

Java 语言概述

Java 是一种面向对象的程序设计语言。本章首先介绍 Java 的发展历程,让读者对 Java 有基本的认识,了解 Java 的特点和运行机制;然后详细讲解 Java 开发环境的搭建和开发流程,包括一些基本的注意事项。

学习目标

- 认识 Java,并了解其发展历程。
- 熟悉 Java 语言的特点及应用领域。
- 掌握安装并配置 Java 开发环境的方法。
- 掌握 Java 中 Path 及 classpath 属性的作用。
- 能够编写并运行一个简单的 Java 程序。
- 掌握 Java 的开发流程。
- 了解常用的 Java 开发工具。

1.1 认识 Java

1.1.1 什么是 Java

在认识 Java 之前,先了解两个基本概念。

1. 软件开发

软件开发是根据用户要求建造软件系统或者系统中的软件部分的过程。软件开发是一项包括需求捕捉、需求分析、设计、实现和测试的系统工程。软件是一系列按照特定顺序组织的计算机数据和指令的集合,分为系统软件和应用软件。软件一般使用某种程序设计语言来实现,通常采用相应的开发工具进行开发。

开发出来的软件系统可以帮助人们解决和处理各种问题,必然产生人机交互。人机交互方式有两种:图形用户界面(graphical user interface,GUI)和命令行界面(command line interface,CLI)。第一种方式简单直观,用户易于接受,容易上手操作,如 Windows 操作系统;第二种方式需要一个控制台,用户输入特定的指令,让计算机完成操作,较为麻烦,需要用户记住一些命令,如 DOS 操作系统。

2. 计算机语言

语言是人与人之间用于沟通的一种方式。例如,中国人与中国人用普通话沟通;中国人要和英国人交流,需要学习英语。计算机语言是人与计算机交流的方式。如果人要与计算机交流,需要学习计算机语言。计算机语言有很多种,如 C、C++、Java、PHP 等。



Java 是 Sun 公司(全称 Stanford university network,于 1982 年成立。2009 年 4 月 20 日,被甲骨文公司以约 74 亿美元收购)开发的一套编程语言,主设计者是 James Gosling(见图 1-1),最早来源于一个叫 Green 的项目。这个项目最初的目的是为家用电子消费产品开发一个小巧、易用、安全稳定、与平台无关的分布式代码系统,以便通过网络对家用电器进行控制。一开始,Sun 公司的工程师们准备采用 C++ 语言来开发,但由于 C++ 过于复杂,安全性差,于是他们决定基于 C++ 开发一种符合自己要求的新语言。1991 年 4 月,历时 18 个月,新语言的第一个版本诞生了。命名时,James Gosling 看到窗口的橡树(见图 1-2),遂为其取名为 Oak,希望它能够有橡树一般坚强的生命力。后来,他发现 Oak 是 Sun 公司另外一种语言的注册商标,于是 1995 年将这种新语言更名为 Java,即太平洋上一个盛产咖啡的岛屿的名字。Java 是一种用于网络的,精巧而安全的语言,使程序能够最大限度地利用网络资源。



图 1-1 James Gosling



图 1-2 印度橡树

Java 语言发展到今天,有多个版本。1995 年 5 月 23 日,Sun 公司推出 JDK 1.0 版,标志着 Java 正式进军 Internet。1998 年,Sun 公司对 JDK 1.0 升级,并推出 JDK 1.2 的开发包,加入大量的轻量级组件包。从此,Java 正式命名为 Java 2。

Java 语言经历了以下 3 个发展方向。

(1) J2SE(Java 2 platform standard edition): 包含构成 Java 语言核心的类,如数据库连接、接口定义、输入/输出和网络编程,主要用于开发一般个人电脑上的应用软件。

(2) J2ME(Java 2 platform micro edition): 包含 J2SE 中的一部分类,用于消费类电子产品的软件开发,如呼机、智能卡、手机、PDA 和机顶盒。

(3) J2EE(Java 2 platform enterprise edition): 即 Java 企业版,包含 J2SE 中的所有类,还包含用于开发企业级应用的类,如 EJB、Servlet、JSP、XML 和事务控制,也是现在 Java 应用的主要方向,用于开发企业级应用软件。

上述 3 项中的核心部分是 J2SE,J2ME 和 J2EE 是在 J2SE 基础上发展起来的。

【注】在 2005 年“Java 十周年大会”之后,上述 3 门技术被重新命名。

(1) J2SE 更名为 Java SE。

(2) J2ME 更名为 Java ME。

(3) J2EE 更名为 Java EE。



1.1.2 Java 语言的特点

Java 总是和 C++ 联系在一起,而 C++ 是从 C 语言派生而来的,所以 Java 语言继承了这两种语言的大部分特性。Java 的语法从 C 语言继承而来,Java 许多面向对象的特性都受到 C++ 的影响。事实上,Java 中的几个自定义特性都来自或可以追溯到它的这些前驱语言。略有不同的是,Java 语言完全面向对象,摒弃了 C 和 C++ 的不足。Java 语言的诞生与过去近 30 年中计算机语言的不断改进和发展密切相关。

Sun 公司在《Java 白皮书》中对 Java 的定义是:“Java: A simple, object-oriented, distributed, interpreted, robust, architecture-neutral, secure, portable, high-performance, multi-threaded, and dynamic language.”即 Java 是一种具有简单、面向对象、分布式、解释型、健壮、安全、与体系结构无关、可移植、高性能、多线程和动态执行等特性的语言。下面简述 Java 的主要特性。

1. 简单易用

Java 语言是一种相当简洁的面向对象程序设计语言,它省略了 C++ 语言中所有难以理解、容易混淆的特性,如头文件、指针、结构、单元、运算符重载和虚拟基础类等,更加严谨、简洁。

Java 源代码的书写不拘泥于特定的环境,可以使用记事本、文本编辑器等;将源文件编译后,可直接运行;再通过调试,得到预期的结果。

此外,Java 可以自动完成垃圾收集工作,回收不再使用的内存,使用户无须担心内存管理之类的事情。

2. 面向对象

面向对象是指以对象为基本粒度,其下包含属性和方法。对象的说明用属性表达,通过使用方法来操作这个对象。可以这么说,面向对象是软件工程学的一次革命,大大提升了人类的软件开发能力,是一个伟大的进步,是软件发展重大的里程碑。作为一种现代编程语言,是不能偏离面向对象这一方向的,Java 语言也不例外。

Java 是一种面向对象的语言,具有面向对象的诸多优点,如代码扩展、代码复用等。

3. 分布式

Java 语言具有强大的、易于使用的联网能力,非常适合开发分布式计算的程序。Java 应用程序可以像访问本地文件系统那样通过 URL 访问远程对象。

使用 Java 语言编写 Socket 通信程序比使用其他任何语言都要简单。它适用于公共网关接口(CGI)脚本的开发,还可以利用 Java 小应用程序(Applet)、Java 服务器页面(Java server page, JSP)、Servlet 等手段构建更丰富的网页。

4. 解释型

Java 是一种解释型语言,相对于 C/C++,用 Java 语言写出来的程序效率低,执行速度慢。但它可以通过在不同平台上运行 Java 解释器,解释 Java 代码,实现“一次编写,到处运行”的目标。为此,牺牲效率是值得的。而且,现在的计算机技术日新月异,运算速度也越来越快,用户不会感到太慢。

5. 健壮

Java 语言在伪编译时做了许多早期潜在问题的检查,在运行时又做了一些相应的检



查,可以说是一种非常严格的编译器。它的这种“防患于未然”的手段将许多程序中的错误扼杀在“摇篮”之中,使许多在其他语言中必须通过运行才会暴露出来的错误,在编译阶段就被发现了。

另外,Java 语言具备保证程序稳定、健壮的特性,有效地减少了错误,使 Java 应用程序更加健壮。

6. 具有较高的安全性

人们设计 Java 语言时,在安全性方面考虑得很仔细,做了许多探究,使其成为目前非常安全的一种程序设计语言。

对 Java 来说,安全性分为 4 个层面,即语言级安全性、编译时安全性、运行时安全性和可执行代码安全性。语言级安全性指 Java 的数据结构是完整的对象,这些封装过的数据类型具有安全性。编译时,要检查 Java 语言和语义,保证每个变量对应一个值,编译后生成 Java 类。运行时,Java 类需要使用类加载器载入,由字节码校验器校验之后才可以运行。Java 类在网络上使用时,对其权限进行了设置,以保证被访问用户的安全性。

7. 可移植性

对于程序员而言,写出来的程序如果不需要修改就能够同时在 Windows、Mac OS、UNIX 等平台上运行,简直就是美梦成真,Java 语言让这个原本遥不可及的事越来越近。使用 Java 语言编写的程序,只需较少的修改,甚至有时根本不需修改,即可在不同的平台上运行。

8. 拥有较高的性能

由于 Java 是一种解释型语言,其执行效率就会低一些,但采取下述两种措施,可使其拥有较高的性能。

(1) Java 语言源程序编写完成后,先使用 Java 伪编译器进行伪编译,将其转换为中间码(也称为字节码)再解释。

(2) 提供了一种即时(just-in-time, JIT)编译器。当需要更快的速度时,使用 JIT 编译器将字节码转换成机器码,将其缓冲下来,速度就会更快。

9. 具有多线程处理能力

线程是一种轻量级进程,是现代程序设计中必不可少的一种特性。多线程是指允许一个应用程序同时存在两个或以上的线程,用于支持事务并发和多任务处理。多线程处理能力使程序具有更好的交互性和实时性。

Java 在多线程处理方面性能超群,除了内置的多线程技术之外,还定义了一些类、方法等来建立和管理用户定义的多线程,具有让用户惊喜的强大功能,而且在 Java 语言中进行多线程处理也很简单。

10. 是一种动态语言

Java 是一种动态的语言,表现在以下两个方面。

(1) 在 Java 语言中,可以简单、直观地查询运行时的信息。

(2) 可以将新代码加入到正在运行的程序中。

1.1.3 Java 程序的运行机制

在 Java 中处理代码的过程如图 1-3 所示。

从图 1-3 中可以看出,Java 源文件的扩展名为.java,通过编译,生成 *.class 文件,在计

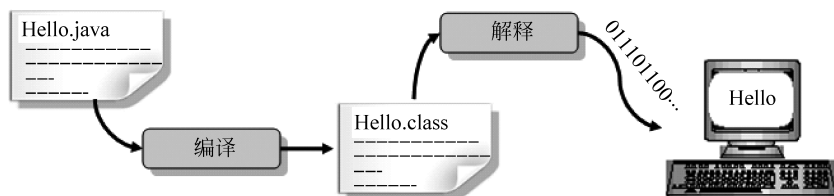


图 1-3 Java 程序的运行机制

计算机上执行。此时执行 *.class 的计算机并不是物理上可见的,而是 Java 自己设计的一台计算机——Java 虚拟机(Java virtual machine,JVM)。Java 通过 JVM 进行可移植性操作。

在 Java 中,所有的程序都在 JVM 上运行。JVM 是在一台计算机上由软件或硬件模拟的计算机,它读取并处理编译过的、与平台无关的字节码 *.class 文件。Java 解释器负责将 Java 虚拟机的代码在特定的平台上运行。JVM 的基本原理如图 1-4 所示。

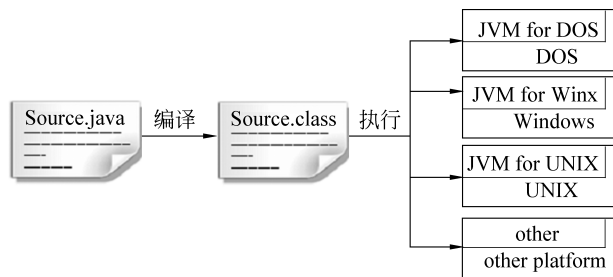


图 1-4 JVM 的基本原理

从图 1-4 中可以发现,所有的 *.class 文件都在 JVM 上运行,即 *.class 文件只需要认识 JVM,由 JVM 去适应各个操作系统。如果不同的操作系统安装了符合其类型的 JVM,那么程序无论在哪个操作系统上都可以正确地执行。

有些读者可能很难理解以上的解释。其实这个过程类似于下述情景:有一位中国富商,同时要和美国、韩国、俄罗斯、日本、法国、德国等几个国家的客户洽谈生意,可是他不懂这些国家的语言,所以他针对每个国家都请了一位翻译。他只对翻译说话,不同的翻译就会将他说的话译给相应的客户。这样,富商只需要对各位翻译说话,就可以同几个国家的客户进行沟通。

【注】Java 的两种核心机制如下所述。

(1) Java 虚拟机(Java virtual machine),是一台利用软件方法实现的虚拟的计算机,具有指令集,并使用不同的存储区域,负责执行指令,管理数据、内存、寄存器。对于不同的平台,有不同的虚拟机。Java 虚拟机屏蔽了底层运行平台的差别,实现了“一次编译,到处运行”。

(2) 垃圾收集机制(garbage collection),即回收不再使用的内存空间。在 C/C++ 等语言中,由程序员负责回收无用内存。Java 语言撤销了程序员回收无用内存空间的责任,它提供一种系统级线程来跟踪存储空间的分配情况,并在 JVM 空闲时,检查并释放那些可被释放的存储空间。垃圾回收在 Java 程序运行过程中自动执行,程序员无法精确控制和干预。

1.1.4 Java 语言的应用领域

Java 技术自 1995 年问世以来,在我国的应用和开发迅速普及。总体来看,主要集中于



企业应用开发。根据有关单位调查显示,从开发领域的分布来看,Web 开发占了一半以上,为 57.9%;Java ME 移动或嵌入式应用占 15%;C/S 应用占 11.7%;系统编程占 15.4%。近 30% 的开发者用 Java 从事 C/S 应用或系统级应用的开发。

Java 语言主要应用在下述几个领域。

1. 行业和企业信息化

由于 Sun、IBM、Oracle、BEA 等国际厂商相继推出基于 Java 技术的应用服务器以及各种应用软件,带动了 Java 在金融、电信、制造等领域日益广泛的应用。例如,清华大学计算机系利用 Java、XML 和 Web 技术研制开发了多个软件平台,东方科技的 Tong Web、金蝶的 Apusic、中创的 Inforweb 等 J2EE 应用服务器,以及和佳 ERP 和宝信 ERP 等 ERP 产品,在许多企业得到应用。

2. 电子政务及办公自动化

东方科技、金蝶、中创等公司开发的 J2EE 应用服务器在电子政务及办公自动化中广泛应用。例如,金蝶的 Apusic 在民政部、广东省市工商局应用;东软电子政务架构 EAP 平台在社会保险、公检法、税务系统应用;中创的 Inforweb 等 Infor 系列中间件产品在国家海事局、山东省政府及中国建设银行、民生银行等金融系统应用;无锡永中科技基于 Java 平台开发的国产化集成办公软件永中 Office 在一些省、市政府部门应用。

3. 嵌入式设备及消费类电子产品

无线手持设备、通信终端、医疗设备、信息家电(如数字电视、机顶盒、电冰箱)、汽车电子设备等是近几年来比较热门的 Java 应用领域。在这方面的应用有中国联通 CDMA 1X 网络中基于 Java 技术的无线数据增值服务——UniJa。

4. 辅助教学

在辅助教学方面,东南大学与中兴通信公司利用 Java 语言联合开发了远程教学系统,用于本地网上教学、课后学习和异地远程教育;清华大学利用 Java 语言进行了计算机软件基础课教学改革,分析、研究 Java 教学软件 BlueJ 的汉化方案;电子科技大学应用 Java RMI 技术进行远程教育;西安电力高等专科学校采用 Java 技术开发了交互式电站仿真系统,实现电站锅炉仿真、锅炉膛火焰仿真,为实现网上仿真进行了有益的探索。

1.2 Java 开发环境与开发工具

1.2.1 Java 开发环境

一台计算机上安装了 JVM,即可运行 Java 程序。但是要开发 Java 程序,还需建立 Java 开发环境。不同领域的 Java 开发应用所需的版本不同,本书使用 Java SE 开发环境。

1. Java SE 的组成

Java SE 是一个包含 Java 开发环境和运行环境的套件,由以下 3 项组成。

(1) Java development kit(JDK): Java 应用程序开发环境。

Java 不仅提供了丰富的语言和运行环境,还提供了一个免费的 Java 开发工具集,以便程序员开发 Java 开发工具包。Java 2 SDK 开发工具集如表 1-1 所示。



表 1-1 Java 2 SDK 开发工具集

工具名称	说 明
javac	Java 编译器,用于将 Java 源程序编译成字节码
java	Java 解释器,用于解释、执行 Java 字节码
appletviewer	小应用程序查看器,用于测试和运行 Java Applet 程序
javadoc	Java 文档生成器
jdb	Java 调试器
javap	Java 类文件反汇编器
javah	C 文件生成器,实现在 Java 类中调用 C++ 代码

JDK 中除了包括 Java 开发工具以外,还包括 JRE,所以安装了 JDK,就不用单独安装 JRE 了。

(2) Java runtime environment(JRE): Java 应用程序运行环境,包括 Java 虚拟机和 Java 程序所需要的核心类库等。如果仅需运行开发好的 Java 程序,计算机只需要安装 JRE。

(3) Java plug-in: Java 插件。

JVM、JRE 与 JDK 的关系如图 1-5 所示。

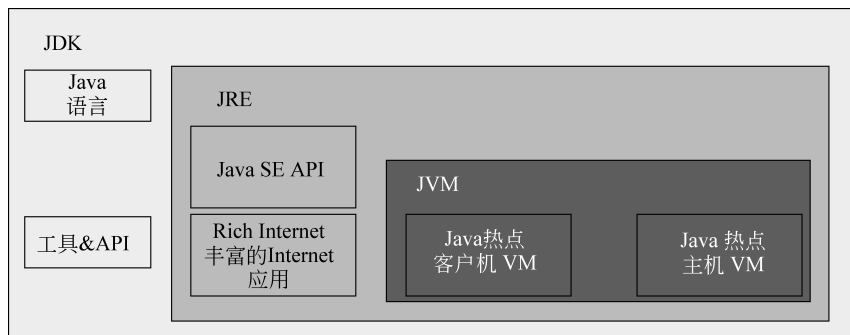


图 1-5 JVM、JRE 与 JDK 的关系

2. 建立 Java SE 开发环境

步骤 1: 安装 JDK(本书使用 JDK 1.6 版本)。

(1) 准备好 JDK 的安装文件 jdk-6u18-windows-i586se.exe。从 Oracle 公司的网站(<http://www.oracle.com/technetwork/java/javase/downloads/index-jsp-138363.html>)下载 JDK 工具包。

(2) 运行.exe 文件,安装 JDK。默认安装在 C:\Program Files\Java 文件夹下,本书更改安装在 D:\Java 文件夹下。

(3) 按照安装向导的提示,完成安装,即可看到如图 1-6 所示的文件夹。

- ① bin: 一些执行文件,包括 Java 的编译器、解释器和工具。
- ② demo: 各种演示的实例。
- ③ lib: 保存的库文件。
- ④ include: Win32 子目录,都是本地文件。

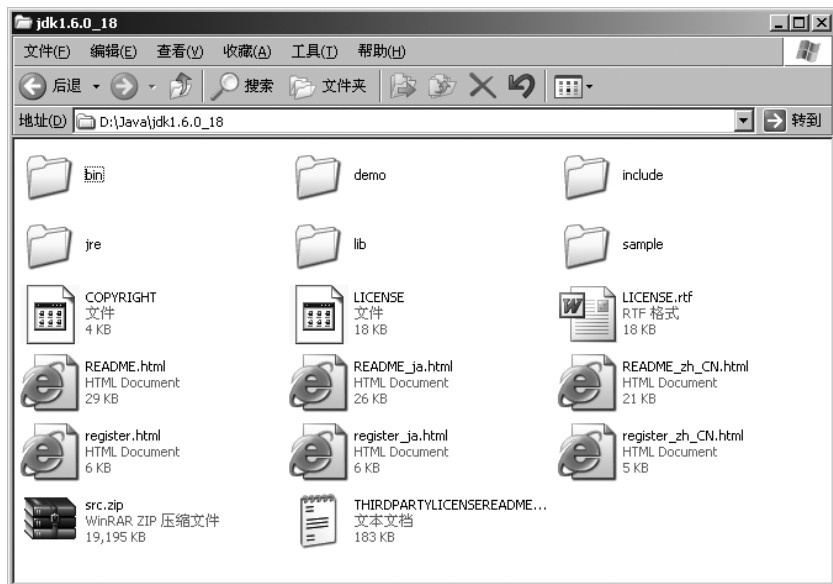


图 1-6 安装 JDK 后的文件夹

⑤ jre: Java 程序运行环境的根目录。

其中, bin 文件夹包含将来要使用的各种 Java 命令,但是这些命令本身并不在 Windows 环境之中。要想使用这些命令,必须先在 Windows 中注册。

步骤 2: 设置环境变量。

在 JDK 安装完毕后,需设置 path 和 classpath 这两个环境变量。这是程序编译和运行的重要保证。

path 指示 Java 命令的路径,如 javac、java、javaw 等。这样,在控制台下面编译、执行程序时就不需要再输入路径了。

由于 JDK 的安装路径多次使用,在此先新建环境变量 JAVA_HOME,操作步骤如下:

(1) 选择“我的电脑”→“属性”命令,在打开的窗口中选择“高级系统设置”命令,打开“系统属性”对话框,如图 1-7 所示。

(2) 选择“高级”选项卡,然后单击“环境变量”按钮,打开“环境变量”对话框,如图 1-8 所示。

(3) 单击“系统变量”栏中的“新建”按钮,弹出“编辑系统变量”对话框。在“变量名”文本框中输入 JAVA_HOME,在“变量值”文本框中输入 C:\Java\jdk1.7.0_03,如图 1-9 所示。

path 环境变量包含在 Windows 系统里,修改一下,使其指向 JDK 的 bin 文件夹,即在“环境变量”对话框中单击“系统变量”栏中的“编辑”按钮,弹出“编辑系统变量”对话框,然后在“变量值”文本框的最前面加上路径,再用“;”将后面的路径分隔开,如“%JAVA_HOME%\bin;”,如图 1-10 所示。

⚠注意:

环境变量的各变量值之间需用分号分隔。



图 1-7 “系统属性”对话框

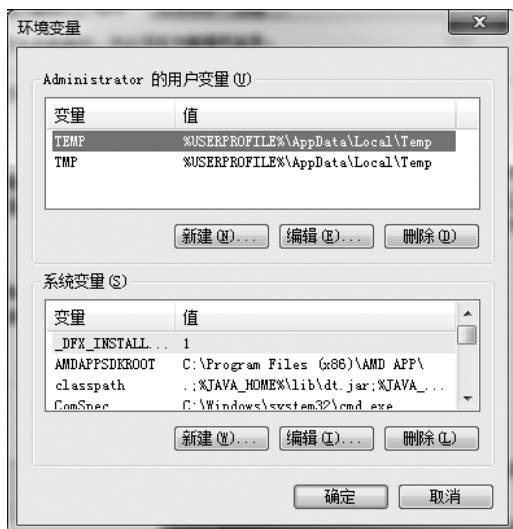


图 1-8 “环境变量”对话框



图 1-9 设置 JAVA_HOME 路径

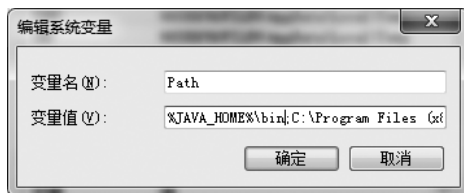


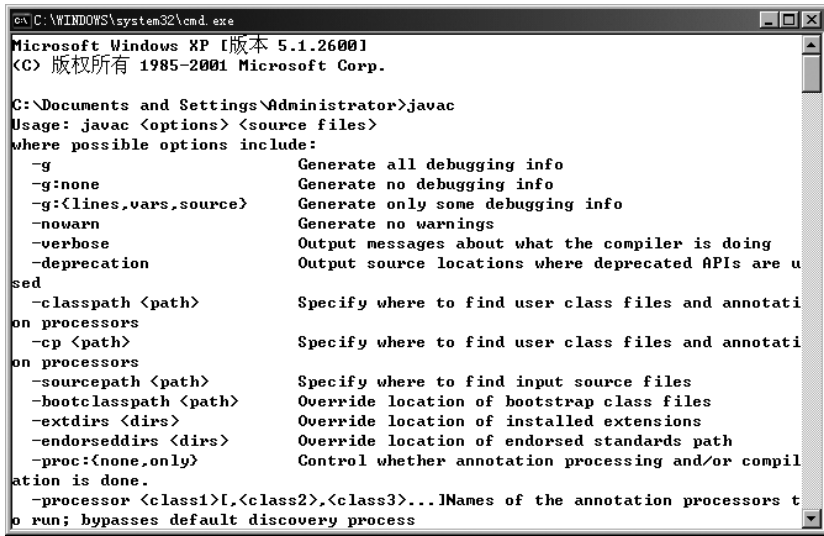
图 1-10 设置 Path 路径

classpath 是类库的默认搜索路径,即告诉 JVM 要使用或者执行的 *.class 文件所在的文件夹。这是专门针对 Java 的,故系统里没有这个环境变量,需要进行手动加入。classpath 的设置流程为:在“环境变量”对话框中单击“系统变量”栏中的“新建”按钮,弹出“编辑系统变量”对话框;然后在“变量名”文本框中输入 classpath,在“变量值”文本框中输入.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar,如图 1-11 所示。注意,最前面是“.”,这告诉 JDK,搜索类时,先查找当前文件夹的.class 文件。



图 1-11 设置 classpath 路径

最后,单击“确定”按钮,保存设置。启动 cmd 命令行方式,输入 javac,如果出现如图 1-12 所示的内容,则表示 JDK 配置成功。



```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [版本 5.1.2600]
(C) 版权所有 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>javac
Usage: javac <options> <source files>
where possible options include:
  -g               Generate all debugging info
  -g:none          Generate no debugging info
  -g:{lines,vars,source}  Generate only some debugging info
  -nowarn          Generate no warnings
  -verbose         Output messages about what the compiler is doing
  -deprecation     Output source locations where deprecated APIs are used
  -classpath <path> Specify where to find user class files and annotation processors
  -cp <path>       Specify where to find user class files and annotation processors
  -sourcepath <path> Specify where to find input source files
  -bootclasspath <path> Override location of bootstrap class files
  -extdirs <dirs>   Override location of installed extensions
  -endorseddirs <dirs> Override location of endorsed standards path
  -proc:{none,only} Control whether annotation processing and/or compilation is done.
  -processor <class1>[,<class2>,<class3>...] Names of the annotation processors to run; bypasses default discovery process

```

图 1-12 JDK 配置成功演示

⚠注意 1: 修改 Path 与 classpath 之后,要重新启动命令行方式。

在进行环境设置时,可能出现“此命令不是系统内部命令”的提示,原因是在配置环境属性之前,命令行方式已经启动,之后再配置 Path 路径和 classpath 路径,于是该环境肯定无法立即生效。此时重新启动命令行方式,可以把新的设置读取进来。

⚠注意 2: javac 与 java 命令的作用。

javac.exe 是 Java 本身提供的编译命令,主要目的是将 *.java 文件编译成 *.class 文件。java.exe 是 Java 提供的解释命令,主要用于解释、执行字节码文件。

1.2.2 Java 开发工具

学过程序设计的人都知道,使用 Basic 语言进行程序设计,可以使用 QBasic、Visual Basic 等开发工具;使用 C 语言进行程序设计,可以使用 Turbo C、Visual C++、C++ Builder 等开发工具。这些开发工具集成了编辑器和编译器,是集成开发工具,使用较方便。学习 Java 程序设计,同样需要方便、易用的开发工具。

Java 的开发工具很多,而且各有优缺点,初学者往往不知道有哪些常用的开发工具,或者由于面临的选择比较多而产生困惑。目前,比较流行的 Java 开发工具有 EditPlus、JCreator、Eclipse、MyEclipse、JBuilder、NetBeans 等。下面介绍几款常用的开发工具,以便初学者掌握并做出选择。

1. 文本编辑器 EditPlus

EditPlus 是功能全面的文本、HTML、程序源代码编辑器,默认支持 HTML、CSS、PHP、ASP、Perl、C/C++、Java、JavaScript 和 VBScript 的语法着色。通过定制语法文件,还可以扩展到其他程序语言。EditPlus 编辑界面如图 1-13 所示。

EditPlus 是共享软件,它的官方网址是 www.editplus.com,最新版本是 EditPlus 3.31。