

“软件蓝领”系列教材

JAVA

及构件技术

广东省教学教材研究室 编



广东教育出版社

“软件蓝领”系列教材

JAVA

及构件技术

广东省教学教材研究室 编



广东教育出版社

“软件蓝领”系列教材

JAVA 及构件技术

广东省教学教材研究室 编

*

广东教育出版社出版发行

(广州市环市东路472号12~15楼)

邮政编码: 510075

网址: <http://www.gjs.cn>

佛山市浩文彩色印刷有限公司印刷

(南海区狮山科技工业园A区)

787毫米×1092毫米 16开本 20印张 385 000字

2003年7月第1版

2005年7月第2版 2006年8月第3次印刷

ISBN 7 - 5406 - 5191 - 1/G·4631

定价: 30.00元

质量监督电话: 020 - 87613102 购书咨询电话: 020 - 34120440

前 言

中国共产党第十六次全国代表大会提出“以信息化带动工业化，以工业化促进信息化”的方针，对加快我国软件产业的发展，具有重大的指导意义。发展软件产业的一个关键是加强软件人才的培养，以使我国软件人才队伍的规模、结构和水平适应国家软件产业发展的实际需求。

根据把广东建设成为信息化强省的要求，广东省教育厅于2002年4月颁布了《广东省中等职业学校计算机软件（“软件蓝领”方向）专业课程设置（试行）》。这套“软件蓝领”系列教材是依据该要求编写的，其中包括《电子电路基础》、《计算机应用基础》、《软件界面设计技术》、《软件开发工具——Visual Basic教程》、《程序设计基础》、《软件开发工具——Delphi教程》、《软件开发工具——Active Server Page教程》、《软件工程技术》、《软件过程管理》、《SQL Server教程》、《Oracle教程》、《实践与案例分析（数据库方向）》、《Visual C++程序设计》、《JAVA及构件技术》、《实践与案例分析（中间件方向）》、《Lotus Domino/Notes教程》、《MS Exchange教程》、《实践与案例分析（群件方向）》、《软件技术英语（上）》、《软件技术英语（下）》等系列教材。

在本系列教材编写中我们努力做到：

一、以软件企业先进的科技力量为依托，注意与国际接轨，充分适应职业学校推行的学业证书和职业资格证书或国际认证的“双证”制度，突出教材的实用性、先进性、科学性和趣味性。

二、教材力求以“任务驱动”方式去编写，体现计算机软件新版本与实用技术，便于学生学习掌握。

三、教材中注意渗透实践性学习、互助性学习和自主性学习，充分调动学生学习积极性，把能力培养和技能培养贯穿于始终。

四、专业教材分成多个不同方向，突出“模块”特点，为不同学校根据自己的师资力量与学生就业取向，灵活选择不同专业方向提供方便。

“软件蓝领”系列教材（第二版）是在教学实践的基础上重新组织材料进行编写的，由李冠英教授任主编。《JAVA及构件技术》由杨培添、林雪锋等编写。

这次编写的“软件蓝领”系列教材（第二版）可能仍有不完善之处，需要在实践中进一步改进，敬请提出宝贵意见。

编 者

编写说明

JAVA及构件技术是软件设计人员的一门重要课程。本书的编写依据是广东省教育厅2002年颁布的《广东省中等职业学校计算机软件(“软件蓝领”方向)专业课程设置(试行)》有关课程的教学大纲。

本教材力图突破传统教材的章节结构体系,以课题为主线,贯穿整个教材结构。本教材分7大模块。

1. 课题描述

让学生在学习本课题之前,从整体上对本课题有一个宏观的了解,以利后续的学习。

2. 课题目标

提炼出本课题的知识点,让学生在学习时明确“学什么”,以调动学生的学习积极性。

3. 课题案例

以案例驱动为教材的编写脉络,通过案例来说明目标。所选择的案例图有典型性和紧贴目标。

4. 课题案例分析

通过案例进行分析,提炼出课题的知识点。

5. 课题实践

这是课题目标的实际应用。让学生通过课题的实践,掌握实际操作能力和加深对理论的理解。

6. 课题总结

归纳出课题要点,把课题中零散的知识归纳为完整的知识体系,便于学生记忆理解。

7. 课题测试

测试学生是否掌握了本课题的基本原理。测试的形式有选择题、填空题和简述题等。

8. 课题设计

这是课题的综合应用。要求学生在学习了本课题后,能够综合应用本课题的知识。

本教材由杨培添主编,参加编写的人员有:第一章由吴卫冬编写,第二章、第三章由胡咏红编写,第四章由陈志宁编写,第五章由林丹编写,第六章由范飞林编写,第七、八章由林雪锋编写,第九章由林雪锋、冯建群编写,第十章由莫炜卓编写,第十一章由苏宁枫、陈志宁编写,第十二章由苏宁枫编写,全书由林雪锋统稿。

本教材在编写过程中,得到华南师范大学李冠英教授等人的悉心指导,在此表示谢意。

采用本教材结构体系编写“软件蓝领”教材,是新的尝试。诚恳希望读者在使用中提出宝贵意见,以便在再版时修正。

编 者

目录

第一章 Java 快速入门	1
课题 1 简单的 Java 程序.....	1
课题 2 Java 的开发工具及安装.....	5
第二章 Java 语言基础	11
课题 1 Java 语言的基本组成.....	11
课题 2 数据类型.....	15
课题 3 运算符.....	24
第三章 Java 流程控制及数组	34
课题 1 分支语句.....	34
课题 2 循环语句.....	44
课题 3 一维数组.....	50
课题 4 多维数组.....	55
第四章 Java 与面向对象编程	60
课题 1 类.....	60
课题 2 对象.....	64
课题 3 多态与继承.....	68
课题 4 接口.....	73
课题 5 包.....	79
课题 6 访问修饰符.....	83
第五章 异常处理	93
课题 1 异常和异常类.....	93
课题 2 异常处理机制.....	97
第六章 多线程编程技术	103
课题 1 线程的实现.....	103
课题 2 线程的状态及调度.....	109
课题 3 线程同步.....	116

第七章 输入 / 输出流	124
课题 1 Java 的输入 / 输出流	124
课题 2 Java 的文件流	132
课题 3 Java 的管道流	141
课题 4 Java 的内存读写流	146
课题 5 Java 的过滤流	149
第八章 Applet 入门	158
课题 1 Applet 创建和 Applet 标记	158
课题 2 Applet 生命周期及其安全基础	162
课题 3 Applet 与 Java Application 的结合	166
课题 4 HTML 与 Applet 参数传递	174
课题 5 Applet 与环境的联系	177
第九章 Java 的多媒体技术	186
课题 1 图形的绘制	186
课题 2 文本信息处理	193
课题 3 颜色的设置	197
课题 4 图像文件的显示	201
课题 5 声音文件的播放	205
课题 6 动画设计与线程机制	210
课题 7 消除动画闪烁现象及放映图像动画	216
课题 8 媒体跟踪器技术及移动型动画	226
第十章 Java 的用户界面设计与事件处理	235
课题 1 UI 的基本元素	235
课题 2 版面设置	242
课题 3 窗口构造构件	246
课题 4 鼠标和键盘事件	253
第十一章 网络编程	261
课题 1 面向连接通讯的实现	261
课题 2 无连接通讯的实现	271
课题 3 利用 URL 访问网站	279
课题 4 利用 URLConnection 与服务器交互	283

第十二章	Java 数据库技术	287
课题 1	数据库的创建.....	287
课题 2	数据库的连接.....	290
课题 3	数据表的创建与删除.....	297
课题 4	数据表记录的查询操作.....	302
课题 5	数据表记录的更改操作.....	307

第一章 Java 快速入门

Java的诞生应追溯到1991年。当时，SUN MicroSystem公司的工程师为在电视、控制烤面包箱等家用电子产品上进行交互式操作而开发了一个名为Oak的软件。起初，Oak并没有引起人们的注意，甚至可以说是默默无闻。到了1994年下半年，Internet的迅猛发展以及WWW的快速增长，SUN的工程师将Oak技术应用于Web上，并且开发出了HotJava的第一个版本。在1995年下半年，SUN公司正式以Java为名称向全球推出。现在，Java已经在各领域获得广泛的应用。本章主要介绍Java的书写格式、特点及安装。

课题1 简单的Java程序

课题描述

Java是一种优秀的程序设计语言。它最大的优点就是与操作系统无关，即在Microsoft Windows、UNIX、Linux以及MacOS等多种操作系统上，都可以使用相同的代码。“一次编写，到处运行”是Java的特色所在，这使得Java在互联网上被广泛采用。本课题主要学习Java的特点。

课题目标

通过本课题的学习，掌握Java语言程序设计最基本的操作过程以及书写格式。

课题案例

下面是一个简单的Java程序的案例。让我们启动记事本，输入如下代码：

```
public class HelloWorld
{
    public static void main(String args[ ])
    {
        // 将 Hello World!输出到屏幕上
        System. out. println("Hello World!");
    }
}
```

将此文件保存为 HelloWorld. java。为方便起见，就存在C盘根目录下。

进入 DOS 命令，输入以下指令：

C: \javac HelloWorld. java

C: \java HelloWorld

其中，C: \javac HelloWorld. java 指令是将 java 源文件 HelloWorld. java 编译为 Java 类文件 HelloWorld. class。类文件如果没有指明存放路径，则默认存放在源文件所在路径下。本例中为 C 盘根目录。

C: \java HelloWorld 指令解释运行编译生成的类文件 HelloWorld. class。

将此程序编译、运行后，便在屏幕上显示程序的运行结果，如图 1 - 1 所示。

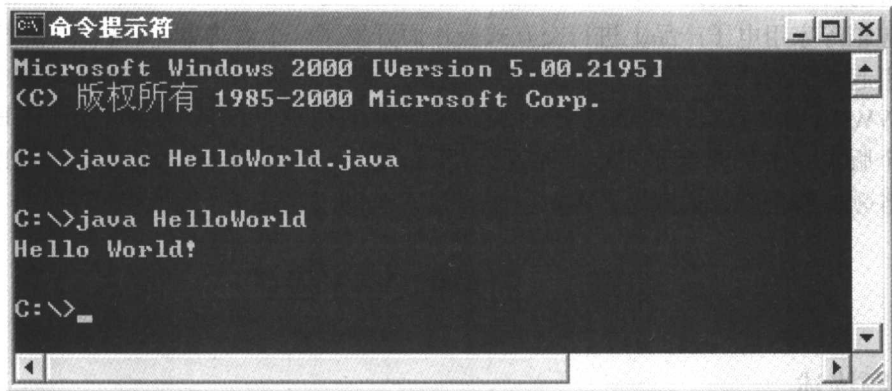


图 1 - 1 HelloWorld 程序的运行结果

课题案例分析

在这个案例中，可以看到每一种编程语言的程序都有其特定的书写格式，Java 也不例外。我们总结 Java 程序的书写格式和特点。

一、Java 程序的书写格式

Java 程序的书写格式：

1. 第 1 行和第 3 行中的 public、class、static、void、main、String 等称为关键字。这些关键字都是由 Java 来定义的，它们具有特定的含义，不能随便引用。

2. public 表示“公有的”，和它对应的是 private，即“私有的”。Java 中通过公有的和私有的来表示外界能否对某个对象或变量进行引用。

3. class 表示“类”，类是 Java 的基础。Java 是基于对象的编程语言，对象通过类创建，因此类和对象有密切的联系。HelloWorld 是类名，不是关键字，可由编程人员选取，由编程人员选取的名字称为标识符。

4. 第 3 行中的 main 称为类 HelloWorld 的方法，包含 main 方法的类称为主类。现在 HelloWorld 类中包含 main 方法，所以 HelloWorld 是主类。public static void 是对 main 方法的说明，其中 static 表示静态的，void 表示无返回值。在 Java 应用程序中，一般情况下，都有一个 main 方法，也只能有一个 main 方法。应将 main 方法说明为 public static void 的。String 表示字符串，args[] 表示 main 方法的参数，是字符串参数。关于 Static、void 和 args[] 的应用后面有详细的说明。

5. 第6行中的out是输出, print是打印, println是System.out类的一个方法, 称为打印方法, 打印后还要进行换行。第6行称为输出打印的语句。输出的内容就是引号中的内容: “Hello World!”。

6. Java是大小写敏感的。如果用UltraEdit编辑器编写时, 窗口中, 蓝色显示的是关键字, 其首字母要小写; 红色显示的是系统定义的类, 其首字母要大写。程序中应注意区分大小写, 如System不要写成system。保存文件, 文件名必须为HelloWorld.java(这里的H是大写)。

7. 由第2行和第8行一对大括号中包含的部分称为HelloWorld类体。

8. 由第4行和第7行一对大括号中包含的部分称为main方法体。

9. 第5行是一个对程序的解释, 对运行结果不起作用。

二、Java程序显著的特点

Java程序的显著特点:

(一) 简单

Java最初是为了对家用电器进行集成控制而设计的一种语言, 因此它必须简单明了。Java语言的简单性主要体现在以下三个方面: ① Java的风格类似于C++, 从某种意义上讲, Java语言是C及C++语言的一个变种; ② Java省略了C++中容易引发程序错误的地方, 如头文件、指针、结构、单元、运算符重载、虚拟基础等; ③ Java提供了丰富的类库。

(二) 面向对象

3

面向对象是软件工程的一次革命, 大大提高了软件开发的效率和水平。Java充分利用了面向对象的思想, 将数据封装于类中, 利用类的优点, 实现了程序的简洁性和便于维护性。

(三) 分布式

Java语言具有功能强大的、易于使用的网络能力。Java应用程序可以像访问本地系统那样通过URL访问远程对象, 非常适合于分布式系统的开发。

(四) 健壮性

Java语言有非常良好的健壮性, 因为Java致力于检查程序在编译和运行时的错误。类型检查帮助检查出许多开发早期出现的错误。Java自己操纵内存减少了内存出错的可能性。Java还实现了真数组, 避免了覆盖数据的可能。

(五) 安全性

在当今的网络环境下, 安全问题越来越成为人们关注的对象。相对于其他编程语言, Java有着不可比拟的安全性, 这也是Java备受青睐的原因之一。Java编译器并不控制内存的分配, 只是在程序运行时, 由操作系统分配, 有效避免了不友善代码的破坏或病毒程序的入侵。

(六) 可移植性

Java的平台无关性造就了Java高度的可移植性。Java程序可以方便地从一种操作系统不加改动或只做很少修改, 移植到另一种系统。Java语言当初设计的理念就是“一次编

写，到处运行”，它的可移植性是一种必然。

(七) 解释性

Java 解释器(运行系统)能直接运行目标代码指令。链接程序通常比编译程序所需资源少，所以程序员可以在创建源程序上花上更多的时间。

另外，为了建立 Java 作为网络的一个整体，Java 将它的程序编译成一种结构中立的中间文件格式。只要有 Java 运行系统的机器都能执行这种中代码，使得 Java 有结构中立的特点，Java 还具有高性能、多线性和动态等特点。



课题实践

上机实践 LifeJava. java，它用于在屏幕上显示文字。代码如下：

```
public class LifeJava
{
    public static void main(String args[ ])
    {
        System. out. println(" 生活就是战斗!");
    }
}
```

程序的运行结果如图 1 - 2 所示。

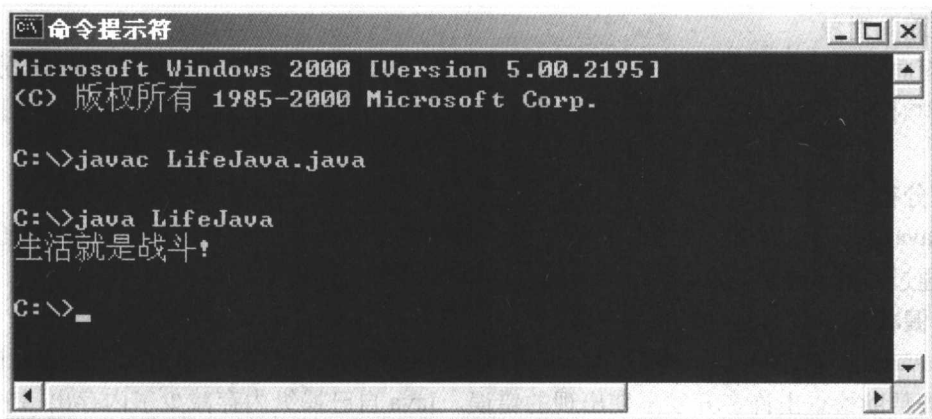


图 1 - 2 LifeJava 程序的运行结果



课题总结

1. Java 的操作过程:

- (1) 使用任意一种文本输入工具，输入代码。
- (2) 用 JDK 编译本程序。
- (3) 编译通过后，键入运行命令执行程序。

2. Java 最显著的特点有：简单；面向对象；分布式；健壮；结构中立；安全性；可移植性；解释性；高性能；多线程；动态等。

课题测试

1. 对照课题案例，简述 Java 程序书写的格式并与 C++ 作比较。
2. 简述 Java 程序显著特点有哪些？

课题设计

请编写一个简单的 Java 程序，在屏幕上显示如下信息：
你好，我是一个中职生！

课题2 Java 的开发工具及安装

课题描述

开发工具就是编辑、编译、调试以及运行程序所需的一整套环境或工具。JDK 是 Java 开发工具的核心工具。JDK 包括所有编译、运行 Java 程序所需要的工具：Java 基本组件、库、Java 解释器、小应用程序浏览器等。Java 是一种解释性语言。经 Java 编译器编译后的文件为字节码，不是机器码，因此不能直接执行。要看到程序的执行结果，就要靠 Java 虚拟机来解释执行。本章主要学习 Java 的安装、JDK 开发包和 Java 虚拟机的应用。

课题目标

通过本课题的学习，掌握 Java 软件环境的安装和了解常用 Java 开发工具的分类。

课题案例

一、安装机器要求

Java 对机器运行要求不高，下面给出的是基于 Windows 平台的机器要求：

1. 硬件要求：CPU P II 以上，64M 内存，100M 硬盘空间。
2. 软件要求：Windows 98/Me/NT/2000，Internet Explorer 5 以上。

二、程序清单

为了建立基于 JDK 的 Java 开发环境，需准备以下一些程序和文档：

1. j2sdk1_3_0 - win. exe，JDK 开发包。
2. j2sdk1_3_0 - doc. zip，相应的帮助文档。
3. UltraEdit 等文本编辑器。

三、安装过程

(一) 安装 JDK 开发包

运行 JDK 开发包 j2sdk1_3_0 - win. exe，安装过程中可以设置安装路径及选择组件，系统默认的安装路径为 C:\jdk1. 3. 0_02，默认的组件选择是全部安装。

安装成功后，C:\jdk1. 3. 0_02 中的文件和子目录结构如表 1 - 1 所示。

表 1 - 1 JDK 的目录结构

文件/目录名	功 能 说 明
src.jar文件	核心API所有类的源文件归档。
bin	包含编译器、解释器等可执行文件。
demo	包含源代码的程序示例。
include	用于本地计算机的C语言头文件。
include-old	头文件，为了向后兼容支持使用旧接口的本地代码编程技术。
jre	Java运行时环境的根目录。
lib	可执行程序使用的归档包文件。

为了能正确使用JDK，需要手工配置一些环境变量，Windows 98与WindowsNT/2000的配置略有不同，下面分别介绍。

(二) Windows 98 环境配置

在C:\autoexec.bat文件中，添加以下内容：

```
set path=%path%; C:\jdk1.3.0_02\bin
```

```
set classpath=.; C:\jdk1.3.0_02\lib; %classpath%
```

C:\jdk1.3.0_02为程序安装路径，如果安装时做了修改，这里必须做出相应的修改。“set path”语句设置Java编译运行程序的路径，“set classpath”语句设置Java包的路径，其中的“.”表示在当前目录下均可执行Java程序。

修改后存盘，然后重新启动计算机，即完成了设置。

(三) Windows NT/2000 环境设置

在桌面“我的电脑”图标上单击右键，选择“属性”，出现系统特性设置界面，系统特性环境变量如图1-3所示。在“高级”选项卡中单击“环境变量”按钮环境变量如图1-4所示。

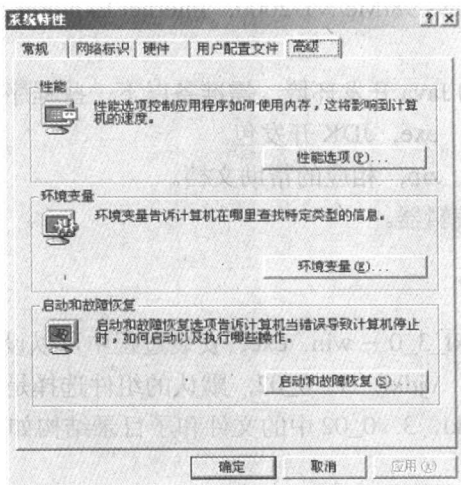


图 1 - 3 系统特性环境变量

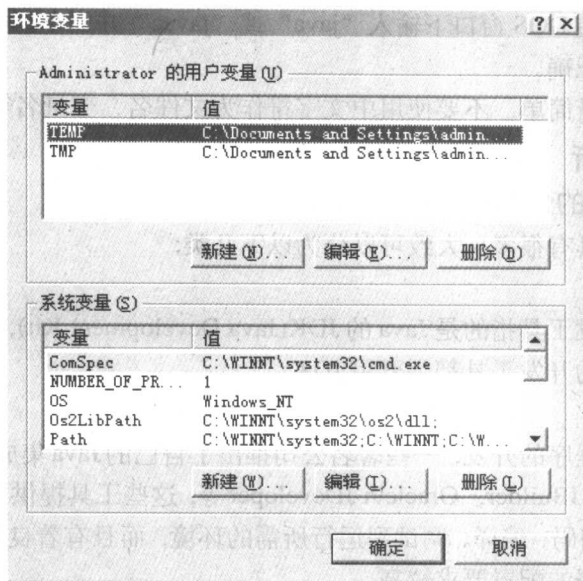


图 1-4 环境变量

在“系统变量”组框中找到“path”，单击“编辑”按钮，将“%PATH%; C:\jdk1.3.0_02\bin”加到“变量值”文本框中，编辑系统变量对话框如图 1-5 所示。也可以在原来的值后面加上以下内容：“; C:\jdk1.3.0_02\bin”。单击“确定”按钮结束编辑变量。

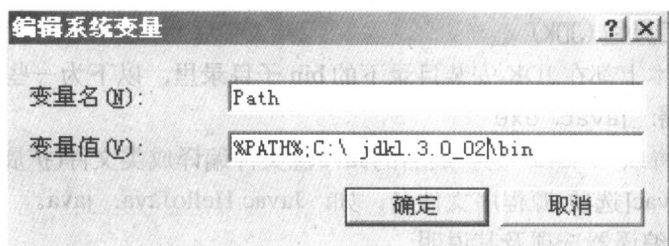


图 1-5 编辑系统变量对话框

若没有找到“Path”，则选择“新建”按钮，设置变量名为“path”，变量值为“C:\jdk1.3.0_02\bin”。

用同样方法设置环境变量“classpath”，其值为

“.; C:\jdk1.3.0_02\lib\tools.jar; C:\jdk1.3.0_02\lib\dt.jar”。

新建系统变量对话框如图 1-6 所示。

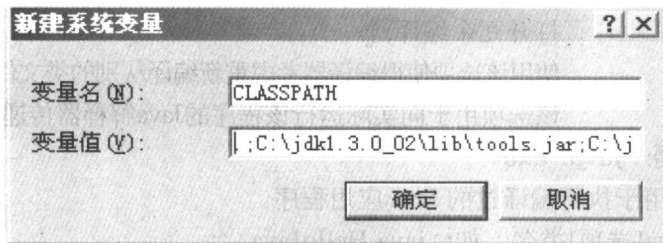


图 1-6 新建系统变量对话框

设置完成后, 在 DOS 窗口下输入 “java” 或 “javac” 并回车, 如果出现其用法参数提示信息, 则安装正确。

Java 程序应尽量简单, 不要使用中文字符作为文件名、变量名等, 便于以后操作。

课题案例分析

一、Java 开发工具的分类

Java 的开发工具有很多, 大致可以分为以下几类:

(一) 核心开发工具

Java 的核心开发工具指的是 Java 的 JDK(Java Development Kit), 它是 Java 开发工具的核心, 任何其他的开发工具都需要用 JDK。

(二) 集成开发工具

为了方便 Java 程序的开发, 一些著名公司推出了自己的 Java 集成开发工具, 如: Java WorkShop, Borland JBuilder, Oracle9i JDeveloper 等, 这些工具提供了 Java 软件开发的一整套环境, 集成了编码、编译、调试到运行所需的环境, 而且有着良好的可视化效果。不过, 这些工具对机器的配置要求较高。

(三) 其他工具

除以上介绍的开发工具外, 还有一些其他的开发工具, 比如: TextPad, KawaPro 等, 这些工具比较小巧, 借助于 JDK 软件包, 加上自己的编辑器, 比较适合于系统配置较低的机器。

二、Java 开发工具包 (JDK)

这些工具基本上都在 JDK 安装目录下的 bin 子目录里, 以下为一些基本工具:

(一) Java 编译器: javac. exe

Java 编译器将以 “. java” 为扩展名的 Java 源文件编译成类文件(扩展名为: “. class”) 语法为: Javac[选项]源程序文件名。如: Javac HelloJava. java。

以下为 Java 编译器选项及其说明:

- classpath<path> 覆盖缺省的 CLASSPATH 环境变量并指定新的 Java 类搜索路径;
- d<directory> 指定存放生成的类文件的目录, 注意使用绝对路径;
- g 使用该选项可以用类文件生成调试表, 供 jdb 使用;
- nowarn 关闭编译器的警告输出;
- O 打开优化处理;
- verbose 打开冗余编译;
- depend 使用该选项使得编译器考虑重新编译从别的类文件中引用的类文件;
- J option 该选项用于向实际运行该程序的 Java 解释器传递一个单独的变量。

(二) Java 解释器: java. exe

Java 解释器用于执行编译过的 Java 应用程序。

语法为: Java[选项]类名。如: java HelloJava。

- help 显示所有选项;

- version 显示 JDK 的版本;
- v 显示被加载的所有类;
- cs 检查源代码是否比类文件新(未编译过), 若新就编译源码;
- debug 使用远程 Java 文件, 该文件要在后面用 jdb 工具调试;
- prof 将配置信息输出到 JAVA. PROF 文件中;
- classpath<path> java 在指定的目录中寻找类文件;
- noasyncgc 关闭异步的无用单元集;
- verify 验证所有被加载的类;
- verifyremote 验证引入的类或继承的类;
- noverify 关闭类验证;
- mx value 设置最大的 java 堆空间为 value 的值;
- ms value 将初始的 Java 堆空间设置为 value 的值;
- ss val 将 C 过程的堆栈空间设置成由 val 指定的值;
- oss val 将 java 过程的堆栈空间设置成由 val 指定的值。

(三) Applet 浏览器: appletViewer. exe

Applet 浏览器用于运行 Java Applet 程序。

语法为: AppletViewer[选项]URLS(或. html 文件)。

(四) java 反编译器: javap. exe

Java 反编译器用于显示编译后的字节代码。

语法为: javap[选项]类名。

(五) C 头文件创建器: javah. exe

是用于当需要使用 C 语言扩展 Java 代码时创建 C 语言头文件。

语法为: javah[选项]类名。

(六) java 文档生成工具: javadoc. exe

它是用于在 Java 源文件中生成基于标志的 HTML 文件。

(七) java 调试器: jdb. exe

它是 Java 环境的高度工具。

课题实践

如果 DOS 命令下输入: java HelloWorld, 出现以下结果: Bad command or file name. 可能是什么原因?

课题总结

Java 开发工具有如下几类: 核心开发工具、集成开发工具和其他工具(如 TextPad、KawaPro 等)。Java 开发工具包(JDK)有: Java 编译器、Java 解释器、Applet 浏览器、Java 反编译器、C 头文件创建器、Java 文档生成工具和 Java 调试器。