

Java 程序设计简明教程

刘克成 郑珂 王晓 编著

中国电力出版社

内容提要

Java 是目前最流行的编程语言之一, 本书以基本概念为起点, 由浅入深、循序渐进地介绍了 Java 程序设计的基本概念、方法和应用。主要内容包括: Java 开发环境, Java 语言基础, 面向对象技术, Java 的核心类库, Java 的异常处理, Java 的多线程机制, 输入/输出、图形界面设计, Java 的数据库编程, Java 的网络编程技术和 Java 的 Web 应用等。

本书把概念和方法与详细的程序代码紧密地联系起来, 并结合图示, 使读者易学易用。本书既可作为大学本科和高职高专院校计算机专业的教学用书, 也可作为从事 Java 应用程序开发的软件网络工程师及 Java 爱好者的参考用书。

图书在版编目(CIP)数据

Java 程序设计简明教程 / 刘克成, 郑珂, 王晓编著. —北京: 中国电力出版社, 2007.7

21 世纪高等学校规划教材

ISBN 978-7-5083-5596-2

I. J... II. ①刘... ②郑... ③王... III. JAVA 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2007) 第 074570 号

丛 书 名: 21 世纪高等学校规划教材

书 名: Java 程序设计简明教程

出版发行: 中国电力出版社

地 址: 北京市三里河路 6 号

邮政编码: 100044

电 话: (010) 68362602

传 真: (010) 68316497, 88383619

服务电话: (010) 58383411

传 真: (010) 58383267

E-mail: infopower@cepp.com.cn

印 刷: 北京市同江印刷厂

开本尺寸: 185mm×260mm 印 张: 17.5 字 数: 425 千字

书 号: ISBN 978-7-5083-5596-2

版 次: 2007 年 7 月北京第 1 版

印 次: 2007 年 7 月第 1 次印刷

印 数: 0001—4000 册

定 价: 27.00 元

敬告读者

本书封面贴有防伪标签, 加热后中心图案消失
本书如有印装质量问题, 我社发行部负责退换

版 权 专 有 翻 印 必 究

前 言

Java 是目前最流行的编程语言之一，基于 Java 语言产生的 Java 体系是世界上很多开发人员用于开发企业级应用的首选。目前，Java 提供三方面的应用，用于编写工作站的 J2SE、可扩展的企业级应用的 Java 2 平台 J2EE 以及用于开发嵌入式 Java 2 电子消费平台的 J2ME。Java 已经不仅仅是一种编程语言，更是一种技术，一个开发平台。

本书共有 12 章。第 1 章介绍 Java 语言的历史和特点，介绍了 Java 虚拟机、Java 平台、Java 的开发工具和开发步骤。第 2 章介绍了 Java 语言的基本要素，包括 Java 语言的标识符、关键字、数据类型、表达式、运算符、数据类型的转换、变量和变量的作用域等。第 3 章介绍了面向对象程序设计编程概念和方法，并且介绍了在 Java 语言中如何实现这些概念和方法。第 4、5 章主要介绍了在编写 Java 程序时用到的主要类库，包括集合、Math 等，另外还介绍了 Java 中数组和字符串的使用。第 6 章介绍了 Java 的异常处理，包括异常的产生、异常的捕获、异常的声明抛出和自定义异常。第 7 章主要介绍 Java 的多线程机制，包括线程的创建、调度、同步和互斥。第 8 章介绍了 Java 的输入和输出，包括字节流、字符流、文件和目录以及过滤流的使用。第 9 章介绍了 Java 的图形用户界面设计，包括 AWT、Swing 组件、容器的布局管理和事件处理。第 10 章介绍 Java 的数据库编程，包括了传统的 JDBC 连接方式和 Hibernate 框架。第 11 章介绍了 Java 的网络应用程序编写，包括 TCP 方式和 UDP 方式。第 12 章介绍了基于 Java 的 Web 编程技术，包括服务器端程序 Servlet、Java 的表示技术 JSP、Web 应用的结构和 Struts 框架。

本书既可作为高校计算机及相关专业的 Java 语言程序设计的课程教材，也可以作为使用 Java 从事软件开发人员的参考书，本书主要有以下特点：

(1) 注重理论和实践相结合，每章中都带有大量的典型实例，每个实例体现本章和前面章节学习过的基本内容，实例都经过调试运行，程序结构严谨。

(2) 注重基本知识的理解和基本技能的培养，讲解深入浅出，通俗易懂，易学易用。

(3) 注重跟随 IT 的发展潮流，讲述最新的经过测试的稳定的技术，使用的工具也是 IT 行业中最流行、最有发展前途的。

(4) 注重弘扬“自由软件”精神，在本书中使用的全部是“开源”项目，从开发工具 Eclipse 到 JSP 和 Servlet 引擎 Tomcat 都是开源代码。

为配合学习，另编写了《Java 程序设计习题解答和实验指导》一书作为本书的配套教材。

本书由刘克成、郑珂、王晓承担主要编写工作。第 1、4、5 章由刘克成编写，第 6、11 章由郑珂编写，第 3 章由王晓编写，第 10 章由彬编写，第 12 章由肖飞编写，第 7 章由张松娟编写，第 8 章由王玉磊编写，第 9 章由李坤编写，第 2 章由张菊艳编写。全书统稿工作由刘克成、郑珂、王晓负责完成。

由于作者水平所限，书中难免存在不足之处，敬请广大读者批评指正。

编 者
2007 年 5 月

目 录

前 言

第 1 章 Java 概述.....	1
1.1 Java 简介.....	1
1.2 Java 语言的开发环境.....	3
1.3 Java 语言的综合开发工具 Eclipse	7
小结.....	12
习题.....	12
第 2 章 Java 语言基础.....	13
2.1 Java 程序的构成.....	13
2.2 数据类型、常量与变量.....	15
2.3 操作符.....	20
2.4 表达式、语句和块.....	25
2.5 流程控制.....	26
小结.....	37
习题.....	37
第 3 章 面向对象程序设计.....	39
3.1 面向对象问题的提出和求解对象.....	39
3.2 面向对象的基本概念.....	40
3.3 对象基础和简单数据对象.....	44
3.4 类.....	48
3.5 继承.....	60
3.6 接口和包.....	69
小结.....	73
习题.....	73
第 4 章 Java 的核心 API.....	76
4.1 java.lang 包中的类.....	76
4.2 数组.....	79
4.3 字符串的处理.....	83
4.4 实用工具类.....	85
小结.....	87
习题.....	87
第 5 章 集合.....	88
5.1 集合框架简介.....	88
5.2 Collection 接口.....	89

5.3	Set 接口	90
5.4	List 接口	92
5.5	Map 接口	93
5.6	算法类	95
	小结	99
	习题	99
第 6 章	异常处理	100
6.1	Java 异常处理的基本知识.....	100
6.2	异常的处理.....	103
6.3	抛出异常的语句.....	106
6.4	自定义异常.....	107
	小结	109
	习题	109
第 7 章	多线程处理	110
7.1	线程	110
7.2	线程生命周期.....	111
7.3	线程的创建.....	112
7.4	线程的调度与控制.....	116
7.5	线程的同步和互斥.....	122
	小结	131
	习题	131
第 8 章	输入/输出	132
8.1	I/O 数据流	132
8.2	文件的处理.....	142
8.3	过滤流	150
8.4	对象的序列化.....	156
	小结	160
	习题	160
第 9 章	图形用户界面	162
9.1	AWT 概述.....	162
9.2	容器	163
9.3	布局管理器.....	167
9.4	Java 的事件处理	173
9.5	常用的 AWT 组件.....	179
9.6	JFC 和 Swing 图形组件.....	191
	小结	197
	习题	198
第 10 章	Java 数据库编程	199
10.1	关系数据库概述.....	199

10.2	常用的 JDBC API.....	206
10.3	使用 ORM 技术操作数据库.....	220
	小结.....	228
	习题.....	228
第 11 章	Java 网络编程.....	229
11.1	URL.....	229
11.2	TCP 网络设计.....	235
11.3	UDP 网络程序设计.....	244
	小结.....	248
	习题.....	249
第 12 章	Java Web 应用.....	250
12.1	Web 应用概述.....	250
12.2	Servlet 技术.....	253
12.3	JSP 网络编程.....	258
12.4	Struts 框架.....	263
	小结.....	270
	习题.....	271
	参考文献.....	272

第 1 章 Java 概 述

1.1 Java 简 介

Java 最早诞生于 1991 年，起初被称为 OAK 语言，它是 Sun 公司为一些消费性电子产品而设计的一个通用环境。他们最初的目的只是为了开发一种独立于平台的软件技术，而且在网络出现之前，OAK 可以说是默默无闻，甚至差点夭折，但是网络的出现改变了 OAK 的命运。在 OAK 出现之前，Internet 上的信息内容都是一些乏味死板的 HTML 文档。这对于那些迷恋于 Web 浏览的人们来说简直不可容忍，他们迫切希望能在 Web 中看到一些交互式的内容，开发人员也极希望能够在 Web 上创建一类无需考虑软硬件平台就可以执行的应用程序，当然这些程序还要有极大的安全保障。对于用户的这种要求，传统的编程语言显得无能为力，Sun 公司的工程师们敏锐地察觉到了这一点，从 1994 年起，他们开始将 OAK 技术应用于 Web 上，并且开发出 Hot Java 的第一个版本。当 Sun 公司 1995 年正式以 Java 这个名字推出的时候，几乎所有的 Web 开发人员都感到兴奋。

Java 的发展速度是惊人的，在 1995 年发布了第一个 Java 版本，而 1996 年 1 月就发布了 Java 的第一个开发包 JDK V1.0，而在 1997 年 2 月发布的 Java 语言的开发包 JDK V1.1，奠定了 Java 语言在计算机语言中的地位。1998 年 12 月 Sun 公司发布 Java 2 平台 JDK V1.2，Java 2 平台是 Java 发展史上的里程碑。1999 年 6 月，Sun 公司重新组织 Java 平台的集成方法，并将企业级应用平台作为 Java 语言的发展方向，它主要由 3 部分组成：

- (1) J2EE——Java 2 Enterprise Edition，可扩展的企业级应用的 Java 2 平台。
- (2) J2SE——Java 2 Standard Edition，用于工作站，PC 机的 Java 2 标准平台。
- (3) J2ME——Java 2 Micro Edition，嵌入式 Java 2 电子消费平台。

Java 技术不仅仅是程序设计语言也是一种技术平台，它还是一种新的计算概念。

1.1.1 Java 语言的特点

Java 程序设计语言是一种高级语言，首先，它简单、面向对象、不依赖机器的结构，不受 CPU 和环境的限制，它具有可移植性、鲁棒性、安全性，并提供了并发机制，具有很高的性能；其次，Java 语言提供了丰富的类库，使程序设计人员可以方便地建立自己的系统。它的主要特点如下：

(1) 平台无关性。平台无关性是指 Java 能运行于不同的平台。Java 引进虚拟机原理，并运行于虚拟机上，使用 Java 编写的程序能在世界范围内共享。这是因为 Java 的数据类型与机器无关，Java 虚拟机 (Java Virtual Machine) 是建立在硬件和操作系统之上，实现 Java 二进制代码的解释执行功能，提供了对于不同平台的接口。

(2) 安全性。Java 的编程类似于 C++，学习过 C++ 的读者将很快掌握 Java 的精髓。但

是 Java 舍弃了 C++ 的指针对存储器地址的直接操作，程序运行时，内存由操作系统分配，这样可以避免病毒通过指针侵入系统，另外 Java 对程序提供了安全管理器，防止程序的非法访问。

(3) 面向对象。Java 吸取了 C++ 面向对象的概念，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，类的封装性、继承性等有关对象的特性，使程序代码只需一次编译，然后通过上述特性反复利用，程序员只需把主要精力用在类和接口的设计和应用上。Java 提供了众多的一般对象的类，通过继承就可使用父类的方法。在 Java 中，类的继承关系是单一的非多重的，一个子类只有一个父类，子类的父类又有一个父类，Java 提供的 Object 类及其子类的继承关系如同一棵倒立的树形，根类为 Object 类，Object 类功能强大，经常会使用到它及其他派生的子类。

(4) 分布式。Java 建立在扩展 TCP/IP 网络平台上，库函数提供了利用 HTTP 和 FTP 协议传送和接收信息的方法，这使得程序员使用网络上的文件和使用本机上的文件一样容易。

(5) 健壮性。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，这些功能特征大大提高了开发 Java 应用程序的周期。

(6) 自动垃圾回收。Java 系统不仅可以分配对象所需的内存空间，而且还需要跟踪资源的使用情况，定期检测出不再使用的内存，由系统自动回收并用于再次分配，所以，Java 程序员不必考虑对象所占用内存的释放问题，这样可以减轻程序员的负担，提高程序的安全性，可以避免因为内存耗尽和资源的重用导致系统瘫痪。

1.1.2 Java 平台

平台是一个程序运行的软硬件环境。大家所熟悉的像 Windows 2000、Linux 等是操作系统和硬件组成的基于硬件的平台，Java 平台不同于这些平台，它是一个运行在一个基于硬件的平台之上的“纯软件”平台。Java 平台有两个主要部分：

(1) Java 虚拟机——The Java Virtual Machine (JVM)。

(2) Java 程序设计接口——The Java Application Programming Interface (JAPI)。

Java 虚拟机 (JVM) 是一个软件模拟实现的虚拟计算机，是 Java 跨平台的基础，它解释运行 Java 程序。执行过程如图 1-1 所示，在使用时 Java 的源文件 (.java)，经编译生成字节码文件 (.class)，然后有 JVM 解释执行字节码文件。从这个过程可以看出，为了运行 Java 程序，计算机上必须安装 JVM 和 Java 解释器。

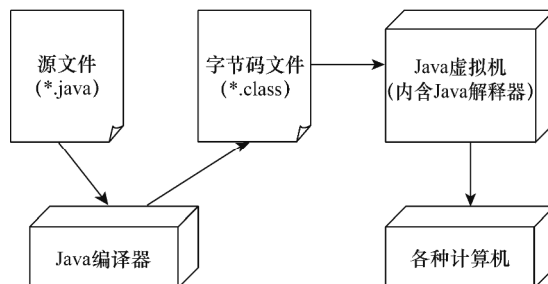


图 1-1 Java 应用程序的编译运行过程

Java API 运行在 JVM 之上，Java API 是一个庞大的软件组件集合，它提供了许多实用的功能，比如用于图形设计的 AWT。Java API 把相关的类和接口组织成一个类库，这些类库称为包（packages）。图 1-2 描述了一个 Java 程序和 Java 平台的关系。

从图 1-2 中可以看出，Java 程序是运行在 Java 平台上的而不是直接在基于硬件的平台上。要运行 Java 程序只需要安装 JVM，但是要开发 Java 程序就需安装所必需的 Java API 和 JVM。因此 Java 平台也可以称为 Java 开发工具集 Java Developer Kit，简称 JDK，Java 2 的开发工具集又被称为 J2SDK。这些可以从 Sun 公司的官方网站 <http://java.sun.com/products> 上免费下载，目前最新的版本是 JDK1.5。

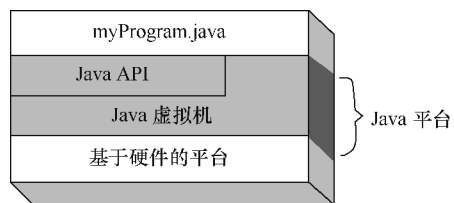


图 1-2 Java 平台

1.2 Java 语言的开发环境

1.2.1 Java 语言的开发环境 JDK

Java 是一个完全跨平台的技术，它的产品可以在不同的平台（像 Windows、Linux）上运行。本章将主要讲解 Java 在 Windows 平台上的应用，下面将介绍在 Windows 下安装和配置 JDK。

1. 下载 JDK

为了在 Windows 环境下建立 Java 的开发环境，需准备下列程序（当然也可以在 Sun 公司的网站上下载更新的版本）。

- (1) jdk-1_5_0_01-windows-i586-p.exe，JDK 的开发包。
- (2) jdk-1_5_0_01-doc.zip，JDK 的帮助文档。

2. 安装 JDK

安装 JDK 非常简单，只要运行 jdk-1_5_0_01-windows-i586-p.exe 文件就可，在安装的过程可以设置安装的路径，还可以选择安装的组件，系统默认的安装路径是 C:\Program Files\Java（假设系统安装在 C 盘）。

在安装目录中有两个子目录 jdk1.5.0_01 和 jre1.5.0_01。前者就是开发 Java 程序时所用的 Java 开发工具集 JDK，在该文件夹内也包含一个 jre 文件夹和后面的 jre1.5.0_01 内容一样。那么 jre 是用来做什么的？JRE 为 Java RunTime Envirment 的简称，包括 Java Runtime Environment 和 Java Plug-in 两部分。JRE 是用来运行、测试和传输 Java 应用程序，它包括 Java 虚拟机、Java 平台核心类和一些支持文件，它不包含开发工具——编译器、调试器和其他工具。因此要开发 Java 程序的话除了 JRE 之外，还要安装其他组件，这些组件包含在 jdk1.5.0_01 目录内。

3. 配置 JDK

安装完 JDK 后，并不能立即使用，要能够进行 Java 程序的编译和调试，必须告诉操作系统完成这些操作的应用程序在什么位置，因此必须先配置环境变量，在进行环境变量设置之前，先来看看在 jdk1.5.0_01 中包含哪些目录文件及它们的功能，如表 1-1 所示。

表 1-1 JDK 的目录结构

文件/目录名	功能描述
bin	包含编译器、解释器等主要的可执行文件
lib	可执行程序使用的归档类包
jre	Java 运行时环境的根目录
src.zip	核心 API 所有类的源文件的归档包
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
sample	范例

从表 1-1 中可以看出，目录 bin 包含了编译和运行 Java 程序的可执行文件，目录 lib 包含了开发 Java 程序所需的基本类库。所以为了能够让系统正确使用 JDK，就必须在系统的环境变量中指定目录 bin 和 lib 的位置。下面以 Windows Server 2003 系统为例介绍环境变量的配置。

在桌面上【我的电脑】单击鼠标右键，选择【属性】，出现系统设置界面，在【高级】选项卡中单击【环境变量】按钮，如图 1-3 所示。

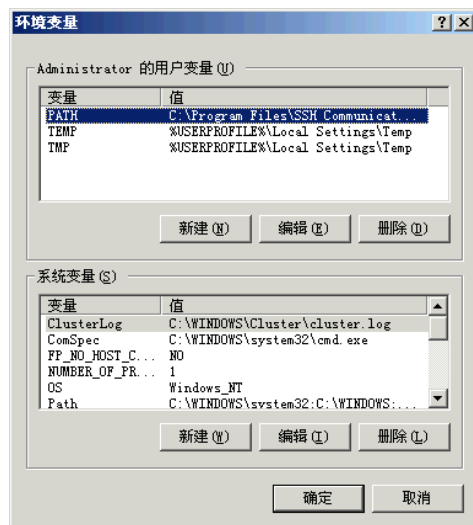


图 1-3 系统的环境变量

在【系统变量】组框内找到“Path”，该变量是用来告诉操作系统一些可执行文件的位置。单击【编辑】按钮，将“C:\Program Files\Java\jdk1.5.0_01\bin”加入到【变量值】文本框内（C:\Program Files\Java 为 JDK 的安装目录）如图 1-4 所示。最后单击【确定】按钮。

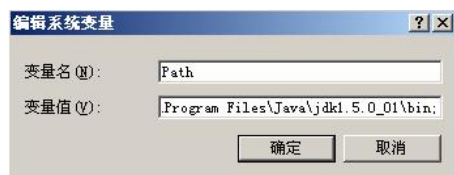


图 1-4 环境变量 Path 的编辑

如果没有找到该变量，可以在图 1-3 中的环境变量对话框中单击“新建”按钮，设置变量名为“Path”，变量值为“C:\Program Files\Java\jdk1.5.0_01\bin”。

为了能够在 Java 程序中使用 Java 类库，可以用同样的方法设置环境变量“classpath”，该变量的值为“.C:\Program Files\Java\jdk1.5.0_01\lib”（注意目录的前面加上“.”）。

4. 验证

如果设置正确，在 DOS 命令提示符下输入“Javac”命令，在屏幕上会出现这些命令的用法参数。否则将会显示“系统找不到 Javac 文件”。设置正确后就可以进行 Java 程序的编译和运行了。

5. 安装 Java 的帮助文档

直接解压缩 jdk-1_5_0_01-doc.zip 文件，运行解压缩后的 api\index.html 文件，就可以打开 Java 帮助文档。

1.2.2 一个简单的 Java 应用程序及其开发过程

Java 可以开发三种程序：Java 应用程序（Application）、Java 小应用程序（Applet）和 Java 服务器端程序（Servlet），Java Application 是可以独立运行的程序；Java Applet 是必须要嵌入到网页中的程序；而 Servlet 是运行在 Web 服务器端的程序。本书主要讲解 Java 应用程序，在第 12 章 Java Web 应用中讲解 Java 的服务器程序，Java 小应用程序本书不作介绍。

首先认识一个简单的 Java 应用程序，然后讲解使用 JDK 开发一个 Java 程序的过程。

【例 1-1】 计算 1~10 这十个数的和，并且显示结果。

```
/**
 * @author zhengke
 *
 */
public class NumberAdd{
    public static void main(String args[]){
        int sum=0; //定义变量并进行初始化
        for(int i=0;i<=10;i++){
            sum+=i;
            System.out.println("1 到 10 的和为: "sum);
        }
    }
}
```

在上面的例子中，以“/**”开头以“*/”结束的部分为 Java 注释部分，Java 的编译器将忽略注释部分，Java 程序中注释有三种类型，分别如下：

- (1) /* text */: 编译器忽略从/*到*/的所有文字。
- (2) /** documentation */: 编译器忽略文档 documentation 注释的所有文字。
- (3) // text: 编译器忽略//之后本行的所有文字。

Java 程序的所有语句必须写在类中，关键字“class”用来声明一个类，NumberAdd 是类名。

方法 main()是程序的入口，它是一个 Java application 程序必须包含的一部分，而且每个应用只能有一个 main()方法。public 指明这个 main 方法可以被所有的对象使用（can be called by

any object), static 指明这个 main 方法是一个类方法。void 指明这个 main 方法不返回任何值; 参数 String[] args 是传递给 main()方法的参数, 称为命令行参数。

下面是以这个程序为例讲解开发 Java Application 的步骤。

1. 准备工作

要进行 Java 程序的开发, 需要使用两种工具:

(1) Java 的开发包 JDK。

(2) 一个文本编辑器。任何一个文本编辑器, 都可以使用, 本节使用 Windows Server 2003 操作系统中自带的记事本。

2. 编写 Java 源程序

(1) 打开记事本, 在编辑窗口输入以下代码。

```
public class NumberAdd{
    public static void main(String args[]){
        int sum=0; //定义变量并进行初始化
        for(int i=0;i<=10;i++){
            sum+=i;
        }
        System.out.println("1 到 10 的和为: "sum);
    }
}
```

在程序中应注意区分大小写, 如 String 不能写成 string, 这是因为 Java 是对大小写敏感的。在程序中 String 和 System 是系统定义的类库, 其首字母必须是大写的; 像 public、static、void、class 等是 Java 的关键字, 其首字母必须是小写的。

(2) 保存 Java 源程序。Java 源程序的扩展名为 java, 文件的名称和程序中类的名称保持完全一致, 在该例中文件的名称即为 NumberAdd.java。进行文件的保存时要注意文件名称的大小写, 另外在保存窗口中保存类型必须选为“所有文件”。



图 1-5 保存 Java 源程序

另外注意记住文件的保存位置, 在本节中作者把它保存在“C:\Java”目录下。

3. 编译源程序

进入 MS-DOS 窗口, 把目录转换到 Java 源文件所在的目录, 然后输入相应的编译命令

“javac”。在本例中应输入：

```
C:\java>javac NumberAdd.java
```

注意文件名称不能写错，同时要注意大小写，而且要带上扩展名“java”。如果编译成功，将会生成字节码文件 NumberAdd.class，可以在 MS-DOS 窗口中使用“dir”命令查看，如图 1-6 所示。如果程序中有错误出现，则系统会终止编译并给出相应的错误信息；如果系统指出未找到 javac.exe 命令，则说明 path 环境变量设置不正确。

```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Administrator>cd c:\java
C:\java>javac NumberAdd.java
C:\java>dir
驱动器 c 中的卷没有标签。
卷的序列号是 5CE1-58A9

C:\java 的目录

2007-05-09  18:02    <DIR>          .
2007-05-09  18:02    <DIR>          ..
2007-05-09  18:02                424 NumberAdd.class
2007-05-09  17:23                237 NumberAdd.java
                2 个文件          661 字节
                2 个目录    1,746,432,000 可用字节

C:\java>_
```

图 1-6 编译 Java 程序

4. 运行字节码文件

经过编译之后，生成的字节码文件就可以由 Java 的解释器解释执行，在 MS-DOS 窗口下输入运行 Java 程序命令并加上字节码文件名，就可以运行程序了。运行成功就可以在该窗口中显示一串字符串“1 到 10 的和为：55”。这样第一个程序就运行成功了。

1.3 Java 语言的综合开发工具 Eclipse

Eclipse 是一个基于 Java 的开源的可扩展的集成开发环境（IDE）。Eclipse 最初是 IBM 公司的一个软件产品，2001 年 11 月它的 1.0 版本正式发布，面世之初，曾经受到业界的批评，但是到了 2003 年 3 月，Eclipse 发布了它的 2.1 版，发布后立刻引起了轰动，人们蜂拥下载该软件，以至于导致 Eclipse 的服务器瘫痪。现在，IBM 公司已将这个经过巨额投资的产品作为一个开源项目捐献给了开源组织 eclipse.org，Eclipse 出色而又独特的平台特性受到了各大公司的关注和支持。

Eclipse 是一个通用的工具平台——普遍使用的开放式的扩展的 IDE。它提供了功能丰富的开放环境，允许开发者高效地创建一些能够无缝集成到 Eclipse 平台中的插件。它的体系结构如图 1-7 所示。

Eclipse 的设计思想是一切皆为插件。Eclipse 的核心是非常小的，其他所有的功能都以插件的形式存在，并且可以天衣无缝地附加到 Eclipse 的核心之上。比如用于图形开发的插件（SWT/Jface）、Java 的开发环境插件（JDT）和插件的开发环境（PDF）等。Eclipse 对这些插

件的管理是动态地载入动态调用的，即只有当真正使用这些插件时才调入内存，不用时会在适当的时候移出内存。

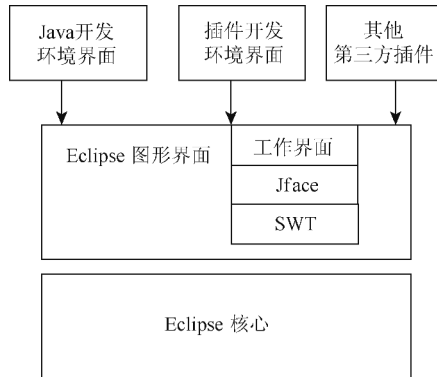


图 1-7 Eclipse 的体系结构

Eclipse 的众多好处使它正在受到更多的关注和使用，本书中将采用这个集成开发工具作为 Java 程序的开发环境。

1.3.1 Eclipse 的安装和配置

Eclipse 是开放源代码的项目，读者可以从 <http://www.eclipse.org/downloads/index.php> 免费下载最新版本，本书使用 eclipse-SDK-3.2.1，它包含了 Eclipse 开发环境、Java 开发环境、Plug-in 开发环境，以及所有源代码和文档。

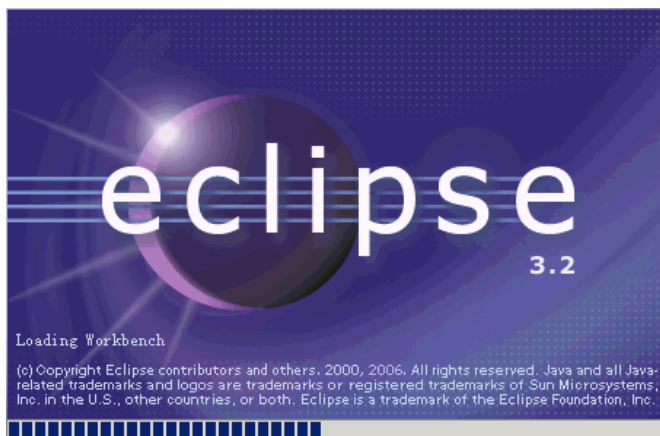


图 1-8 Eclipse 的启动界面

安装 Eclipse 的步骤非常简单，只需将下载的压缩包 eclipse-SDK-3.2.1-win32.zip 按原路径直接解压即可。在解压缩之后到相应的安装路径（可以创建一个快捷方式，或者定义它的路径）去找 eclipse.exe 文件双击运行启动 Eclipse，即可启动 Eclipse。屏幕将显示如图 1-8 所示的启动界面，闪现一个很酷的月蚀（Eclipse）图片。

然后可以设置个人的工作空间，以后你所作的 Java 工程和应用程序可以放置在这个目录下，如图 1-9 所示。

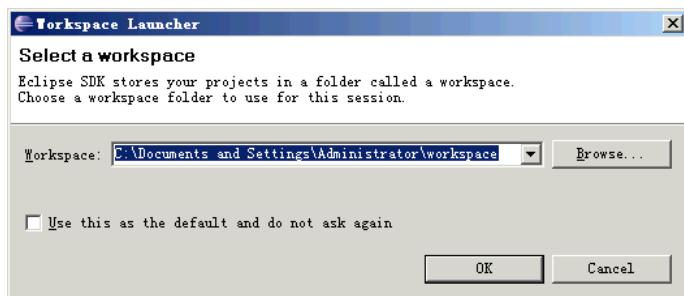


图 1-9 设置个人的工作环境

Eclipse 开发环境的基本构成包括编辑器窗口、大纲窗口、任务窗口和包资源管理器窗口。如图 1-10 所示为 Eclipse 启动时的默认视图——Java 透视图。

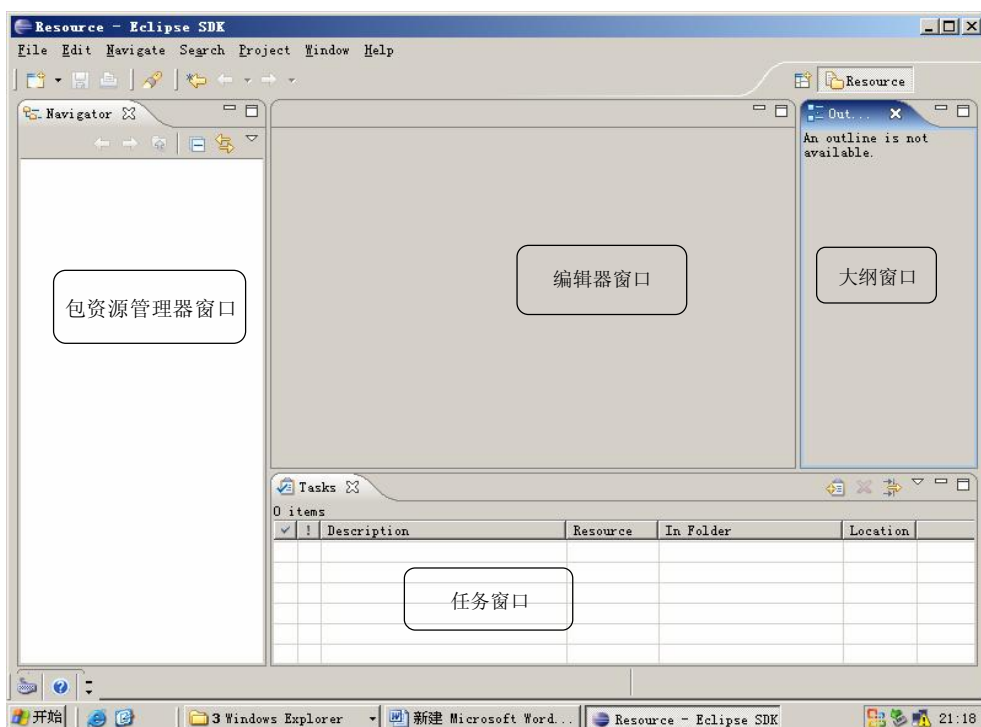


图 1-10 Eclipse 开发环境的基本构成

(1) 包资源管理器窗口。该窗口中默认显示包资源管理器，是使用 Eclipse 建立项目和工作开始的地方，它显示 workspace 中的级次显示和所有的项目。

(2) 编辑器窗口。该窗口是进行代码编写和调试的核心区域，在该区域中开发人员可以方便地进行代码的编写和查看，并且可以随时获得 Eclipse 的提示和帮助。

(3) 大纲窗口。在大纲窗口中可以层次清晰地反映程序源代码的基本结构，在大代码量的程序编写调试时，对于代码的迅速定位会非常有帮助。

(4) 任务窗口。在开发过程中查看错误提示、程序运行过程中的相关控制台输出等信息的显示。

Eclipse 的主菜单显示在工作台窗口的顶端，在标题栏的下方。你可以使用主菜单和其子菜单调用 Eclipse 的大部分功能。例如，如果 `HelloWorld.java` 文件正在被编辑，你可以选择【文件】|【保存】，来保存 `HelloWorld.java` 文件。

在主菜单下面的工具栏是主工具栏，它包含一些普通功能的快捷方式按钮。比如，上面有一个软盘图标的按钮，它的功能就是保存目前正在编辑的文件。当将鼠标移到图标的上面的时候，它将显示这个按钮的提示标签，提示这个按钮的功能。

在屏幕左侧的另外一个工具条是快捷方式工具条，这里的按钮提供了有一个快捷的方式用来新建一个透视图，或者在透视图之间进行切换。最顶端的按钮是打开一个透视图，相当于“窗口”→“打开透视图”的菜单功能。在其下方是资源透视图快捷方式。

1.3.2 使用 Eclipse 建立 Java 程序

在集成开发环境中的任何一个 Java 程序，都必须存在一个 Java 项目中，Java 项目包括用于构建 Java 程序的源代码和其他资料。下面以在屏幕上输出“Hello World”应用程序为例，讲述在 Eclipse 中创建 Java 应用的步骤：

(1) 新建一个空的 Java 项目。选择主菜单【File】|【New】|【Project】。在弹出的窗口中选择【Java】|【Java Project】选项，如图 1-11 所示。

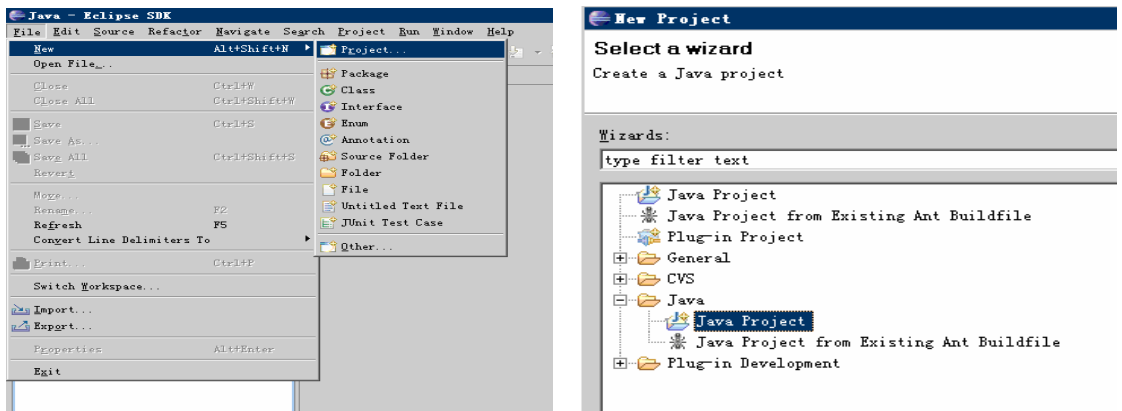


图 1-11 创建 Java 项目

单击“Next”按钮，出现如图 1-12 所示的窗口。在【Project name】文本框中输入项目名称，比如 `MyProject`，Eclipse 会自动地在工作目录中建立同名的目录。在项目布局窗口【Project Layout】中选择“Use project folder as a root for sources and class files”，这样项目中就会创建两个不同的目录 `src` 和 `bin`，分别存放源文件和类文件。

单击【Finish】按钮，就会发现在导航器视图中出现一个新的项目，该项目包含两个空目录 `src` 和 `bin`。

(2) 新建一个 Java 类。在包资源管理器中选择要添加程序的 Java 项目，并单击鼠标右键，在弹出的菜单中选择【New】|【Class】，进入建立 Java 类向导。

在如图 1-13 所示的窗口创建 Java 类中输入类名 `FirstClass`，包名 `net.nyist.java.java2`，并选中“public static void main (String args[])”来自动创建 `main` 方法。

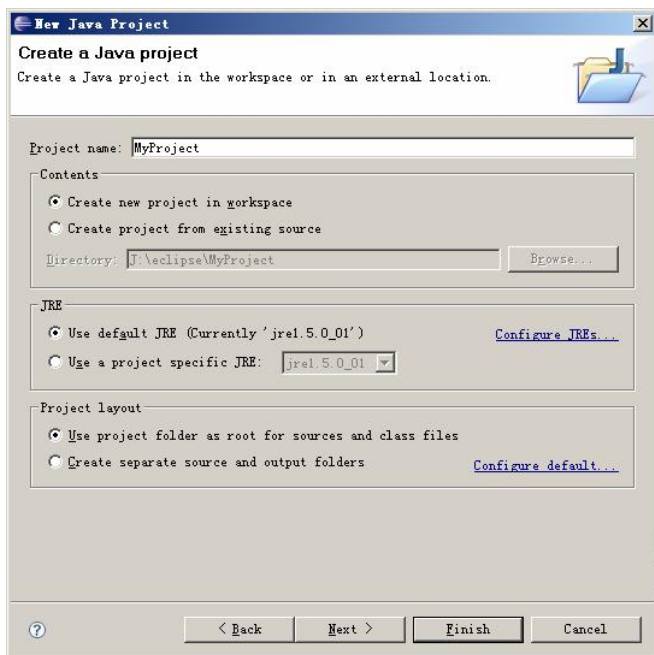


图 1-12 设置 Java 项目选项

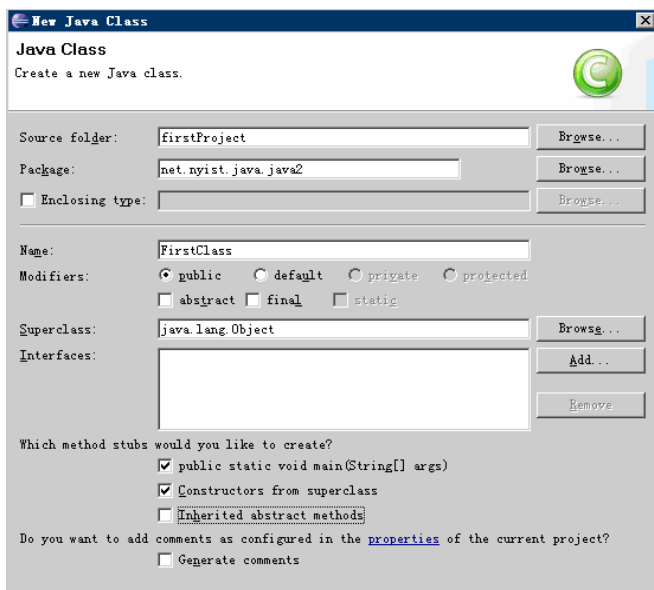


图 1-13 设置类名及包名

单击“Finish”按钮，一个代码编辑器被打开，同时可以在左边 src 看到几个和包名一致的子目录，bin 目录自动和 src 目录保持同步。在编辑器中可以看到很多注释，程序员可以自己修改和添加这些注释。

在编辑器中加入一行代码“System.out.println(“Hello World!”)”。在输入的过程中 Eclipse 可以自动地提示帮助，如图 1-14 所示，程序员可以作为参考。