

单元 1 搭建程序开发环境与运行 Java 程序

Java 是一种可以撰写跨平台应用程序的面向对象的程序设计语言,Java 技术具有卓越的通用性、高效性、平台移植性和安全性,是目前使用最为广泛的编程语言。NetBeans 是 Sun 公司使用 Java 语言开发的一个开源工具,是目前使用非常广泛的 Java 开发工具。

教学导航

教学目标	(1) 初步了解几个基本概念: Java、JDK、JRE、JVM、Java API、NetBeans (2) 初步了解 Java 程序的运行机制、Java 程序的编译与运行方法 (3) 学会下载与安装 JDK、NetBeans IDE (4) 学会 Windows 操作系统中 Java 运行环境的配置 (5) 熟悉 NetBeans IDE 集成开发环境的组成及其功能
教学重点	(1) Java 运行环境的配置 (2) NetBeans IDE 集成开发环境 (3) Java 程序的运行机制

前导知识

1. 初识 Java

Java 是由 Sun Microsystems 公司推出的 Java 面向对象程序设计语言(以下简称 Java 语言)和 Java 平台的总称,由 James Gosling 与他的同事们共同研发,并在 1995 年正式推出。Java 最初被称为 Oak,是 1991 年为消费类电子产品的嵌入式芯片而设计的。1995 年更名为 Java,并重新设计用于开发 Internet 应用程序。用 Java 实现的 HotJava 浏览器(支持 Java applet)显示了 Java 的魅力:跨平台、动态 Web、Internet 计算。从此,Java 被广泛接受并推动了 Web 的迅速发展,常用的浏览器均支持 Java applet。Java 自面世后就非常流行,发展迅速,Java 技术也不断更新。在全球云计算和移动互联网的产业环境下,Java 更具备了显著优势和广阔前景。2010 年 Oracle 公司收购 Sun Microsystems,于是 Java 便成为 Oracle 旗下的产品。

2. 初识 JDK

JDK(Java Development Kit)是 Java 语言的软件开发工具包(SDK),如果没有 JDK,则无法编译 Java 程序,如果只想运行 Java 程序,要确保已安装相应的 JRE。JDK 主要分为标准版、企业版和微型版。

(1) SE(J2SE)(standard edition 标准版):这是我们通常用的一个版本,从 JDK 5.0 开始,改名为 Java SE。

(2) EE(J2EE)(enterprise edition,企业版):使用这种 JDK 可以开发 J2EE 应用程序。从 JDK 5.0 开始,改名为 Java EE。

(3) ME(J2ME)(micro edition,微型版):主要用于移动设备、嵌入式设备上的 Java 应用程序。从 JDK 5.0 开始,改名为 Java ME。

3. 初识 JRE

JRE(Java Runtime Environment,Java 运行环境),是运行 Java 程序所必须的环境集合,包含 JVM 标准实现及 Java 核心类库。JRE 是可以在其上运行、测试和传输应用程序的 Java 平台,它包括 Java 虚拟机(JVM)、Java 核心类库和支持文件。但它不包含开发工具(JDK)的编译器、调试器和其他工具。JRE 需要辅助工具 Java Plug-in,以便在浏览器中运行 applet。

4. 初识 JVM

JVM 是 Java Virtual Machine(Java 虚拟机)的缩写,JVM 是一个虚构出来的计算机,通过仿真模拟各种计算机功能来实现其功能。Java 语言的一个非常重要的特点就是与平台的无关性,而使用 Java 虚拟机是实现这一特点的关键。一般的高级语言如果要在不同的平台上运行,至少需要编译成不同的目标代码。而引入 Java 语言虚拟机后,Java 语言在不同平台上运行时不需要重新编译。Java 语言使用 Java 虚拟机屏蔽了与具体平台相关的信息,使 Java 语言编译程序只需生成在 Java 虚拟机上运行的目标代码(字节码),就可以在多种平台上不加修改地运行。Java 虚拟机在执行字节码时,把字节码解释成具体平台上的机器指令执行。这就是 Java 能够“一次编译,到处运行”的原因。

5. 初识 Java API

Java API(Java Application Interface)是编程人员使用 Java 语言进行程序开发时相关类的集合,是 Java 平台的一个重要组成部分。Java API 中的类按照用途被分为多个包(package),每个包又是一些相关类或接口的集合,其中“java. * ”包是 Java API 的核心。

6. 初识 NetBeans

NetBeans 是 Sun 公司使用 Java 语言开发的一个开源工具,是 Java 为开发图形化用户界面(GUI)提供的第三代技术,是目前使用广泛的开源且免费的 Java 集成开发环境,NetBeans IDE 可以使用 Swing 组件快捷开发具有 GUI 的 Java 应用程序。作为 Sun 公司官方认定的 Java 开发工具,NetBeans 的开发过程被认为最符合 Java 的开发理念。NetBeans 主要包括 IDE(集成开发环境)和 platform(平台)两部分,其中 IDE 是在平台基础上实现的,并且平台本身也开放给开发人员使用。

NetBeans IDE 是一个与 Microsoft Visual Studio 类似的集成开发环境,可以方便地

在 Windows、Mac、Linux 和 Solaris 中运行。NetBeans IDE 可以使开发人员利用 Java 平台快速创建 Web、企业、桌面以及移动的应用程序,NetBeans IDE 目前支持 PHP、Ruby、JavaScript、Ajax、Groovy、Grails 和 C/C++ 等开发语言。

利用 NetBeans IDE 可以快速、方便地开发 Java 应用程序,在 NetBeans IDE 中所有的开发工作都基于项目完成的。项目由一组源文件组成,即一个项目可以包含一个或多个源代码文件。此外,项目还包含了用来生成、运行和调试这些源文件的配置文件。

7. 初识 Java 程序的运行机制

Java 程序分为 Java 应用程序(Java Application)和 Applet 小程序(Java Applet)两类。Java 应用程序只有通过编译器编译生成 .class 文件,然后才能由 Java 解释器解释执行;Applet 小程序不能独立运行,必须嵌入 HTML 代码中,是由 Web 浏览器内含的 Java 解释器解释运行的非独立程序。

对于多数程序设计语言来说,其程序执行方式要么采用编译执行方式,要么采用解释执行方式。而 Java 的特殊之处在于,程序的运行既要经过编译又要进行解释。

Java 根据自身的实际需要将解释型和编译型相结合,采用一种“半编译半解释型”的执行机制,即 Java 程序的最终执行需要经过编译和解释两个步骤。首先,它使用 Java 编译器将 Java 源程序编译成与操作系统无关的字节码(二进制代码),而不是本机代码;其次,这种字节码必须通过 Java 解释器来执行。任何一台机器,无论安装什么类型的操作系统,只要配备了 Java 解释器,就可以执行 Java 字节码,而不必考虑这种字节码是在哪种类型的操作系统上生成的。

Java 通过预先把源程序编译成字节码,避免了传统的解释型语言执行效率低的性能瓶颈。但是,Java 字节码不能在操作系统上直接执行,必须在一个包含 Java 虚拟机(JVM)的操作系统上才能执行。

JVM 是一种可执行 Java 代码的虚拟计算机,这台虚拟机器在任何操作系统上都能提供给编译程序一个共同的接口。编译程序只需要面向虚拟机并生成其能够解释的代码,然后由解释器将虚拟机代码转换为特定操作系统的机器码执行即可。JDK 针对每一种操作系统平台提供的解释器是不同的,但是 JVM 的实现却是相同。

Java 应用程序执行过程:Java 源程序(.java 文件)经过 Java 编译器(javac.exe)编译后生成的字节码文件(.class 文件),由 Java 虚拟机(JVM)解释执行,然后 Java 解释器(java.exe)将其解释成特定机器上的机器码,并在特定的机器上执行,如图 1-1 所示。利用 JVM 把 Java 字节码与具体的软硬件平台隔离,就能保证在任何操作系统中编译的 Java 字节码文件都能在 JVM 上执行,即通常所说的“Write once,run everywhere”。



图 1-1 Java 程序运行机制

引导编程

1.1 Java 程序开发环境的搭建与配置

任务 1-1 安装与配置 JDK

【任务描述】

1. 下载与安装 JDK

从网上下载最新版本的 JDK,然后正确安装 JDK。

2. Windows 操作系统中配置 Java 运行环境

JDK 安装完成后,需要对 JAVA_HOME 以及 Path、ClassPath 系统环境变量进行正确地配置。其中 JAVA_HOME 值设置为 JDK 所在路径,例如作者计算机中为“C:\Program Files\Java\jdk1.8.0_60”,Path 环境变量为 bin 文件夹所在路径,那么计算机中 Path 变量应设置为“C:\Program Files\Java\jdk1.8.0_60\bin;”。注意,“;”是与其他路径之间的分隔符。ClassPath 环境变量设置为编译 Java 程序时需要的一些外部 class 文件所在路径,例如作者计算机中 ClassPath 变量便设置为“C:\Program Files\Java\jdk1.8.0_60\jre\lib;”。

【任务实现】

1. 下载 JDK

本书使用的是 Java SE Development Kit 8 版本,安装程序名为 jdk-8u20-windows-x64.exe。下载 Java Platform (JDK) 8u60 的网址为: <http://www.oracle.com/technetwork/cn/java/javase/downloads/jdk8-downloads-2133151-zhs.html>。

首先输入网址 <http://www.oracle.com/technetwork/cn/java/javase/downloads/index.html>,打开下载主页面,在该页面中单击“下载”选项卡,显示如图 1-2 所示的 Java SE 下载页面,然后单击“Java”按钮,打开如图 1-3 所示的 Java SE Development Kit 8u20 下载列表,在该下载列表中双击 jdk-8u20-windows-x64.exe 超链接,下载适用于 64 位 Windows 的 Java SE。

2. 安装 JDK

下载完 JDK 的安装文件后,双击 JDK 的安装文件 jdk-8u20-windows-x64.exe,启动安装向导,然后根据安装向导的提示信息一步一步完成 JDK 的安装即可。

3. 在 Windows 操作系统中配置 Java 运行环境

在开发 Java 程序之前,必须在计算机上安装并配置 Java 开发环境,本书采用 jdk1.8.0 版本。jdk1.8.0 成功安装后,在指定的安装位置将会出现 jdk1.8.0_60 文件夹,如图 1-4 所示。

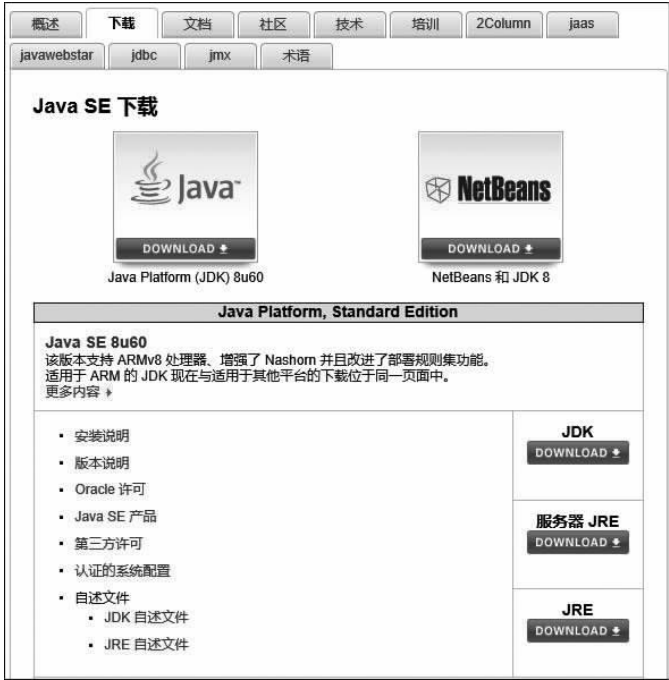


图 1-2 Java SE 的下载页面

Java SE Development Kit 8u20		
您必须接受 针对 Java SE 的 Oracle 二进制代码许可协议才能下载该软件。		
☐ 接受许可协议 ☒ 拒绝许可协议		
产品/文件说明	文件大小	下载
Linux x86	135.24 MB	jdk-8u20-linux-i586.rpm
Linux x86	154.87 MB	jdk-8u20-linux-i586.tar.gz
Linux x64	135.6 MB	jdk-8u20-linux-x64.rpm
Linux x64	153.42 MB	jdk-8u20-linux-x64.tar.gz
Mac OS X x64	209.11 MB	jdk-8u20-macosx-x64.dmg
Solaris SPARC 64 位 (SVR4 软件包)	137.02 MB	jdk-8u20-solaris-sparcv9.tar.Z
Solaris SPARC 64 位	97.09 MB	jdk-8u20-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 软件包)	137.16 MB	jdk-8u20-solaris-x64.tar.Z
Solaris x64	94.22 MB	jdk-8u20-solaris-x64.tar.gz
Windows x86	161.08 MB	jdk-8u20-windows-i586.exe
Windows x64	173.08 MB	jdk-8u20-windows-x64.exe

图 1-3 Java SE Development Kit 8u20 的下载列表

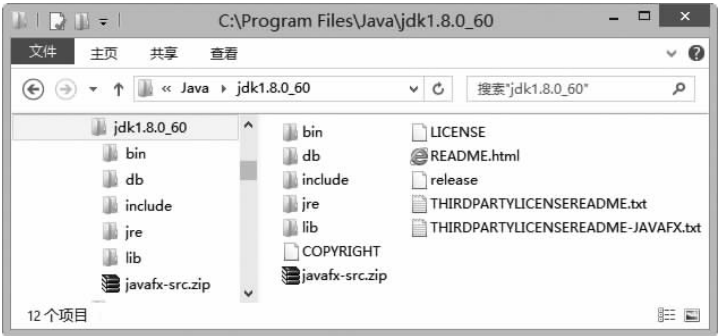


图 1-4 jdk1.8.0_60 文件夹中包含子文件夹和文件

在jdk1.8.0文件夹中包含了bin、db、include、jre、lib等多个子文件夹。bin子文件夹用来存放开发Java程序所用到的工具,例如编译指令javac、执行指令java等。db子文件夹用来存放使用嵌入式数据库所用到的资源。include子文件夹用来存放编译本地方法的头文件。jre子文件夹用来存放Java运行时环境(JRE)。lib子文件夹用来存放开发工具包的类库文件。

编译和执行Java程序必须经过两个步骤:第一步,将Java源代码文件(扩展名为.java)编译成字节码文件(扩展名为.class);第二步,解释执行字节码文件。

实现以上两个步骤要使用javac和java命令,在Windows操作系统的命令行窗口提示符下输入javac命令,则会输出以下提示信息:“javac不是内部或外部命令,也不是可运行的程序或批处理文件。”同理,输入java命令也会输出类似的提示信息。这说明虽然安装了JDK软件包,但是操作系统并不知道这两个命令所在的路径,也就无法执行命令了。Windows操作系统根据Path环境变量来查找命令的,Path的值就是一系列路径。因此,安装完JDK之后,还要设置Windows操作系统的环境变量并测试JDK的配置是否成功,才能正确编译和执行Java程序。

在使用JDK编译和运行程序之前,必须先设置环境变量,使系统知道JDK的安装路径。

这里以Windows 8操作系统为例说明环境变量的设置方法,首先右击桌面上的“这台电脑”图标,在弹出的快捷菜单中选择“属性”命令,打开“系统”窗口,在“系统”窗口左侧的导航栏区域单击“高级系统设置”链接,弹出“系统属性”对话框,并自动切换到“高级”选项卡,如图1-5所示。



图 1-5 “系统属性”对话框

(1) 设置 Path 环境变量

在“系统属性”对话框的“高级”选项卡中单击“环境变量”按钮,打开“环境变量”对话框,在该对话框的“系统变量”列表框中选择 Path。

然后单击“编辑”按钮,弹出“编辑系统变量”对话框,在该对话框“变量值”文本框中添加路径“C:\Program Files\Java\jdk1.8.0_60\bin;”,如图 1-6 所示。单击“确定”按钮,返回“环境变量”对话框,如图 1-7 所示。通过上述设置,系统将会顺利地在安装路径下的 bin 文件夹中找到 Java 编译器命令 javac、解释器命令 java 以及其他的工具命令。

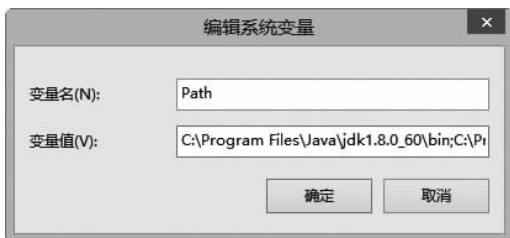


图 1-6 “编辑系统变量”对话框



图 1-7 “环境变量”对话框

注意：“C:\Program Files\Java\jdk1.8.0_60\bin;”为 Java 编译器命令 javac 和解释器命令 java 的路径。“;”为半角字符,用于与其他路径分隔,不要遗漏“;”,也不要删除 Path 系统变量中原有的变量值。

系统环境变量 Path 用来指定一个系统搜索的路径列表,以便自动搜索文件。如果待

搜索的文件在当前文件夹中没有找到,则依次搜索 Path 中的每一个路径;如果仍没有搜索到,则会给出提示信息。注意,用户变量只对当前用户有效,而系统变量则对所有用户都有效,系统变量的优先级高于用户变量。

(2) 创建环境变量 ClassPath

在“环境变量”对话框的“系统变量”区域单击“新建”按钮,弹出“新建系统变量”对话框,在“变量名”文本框中输入 ClassPath,然后在“变量值”文本框中输入“.; C:\Program Files\Java\jdk1.8.0_60\jre\lib”,如图 1-8 所示。单击“确定”按钮,这样就完成了 JDK 的安装和配置。

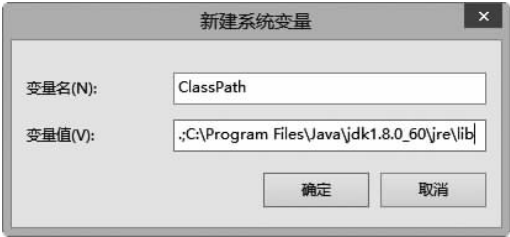


图 1-8 “新建系统变量”对话框

提示：ClassPath 变量用于设置搜索类文件的路径,ClassPath 的变量值中“.”不能缺少,代表当前文件夹。

当使用 Java 解释器执行 Java 程序时,对于 JDK1.4 及以前版本,如果没有设置环境变量 ClassPath 的值,可能会无法正常编译和运行 Java 程序。对于 JDK1.5 以上版本,即使不设置环境变量 ClassPath 的值,也可以在任何路径下正常编译和运行 Java 程序,因为对于 JDK1.5 以上版本,JRE 会自动搜寻当前路径下的类文件,而且使用 Java 的编译和运行工具时,系统可以自动加载 dt.jar 和 tools.jar 文件中的 Java 类。

(3) 测试 JDK 的环境配置

单击“开始”按钮,再选择“运行”命令,打开“运行”对话框,再“打开”下拉列表框中输入命令 cmd,如图 1-9 所示,单击“确定”按钮,打开“命令行”窗口(DOS 窗口)。在该窗口命令提示符后面输入命令“java-version”,再按 Enter 键,如果环境配置正确,就会显示 JDK 的版本信息,如图 1-10 所示。

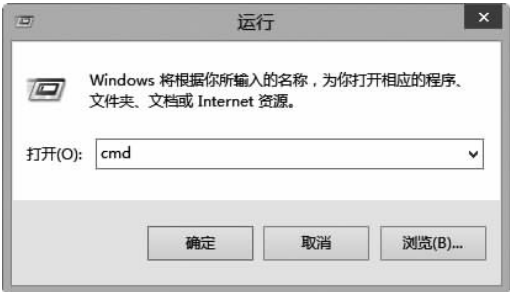


图 1-9 “运行”对话框



图 1-10 测试 JDK 的安装与配置

任务 1-2 下载与安装 NetBeans IDE

【任务描述】

从网上下载最新版本的 NetBeans IDE,然后进行正确安装。

【任务实现】

1. 下载 NetBeans IDE

在如图 1-2 所示的 Java SE 的下载页面中单击 NetBeans 的下载按钮,打开如图 1-11 所示的 JDK 8u60 with NetBeans 8.0.2 下载列表,在下载区域中选中复选框“Accept License Agreement”,然后在“Java SE and NetBeans Cobundle (JDK 8u60 and NB 8.0.2)”区域单击超链接“jdk-8u60-nb-8_0_2-windows-x64. exe”,再按照提示逐步下载适用于 64 位 Windows 的 NetBeans 和 JDK 升级版即可。

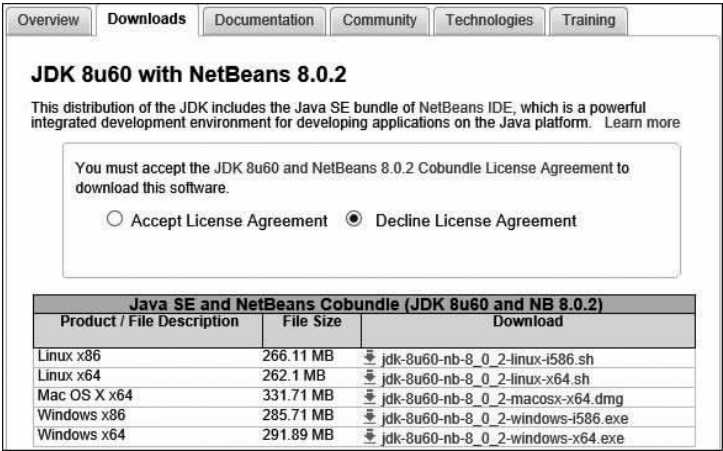


图 1-11 JDK 8u60 with NetBeans 8.0.2 下载列表

也可以直接输入网址 <https://netbeans.org/downloads/index.html>,在打开如图 1-12 所示的 NetBeans IDE 下载页面的 NetBeans IDE 8.0.2 下载列表中选择所需的下载包下载即可。



图 1-12 NetBeans IDE 8.0.2 下载列表

2. 安装 NetBeans IDE

下载完成 NetBeans IDE 的安装文件后,双击 NetBeans IDE 的安装文件 jdk-8u60-nb-8_0_2-windows-x64.exe,启动安装向导,然后根据安装向导的提示信息一步一步完成 NetBeans IDE 的安装和 JDK 升级即可。

1.2 NetBeans IDE 集成开发环境的熟悉与使用

任务 1-3 熟悉与使用 NetBeans IDE 的集成开发环境

【任务描述】

- (1) 启动 NetBeans IDE,熟悉 NetBeans IDE 集成开发环境的基本组成。
- (2) 在 NetBeans IDE 中试着创建 Java 标准项目、Java 包、Java 主类和 Java 类,然后试着更改运行的主类。
- (3) 在 NetBeans IDE 中试着运行 Java 标准项目。

【任务实现】

1. 启动 NetBeans IDE

在“开始”菜单中选择“程序”→NetBeans→NetBeans IDE 8.0.2,也可以直接双击桌面快捷方式 NetBeans IDE 8.0.2,开始启动 NetBeans IDE,显示如图 1-13 所示的启动过程界面。