



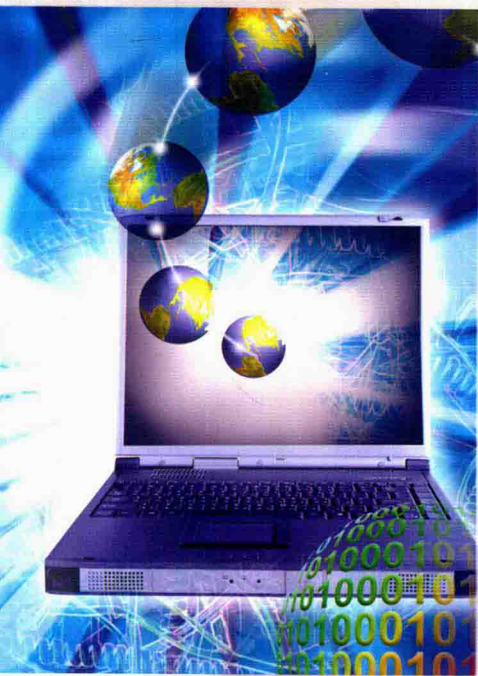
软件职业技术学院“十二五”规划教材

Java高级程序设计

(第二版)

主 编 肖 英 刘 洁

副主编 李文蕙 刘 嵩 李 唯



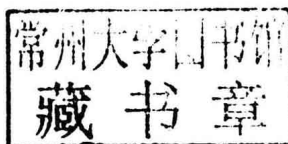
中国水利水电出版社
www.waterpub.com.cn

软件职业技术学院“十二五”规划教材

Java 高级程序设计（第二版）

主 编 肖 英 刘 洁

副主编 李文蕙 刘 嵩 李 唯



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

Java 语言具有简单、面向对象、分布式、健壮性、安全性、可移植性、多线程、高性能等诸多优势特点,也是近十年来计算机编程语言排行榜上的佼佼者,可以用于开发各种领域的应用软件。熟练掌握 Java 语言是软件从业人员的必备技能。

本书共 9 章内容,通过实例讲解了如何使用 Java 语言开发数据库应用程序、多媒体应用程序、网络应用程序等。这些程序典型简洁,主要功能突出,所涉及的技术可以解决同类问题。

本书实例丰富,内容充实,各章节都提供了大量的实例帮助读者学习;由浅入深,逐步讲解,实践内容步骤详细,符合高职学生的学习特点;语言通俗易懂,有助于初学者理解学习。

本书配有全部实例的源代码和电子课件,读者可以从中国水利水电出版社网站和万水书苑免费下载,网址为: <http://www.waterpub.com.cn/softdown/>和 <http://www.wsbookshow.com>。

图书在版编目(CIP)数据

Java 高级程序设计 / 肖英, 刘洁主编. — 2 版. —
北京: 中国水利水电出版社, 2015. 3
软件职业技术学院“十二五”规划教材
ISBN 978-7-5170-2891-8

I. ①J… II. ①肖… ②刘… III. ①JAVA 语言—程序
设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2015)第020905号

、策划编辑: 杨庆川

责任编辑: 李 炎

封面设计: 李 佳

书 名	软件职业技术学院“十二五”规划教材 Java 高级程序设计(第二版)
作 者	主 编 肖 英 刘 洁 副主编 李文蕙 刘 嵩 李 唯
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路 1 号 D 座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 68367658 (发行部)、82562819 (万水)
经 售	北京科水图书销售中心(零售) 电话: (010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	三河市铭浩彩色印装有限公司
规 格	184mm×260mm 16 开本 16.5 印张 406 千字
版 次	2006 年 8 月第 1 版 2006 年 8 月第 1 次印刷 2015 年 3 月第 2 版 2015 年 3 月第 1 次印刷
印 数	0001—3000 册
定 价	30.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

前 言

Java 语言具有简单、面向对象、分布式、健壮性、安全性、可移植性、多线程、高性能等诸多优势特点,也是近十年来计算机编程语言排行榜上的佼佼者,可以用于开发各种领域的应用软件。熟练掌握 Java 语言是软件从业人员的必备技能。

本书的目标是帮助广大高职高专学生学习和掌握利用 Java 语言进行应用系统开发的核心知识和技术。本书知识点系统连贯、逻辑性强;重视实际应用,案例丰富;学习模块划分合理,重难点突出,利于组织教学。整本书在内容组织上注意承上启下、循序渐进地介绍 Java 语言的常用知识和技术。在编排上也做了特别的设置,各章节大多首先演示了典型任务模块,读者可以清楚地知道运用本章节的知识和技术可以达到怎样的实践目标,从而可以更目的明确、有的放矢地学习;接下来详细介绍相关的知识和技术引导读者深入学习和实践,最后通过实训任务令读者掌握相应的开发技术;最后一章通过一个综合运用的案例将本书介绍的核心知识和技术融会贯通,且提供了详尽的开发指导,可以引导读者一步一步自行完成系统开发,有益于提高读者的学习兴趣和信心,达到更好的学习效果。

本书共包含 9 章内容。第 1 章“Java 入门基础”,带领读者熟悉 Java 面向对象编程的主要知识和技术;第 2 章“异常处理”,帮助读者在今后的应用开发过程中令程序更健壮;第 3 章“Java 常用 API”,介绍 Java 语言的一些实用工具类和接口,读者在开发中可能经常会用到它们;第 4 章“GUI 程序设计”,引导读者创建有窗体的桌面应用程序;第 5 章“IO 输入输出流”,实现 Java 程序与外部数据的交换;第 6 章“JDBC 数据库编程”,令 Java 程序可以连接和操作数据库,为开发基于数据库的应用系统打下良好的基础;第 7 章“多线程”,介绍在程序中创建和启动多条执行线程;第 8 章“Java 网络编程”,介绍网络环境中客户机/服务器模式网络应用程序的开发;第 9 章“综合应用”,运用 Java 面向对象编程、GUI 程序设计及 JDBC 编程技术等知识与技术进行 Java 数据库应用系统开发,锻炼读者综合运用本书介绍的知识和技术解决问题的能力。

本书的主要特点是:①实例丰富,内容充实,各章节都提供了大量的实例帮助读者学习;②由浅入深,逐步讲解,实践内容步骤详细,符合高职学生的学习特点;③语言通俗易懂,有助于初学者理解学习;④本书配有全部实例的源代码和电子课件,方便读者学习和教师教学。

本书由肖英、刘洁担任主编,李文蕙、刘嵩、李唯担任副主编,杨国勋、库波参与编写,肖英统编全稿。

由于时间仓促,加之编者水平有限,书中不妥或错误之处在所难免,殷切希望广大读者批评指正。同时,恳请读者一旦发现错误,于百忙之中及时与编者联系,以便尽快更正,编者将不胜感激。

编 者

2014 年 12 月

目 录

前言

第1章 Java 入门基础	1	实训任务	41
本章目标	1	第3章 Java 常用 API	42
本章任务	1	本章目标	42
1.1 Java 开发环境的安装与配置	1	本章任务	42
任务1 安装与配置 JDK、Eclipse	1	3.1 API 的概念	42
任务2 Eclipse 中开发 Java Application	4	3.2 字符串处理	44
知识点: Java Application 程序	6	任务1 字符串加密与解密	44
1.2 Java 语言基础	7	知识点: String、StringBuffer	45
任务3 实现四则运算	7	任务2 统计单词及其数量	52
知识点: Java 语法基础	8	知识点: StringTokenizer 类	52
任务4 找出 100 以内的所有素数	12	3.3 数学运算工具 Math	53
知识点: Java 的流程控制语句	13	任务3 常见数学运算	53
1.3 Java 面向对象编程基础	17	知识点: Math 类	54
任务5 定义类、使用对象	17	3.4 日期时间处理	55
知识点: 类与对象	20	任务4 格式化日期	55
任务6 运用继承	21	知识点: Date、DateFormat、Calendar	55
知识点: 包和访问控制符、继承、接口	25	3.5 集合类	57
实训任务	27	任务5 保存和显示学生列表	59
第2章 异常处理	29	知识点: List 接口及实现类	60
本章目标	29	任务6 使用 Set 保存和显示学生集合	64
本章任务	29	知识点: Set 接口及实现类	65
2.1 什么是异常	29	任务7 使用 Map 保存和显示学生集合	67
任务1 识别异常	29	知识点: Map 接口及实现类	68
知识点: 异常的概念、异常处理机制	30	3.6 泛型	70
2.2 处理异常	31	实训任务	72
任务2 处理异常	31	第4章 GUI 程序设计	73
知识点: try-catch-finally 语句处理异常	32	本章目标	73
2.3 自定义异常与 throw 关键字	35	本章任务	73
任务3 自定义百分制分数异常	35	4.1 认识 GUI 程序	73
知识点: 自定义异常、throw 关键字	37	任务1 登录窗体	73
2.4 throws 关键字	38	知识点: Swing 包、创建窗体	75
任务4 使用 throws 关键字	38	任务2 登录验证功能	80
知识点: throws 关键字为方法声明异常	39	知识点: 事件处理机制	82
2.5 Java 的内置异常	39	4.2 容器与布局	90

任务 3 图形用户界面的四则运算·····	90	6.1 JDBC 数据库编程概述·····	156
知识点: JPanel、布局管理器·····	92	任务 1 基于数据库的学生信息检索程序··	156
4.3 GUI 基本元素·····	100	知识点: JDBC 概述、MySQL、连接	
任务 4 文本复制·····	100	MySQL 数据库·····	162
知识点: JTextArea·····	101	6.2 JDBC 基本操作: CRUD·····	168
任务 5 字体选择器·····	102	任务 2 查询联系人·····	169
知识点: 选择型组件·····	105	知识点: 查询操作·····	171
4.4 菜单与工具栏·····	109	任务 3 添、删、改联系人·····	172
任务 6 简单记事本界面·····	109	知识点: 添删改操作、PreparedStatement、	
知识点: 菜单、工具栏·····	112	可滚动结果集·····	175
4.5 对话框·····	117	6.3 JDBC 进阶·····	182
任务 7 确认退出对话框·····	117	任务 4 运用事务·····	182
知识点: JDialog、JOptionPane、		知识点: 事务·····	184
FileChooser·····	119	实训任务·····	184
4.6 表格·····	124	第 7 章 多线程·····	185
任务 8 使用表格·····	124	本章目标·····	185
知识点: JTable·····	128	本章任务·····	185
实训任务·····	131	7.1 线程概述·····	185
第 5 章 IO 输入输出流·····	133	任务 1 创建并启动多个线程·····	185
本章目标·····	133	知识点: 线程概念、线程的创建和启动··	187
本章任务·····	133	7.2 线程的生命周期·····	190
5.1 IO 流的概念·····	133	7.3 线程的优先级·····	193
任务 1 使用预定义流·····	133	7.4 线程同步·····	194
知识点: 流的概念和分类、预定义流··	134	任务 2 模拟多窗口售票·····	194
5.2 文件读写·····	138	知识点: 线程同步·····	196
任务 2 文件类型过滤器·····	138	7.5 线程通信·····	197
知识点: File 类·····	139	实训任务·····	198
任务 3 文件复制·····	141	第 8 章 Java 网络编程·····	200
知识点: 文件字节 IO 流·····	142	本章目标·····	200
任务 4 记事本打开、另存为功能·····	144	本章任务·····	200
知识点: 文件字符 IO 流·····	147	8.1 网络编程基础·····	200
5.3 使用过滤流·····	148	8.1.1 统一资源定位符(URL)·····	200
任务 5 学生信息存至磁盘文件·····	148	8.1.2 Internet 编址与端口号·····	200
知识点: 节点流和过滤流、常用过滤流··	151	8.1.3 客户机/服务器模式·····	202
5.4 使用 RandomAccessFile·····	154	8.1.4 TCP/IP 与 UDP 协议·····	202
实训任务·····	155	8.2 Java 网络编程 API·····	203
第 6 章 JDBC 数据库编程·····	156	任务 1 读取网页文件·····	203
本章目标·····	156	知识点: URL 类、URLConnection 类、	
本章任务·····	156	InetAddress 类·····	204

8.3 Java 编写 TCP 网络程序·····	208	9.2 系统实现及编码·····	228
任务 2 一对一聊天室·····	208	编码规范·····	228
知识点: TCP/IP 套接字·····	212	任务 3 开发公共模块·····	229
任务 3 多人聊天室·····	215	任务 4 开发登录模块·····	234
8.4 Java 编写 UDP 网络程序·····	220	任务 5 开发主界面·····	236
实训任务·····	224	任务 6 开发查找联系人模块·····	240
第 9 章 综合应用·····	225	任务 7 开发添加联系人模块·····	244
本章目标·····	225	任务 8 开发编辑联系人模块·····	248
本章任务·····	225	任务 9 开发删除联系人模块·····	253
9.1 需求分析及系统设计·····	225	开发小结·····	256
系统目标·····	225	实训任务·····	256
任务 1 数据库设计及实施·····	226	参考文献·····	257
任务 2 界面设计与功能模块·····	227		

第 1 章 Java 入门基础



本章目标

- Java Application 开发环境的安装与配置
- Java 语言基础
- Java 面向对象编程基础



本章任务

- 任务 1 安装和配置 JDK、Eclipse
- 任务 2 Eclipse 中开发 Java Application
- 任务 3 实现四则运算
- 任务 4 找出 100 以内的所有素数
- 任务 5 定义类、使用对象
- 任务 6 运用继承

1.1 Java 开发环境的安装与配置

任务 1 安装与配置 JDK、Eclipse

“工欲善其事，必先利其器”，在开始 Java Application 的编程之旅前，需要先准备好开发环境。

Step1: 下载 JDK

登录 Oracle 公司官网，找到 JavaSE 的下载页面（<http://www.oracle.com/technetwork/java/javase/downloads/index.html>）选择“Java SE Downloads”，如图 1-1 所示。在打开的下载中，根据自己的操作系统平台选择合适的 JDK 安装文件，如图 1-2 所示，64 位的 Windows 操作系统可选择文件名为“jdk-8u25-windows-x64.exe”的 JDK 下载安装。JDK 下载完成后，直接双击安装程序，程序会自动解压缩并进行安装，按照向导一步一步进行安装即可，安装过程中需记住安装的路径，在配置环境变量时将会用到这个安装路径。如 JDK 安装的目录为 C:\Program Files\Java\jdk1.8.0_25，JRE 安装的目录为 C:\Program Files\Java\jre1.8.0_25。

Step2: 配置 JDK

在 Windows 系统下，环境变量的配置过程如下：

（1）打开【控制面板】，双击【系统】，选择【高级系统设置】，在打开的对话框中选择【高级】选项卡，然后单击【环境变量】按钮。

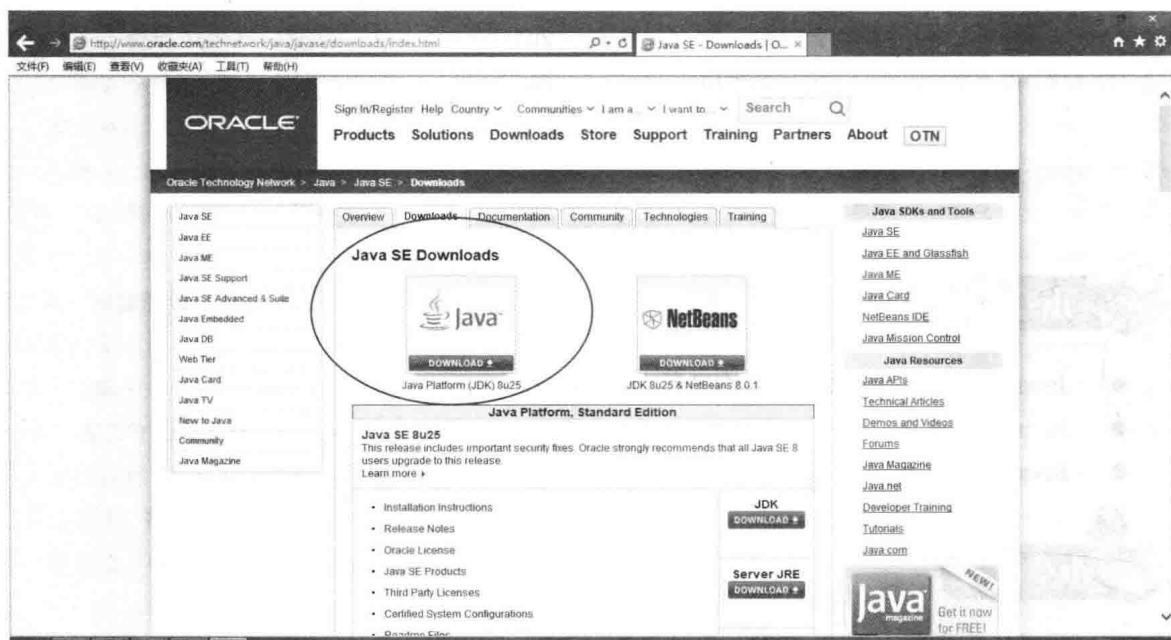


图 1-1 Java SE 下载

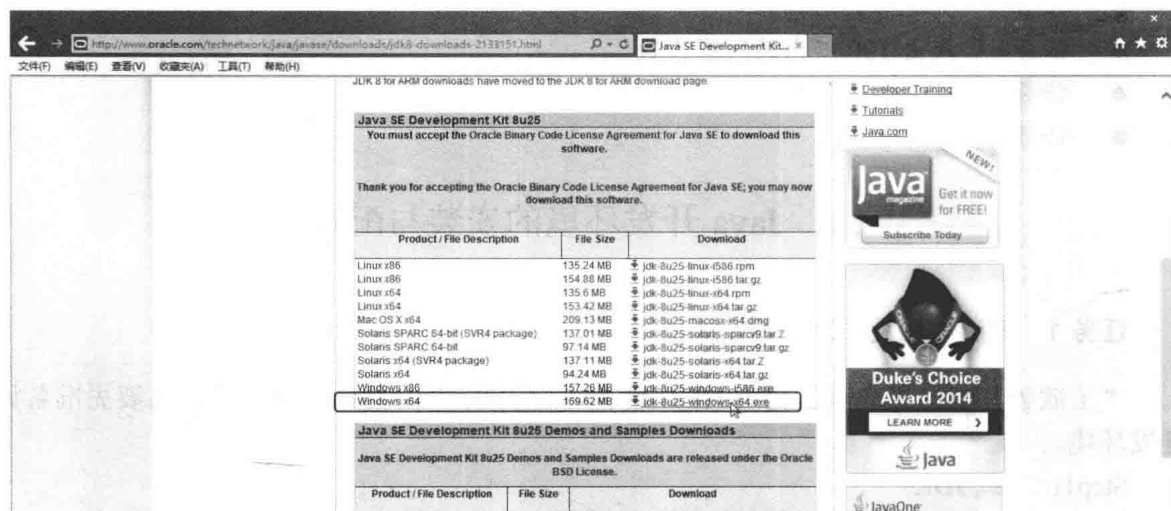


图 1-2 选择适当的 JDK 安装文件

(2) 在【环境变量】的【系统变量】中新建一个名字为“java_home”的变量，配置其值为“C:\Program Files\Java\jdk1.8.0_25”，即 JDK 所在的路径，创建了该变量后，别的变量就可以使用 %java_home% 来引用它的值。

(3) 在【系统变量】中找到变量“classpath”（若无 classpath，则新建之），配置其值为“.;%java_home%\lib\dt.jar;%java_home%\lib\tools.jar”。注意：变量 classpath 中的第一个值是英文句号“.”，表示当前目录。

(4) 在【系统变量】中找到变量“path”（若无 path，则新建之），在 path 变量值的最后添加“%java_home%\bin”。注意：变量值的各个值之间要以英文分号“;”进行分隔。

Step3: 下载 Eclipse

登录 Eclipse 官网，进入“Eclipse IDE for Java Developers”的下载页面（<http://www.eclipse.org/downloads/>），根据自己的操作系统平台选择合适的 Eclipse IDE 文件，如图 1-3 所示。

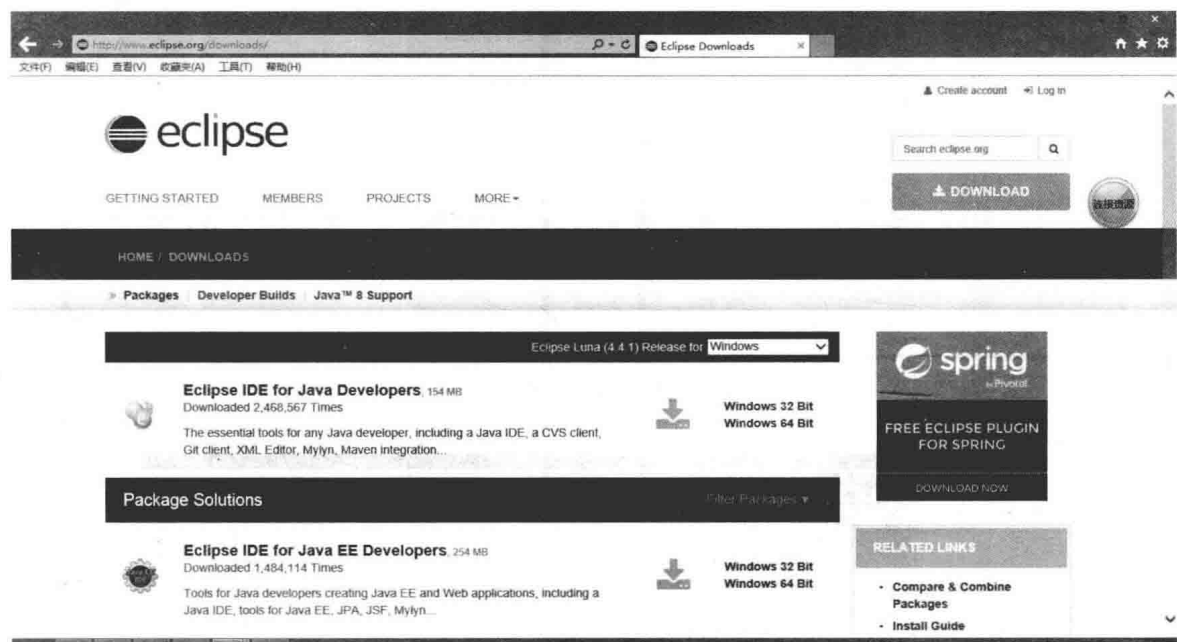


图 1-3 下载 Eclipse

下载 64 位的 Windows 操作系统适用的 Eclipse IDE 得到名为“eclipse-java-luna-SR1-win32-x86_64.zip”的压缩包（若发生版本变更，名称可能会不同），无需安装，将其解压缩后即可使用。如解压缩至“C:\Program Files\eclipse”目录，双击该目录下的“eclipse.exe”即可启动 Eclipse 开发环境。

Step4: 配置 Eclipse 工作空间

启动 Eclipse 时，出现如图 1-4 所示的“工作空间设置”对话框，修改默认地址并重新配置为“D:\eclipse\workspace”，则后续在 Eclipse 中创建的项目均会保存到此路径下。点击 OK 按钮开始启动 Eclipse，首次启动后出现如图 1-5 所示的欢迎页，点击右上角 Workbench 按钮打开如图 1-6 所示的 Eclipse 工作台界面，即可开始使用 Eclipse 开发 Java Application 了。

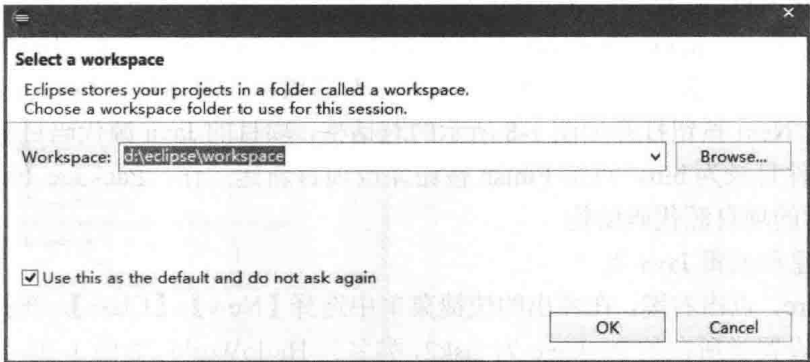


图 1-4 配置 Eclipse 工作空间

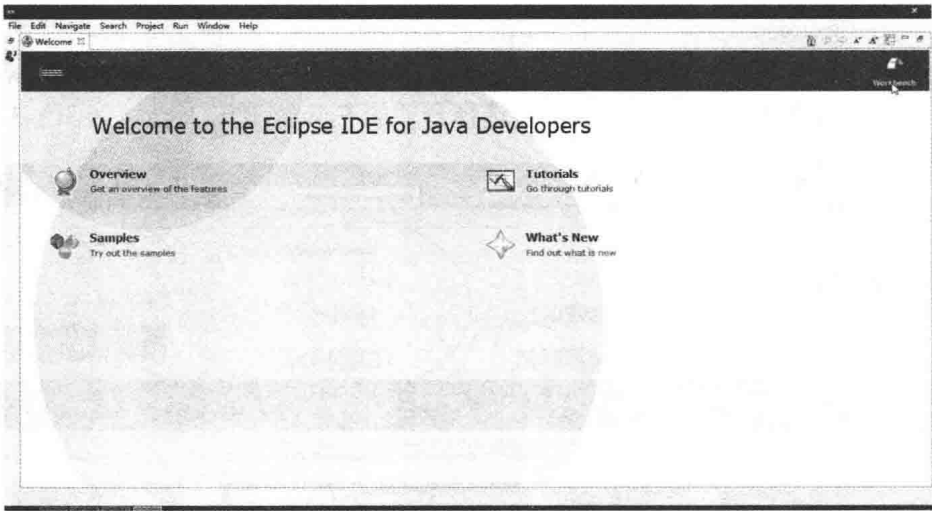


图 1-5 Eclipse 欢迎页

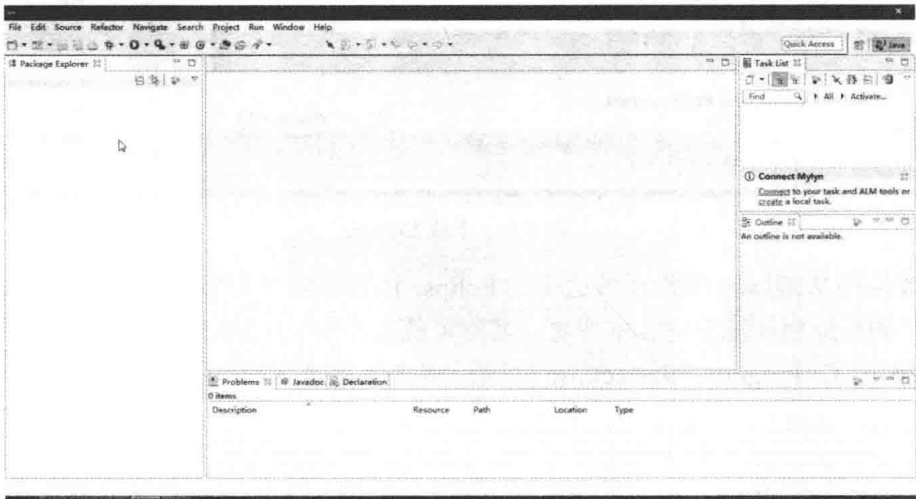


图 1-6 Eclipse 工作台界面

任务 2 Eclipse 中开发 Java Application

Step1: 新建项目

在 Eclipse 的 Package Explorer 视图空白处单击鼠标右键，在弹出的快捷菜单中选择【New】|【Java Project】，打开如图 1-7 所示的“新建项目”对话框，在 Project Name 处填写项目名称，点击 Next 按钮打开如图 1-8 所示的对话框，项目的 Java 源代码目录为 src，编译后得到的 class 文件目录为 bin，点击 Finish 按钮完成项目新建工作，Package Explorer 视图中得到如图 1-9 所示的项目源代码结构。

Step2: 创建和编辑 Java 类

选中包名 src，点击右键，在弹出的快捷菜单中选择【New】|【Class】，在打开的 New Java Class 对话框中设置类所在的 Package 为 task2，类名为 HelloWorld，如图 1-10 所示。点击 Finish 按钮确定后会自动打开该类的编辑窗口，如图 1-11 所示。

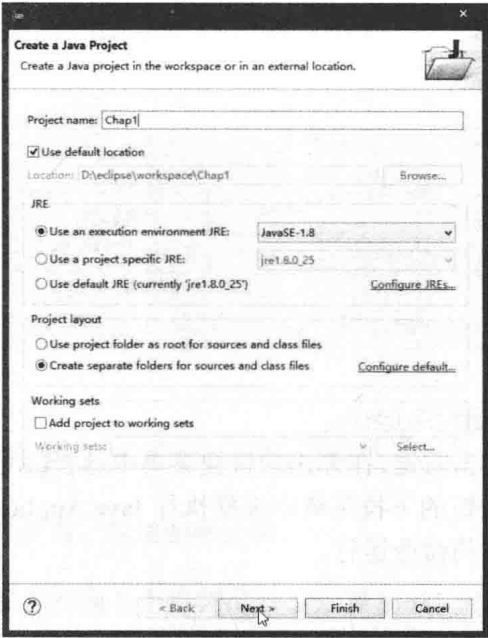


图 1-7 新建项目

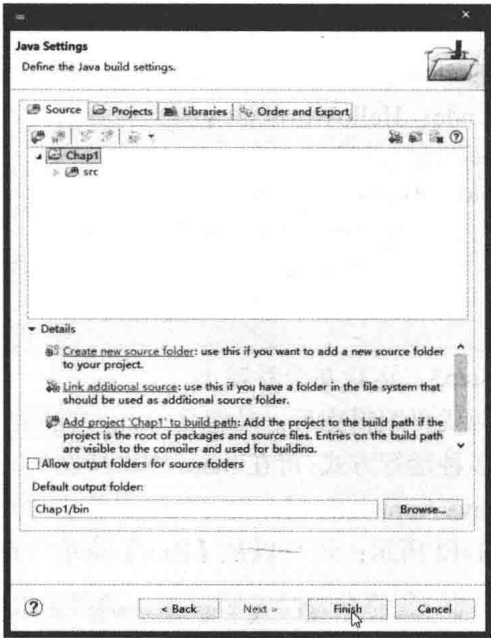


图 1-8 新建项目

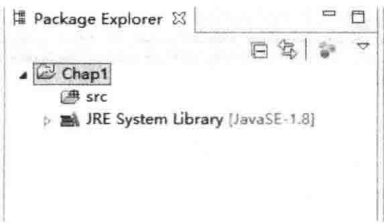


图 1-9 新建项目



图 1-10 新建类

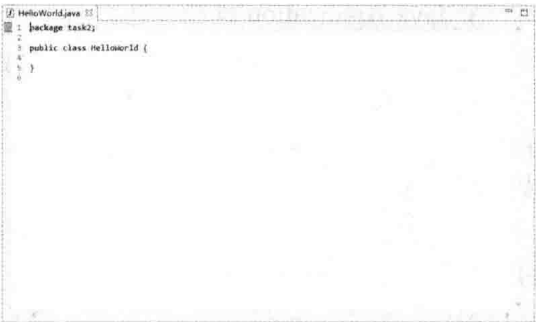


图 1-11 编辑类

在 HelloWorld.java 的编辑窗口编辑源代码，在编辑过程中，Eclipse 会同时进行编译工作，生成的 class 文件位于项目根目录的 bin 目录下。

Code: HelloWorld.java

```
package task2;
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Welcome to Java World!");
    }
}
```

Step3: 运行并查看输出

源代码编辑完后，从包含 main 方法的主类启动程序运行。

多种运行方式：可在 main 方法所在的主类中单击右键，在弹出的快捷菜单中选择【Run As】|【1 Java Application】；也可从工具栏上“运行”按钮的下拉菜单中选择执行 Java Application，如图 1-12 所示；还可以从【Run】菜单中选择合适的命令运行。

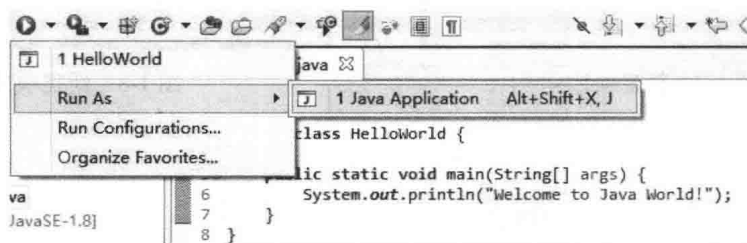


图 1-12 运行

在 Eclipse 的 Console 视图查看程序运行的控制台输出结果，如图 1-13 所示。

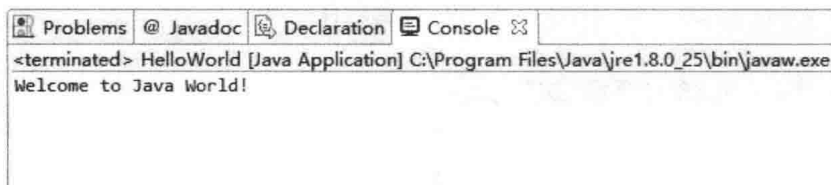


图 1-13 查看运行输出

知识点：Java Application 程序

一个 Java Application 程序往往由若干个类组成，其中包含 public static void main(String[] args)方法的类是主类，main()方法是 Java Application 程序的执行入口，由 JVM 调用执行。

本书并没有为每一个示例程序创建一个单独的 Java Project，而是主要按章节来创建 Java Project，再按任务来创建 package 和类，因此一个 Java Project 中可能包含多个示例程序，读者阅读时请注意。

Java Application 程序中定义类的源文件的后缀名为.java，如上例的 HelloWorld.java。源文件的命名规则是：如果源文件中有多类，那么至多只能有一个类是访问控制符为 public 的类；如果确实有一个类是 public 的，那么源文件的名称必须与这个类的名称完全相同，包括大小写；如果源文件没有 public 的类，那么源文件的名称只要和某个类的名称相同即可。

程序开发时，建议为 Java Application 程序的每一个 Java 类单独创建源文件。

当创建了 Java Application 的源文件之后，需对其进行解释和运行，使它成为机器能够读懂的二进制文件。图 1-14 展示了 Java Application 的解释执行过程。

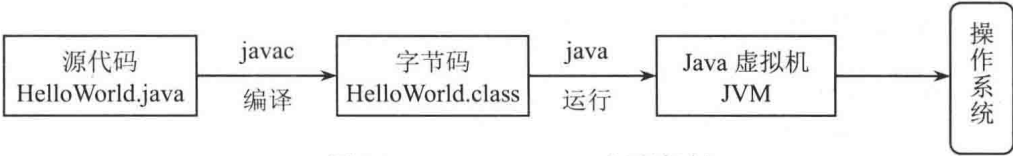


图 1-14 Java Application 运行机制

1.2 Java 语言基础

任务 3 实现四则运算

编程实现 2 个数据的四则运算功能：用户输入 2 个数据，显示这 2 个数分别进行加、减、乘、除法运算后的结果，运行效果如图 1-15 所示。

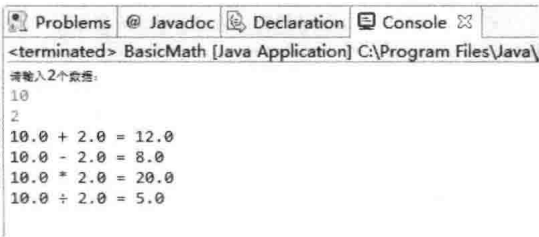


图 1-15 四则运算运行结果

在项目 Chap1 中创建包 task3，在此包中创建类 BasicMath，实现 2 个数的四则运算功能。

Code: BasicMath.java

```
package task3;
import java.util.Scanner;

public class BasicMath {
    public static void main(String[] args) {
        System.out.println("请输入 2 个数据: ");

        Scanner scan = new Scanner(System.in);

        //接收键盘输入 2 个浮点数
        double num1 = scan.nextDouble();
        double num2 = scan.nextDouble();

        //result: 存四则运算结果
        double result = num1 + num2;
        System.out.println(num1 + " + " + num2 + " = " + result);
    }
}
```

```
        result = num1 - num2;
        System.out.println(num1 + " - " + num2 + " = " + result);

        result = num1 * num2;
        System.out.println(num1 + " * " + num2 + " = " + result);

        result = num1 / num2;
        System.out.println(num1 + " ÷ " + num2 + " = " + result);
    }
}
```

知识点：Java 语法基础

一、标识符和关键字

标识符是用来标识类名、变量名、方法名、类型名、数组名、文件名等的有效字符序列。

Java 语言规定标识符由字母、下划线、美元符号和数字组成，并且需遵循以下规则：

- (1) 区分大小写。
- (2) 不能以数字开头。
- (3) 以字母或下划线_或\$符号开头。
- (4) 不能使用 Java 的关键字。

以下为合法的标识符：

aIdentifier、_Identifier、a 变量、\$Identifier、elseother。

以下为不合法的标识符：

2002WorldCup、Identifier@Java、else。

基本上，一般用于命名的符号包括字母 A-Z、字母 a-z、数字 0-9 以及 Unicode 字符集中序号大于 0xC0 的所有符号。

关键字是 Java 语言中已经被赋予特定意义的单词，对 Java 的编译器有特殊的作用，一般用来表示一种数据类型，或者程序的结构等。保留字是为 Java 预留的关键字，它们虽然现在没有作为关键字，但在以后的升级版本中很有可能成为关键字。所有的关键字都是小写。目前 Java 语言定义的关键字如下：

const（保留字） goto（保留字） abstract finally int public this boolean continue
float interface return throw break default for long short throws byte do native
static transient case double if new strictfp try catch else implements package
super void char extends import private switch volatile class final instanceof
protected synchronize while

另外，不能将关键字或保留字当作标识符来使用。

二、数据类型

Java 共有八种基本类型：四种整型，两种浮点数值型，一种字符型 char（用于 Unicode 编码中的字符），以及用于表示真假的布尔类型（boolean）。

1. 整型

Java 定义了 4 种整数类型：字节型（byte），短整型（short），整型（int），长整型（long）。

这些整数类型的长度和变化范围如表 1-1 所示。

表 1-1 整数的各种类型

名称	长度	数的范围
长整型	64	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
整型	32	-2,147,483,648 ~ 2,147,483,647
短整型	16	-32,768 ~ 32,767
字节型	8	-128~127

整数类型的范围与 Java 代码运行的机器无关。长整型数字（long）有一个后缀 L（如，40000000000L）。十六进制数据有一个前缀 0x（如，0xCAEF）；八进制数据有一个前缀 0，例如：010 就是十进制中的 8。

2. 浮点类型

Java 中定义了两种浮点类型：单精度（float）浮点型及双精度（double）浮点型，见表 1-2。

表 1-2 浮点数的各种类型

名称	位数	数的范围
double	64	1.7e-308~1.7e+308
float	32	3.4E-038~3.4E+038

大部分程序中使用 double 类型，因此表示 float 类型数据时需要在后面添加后缀 F，比如，3.402F。没有后缀的浮点数据（如 3.402）总被认为是 double 类型。也可以在数字后面加上后缀，如，3.402D 表示 double 类型。所有浮点计算都遵从 IEEE754 规范。

有三种特殊的浮点值：正无穷大、负无穷大、NaN（非数字），用于表示溢出和出错。比如，用 0 去除一个整数，所得的结果为正无穷大，而计算 0/0 或者对一个负数开平方则会产生 NaN。浮点类型的数据不适合在不容许舍入误差的金融计算领域使用。

例程 1-1：计算圆的面积——CircleArea.java。

```
package task3;

public class CircleArea {
    public static void main(String[] args) {
        double radio = 2;//圆半径
        double area = Math.PI * radio * radio;
        //常量 Math.PI 为 π 的值
        System.out.println("半径为 2 的圆面积是: " + area);
    }
}
```

3. 字符类型

char 类型用来表示 Unicode 编码表中的字符。

Unicode 定义的国际化的字符集能表示迄今为止人类语言的所有字符集。它是几十个字符集的统一，例如拉丁文、希腊语、阿拉伯语、古代斯拉夫语、希伯来语、日文片假名、匈牙利

语等，因此它要求 16 位。这样，Java 中的 char 类型是 16 位，其范围是 0~65,536，没有负数的 char。人们熟知的标准字符集 ASCII 码的范围仍然是 0~127，扩展的 8 位字符集 ISO-Latin-1 的范围是 0~255。既然 Java 被设计为允许其开发的 applet 在世界范围内使用，因此使用 Unicode 码代表字符是说得通的。当然，Unicode 的使用对于英语、德语、西班牙语或法语的语言来说是有低效，因为它们的字符能容易地被包含在 8 位以内。但是为了全球的可移植性，这一点代价是必需的。

4. 布尔类型

只有两个状态值，真（true）和假（false）。

例程 1-2：在工作时间输出消息——SayHello.java。

```
package task3;

import java.util.Calendar;
import java.util.GregorianCalendar;

public class SayHello {

    public static void main(String[] args) {
        Calendar now = new GregorianCalendar();
        //当前时间的日历对象

        boolean sayHello=false;

        int hour = now.get(Calendar.HOUR_OF_DAY);
        //当前时间（小时）

        if(hour>=8&&hour<18){
            sayHello=true;
        }

        if(sayHello){
            System.out.println("Hello , now is working time!");
        }
    }
}
```

三、运算符

Java 提供了丰富的运算符环境。Java 有四大类运算符：算术运算、位运算、关系运算和逻辑运算。Java 还定义了一些附加的运算符用于处理特殊情况。

算术运算符用在数学表达式中，其用法和功能与代数学中一样，Java 定义了下列算术运算符，如表 1-3 所示。

关系运算符决定值和值之间的关系，例如决定相等不相等以及排列次序。关系运算符如表 1-4 所示。