

●Java 语言基础(JDK 的下载和安装过程图解)(了解)

- A:JDK 的下载
 - a:官网 <http://www.oracle.com>
 - B:JDK 的安装
 - a:傻瓜式安装
 - 双击安装程序，然后一路 next 即可
 - b:安装的推荐方式
- C:验证安装是否成功

常见的 DOS 命令讲解

进入 DOS 窗口：

- a:通过 DOS 命令，切换到 JDK 安装的 bin 目录下。
 - C:\Program Files\Java\jdk1.8.0_131\bin
- b:然后分别输入 javac 和 java，如果正常显示一些内容，说明安装成功

●path 环境变量的作用及配置方式 1(掌握)

- A:在 JDK 的 bin 目录下开发程序容易产生问题
 - a:如果文件过多，不方便管理
 - b:删除自己写过的不需要的文件，可能不小心把 JDK 自带的工具给删除了
- B:如何解决问题呢
 - notepad 这样的命令为什么在任何路径下都能够执行, 配置 path 环境变量
- C:配置方式

●classpath 环境变量的作用及其配置(了解)

- A:为什么要配置 classpath
- B:classpath 配置的原理
- C:如何配置 classpath
- path 和 classpath 的区别(面试的时候可能会问)
 - path 配置的是可执行的文件.exe (javac 编译工具 和 java 运行工具),配置后可以在不同的盘符下访问 path 路径下的可执行文件
 - classpath 配置的 java 环境搜索类和其他资源文件 (比如 jar\zip 等资源) 的路径

●JDK 安装路径下的目录解释(了解)

- a:bin 目录: 该目录用于存放一些可执行程序。
 - 如 javac.exe (java 编译器)、java.exe (java 运行工具), jar.exe (打包工具)和 javadoc.exe (文档生成工具)等。
- b:jre 目录: "jre"是 Java Runtime Environment 的缩写,意为 Java 程序运行时环境。此目录是 Java 运行时环境的根目录,它包括 Java 虚拟机,运行时的类包,Java 应用启动器以及一个 bin 目录,但不包含开发环境中的开发工具。
- c:include 目录: 由于 JDK 是通过 C 和 C++实现的,因此在启动时需要引入一些 C 语言的头文件,该目录就是用于存放这些头文件的。
- d:lib 目录: lib 是 library 的缩写,意为 Java 类库或库文件,是开发工具使用的归档包文件。
- e:lib 目录下的 src.zip 文件:src.zip 为 src 文件夹的压缩文件,放置的是 JDK 核心类的源代码,通过该文件可以查看 Java 基础类的源代码。
- f:jdk11 内含有 JavaFX 是一个强大的图形和多媒体处理工具包集合,它允许开发者来设计、创建、测试、调试和部署富客户端程序,并且和 Java 一样跨平台
- g:jmods 模块系统,在 Java 中支持模块化程序设计
- h:legal 每个模块的版权和许可文件

●Java 开发工具工具介绍(了解)

- A:windows 系统自带的“文本文档”
- B:高级文本 Editplus/Notepad 等等
- C:集成开发软件: IntelliJ Idea2019、Eclipse

操作 DOS 命令行

win+r 键, 输入 cmd 进入 dos 命令行界面



操作命令:

cd\ :回到当前磁盘的根目录

tab: 补全命令

cd 目录名称 : 表示进入指定的目录中

dir:列出当前目录下所有的子目录

●第一个 java 程序: HelloWorld 案例的编写和运行(掌握)

- A:定义类 class
- B:写 main 方法(主方法)
- C:写输出语句
- D:Java 程序开发运行与工作原理
- E:编译和运行程序

```
public class HelloWorld{           //这是注释, 后面的内容忽略; HelloWorld 这个是取了一个类名,
要跟文件名一致, 大小写也要一致; class 类
```

```
public static void main(String[] args) { //这是 java 程序的运行入口（也称为 main 方法）
String s 要大写
    //输出一句话
    System.out.println("你好 JAVA!"); //这是一个打印语句，往控制台输出信息，打印语
句结尾要用英文输入法的分号结尾
}
}
```

// public : 公共的
// class:类
// static :静态的
// void :没有返回值
// String[] args :是一个数组
// System:系统
// out:输出
// println: 打印并换行
// 注意:

环境配置

方式一:

1 我的电脑右键选属性

2 选高级系统设置

系统

查看有关计算机的基本信息

Windows 版本

Windows 10 专业版

© 2018 Microsoft Corporation。保留所有权利。

系统

处理器: Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz 2.30 GHz

已安装的内存(RAM): 16.0 GB (15.8 GB 可用)

系统类型: 64 位操作系统, 基于 x64 的处理器

笔和触控: 没有可用于此显示器的笔或触控输入

计算机名、域和工作组设置

计算机名: shenlan [更改设置](#)

计算机全名: shenlan

计算机描述: shenlan

工作组: WORKGROUP

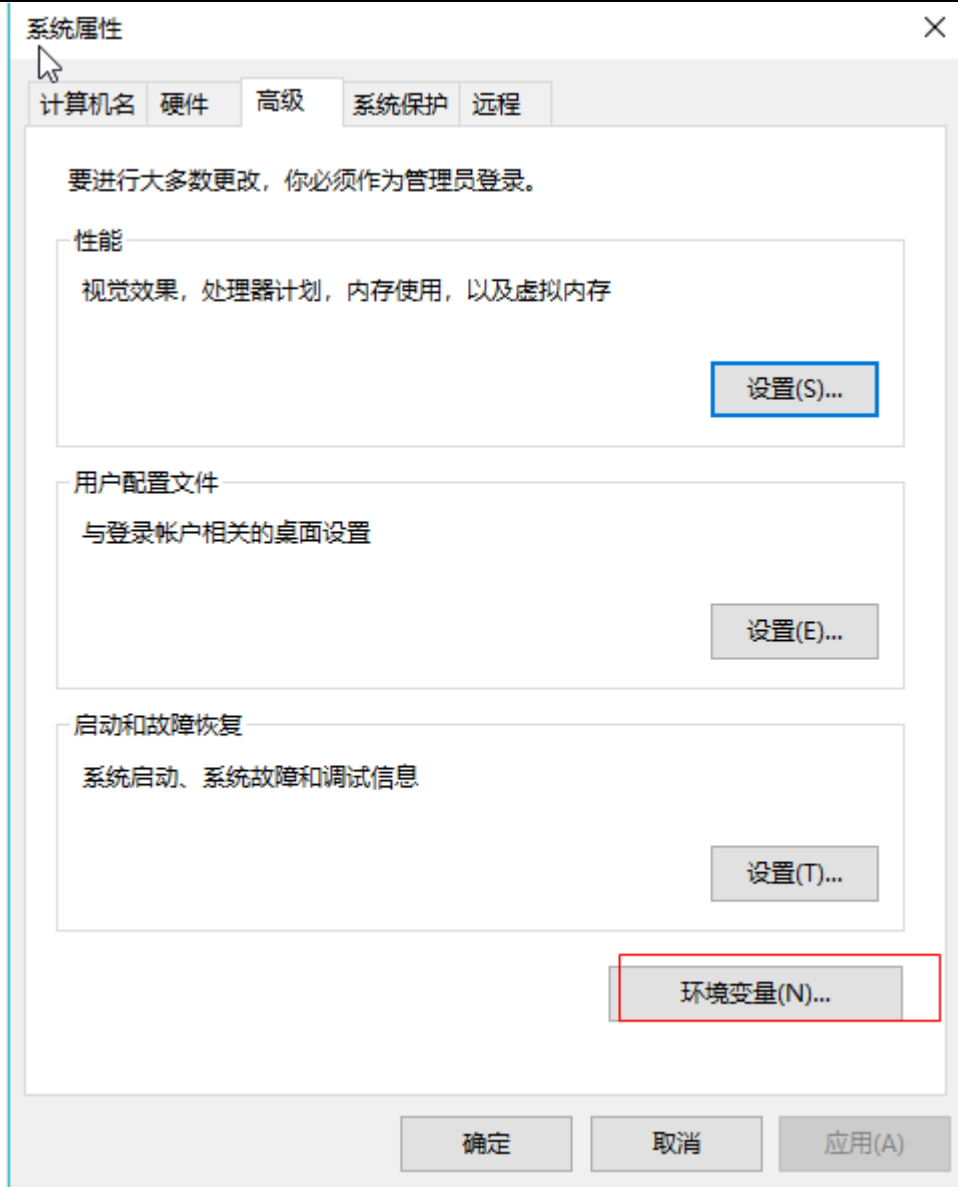
Windows 激活

Windows 已激活 [阅读 Microsoft 软件许可条款](#)

产品 ID: 00331-10000-00001-AA504 [更改产品密钥](#)

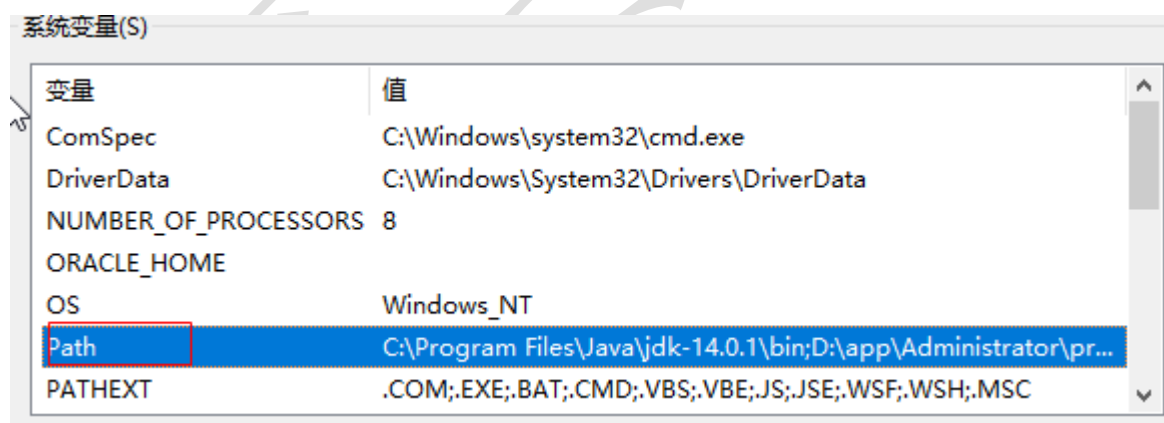
另请参阅

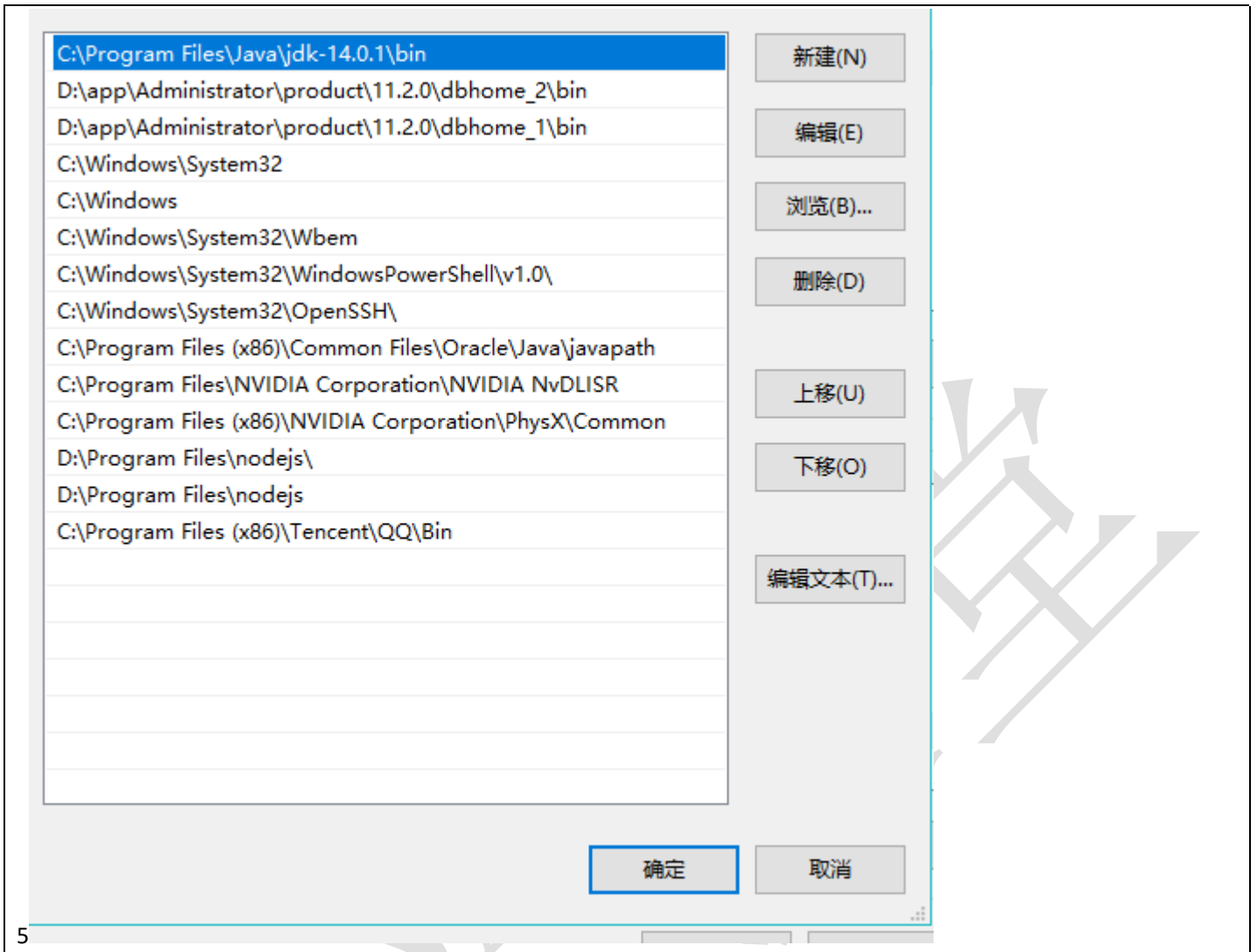
[安全和维护](#)



3.

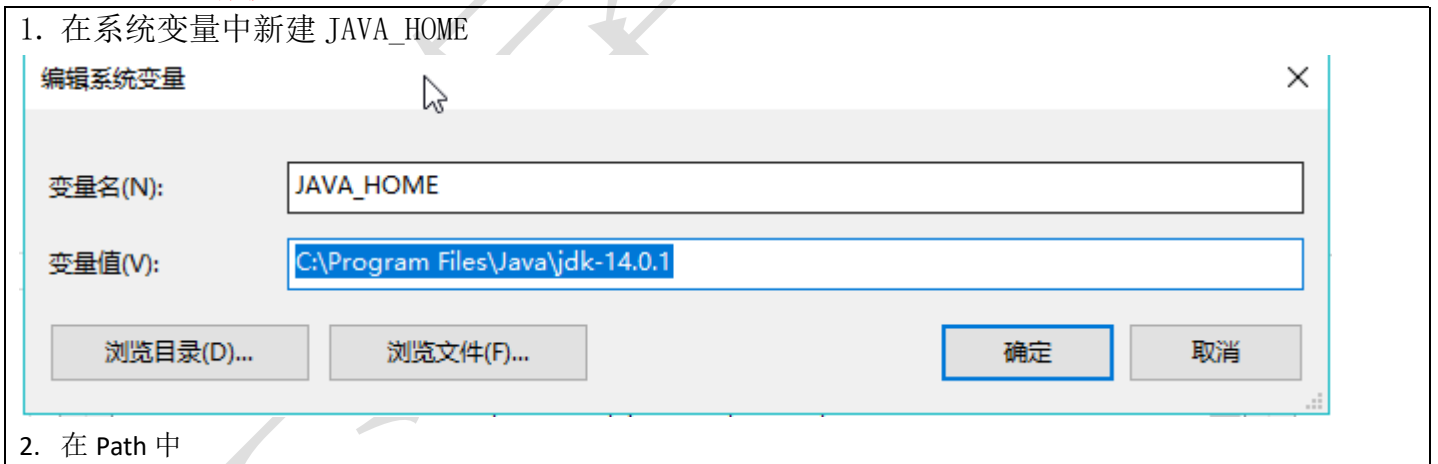
4





方法二：（推荐使用）

1. 在系统变量中新建 JAVA_HOME



编辑环境变量

%JAVA_HOME%\bin

D:\app\Administrator\product\11.2.0\dbhome_2\bin

D:\app\Administrator\product\11.2.0\dbhome_1\bin

C:\Windows\System32

C:\Windows

C:\Windows\System32\Wbem

C:\Windows\System32\WindowsPowerShell\v1.0\

C:\Windows\System32\OpenSSH\

C:\Program Files (x86)\Common Files\Oracle\Java\javapath

C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR

C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common

D:\Program Files\nodejs\

D:\Program Files\nodejs

C:\Program Files (x86)\Tencent\QQ\Bin

新建(N)

编辑(E)

浏览(B)...

删除(D)

上移(U)

下移(O)

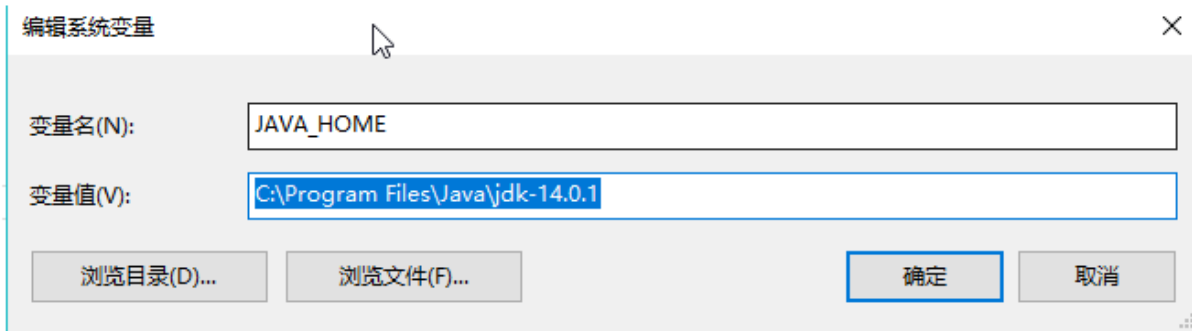
编辑文本(T)...

确定

取消

%JAVA_HOME%的解释:

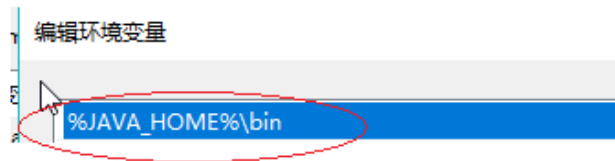




JAVA_HOME: C:\Program Files\Java\jdk-14.0.1

C:\Program Files\Java\jdk-14.0.1\bin

%JAVA_HOME% 是引入 \bin



● HelloWorld 案例常见问题(掌握)

- A: 找不到文件
 - a: 文件扩展名隐藏导致编译失败
 - b: 文件名写错了
- B: 单词拼写问题
 - a: class 写成 Class
 - b: String 写成 string
 - c: System 写成 system
 - d: main 写成 mian
- C: 括号匹配问题

- a:把类体的那对大括号弄掉一个
- b:把方法体的那对大括号弄掉一个
- c:把输出语句的那对小括号弄掉一个
- D:中英文问题
 - 注意: java 编程中需要的基本上都是英文字符

●注释概述及其分类(掌握)

- A:什么是注释
- B:注释的分类及讲解
 - 文档注释目前不讲, 说后面讲解
- 注释的作用
 - A:解释说明程序
 - B:帮助我们调试错误

```
public class HelloJava {  
    public static void main(String[] args) {  
        System.out.println("你好 java");  
        //      System.out.println(1/0);  // 选定行注释快捷键: Ctrl+/  
    }  
}  
  
// 单行, 双斜杠后整行数据都会被编译器忽略, 可以嵌入多个单行注释  //  
/*  
这是多行注释, 为了增加代码的可读性  
在注释之间的任何内容都被忽略  
多行注释里可以嵌套单行注释  
//  
//  
//  
多行注释不能嵌套  
*/
```

●关键字的概述和使用(掌握)

- A:什么是关键字
 - 被 Java 语言赋予特定含义的单词
- B:关键字的特点
 - 组成关键字的字母全部小写（一般情况下）
- C:常见关键字
 - `public static void class` 等
- D:关键字的注意事项
 - `goto` 和 `const` 作为保留字存在, 目前并不使用, 类似 Editplus 这样的高级记事本, 针对关键字有特殊的颜色标记, 非常直观

●标识符的概述和组成规则(掌握)

- A:什么是标识符
 - 就是给类, 接口, 方法, 变量等起名字时使用的字符序列
- B:标识符的组成规则
 - 英文大小写字母
 - 数字字符
 - `$`和`_`
- C:标识符注意事项
 - 1, 不能使用关键字
 - 2, 不能数字开头

HelloWord	Hello word
113	2b
Demo	Demo#
Bj_czd&45495	a34.7

●标识符中常见的命名规则(了解)

- 见名知意
- A:包
 - 最好是域名倒过来, 要求所有的字母小写 `com.baidu.user`
- B:类或者接口
 - 如果是一个单词首字母大写
 - 如果是多个单词每个单词首字母大写(驼峰标识) `StudentName`(大驼峰标识)
- C:方法或者变量
 - 如果是一个单词全部小写
 - 如果是多个单词, 从第二个单词首字母大写(`doThing`) (小驼峰原则)
- D:常量
 - 如果是一个单词, 所有字母大写
 - 如果是多个单词, 所有的单词大写, 用下划线区分每个单词 `MATH_PI=3.14;`