

# 第01节：计算机基本概念和JDK的安装

## 课程概要

计算机的基本概念

计算机功能键和常用快捷键

JDK相关的基本概念

JDK的下载、安装和配置

## 学习目标

了解计算机相关的基本概念

认识键盘的常用功能键

**能够熟练使用常用的快捷键**

**英文打字速度不低于150字/min**

理解JDK、JRE、JVM的概念和关系

**能够在官网下载JDK**

**能够正确安装JDK**

**能够理解PATH环境变量的作用并正确配置**

## 计算机的基本概念（了解）

计算机的概念：Computer，全称电子计算机，俗称电脑。是一种能够按照程序运行，自动、高速处理海量数据的现代化智能电子设备。由硬件和软件所组成。

计算机的分类：常见的形式有台式计算机、笔记本计算机、大型计算机等

硬件：是指计算机中由电子、机械和光电元件等组成的各种物理装置的总称。这些物理装置按系统结构的要求构成一个整体为计算机软件运行提供物质基础。

计算机硬件的五大组成部分：运算器、控制器、存储器、输入设备和输出设备。运算器和控制器是计算机的核心，合称中央处理单元（Central Processing Unit, CPU）或处理器，也就是我们常说的CPU；存储器用来存储数据，比如内存、硬盘；输入设备和输出设备，是用来向我们展示信息和对计算机进行操作的。常见的输入设备，比如鼠标、键盘；常见的输出设备，如显示器、打印机等。

软件：按照特定顺序组织的计算机数据和指令的集合。计算机软件是使用计算机过程中必不可少的东西，通俗来说，计算机软件就是使计算机按照事先预定好的顺序完成的特定的功能。计算机软件，按照其功能划分为系统软件与应用软件。常见的系统软件，有DOS, Windows, Linux, Unix, Mac, Android, iOS。常见的应用软件，如办公软件Office，聊天软件微信、QQ，游戏等。

计算机语言：人与计算机之间进行交流沟通的一种特殊语言。我们可以通过计算机语言与计算机传递信息。

软件开发：借助开发工具与计算机语言制作软件的过程

计算机语言的分类：机器语言，汇编语言和高级语言。机器语言，是直接用二进制代码指令表示的计算机语言，指令由0和1组成，它们有一定的位数，并分成若干段，各段的编码表示不同的含义。机器语言：阅读和编写都很麻烦，非常不利于开发工作。汇编语言：使用一些特殊的符号来代替机器语言的二进制码，计算机不能直接识别，需要用一种软件将汇编语言翻译成机器语言。相对于机器语言，阅读性大大提升，但是编码量依然非常大。高级语言：开发的程序使用普通英语进行编写，编写完成的内容我们称为源代码，通过编译器将源代码翻译成计算机能够直接识别的机器语言，之后再由计算机执行。常见的高级语言就很多了，比如：C，C++，Java等等

## 计算机功能键和常用快捷键

### 计算机功能键

Tab：制表符，用于缩进

Shift：上档键

Ctrl：控制键，和其它键配合使用

Alt：改变，单独使用可以调用窗口菜单，一般和其它键配合使用

space：空格键，输入一个空白字符，即空格

Enter：回车键，用于换行

Windows：windows图标，呼出系统菜单

### 计算机常用快捷键

Ctrl+A：全选

Ctrl+C：复制

Ctrl+V：粘贴

Ctrl+X：剪切

Ctrl+Z：撤销

Ctrl+S：保存

### 打字速度要求

最低速度：150字/min。英文。

## JDK相关的基本概念

### JDK

Java Development Kit，即Java开发工具包。用于开发Java程序

### JRE

Java Runtime Environment，即Java运行时环境。用于运行Java程序

### JVM

Java Virtual Machine, Java虚拟机, 将Java程序翻译成机器语言, 交给底层操作系统执行, 并保证相同的运行效果

## JDK的下载、安装和配置

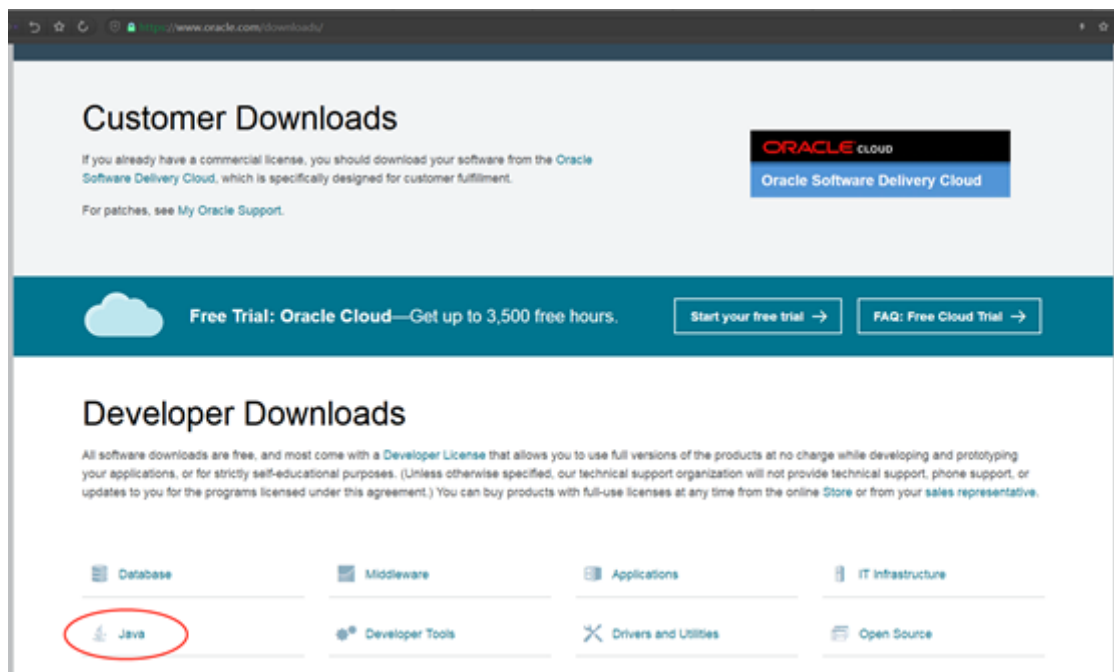
### JDK的下载

#### 第一步

输入下载地址: <https://www.oracle.com/downloads/>

#### 第二步

向下滚动鼠标, 点击进入Java产品下载列表页面



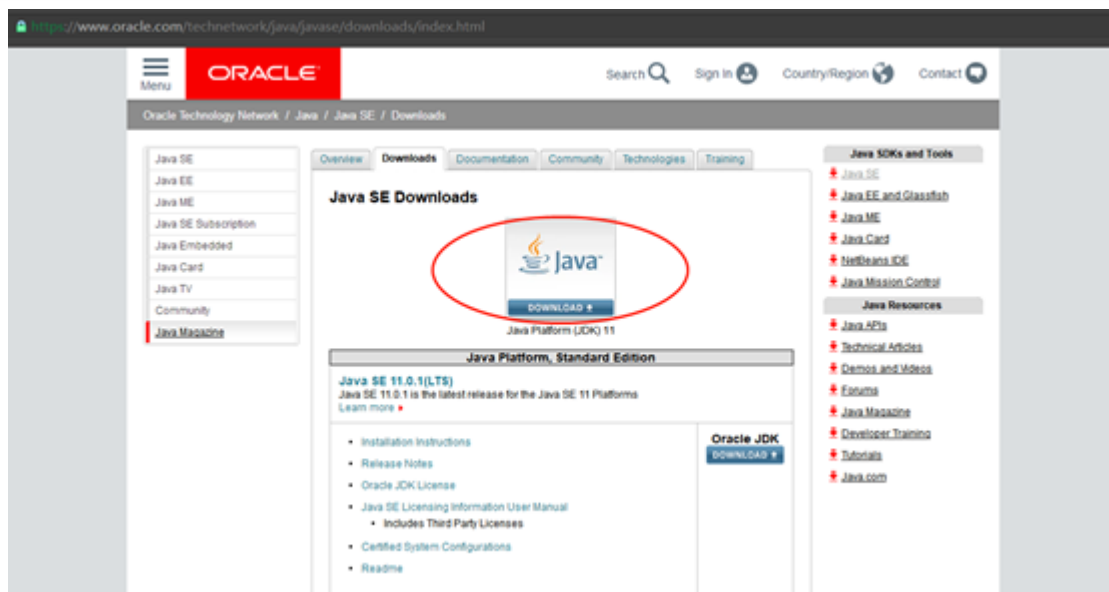
#### 第三步

点击第二个条目“Java (JDK) for Developers”并进入JDK下载首页



#### 第四步

点击Java图标进入最新版JDK下载页



## 第五步

根据自己操作系统下载对应版本程序包



## JDK的安装

### 第一种方式

傻瓜式安装。一直点下一步。

### 第二种：推荐的方式

第一步：双击运行安装包，等待安装程序就绪



## 第二步：修改JDK安装目录



### 注意事项

1. 路径中不要有中文或特殊符号
2. 建议所有开发相关软件路径统一
3. 举例：我的JDK安装路径：D:\DevelopTools\Java\jdk-11.0.1\

## 第三步：根据提示安装完成



## JDK的配置

### 环境变量

系统或应用程序运行过程中使用到的一些信息（参数）。可能是某个值，有可能是某个路径。

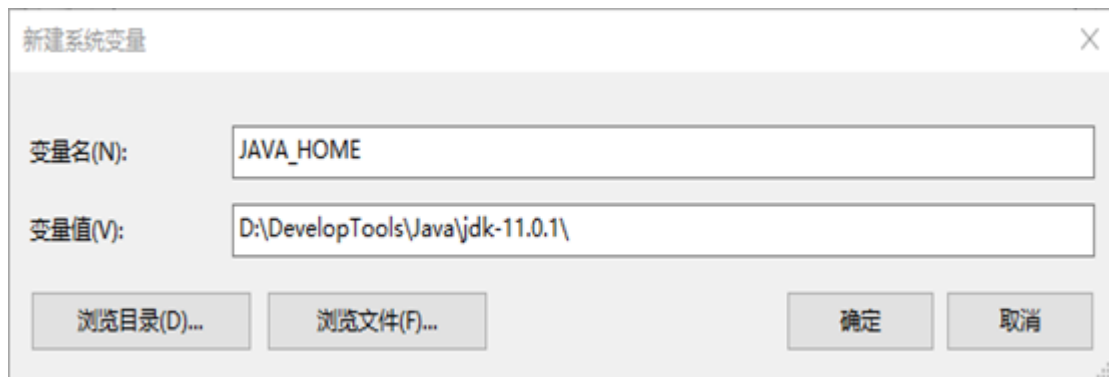
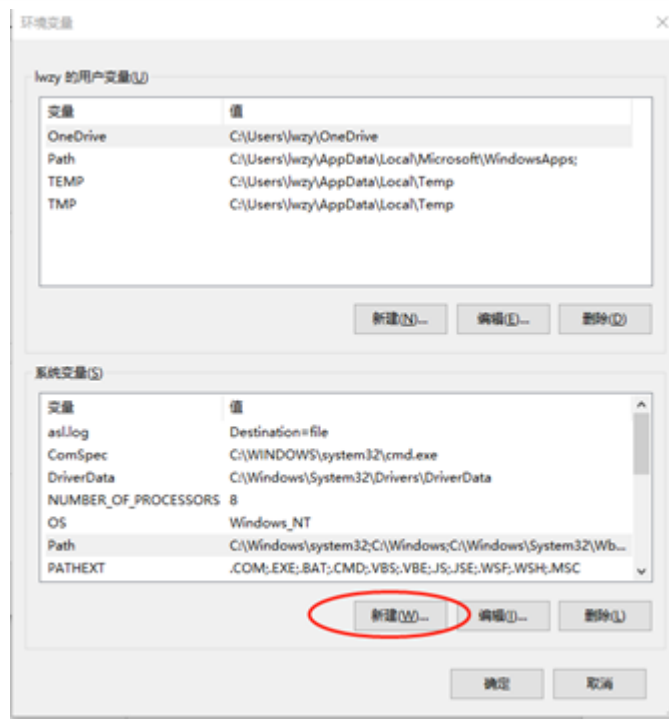
#### 第一步：打开控制面板，进入计算机属性界面



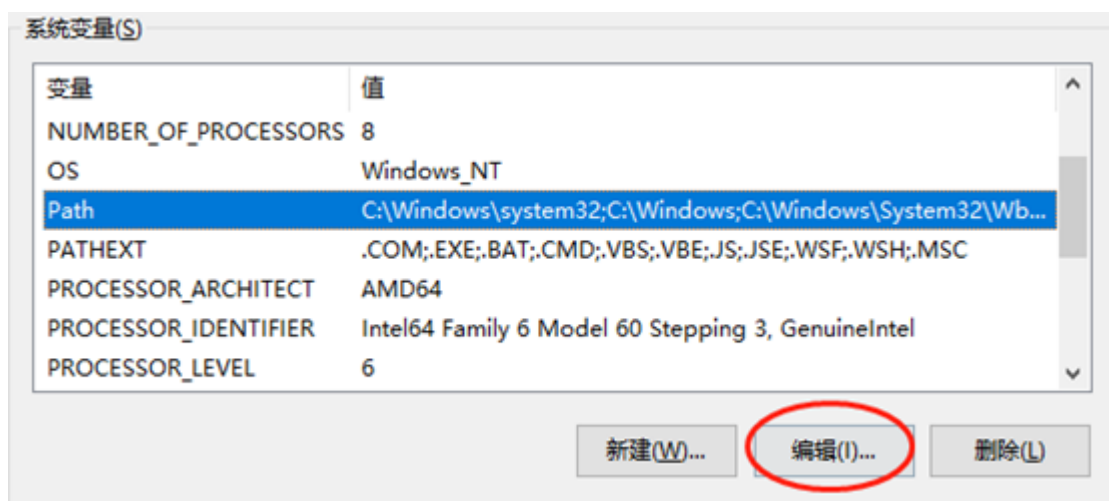
## 第二步：打开环境变量设置窗口



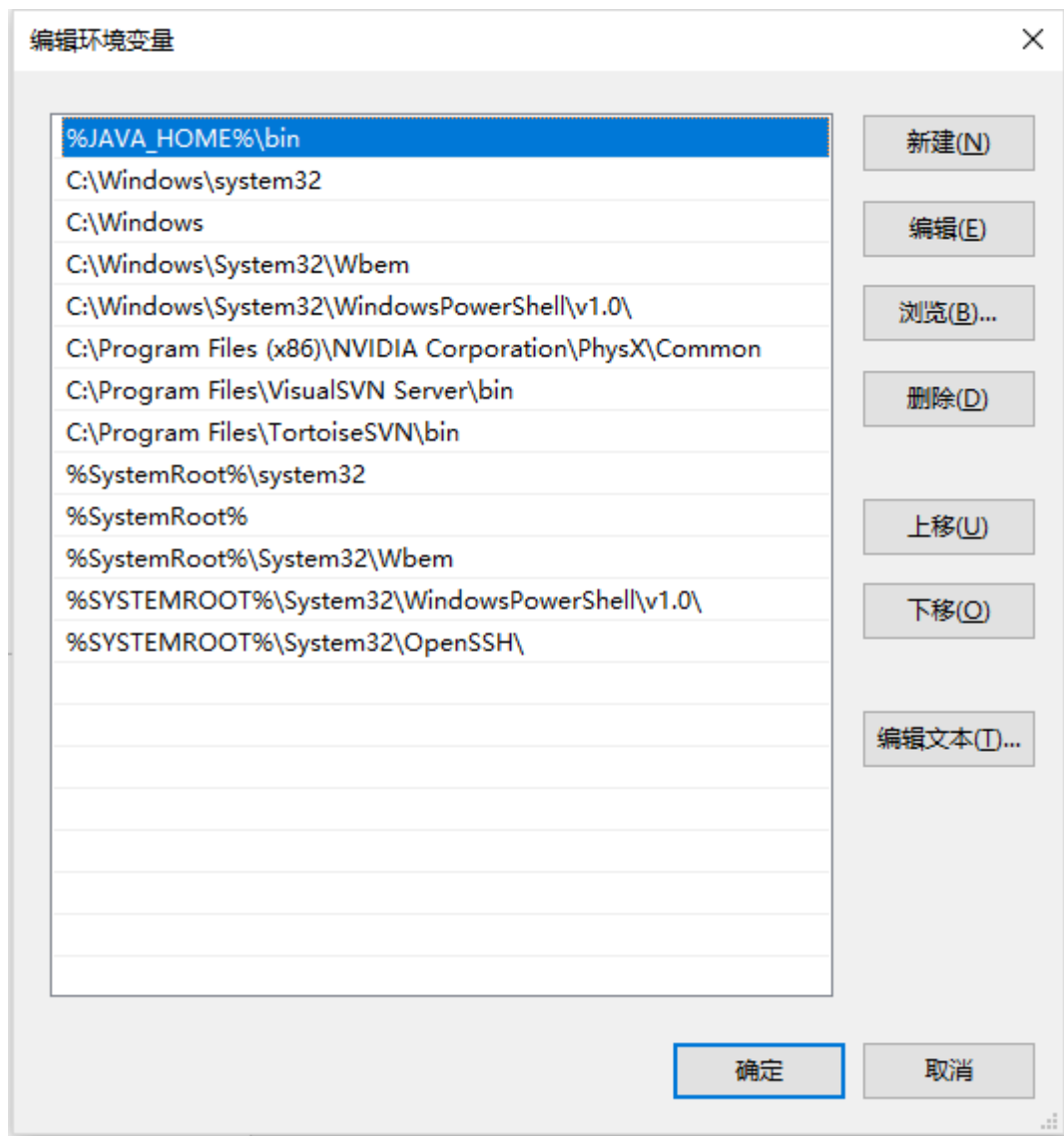
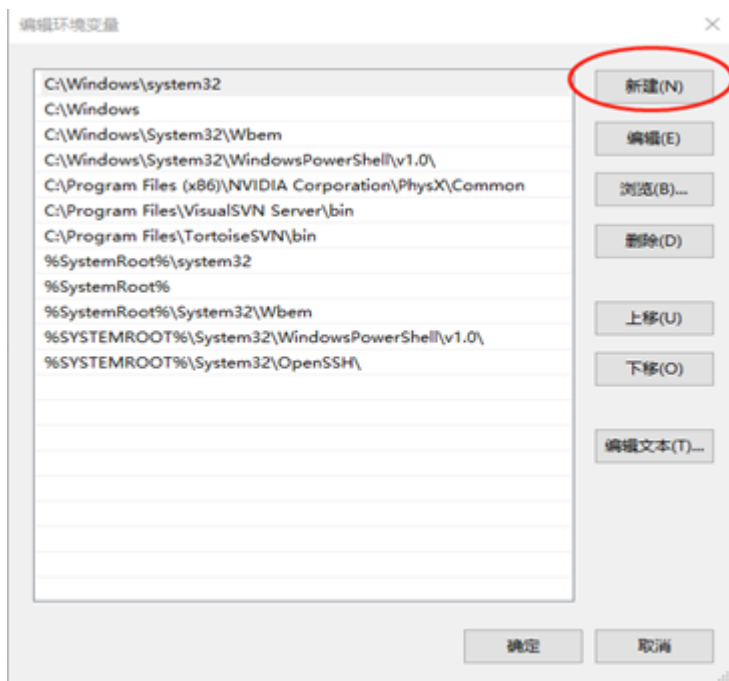
## 第三步：新建系统变量JAVA\_HOME，值为JDK安装根目录



第四步：编辑PATH变量，将新建的JAVA\_HOME变量加上bin目录设置到PATH中





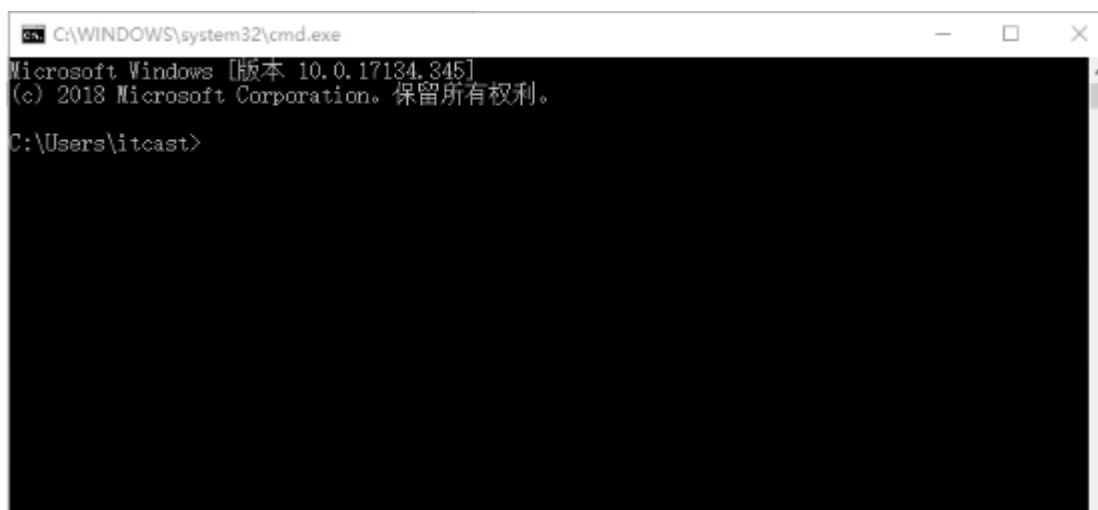
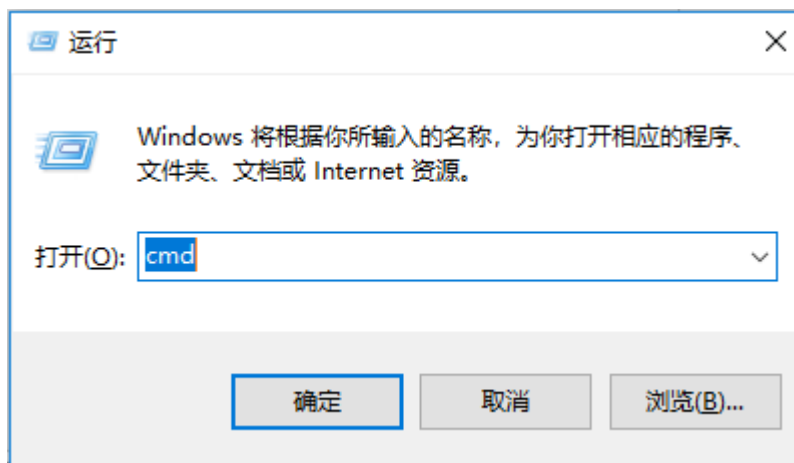


第五步：依次点击确定，关闭窗口

## JDK的配置验证

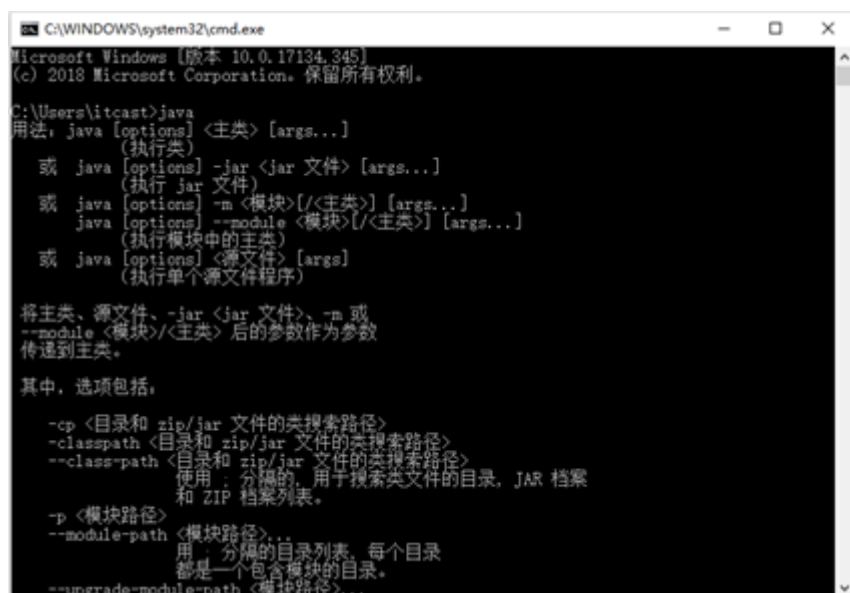
### 第一步

同时按住win + R，输入cmd，回车，进入命令提示符窗口



### 第二步

分别输入java和javac后回车，如果都能看到java相关的一些信息，就说明配置成功了



```
C:\WINDOWS\system32\cmd.exe

C:\Users\itcast>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]              传递给注释处理程序的选项
  --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注释处理程序的位置
  -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>       指定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>             指定放置生成的本机标头文件的位置
  --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>)*
                             限制可观察模块的领域
```