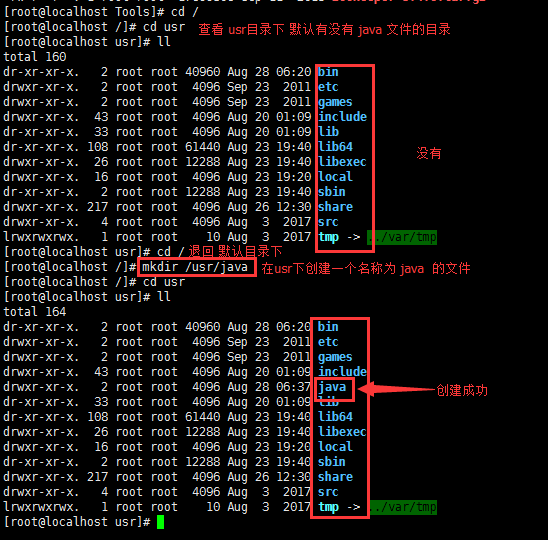
tar.gz安装不同于rpm安装,rpm自动安装好配置JDK信息

tar.gz安装需要手动创建相关的文件目录,普遍是在/usr目录下创建目录名称为java的文件,然后将下载好的JDK复制进去解压安装配置

**/usr目录下创建目录名称为java的文件**

命名行:

mkdir /usr/java



创建java目录文件成功,接下来将下载好的JDK文件使用命令行复制过去

**将准备好的JDK文件复制到 /usr/java 目录下**

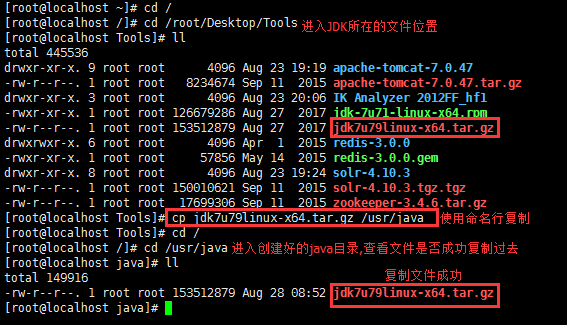
这种操作可以使用WinSCP这种方便的工具,尽量避免权限问题

这里使用Xshell工具

进入JDK所在目录,使用命名行:

cp jdk7u79linux-x64.tar.gz /usr/java

含义是将 jdk7u79linux-x64.tar.gz 复制到/usr/java 目录下



文件复制完成,解压复制过来的JDK文件

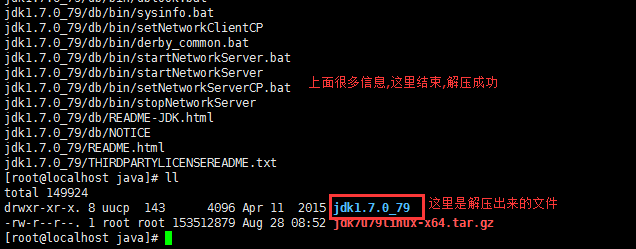
**解压文件**

解压已经成功复制过后的JDK文件,命名行:

tar xvf jdk7u79linux-x64.tar.gz

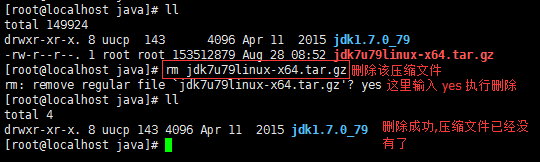
graphic

解压完毕,并查看java目录



解压成功,删除 jdk7u79linux-x64.tar.gz ,也可以留着

删除的命名行:rm jdk7u79linux-x64.tar.gz



## 配置服务器JDK环境变量

tar.gz格式文件不会想rpm安装文件那养自动配置好JDK环境变量,tar.gz格式文件需要手动配置

## 设置对所有用户有效的方式

修改 /etc/profile 目录文件系统变量,在下面末尾添加JDK环境变量,此方法对该服务器任何人有效,都可以使用JDK

使用VI编辑 /etc/profile 文件,在末尾添加以下配置参数

=============我是分割线他哥======================

# set java environment

export JAVA\_HOME=/usr/java/jdk1.7.0\_79

export JRE\_HOME=$JAVA\_HOME/jre

export PATH=$PATH:$JAVA\_HOME/bin

export CLASSPATH=./:$JAVA\_HOME/lib:$JAVA\_HOME/jre/lib

=============我是分割线他哥======================

命名行: vi /etc/profile

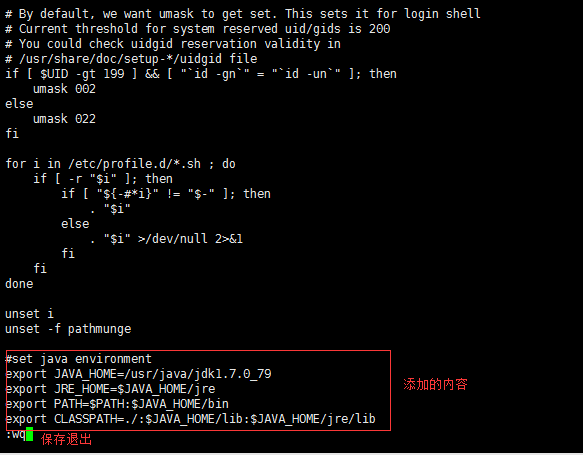
解锁大写建,按两下GG,跳到编辑文件的末尾

点击键盘 Insert 键位,开始编辑,添加上面分割线以内的内容

添加完毕后,点击键盘 Esc 退出编辑模式

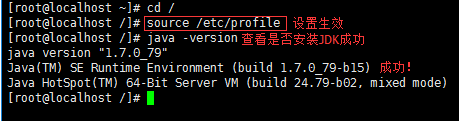
然后使用 Shift + : 随后输入 wq 然后回车,含义是保存退出

如果操作有问题, Ctrl + z 可以取消编辑并退出



输入使profile配置文件生效的命令: source /etc/profile

或者重启服务器即可,使用 source /etc/profile 无需重启

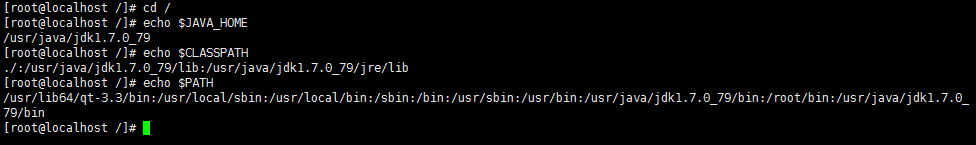


**使用查看环境变量配置,命令行如下:**

echo $JAVA\_HOME

echo $CLASSPATH

echo $PATH



tar.gz格式的JDK就此安装完毕,成功！

## 其他设置环境变量方式

**设置对当前用户有效方式**

修改.bashrc文件设置用户变量,设置jdk环境变量,该方式仅对当前用户有效,对其它用户无效,其它用户没有运行java命令的权限

命令行：vi ~/.bashrc

在bashrc文件中添加下面配置

=============我是分割线他哥======================

export JAVA\_HOME=/usr/java/jdk1.7.0\_79

export JRE\_HOME=$JAVA\_HOME/jre

export PATH=$PATH:$JAVA\_HOME/bin

export CLASSPATH=./:$JAVA\_HOME/lib:$JAVA\_HOME/jre/lib

=============我是分割线他哥======================

解锁大写建,按两下GG,跳到编辑文件的末尾

点击键盘 Insert 键位,开始编辑,添加上面分割线以内的内容

添加完毕后,点击键盘 Esc 退出编辑模式

然后使用 Shift + : 随后输入 wq 然后回车,含义是保存退出

如果操作有问题, Ctrl + z 可以取消编辑并退出

使bashrc配置生效,命令行: source ~/.bashrc

**设置对某个用户有效方式**

如果要使JDK对某个用户有效,某个用户具有运行java命令的权限,修改该用户的.bashrc文件。

例如：ROOT用户给Super用户名运行java命令的权限

命令行：vi /home/Super/.bashrc

在bashrc文件中添加下面配置

=============我是分割线他哥======================

export JAVA\_HOME=/usr/java/jdk1.7.0\_79

export JRE\_HOME=$JAVA\_HOME/jre

export PATH=$PATH:$JAVA\_HOME/bin

export CLASSPATH=./:$JAVA\_HOME/lib:$JAVA\_HOME/jre/lib

=============我是分割线他哥======================

解锁大写建,按两下GG,跳到编辑文件的末尾

点击键盘 Insert 键位,开始编辑,添加上面分割线以内的内容

添加完毕后,点击键盘 Esc 退出编辑模式

然后使用 Shift + : 随后输入 wq 然后回车,含义是保存退出

如果操作有问题, Ctrl + z 可以取消编辑并退出

如果当前登录用户是Super

使bashrc配置生效,命令行: source ~/.bashrc

the end..