Path：当用javac、java等命令编译、运行Java程序时，操作系统需要查找javac.exe、java.exe这些可执行文件的位置，然后调用这些可执行程序对Java程序进行对应的操作。我们开发程序代码之后，通常在代码文件夹所处的路径编译代码文件，操作系统会在用户所在当前路径和系统环境变量这两个地方查找对应的Java可执行文件，但是编译、运行Java代码的可执行文件所在的路径（即我们安装JDK的bin所在的路径）通常跟我们编写的代码是不再同一个文件夹目录下的，所以只能将其添加到系统的环境变量中，才能够找到对应的可执行文件，从而顺利编译、运行我们的代码。

JAVA\_HOME：直接在Path中编辑JDK安装的绝对路径，对大多数Java应用程序以及我们自己编写的Java代码都是可以正常编译、运行的，但是如果我们对本机安装的JDK进行版本修改或者升级，甚至对其安装路径进行改动，我们就需要更改Path中相应的内容。如果不小心误删了其它部分，很有可能导致我们的操作系统无法正常运行甚至崩溃，因此将bin之前的路径另外赋值给JAVA\_HOME是很有必要的。

CLASSPATH：classpath是一个环境变量，是给谁指路的呢？是给“类加载器”指路的。而且classpath环境变量不属于Windows操作系统，classpath环境变量隶属于Java，是Java特有的。我们把classpath配置一下（注意这个环境变量在windows中没有，需要新建），桌面-->计算机/此电脑-->右键-->属性-->高级系统设置-->环境变量-->新建-->classpath=A路径;B路径;C路径...。这里classpath是一个变量名，A路径;B路径;C路径.....是变量值。

Classpath：.;%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\lib\tools.jar

首先，我们一起来研究一下，“java HelloWorld”的执行过程以及原理。

第一步：会先启动JVM（java虚拟机）。

第二步：JVM启动之后，JVM会去启动类加载器（classloader）。类加载器的作用：加载类的。本质上类加载器负责去硬盘上找“类”对应的“字节码”文件。举个例子,假设是“java HelloWorld”,那么类加载器会去硬盘上搜索：HelloWorld.class文件。

第三步：类加载器如果在硬盘上找不到对应的字节码文件，会报错，那报的什么错？“错误: 找不到或无法加载主类”。类加载器如果在硬盘上找到了对应的字节码文件,类加载器会将该字节码文件装载到JVM当中，JVM启动“解释器”将字节码解释为“101010000...”这种二进制码,操作系统执行二进制码和硬件交互。

这里有个问题，默认情况下，类加载器去硬盘上找“字节码”文件的时候，默认从哪找？默认情况下类加载器（classloader）会从当前路径下找。那么我们能不能给类加载器指定一个路径，让类加载器去指定的路径下加载字节码文件呢？答案是可以的，但是我们需要设置一个环境变量，叫做：classpath。