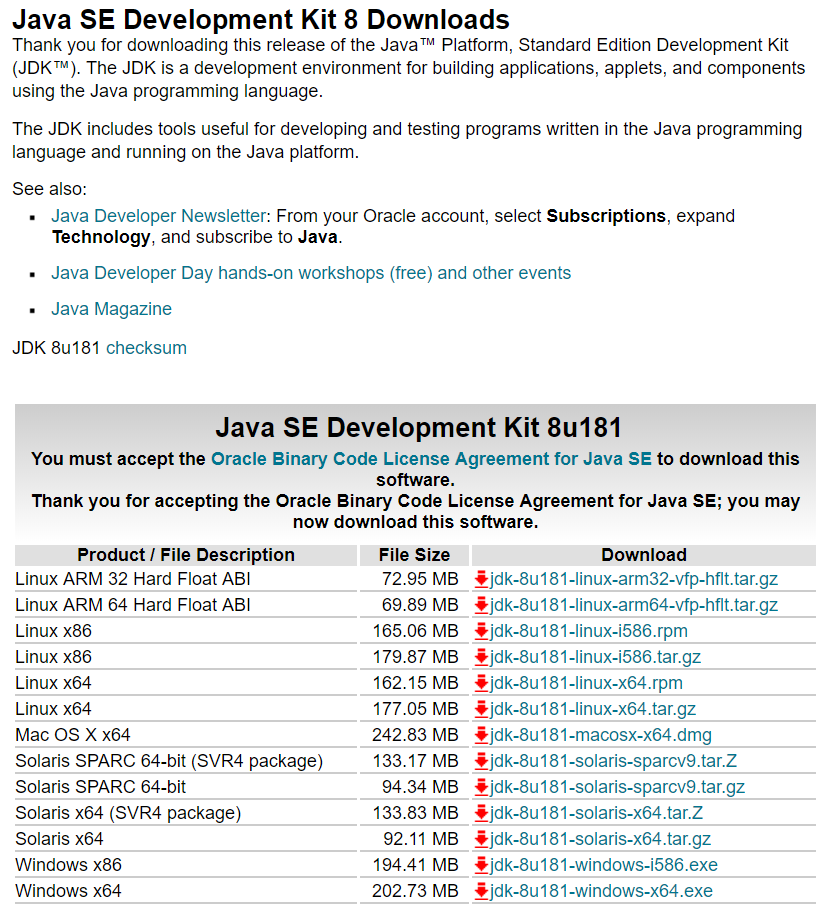
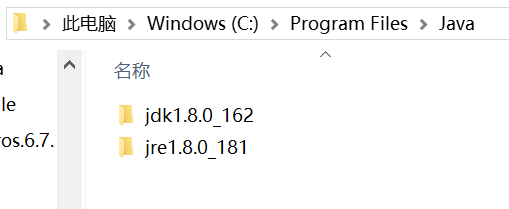
1.JDK软件包下载与安装

下载网址：<https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

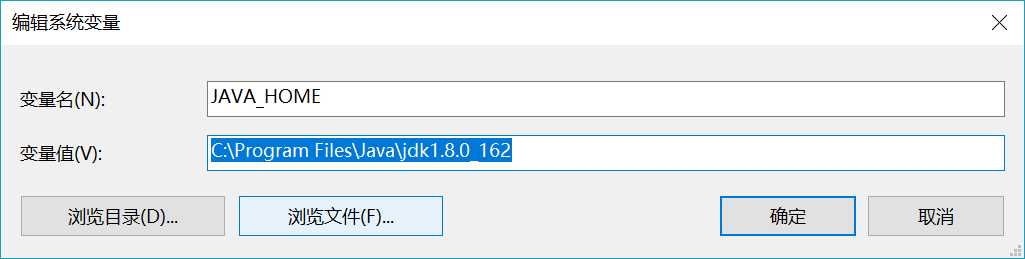
选择Windows x64上运行的Java8版JDK软件包并进行下载；

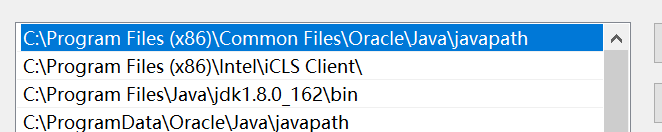


选择路径，在引导下进行安装JDK与JRE；

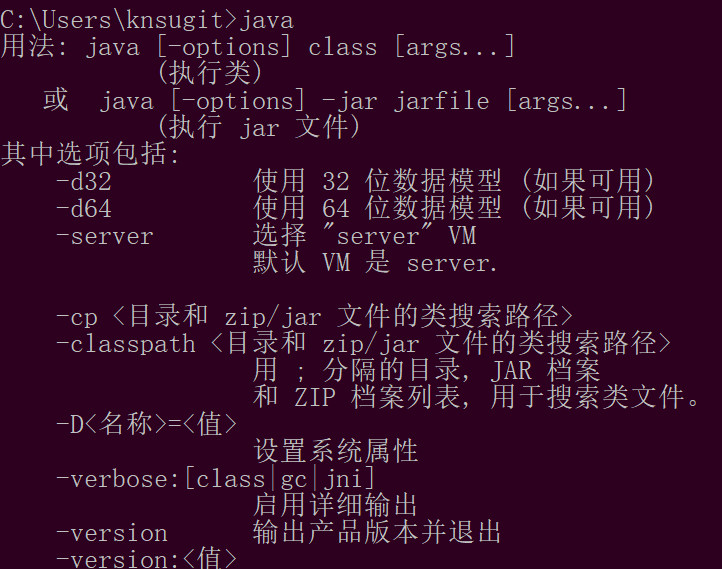


接着进行环境变量配置





最后通过cmd运行javac和java验证配置是否正常。



尝试阅读JDK文档：

在<https://docs.oracle.com/javase/8/index.html有Java>各个版本的完整的文档。通过阅读JDK文档，我们可了解Java各版本的特性，安装方法，快速入门等基本信息，也可以查阅Java API等实用信息。对我们学习和使用Java具有重要意义。



2.区分Java Application和Applet程序

区别①：运行环境不同。Application经过编译器编译，通过Java解释器和JVM（Java虚拟机）就可独立运行，不需要借助其他组件；Applet必须嵌入在HTML网页中，并借助支持Java Applet的浏览器或AppletViewer（Applet观察器）运行。

区别②：程序结构不同。Application中必定有且仅有一个main方法，作为程序运行的入口，含有main方法的类称为主类；而Applet中不存在含有main方法的主类，但一定含有一个从java.applet.Applet派生的类。

区别③：应用方向不同。Application主要用于桌面应用开发，而Applet一般用于B/S页面上作为插件式的开发。

3.Java语言主要特点

①简洁性：Java在C++的基础上进行了很大的简化和改进。例如虽然Java只支持单重继承，但支持接口，一个类可以实现多个接口，利用接口可以得到多继承的优点，又没有多继承混乱、复杂的问题；相对于C++，Java采用自动内存分配和回收大大简化了内存管理工作。另外，Java采用面向对象编程使得程序结构简洁，易于阅读。

②面向对象：Java语言具有真正的面向对象语言的特点，它完全基于类、对象，以类的形式组织代码，数据类型。它支持静态和动态的代码继承及重用，也具有面向对象所共有的特性：封装、继承、多态。

③动态性：Java语言的设计能够适应于发展变化的环境。

④安全性：Java虚拟机对字节码进行加密传输，客户端校验以及对解释器在客户端临时分配布置内存管理来保障其安全性。

⑤平台无关性和可移植性：用Java编写的程序，可以不经任何改动就在不同的硬件或软件平台上执行。

⑥高性能性：Java字节码格式设计中充分考虑到它的机械码执行效率，很容易直接转换成对应于特定处理器的高性能机器码。因此Java字节码的执行效率非常接近与C或C++生成的机械码执行效率。

⑦多线程：Java多线程支持表现在两个方面，一个是它自身的多线程，可以利用系统的空闲执行一些常规处理等；另一方面，提供对多线程的语言级支持，提高程序执行效率。

⑧分布式：Java是一个适用于网络的语言，它的设计使分布式计算变得容易起来。

⑨健壮性：Java不支持指针，并提供异常控制机制，能够更好地适应各种情况的出现。

4.Java三个版本：

①Java SE（Java标准版）

Java SE基于JDK和JRE，用于开发和部署桌面、服务器以及嵌入设备和实时环境中的Java应用程序。Java SE包括用于开发Java Web服务的类库，同时，Java SE为Java EE和Java ME提供了基础。

②Java EE（Java企业版）

Java EE是一种利用Java2平台来简化企业解决方案的开发、部署和管理相关的复杂问题的体系结构。

③Java ME（Java微型版）

Java ME是一个技术和规范的集合，它为移动设备（包括消费类产品、嵌入式设备、高级移动设备等）提供了基于Java环境的开发与应用平台。

5.

|  |
| --- |
| JavaWorld.java |
| public class JavaWorld{  public static void main(String args[]){  System.out.print("Welcome to Java World!");  }  } |
|  |

6.

|  |
| --- |
| HelloJavaApp.java |
| import java.applet.Applet;  import java.awt.Graphics;  public class HelloJavaApp extends Applet{  public void paint(Graphics g){  g.drawString("Welcome to Java applet World!", 100, 100);  }  } |
| HelloJava.html |
| <html>  <body>  <applet code="HelloJavaApp.class"WIDTH=1000 HEIGHT=500>  </applet>  </body>  </html> |
|  |