

■ 高职高专计算机规划教材

Java 教程

郑阿奇 主编



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

高职高专计算机规划教材

Java 教程

郑阿奇 主 编

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本教程以 Java 最新的版本 Java SE Development Kit (JDK) 6 Update 10 为基础, 内容循序渐进、深入浅出, 精心设计每一个示例, 结构安排合理, 使读者准确把握 Java 的知识点。

本教程包括 Java 基础、习题、实验和习题答案四部分。本教程在讲解内容后紧跟实例, 每章的最后还配备了一个综合实例, 对已经学习的主要知识进行综合应用。实验部分通过实例引导读者进行学习, 并提出思考和练习。实例程序均通过上机调试。全书以开源软件 Eclipse 作为 Java 的集成开发环境, 使得编写、调试、运行 Java 程序变得更为简便。

本书专为高职高专设计, 可作为高职高专 Java 语言课程的教材, 也可作为 Java 自学者或者应用开发者的参考书。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目(CIP)数据

Java 教程 / 郑阿奇主编. —北京: 电子工业出版社, 2009.9
高职高专计算机规划教材
ISBN 978-7-121-09547-4

I. J... II. 郑... III. Java 语言—程序设计—高等学校: 技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2009) 第 168267 号

策划编辑: 赵云峰

责任编辑: 刘真平

印 刷: 涿州市京南印刷厂

装 订: 涿州市桃园装订有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 21.75 字数: 556 千字

印 次: 2009 年 9 月第 1 次印刷

印 数: 4 000 册 定价: 32.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

Java 是 Sun 公司开发的新一代编程语言，因其具有简单、面向对象、多线程、跨平台等特性而深受人们的欢迎。当前学习 Java 不仅是一种时尚，而且是一种潮流。

Java 教程以 Java 最新的版本 Java SE Development Kit (JDK) 6 Update 10 为基础，内容循序渐进、深入浅出，精心设计每一个示例，结构安排更为合理，使读者准确把握 Java 的知识点。实用教程一般在讲解一项内容后紧跟相关的实例演示。每章的最后还配备了一个综合实例，使学习者综合应用已经学过的主要知识。实验部分也是通过实例先引导读者进行学习，再提出思考练习。实例程序均通过上机调试，运行结果采用屏幕截图，避免代码与结果不一致的错误。全书以开源软件 Eclipse 作为 Java 的集成开发环境，这使得编写、调试、运行 Java 程序变得更为简便。

实际上，本教程不仅适合于教学，也非常适合于 Java 的各类培训和用 Eclipse 编程开发应用程序的用户学习和参考。只要阅读本书，结合上机操作进行练习，就能在较短的时间内基本掌握 Java 知识及其应用技术。

与本书配套的同步 PowerPoint 课件，可直接用于课堂教学。书中的源代码和 PowerPoint 课件，可从 <http://www.hxedu.com.cn> 或 <http://www.huaxin.edu.cn> 网站上免费下载。

本书专为高职高专设计，一些同志参加本书的基础工作，最后由南京师范大学郑阿奇统稿、定稿。

由于作者水平有限，不当之处在所难免，恳请读者批评指正。

意见建议邮箱：easybooks@163com。

编 者
2009年6月

目 录

第 1 部分 Java 实用教程

第 1 章 Java 和 Eclipse 集成开发环境	1
1.1 Java 语言简介	1
1.2 第一个 Java 程序	2
1.3 Java 开发环境的搭建	3
1.4 Eclipse 集成开发环境	6
第 2 章 Java 语法基础	9
2.1 常量和变量	9
2.1.1 数据类型	9
2.1.2 标识符和关键字	9
2.1.3 常量	10
2.1.4 变量	12
2.1.5 类型转换	13
2.2 运算符和表达式	14
2.2.1 运算符	14
2.2.2 表达式	19
2.3 流程控制	19
2.3.1 分支语句	19
2.3.2 循环语句	21
2.3.3 流程跳转语句	25
2.4 方法与递归	26
2.4.1 方法	26
2.4.2 参数传递	26
2.4.3 递归	27
2.5 命名规范	28
2.6 注释语句	29
2.7 综合实例	29
第 3 章 Java 面向对象编程（上）	32
3.1 类的定义及成员变量初始化	32
3.1.1 类的定义	33
3.1.2 变量的初始化	33
3.2 创建对象	33
3.2.1 构造方法	33
3.2.2 默认构造方法	35
3.2.3 构造方法重载	36
3.2.4 普通方法重载	37
3.3 this 引用	38

3.4	静态成员	39
3.4.1	static 关键字	39
3.4.2	main()方法	41
3.4.3	类的初始化	42
3.5	package 与 import 语句	43
3.6	内部类	47
3.6.1	实例内部类	47
3.6.2	匿名类	49
第 4 章	Java 面向对象编程 (下)	51
4.1	继承	51
4.1.1	继承的定义	51
4.1.2	初始化基类	52
4.1.3	方法的重写	53
4.1.4	super 关键字	56
4.2	对象的转型	56
4.3	多态	59
4.4	抽象类	61
4.5	接口	63
4.6	final 关键字	64
4.7	权限修饰符	65
4.7.1	类内部	66
4.7.2	同一个包的类	67
4.7.3	不同包的子类	68
4.7.4	通用性	69
4.8	综合实例: 航班管理	69
第 5 章	常用类与异常处理	78
5.1	Object 类	78
5.1.1	equals()方法	78
5.1.2	hashCode()方法	80
5.1.3	toString()方法	80
5.2	字符串类	81
5.2.1	String 类	81
5.2.2	StringBuffer 类	83
5.3	包装类	85
5.4	Math 类	87
5.5	日期类	88
5.6	正则表达式	90
5.6.1	替换操作	95
5.6.2	Scanner 定义符	95
5.7	异常处理	96

571	5.7.1	异常的捕获与处理	97
121	5.7.2	声明抛出异常子句	99
121	5.7.3	抛出异常语句	100
121	5.7.4	自定义异常类	101
121	5.8	综合实例	103
第6章 数组与枚举			105
101	6.1	一维数组	105
101	6.2	多维数组	108
101	6.3	访问数组	111
101	6.4	数组实用类:Arrays	113
101	6.4.1	复制数组	113
101	6.4.2	数组排序	114
101	6.4.3	数组元素的查找	115
101	6.5	枚举	116
101	6.6	enum 的构造方法	117
101	6.7	综合实例	118
第7章 容器和泛型			121
121	7.1	Collection 与 Iterator	121
121	7.2	实用类 Collections	123
121	7.3	Set (集)	124
121	7.3.1	HashSet	124
121	7.3.2	TreeSet	126
121	7.4	List (列表)	129
121	7.4.1	ArrayList	129
121	7.4.2	LinkedList	130
121	7.5	Map (映射)	131
121	7.5.1	HashMap	132
121	7.5.2	TreeMap	133
121	7.6	泛型	135
121	7.7	通配符与受限通配符	137
121	7.8	综合实例	139
第8章 Java 输入/输出系统			142
142	8.1	字节流	142
142	8.1.1	文件输入流	143
142	8.1.2	文件输出流	144
142	8.2	过滤流	146
142	8.2.1	缓冲流类	146
142	8.2.2	数据流类	148
142	8.2.3	PrintStream 类	150
142	8.3	字符流	151

8.3.1	FileReader 和 FileWriter	152
8.3.2	BufferedReader 和 BufferedWriter	153
8.4	标准 I/O	154
8.5	File 类	156
8.6	综合实例	157
第 9 章	AWT 组件及应用	160
9.1	AWT 容器	160
9.1.1	Window 和 Frame	160
9.1.2	Panel	161
9.2	布局管理器	162
9.3	事件处理机制	164
9.3.1	AWT 事件与监听器	164
9.3.2	窗口事件	166
9.3.3	内部类实现监听接口	166
9.3.4	类自身实现监听接口	167
9.3.5	外部类实现监听接口	169
9.3.6	采用事件适配器	171
9.4	综合实例	172
第 10 章	Swing 组件及应用	177
10.1	窗口 JFrame	177
10.2	按钮	178
10.3	JTextField 与 JTextArea	180
10.4	JCheckBox 和 JRadioButton	183
10.5	菜单条 (JMenuBar)	186
10.6	弹出式菜单 (JPopupMenu)	189
10.7	综合实例	191
第 11 章	并发	195
11.1	线程的创建与启动	195
11.1.1	继承 java.lang.Thread 类	195
11.1.2	实现 Runnable 接口	197
11.2	线程的状态转换	198
11.3	线程调度	200
11.3.1	调整各个线程的优先级	200
11.3.2	线程让步	201
11.3.3	合并线程	202
11.4	后台线程	203
11.5	综合实例	204
第 12 章	综合实例	207
12.1	设计思路	207
12.2	汉诺塔上盘子模拟	207

12.3	汉诺塔上对象的定位及盘子的存放	208
12.4	创建汉诺塔及实现手动搬运盘子	209
12.5	自动搬运盘子	216
第 13 章	Java 网络编程	222
13.1	网络程序设计基础	222
13.1.1	TCP 和 UDP	222
13.1.2	端口和套接字	222
13.2	TCP 网络编程	223
13.2.1	InetAddress 类	225
13.2.2	TCP 通信程序	227
13.3	UDP 网络编程	230
13.3.1	UDP 通信程序	232
13.3.2	组播	234
13.4	URL	236
13.5	综合实例	238
第 14 章	JDBC 编程	243
14.1	SQL 语言	243
14.2	JDBC	245
14.3	MySQL 数据库	248
14.3.1	MySQL 服务器的安装	248
14.3.2	MySQL 服务器的配置	248
14.3.3	MySQL 的环境	250
14.4	访问数据库	251
14.4.1	加载并注册数据库驱动	252
14.4.2	建立到数据库的连接	253
14.4.3	访问数据库	254
14.5	JDBC 编程	256
14.6	批处理	261
14.7	事务处理	262
14.8	综合实例	264

第 2 部分 习题集

第 1 章	Java 和 Eclipse 集成开发环境	267
第 2 章	Java 语法基础	267
第 3 章	Java 面向对象编程 (上)	267
第 4 章	Java 面向对象编程 (下)	268
第 5 章	常用类与异常处理	268
第 6 章	数组与枚举	268
第 7 章	容器和泛型	268
第 8 章	Java 输入/输出系统	269
第 9 章	AWT 组件及应用	269

第 10 章 Swing 组件及应用	269
第 11 章 并发	270
第 13 章 Java 网络编程	270
第 14 章 JDBC 编程	270
第 3 部分 实 验	
实验 1 Java 和 Eclipse 集成开发环境	271
实验目的	271
实验准备	271
实验内容	271
思考与练习题	273
实验 2 Java 语法基础	273
实验目的	273
实验准备	273
实验内容	273
思考与练习题	276
实验 3 Java 面向对象编程（上）	276
实验目的	276
实验准备	276
实验内容	277
思考与练习题	278
实验 4 Java 面向对象编程（下）	279
实验目的	279
实验准备	279
实验内容	279
思考与练习题	281
实验 5 常用类与异常处理	281
实验目的	281
实验准备	281
实验内容	282
思考与练习题	284
实验 6 数组与枚举	284
实验目的	284
实验准备	284
实验内容	284
思考与练习题	287
实验 7 容器和泛型	287
实验目的	287
实验准备	287
实验内容	287
思考与练习题	290

实验 8 Java 输入/输出系统	290
实验目的	290
实验准备	290
实验内容	290
思考与练习题	293
实验 9 AWT 组件及应用	293
实验目的	293
实验准备	293
实验内容	294
思考与练习题	298
实验 10 Swing 组件及应用	299
实验目的	299
实验准备	299
实验内容	299
思考与练习题	303
实验 11 并发	303
实验目的	303
实验准备	303
实验内容	303
思考与练习题	306
实验 12 综合实例	306
实验目的	306
实验准备	306
实验内容	306
思考与练习题	306
实验 13 Java 网络编程	306
实验目的	306
实验准备	307
实验内容	307
思考与练习题	310
实验 14 JDBC 编程	310
实验目的	310
实验准备	310
实验内容	310
思考与练习题	314
第 4 部分 习题答案	
第 1 章 习题答案	315
第 2 章 习题答案	315
第 3 章 习题答案	317
第 4 章 习题答案	318

第 5 章 习题答案	320
第 6 章 习题答案	320
第 7 章 习题答案	323
第 8 章 习题答案	324
第 9 章 习题答案	325
第 10 章 习题答案	328
第 11 章 习题答案	331
第 13 章 习题答案	331
第 14 章 习题答案	333

第 1 部分 Java 实用教程

第 1 章 Java 和 Eclipse 集成开发环境

1.1 Java 语言简介

1991 年, Sun 公司开发了新一代编程语言 Java, 初衷是为家用消费类电子产品开发一个分布式代码系统。为了使整个系统与平台无关, 采用了虚拟机器码 (Virtual Machine Code) 方式, 虚拟机运行在一个解释器上, 每一个操作系统均有一个解释器。1995 年 3 月发布了 Java 的 Alpha1.0a2 版本, 1996 年 1 月发布了 Java 的第一个开发包 JDK v1.0, 1997 年 2 月发布了 Java 语言的开发包 JDK v1.1, 从此奠定了 Java 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, 这是 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 发展方向, 主要有 3 个成员:

- J2ME——Java 2 Micro Edition, 用于嵌入式应用的 Java 2 平台。
- J2SE——Java 2 Standard Edition, 用于工作站、PC 的 Java 2 标准平台。
- J2EE——Java 2 Enterprise Edition, 可扩展的企业级应用的 Java 2 平台。

2004 年, J2SE1.5 发布, 为了表示这个版本的重要性, J2SE1.5 更名为 J2SE5.0。2005 年, JavaOne 大会召开, Sun 公司公开 JavaSE6, 此时, Java 的各种版本被更名, 取消其中的数字“2”, J2SE 更名为 Java SE, J2EE 更名为 Java EE, J2ME 更名为 Java ME。

Java 是一个广泛使用的网络编程语言, 它简单, 面向对象, 不依赖于机器的结构, 不受 CPU 和环境的限制, 具有可移植性、安全性, 并且提供了并发的机制, 具有很高的性能。此外, Java 还提供了丰富的类库, 使程序设计人员可以方便地建立自己的系统。

Java 有两种核心机制: 一种是 Java 虚拟机 (Java Virtual Machine), 另一种是垃圾收集机制 (Garbage Collection)。

1. Java 虚拟机

Java 程序是如何做到“一次编译, 到处运行”的呢? 这正是通过 Java 虚拟机来实现的。JVM 可以理解成一个以字节码为机器指令的 CPU。首先, Java 编译程序将后缀名为 .java 的 Java 源程序编译为 JVM 可执行的代码, 即后缀名为 .class 的 Java 字节码文件, 如图 1.1 所示。运行 JVM 字节码的工作是由解释器来完成的。解释执行过程分代码的装入、代码的校验和代码的执行 3 步进行。装入代码的工作由“类装载机”完成, 类装载机负责装入一个程序运行需要的所有代码。字节码校验器负责代码的校验。每种类型的操作系统都有一种对应的 Java 虚拟机, Java 虚拟机屏蔽了底层操作系统的差异。所以 Java 程序能够做到“一次编译, 到处运行”。

2. 垃圾收集机制

垃圾回收器能够自动回收垃圾，即无用的对象所占据的内存空间被回收。在 C/C++ 语言中，这些工作由程序员负责，无疑增加了程序员的负担。而 Java 语言消除了程序员回收垃圾的责任：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 的空闲时，检查并释放那些可被释放的存储空间。在 Java 中，对象被创建后，就会在堆区中分配一块内存。当对象不再被程序引用时，它就变成一个垃圾，所占用的堆空间可以被回收，以便空间被后续的新对象所使用。Java 的垃圾回收器能断定哪些对象不再被引用，并且能够把它们所占据的堆空间释放出来。

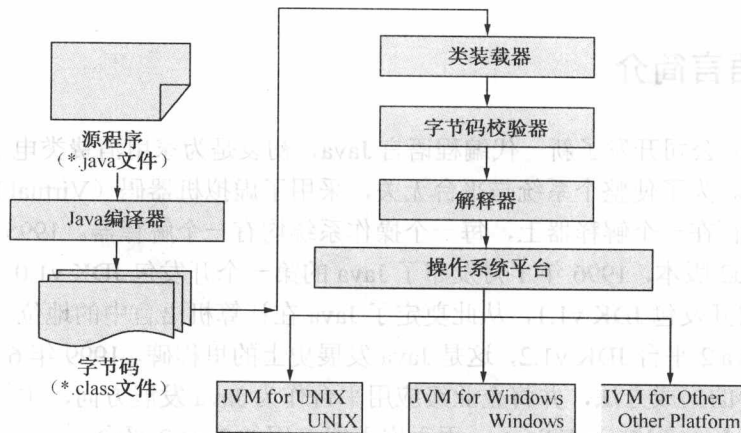


图 1.1 Java 程序执行流程

1.2 第一个 Java 程序

下面先看一个 Java 小程序，对 Java 编程有一个初步的认识。

【例 1.1】 求圆面积。

用文本编辑器（如 Windows 的记事本）编辑下列程序，文件名为 Area.java。

Area.java

```
/** 求圆的面积 */
public class Area {
    double pi = 3.1415; // 定义变量 pi
    public static void main(String[] args) {
        double r, area;
        r = 3;
        area = pi * r * r; // 求圆的面积
        System.out.println("圆的面积="+area); // 在屏幕上显示结果
    }
}
```

下面对程序进行简要说明。

(1) /**到*/之间的内容为注释。

(2) 保留字 `class` 声明一个类，其类名为 `Area`，保留字 `public` 表示它是一个公共类。类定义由花括号 `{}` 括起来。当编写一个 Java 源代码文件时，此文件通常被称为编译单元。每个编译单元都必须有一个后缀名 `.java`，而在编译单元内则可以有一个 `public` 类，该类的名称必须与文件的名称相同（包括大小写）。每个编译单元最多只能有一个 `public` 类，也可以没有，否则编译器就不会接受。

(3) 语句 “`double pi = 3.1415;`” 表示定义实型变量 `pi` 的值为 3.1415。

(4) 在该类中定义了一个 `main()` 方法，其中 `public` 表示访问权限，指明所有的类都可以使用这一方法；`static` 指明该方法是一个类方法，它可以通过类名直接调用；`void` 则指明 `main()` 方法不返回任何值。对于一个应用程序来说，`main()` 方法是必需的，而且必须按照如上的格式来定义。Java 解释器在没有生成任何实例的情况下，以 `main()` 方法作为入口来执行程序。Java 程序中可以定义多个类，每个类中可以定义多个方法，但是最多只能有一个公共类，`main()` 方法也只能有一个。

1.3 Java 开发环境的搭建

要想编译和运行上面的程序，离不开 Java 的编译和运行环境。Sun 公司提供了自己的一套 Java 开发环境，通常称为 JDK (Java Development Kit)，又称为 J2SDK。目前最新的版本是 Java SE Development Kit (JDK) 6 Update 10，可以到 Sun 公司的网站下载。输入网址 <http://java.sun.com/javase/downloads/index.jsp>，进入第二栏，单击“Download”按钮，如图 1.2 所示。

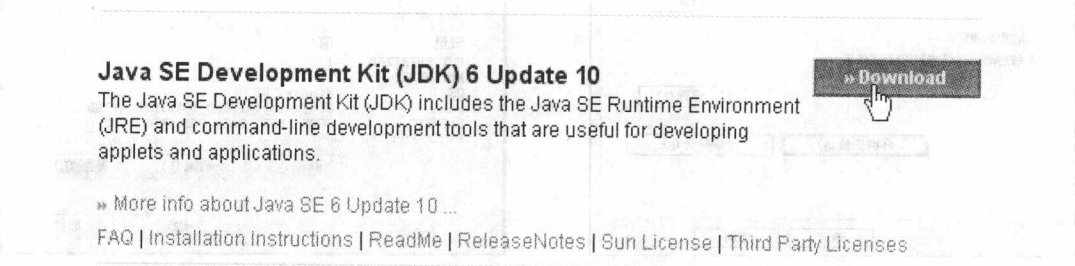


图 1.2 选择操作系统

在“Platform”栏中选择 Windows，如果自己的平台是 Linux 操作系统，请选择 Linux，单击“Continue”按钮。之后进入下一屏，选中“Windows offline installation”单选按钮，单击“jdk-6u10-windows-i586-p.exe”项，JDK 就可以开始下载了。

下载完成后，双击可执行文件 `jdk-6u10-windows-i586-p.exe`，按照提示完成安装。这里 JDK 的安装路径改为“`C:\Java\jdk1.6.0_10\`”，如图 1.3 所示。

在 JDK 的安装包里带有 JRE 安装包，JRE 是 Java 运行时的环境。JRE 的安装路径改为“`C:\Java\jre6\`”，如图 1.4 所示。

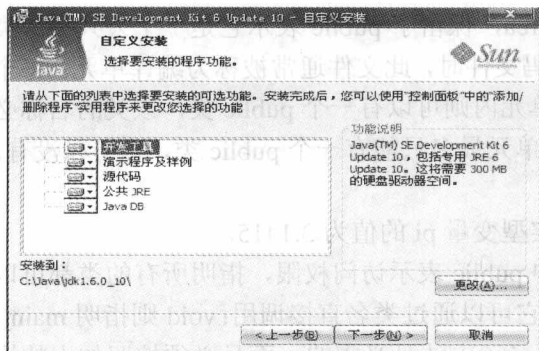


图 1.3 JDK 的安装

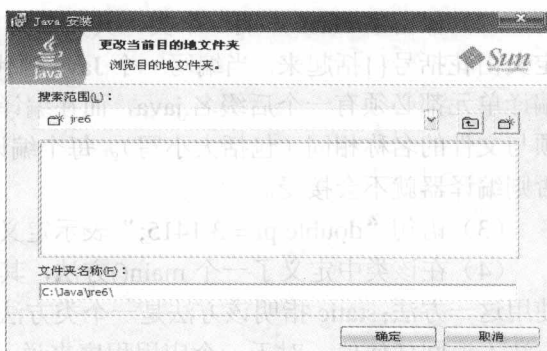


图 1.4 JRE 的安装

安装完成之后，还需要配置环境变量。在 Windows 系统中，在桌面上右击“我的电脑”，从打开的快捷菜单中选择“属性”命令，在打开的“系统属性”对话框中单击“高级”选项卡，如图 1.5 所示，单击“环境变量”按钮，打开“环境变量”对话框，如图 1.6 所示。

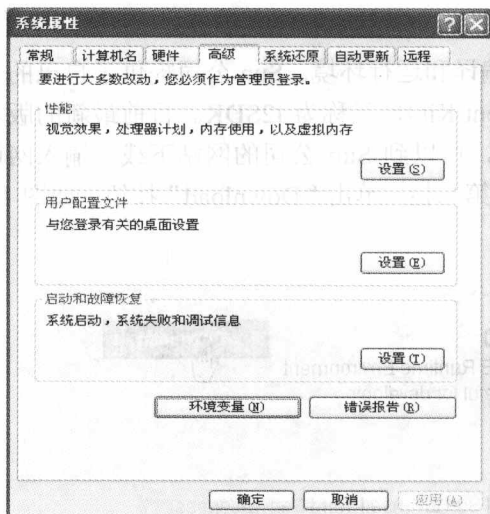


图 1.5 “系统属性”对话框

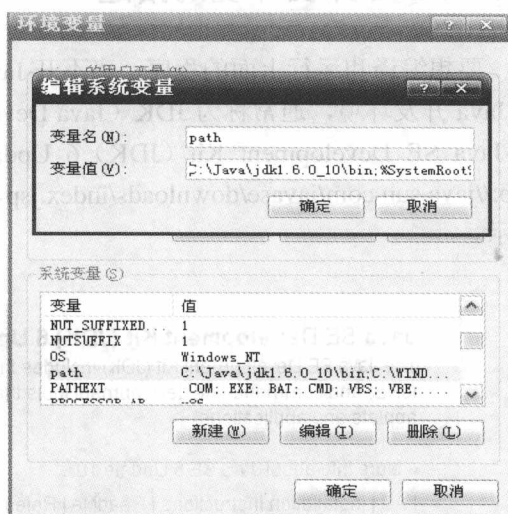


图 1.6 “环境变量”对话框

新建环境变量：在系统变量中双击“path”变量，在变量值栏内填入“C:\Java\jdk1.6.0_10\bin;”，在系统变量中再双击“CLASSPATH”变量，输入值为“.”，表示当前路径。如果变量值栏内还没有“CLASSPATH”变量，请新建“CLASSPATH”变量。最后，单击“确定”按钮，JDK 就配置完成并可以使用了。

下载的 JDK 开发工具包并没有包含 JDK API 文档，API 文档中提供了 JDK 中的类的完整使用说明，应下载下来以备随时查阅。API 文档的首页如图 1.7 所示。

JDK	Java Language	Java Language								
	Tools & Tool APIs	java	javac	javadoc	apt	jar	javap	JPDA	JConsole	Java VisualVM
		Security	Int'l	RMI	IDL	Deploy	Monitoring	Troubleshoot	Scripting	JVM TI
	Java Web (Deployment)	Java Web App Development/Distribution					Java Web Start		Applet(Plug-In)	
	User Interface Toolkits	AWT			Swing			Java 2D		
		Accessibility	Drag n Drop	Input Methods		Image I/O	Print Service	Sound		
	Integration Libraries	IDL	JDBC _{TM}	JNDI _{TM}	RMI	RMI-IIOP		Scripting		
	Other Base Libraries	Beans	Intl Support		I/O	JMX	JNI		Math	
		Networking	Override Mechanism	Security	Serialization	Extension Mechanism	XML JAXP			
	Lang and util Base Libraries	lang and util	Collections	Concurrency Utilities		JAR	Logging	Management		
		Preferences API	Ref Objects	Reflection		Regular Expressions	Versioning	Zip	Instrument	
	Java Virtual Machine	Java Hotspot _{TM} Client VM				Java Hotspot _{TM} Server VM				
	Platforms	Solaris _{TM}		Linux		Windows		Other		

图 1.7 API 文档的首页

进入 JDK 的安装路径，将看到如表 1.1 所示的目录。

表 1.1 JDK 的常用目录结构

目 录	描 述
src 子目录	存放 Java 源文件
bin 子目录	存放 JDK 的工具程序
lib 子目录	存放 JAR 文件
demo 子目录	存放 Java 示范程序

JDK 主要包括以下内容。

- (1) Java 虚拟机：负责解释和执行 Java 程序，Java 虚拟机可以运行在各种操作系统平台上。
- (2) JDK 类库：提供了最基础的 Java 类库，即各种实用类，包括最常用的类库，如 java.lang、java.io、java.util、javax.swing 和 java.sql 等。
- (3) 开发工具：这些开发工具都是可执行程序，主要包括 javac.exe（编译工具）、java.exe（运行工具）、javadoc.exe（生成 Javadoc 文档的工具）和 jar.exe（打包工具）等。

Java 的环境变量已配置好，现在就可以编写 Java 程序了。用记事本就可以编写 Java 程序。首先建立一个名为“Area.java”的源文件，存放在 d 盘根目录中。

打开命令提示符，进入 d 盘根目录，输入“javac Area.java”命令，这时在 d 盘根目录下出现一个“Area.class”文件，输入“java Area”命令。在控制台上显示出“圆的面积=28.2735”，如图 1.8 所示。javac 命令把 Java 源文件编译成“.class”二进制文件，java 命令解释执行“.class”文件。