

JAVA环境配置-Windows

一、安装 JDK

1.1 下载JDK

1.2 安装JDK

一、安装 JDK

1.1 下载JDK

打开浏览器，访问

<https://www.oracle.com/java/technologies/downloads/archive/>

2. 选择版本

选择适合你的 JDK 版本，我们建议使用11版本之前，如JAVA8，并下载适用于 Windows 的安装包（通常是 .exe 文件）。在网站右侧选择版本，选择windows 64位系统的安装包

Java SE

GraalVM

Java EE

Java ME

JavaFX

Java SE

Java Client Technologies

Java 3D, Java Access Bridge, Java Accessibility, Java Advanced Imaging, Java Internationalization and Localization Toolkit, Java Look and Feel, Java Media Framework (JMF), Java Web Start (JAWS), JIMI SDK

Java Platform Technologies

Java Authentication and Authorization Service (JAAS), JavaBeans, Java Management Extension (JMX), Java Naming and Directory Interface, RMI over IIOP, Java Cryptography Extension (JCE), Java Secure Socket Extension

Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files

The Java Cryptography Extension enables applications to use stronger versions of cryptographic algorithms. JDK 9 and later offer the stronger cryptographic algorithms by default.

The unlimited policy files are required only for JDK 8, 7, and 6 updates earlier than 8u161, 7u171, and 6u181. On those versions and later, the stronger cryptographic algorithms are available by default.

JVM Technologies

jvmsat

Java Database

Java DB Connectivity (JDBC), Java Data Objects (JDO)

Misc. tools and libraries

JDK Demos and Samples, Sample Code for GSSAPI/Kerberos, Java Communications API

Java SE downloads

> Java SE 24

> Java SE 23

> Java SE 22

> Java SE 21

> Java SE 20

> Java SE 19

> Java SE 18

> Java SE 17 (17.0.13 and later)

> Java SE 17 (17.0.12 and earlier)

> Java SE 16

> Java SE 15

> Java SE 14

> Java SE 13

> Java SE 12

> Java SE 11

> Java SE 10

> Java SE 9

> Java SE 8 (8u211 and later)

> Java SE 8 (8u202 and earlier)

> Java SE 7

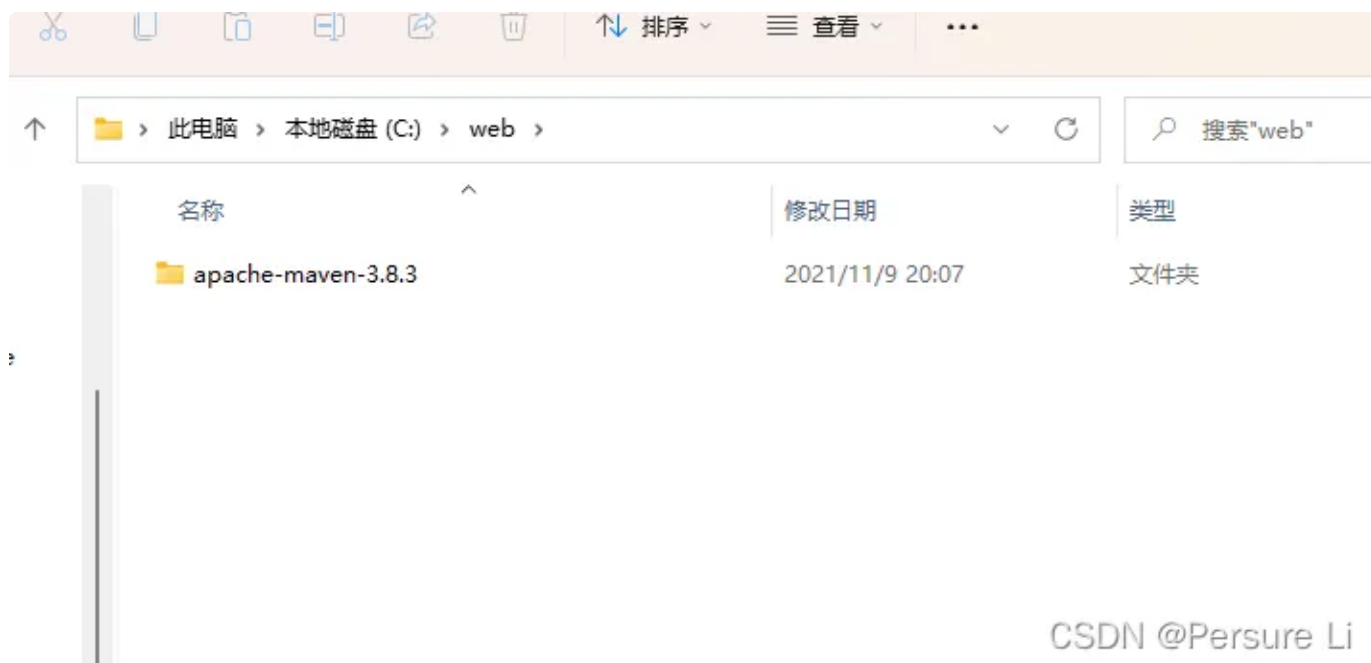
> Java SE 6

> Java SE 5

1.2安装JDK

开发环境能放到自定义目录就放到自定义目录，这样才能更好的管理，所以在开始之前，在C盘目录下新建一个 **web** 目录，用于存放Java环境。

新建的web目录（apache-maven文件请忽略，和本次实验无关）



CSDN @Persure Li

下载完成后，双击下载的 .exe 文件启动安装程序。

1.运行安装程序



点击更改：

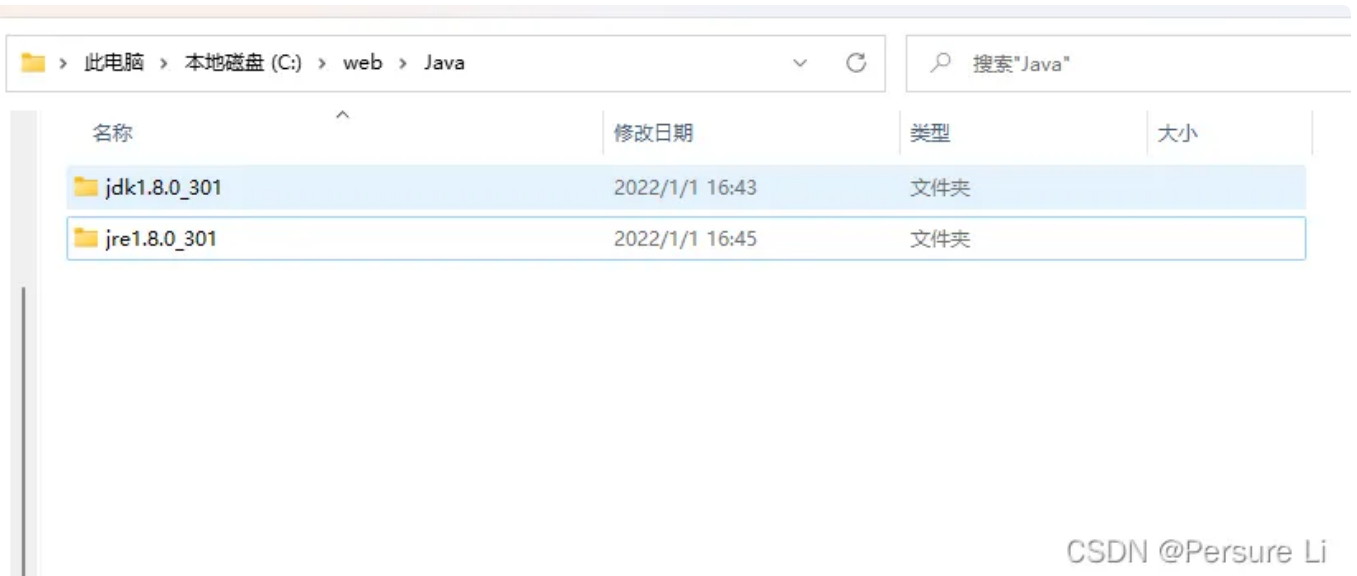


将文件目录修改成: `C:\web\Java\jdk1.8.0_301\`



(注意这个文件夹的位置，后面配置环境变量时要用到)

点击确定并下一步，等待安装，此时在 `C:\web\Java\` 下面创建文件夹 `jre1.8.0_301`，与 `C:\web\Java\jdk1.8.0_301\` 平级：



CSDN @Persure Li

将此目录更改成创建的目录，点击更改选择即可



CSDN @Persure Li

3.完成安装

按照提示完成安装过程，点击“Finish”退出安装向导。



4.配置环境变量

设置 JAVA_HOME

右键点击“此电脑”或“计算机”，选择“属性” -> “高级系统设置” -> “环境变量”；



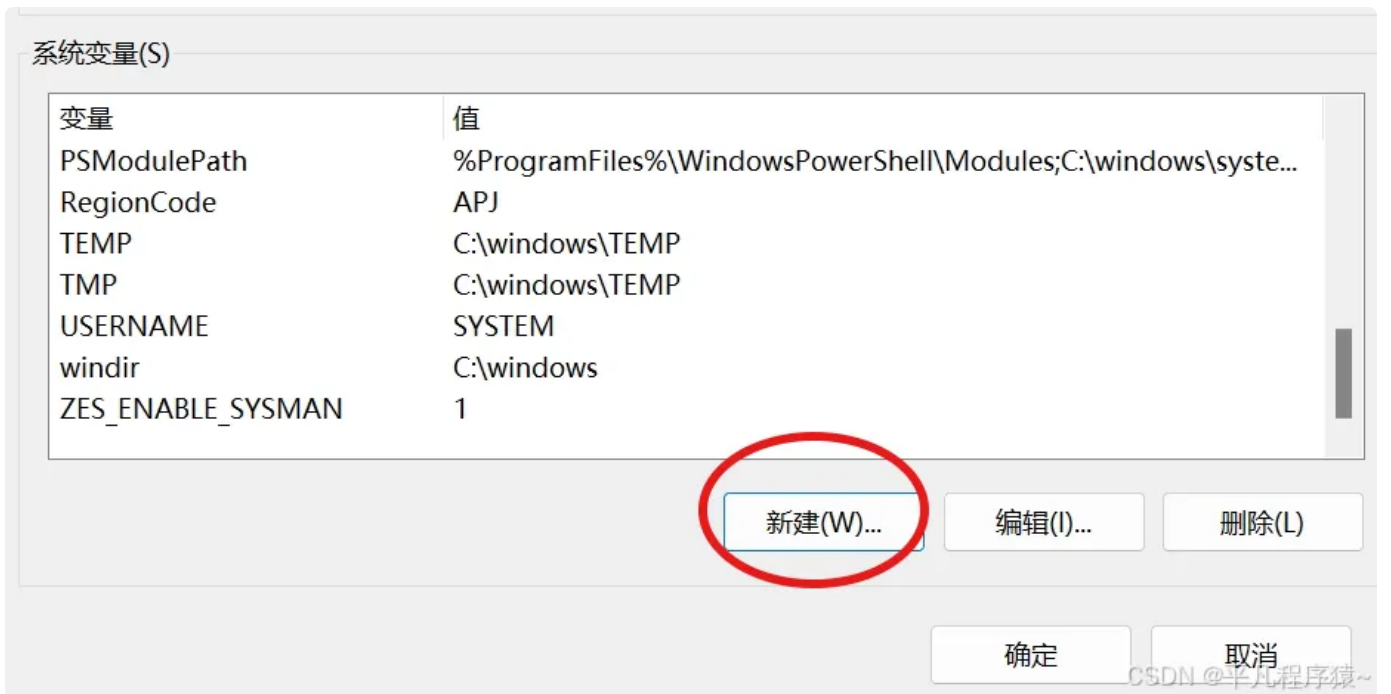


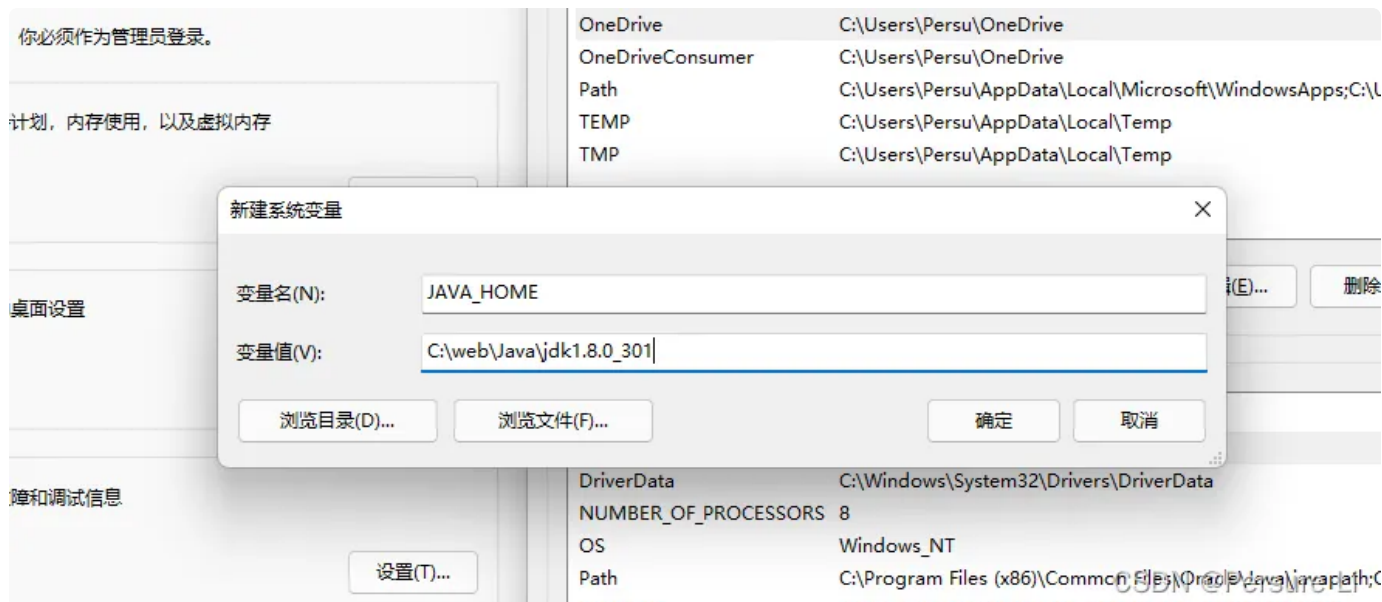
(也可以在控制面板直接搜索“环境变量”)



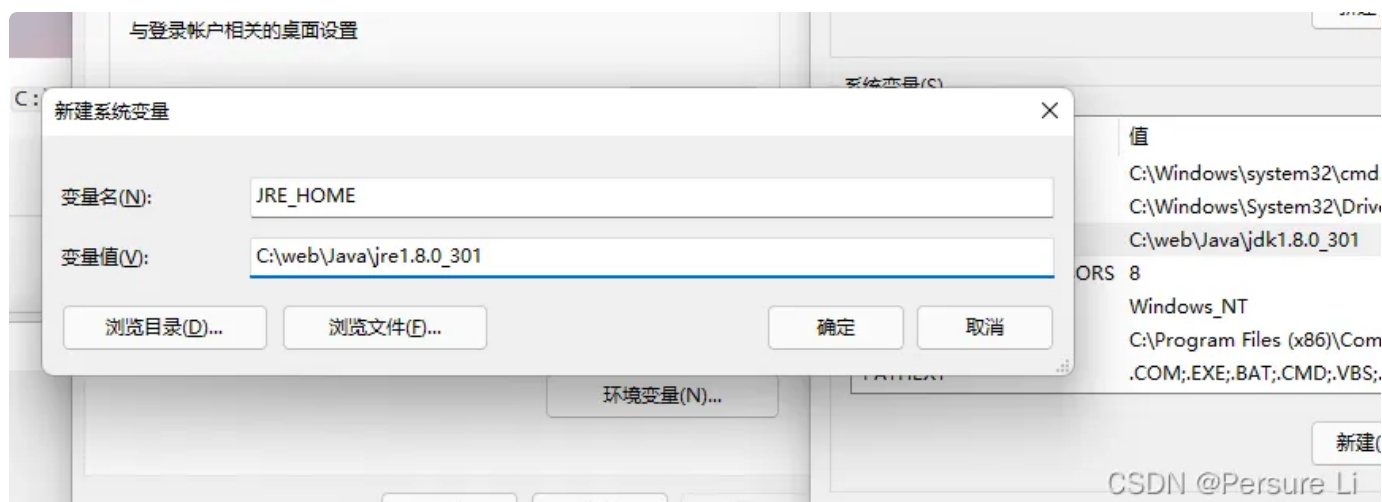
在系统变量中，点击“新建”，添加 JAVA_HOME 变量，值为 JDK 安装目录（例如 C:\web\Java\jdk1.8.0_301）；

请按照自己的安装目录来填写！下同

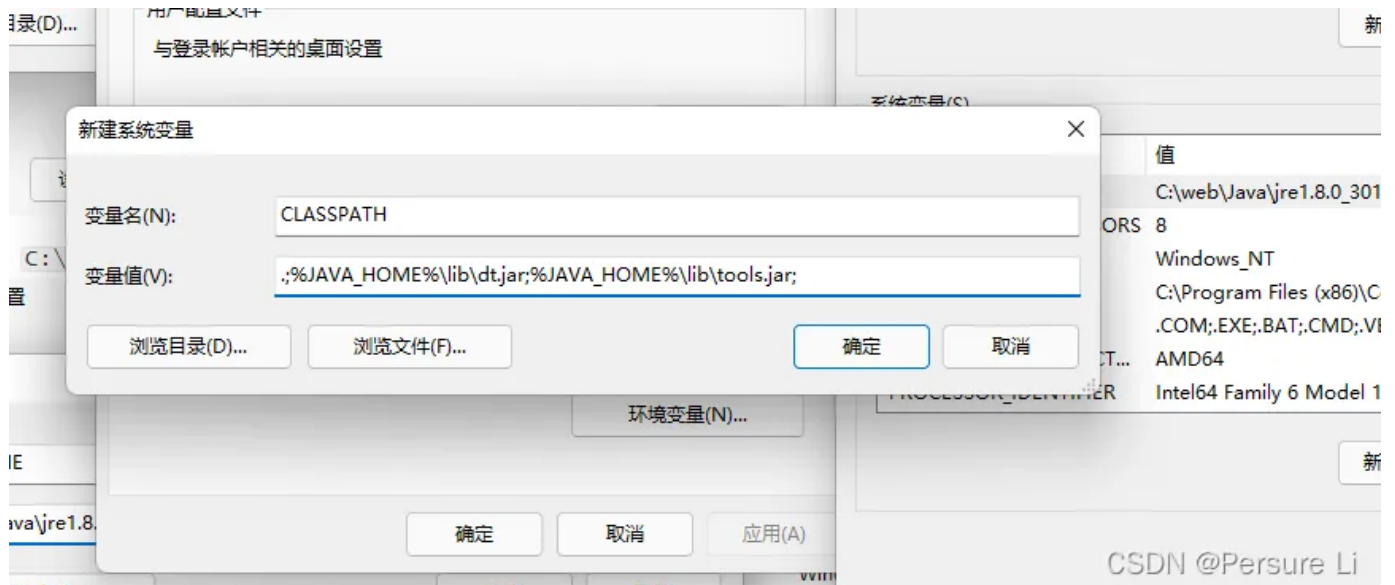




变量名: **JRE_HOME** , 变量值: **C:\web\Java\jre1.8.0_301** , 点击确定;

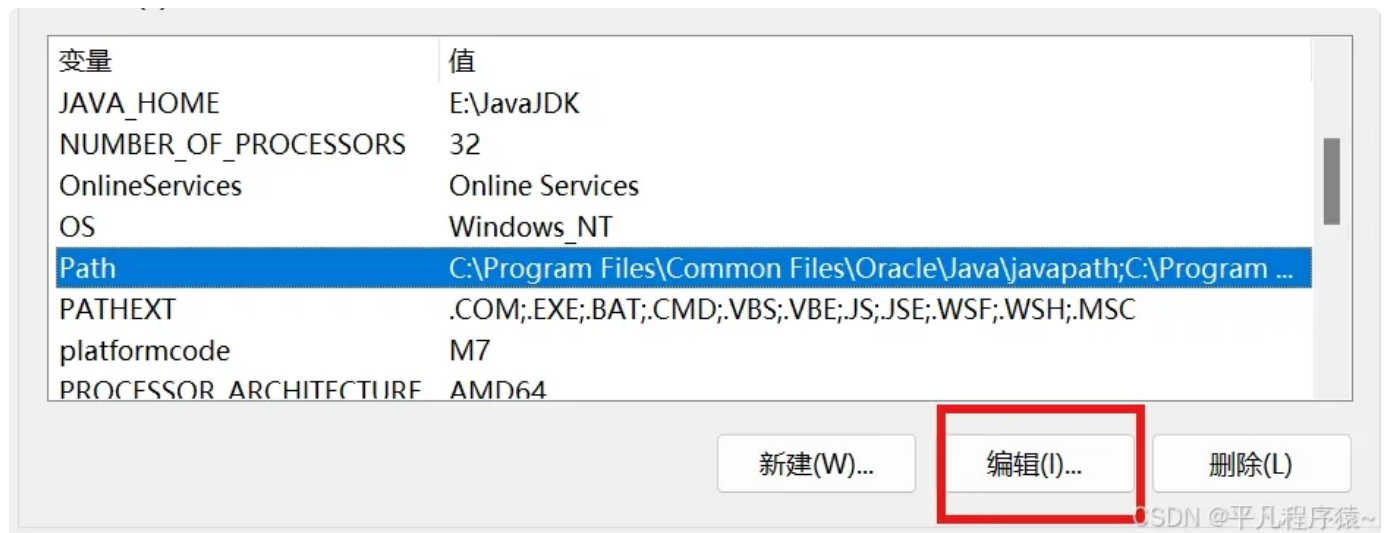


变量名: **CLASSPATH** , 变量值: **.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar;**
, 注意变量值的前面有一个 **.** , 点击确定;

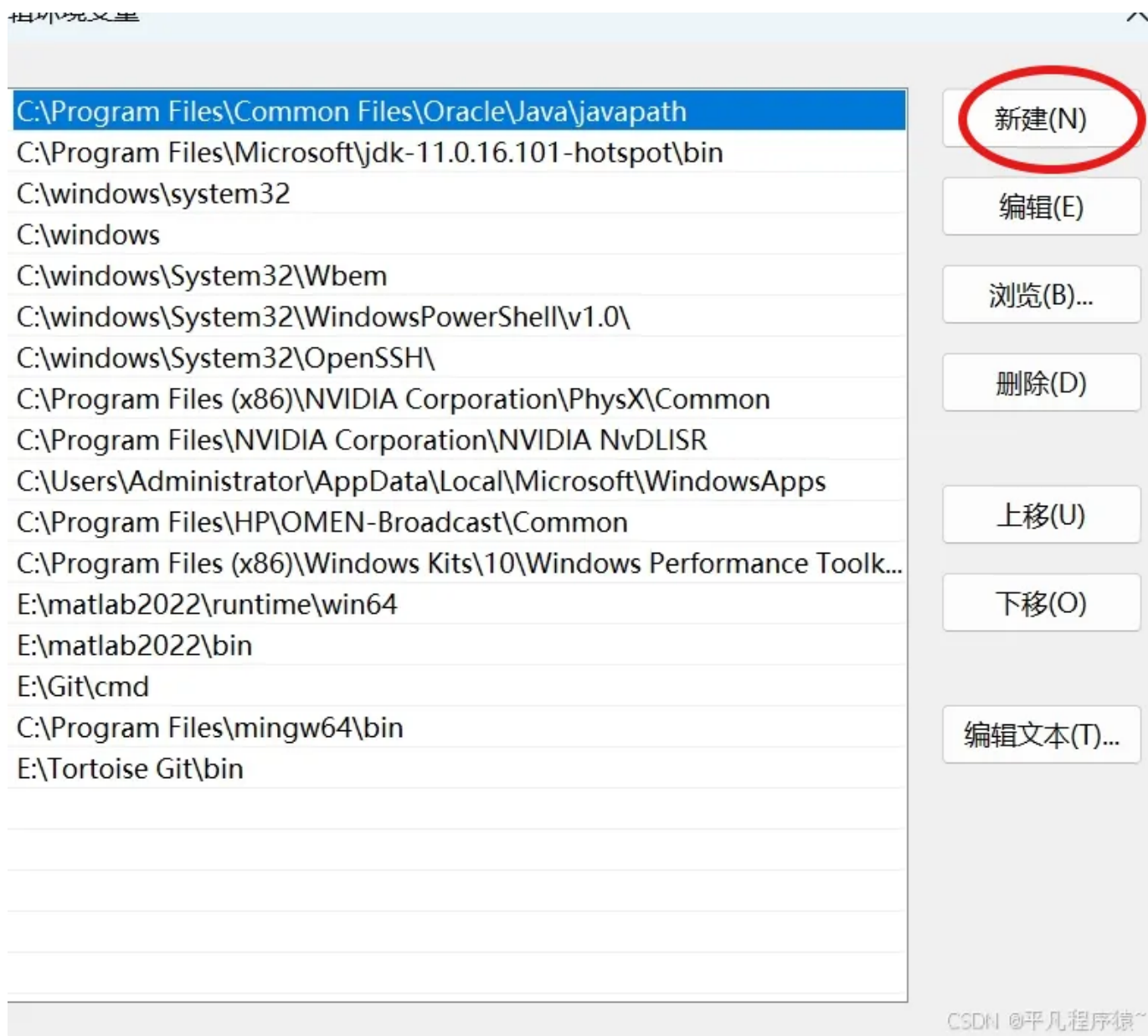


5.更新 PATH 变量

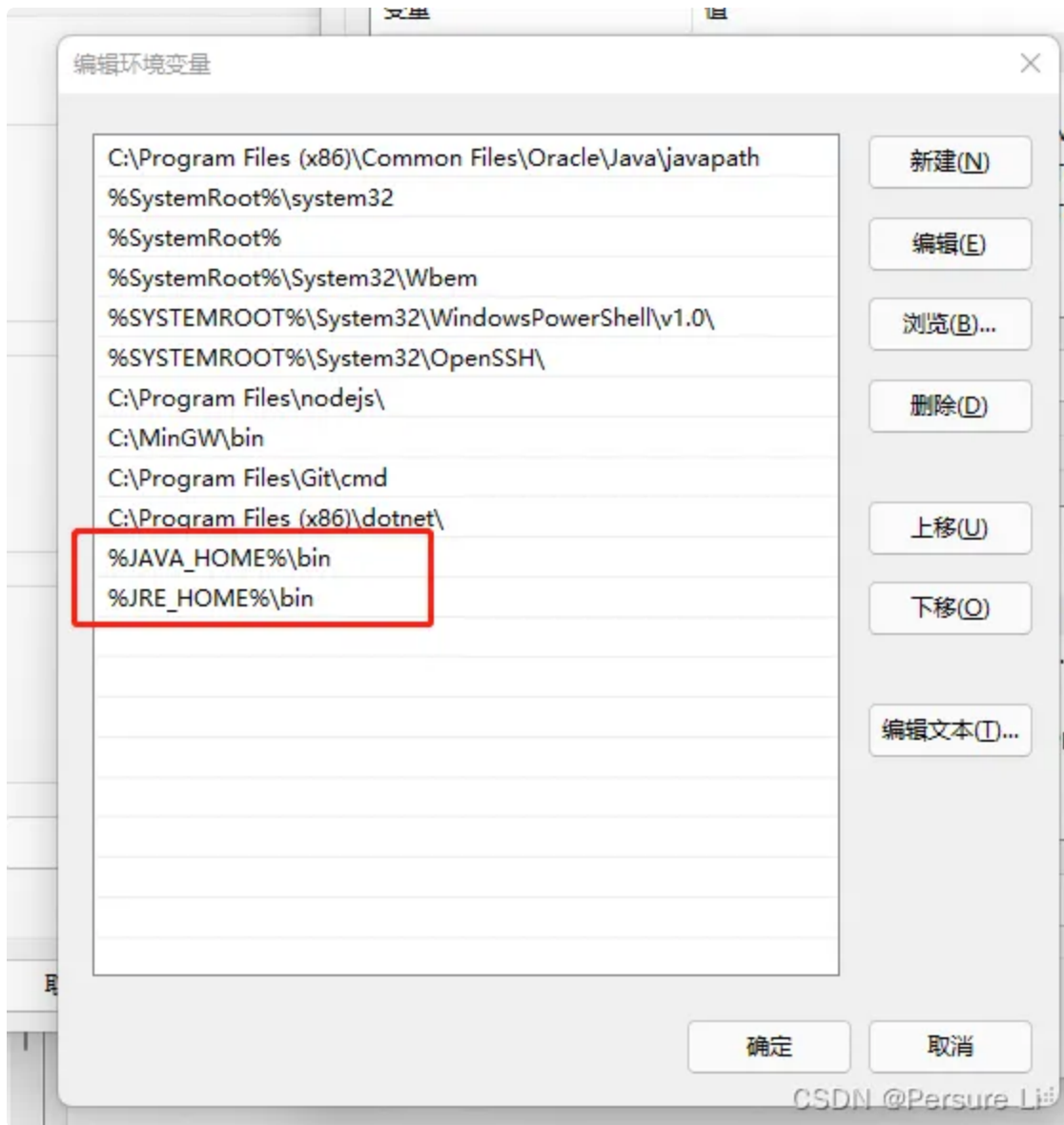
在“系统变量”中找到 Path 变量，点击“编辑”；



再点击“新建”；

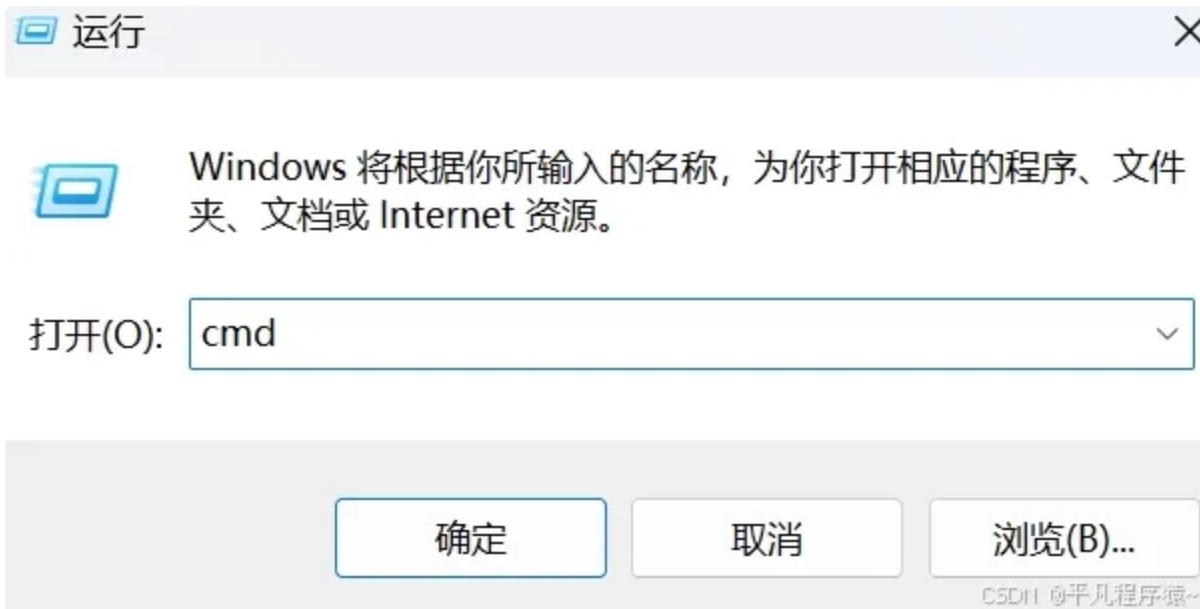


点击新建添加两个变量值： `%JAVA_HOME%\bin` ， `%JRE_HOME%\bin`



6.验证安装

按下“Win+R”键，输入“cmd”打开命令提示符；



输入“java”后回车看看是否显示出如下面样式；

```
C:\Users\qwer>java
用法: java [options] <主类> [args...]
      (执行类)
或 java [options] -jar <jar 文件> [args...]
      (执行 jar 文件)
或 java [options] -m <模块>[/<主类>] [args...]
   java [options] --module <模块>[/<主类>] [args...]
      (执行模块中的主类)
或 java [options] <源文件> [args]
      (执行单个源文件程序)

将主类、源文件、-jar <jar 文件>、-m 或
--module <模块>/<主类> 后的参数作为参数
传递到主类。

其中, 选项包括:

-cp <目录和 zip/jar 文件的类搜索路径>
-classpath <目录和 zip/jar 文件的类搜索路径>
--class-path <目录和 zip/jar 文件的类搜索路径>
              使用 ; 分隔的, 用于搜索类文件的目录, JAR 档案
              和 ZIP 档案列表。
-p <模块路径>
--module-path <模块路径>...
              ; 分隔的元素列表, 每个元素都是
              模块或包含模块的目录的文件路径。每个模块都是
              模块化 JAR 或展开的模块目录。
CSDN @平凡程序猿~
```

再输入“javac”验证；

```

C:\Users\qwer>javac
用法: javac <options> <source files>
其中, 可能的选项包括:
  @<filename>                从文件读取选项和文件名
  -Akey[=value]              传递给批注处理程序的选项
  --add-modules <模块>(<模块>)*
                             除了初始模块之外要解析的根模块; 如果 <module>
                             为 ALL-MODULE-PATH,
                             则为模块路径中的所有模块。
  --boot-class-path <path>, -bootclasspath <path>
                             覆盖引导类文件的位置
  --class-path <path>, -classpath <path>, -cp <path>
                             指定查找用户类文件和批注处理程序的位置
  -d <directory>             指定放置生成的类文件的位置
  -deprecation                输出使用已过时的 API 的源位置
  --enable-preview            启用预览语言功能。要与 -source 或 --release 一起使用。
  -encoding <encoding>       指定源文件使用的字符编码
  -endorseddirs <dirs>        覆盖签名的标准路径的位置
  -extdirs <dirs>             覆盖所安装扩展的位置
  -g                           生成所有调试信息
  -g:{lines,vars,source}      只生成某些调试信息
  -g:none                     不生成任何调试信息
  -h <directory>              指定放置生成的头文件的位置
  --help, -help, -?           输出此帮助消息
  --help-extra, -X            输出额外选项的帮助
  -implicit:{none,class}      指定是否为隐式引用文件生成类文件
  -J<flag>                    直接将 <标记> 传递给运行时系统
  --limit-modules <模块>(<模块>)*

```

CSDN @平凡程序猿~

最后输入“java -version”(中间有一个空格)

```

C:\Users\qwer>java -version
java version "21.0.4" 2024-07-16 LTS
Java(TM) SE Runtime Environment (build 21.0.4+8-LTS-274)
Java HotSpot(TM) 64-Bit Server VM (build 21.0.4+8-LTS-274, mixed mode, sharing)
C:\Users\qwer>

```

CSDN @平凡程序猿~

到这里就说明JDK配置成功了。