

普通高等教育“十三五”规划教材 (软件工程专业)

Java面向对象程序设计

JAVA MIANXIANGDUIXIANG
CHENGXUSHEJI

主 编 肖 奎 姚 竞
副主编 赵明俊 何 鹏 关培超
主 审 余敦辉

Java

- ◎ 取材得当、循序渐进、通俗易懂、结构清晰、层次分明
- ◎ 通过典型实例验证和说明语法结构、程序设计的思想和方法
- ◎ 始终强调以面向对象的思想来分析问题与解决问题



中国水利水电出版社
www.waterpub.com.cn

普通高等教育“十

件工程专业)

Java 面向对象程序设计

主 编 肖 奎 姚 竞

副主编 赵明俊 何 鹏 关培超

主 审 余敦辉



中国水利水电出版社
www.waterpub.com.cn

• 北京 •

内 容 提 要

本书主要介绍 Java 语言的基本语法以及桌面应用程序开发的相关知识,在讲解知识的过程中,始终强调以面向对象的思想来分析问题与解决问题。

本书共 10 章,第 1 章介绍了 Java 语言的特性以及几种 Java 开发工具,第 2 章讲解 Java 基本语法,第 3 章介绍了面向对象的基本概念,第 4 章至第 10 章分别介绍 Java 语言的异常处理机制、多线程与并发处理、输入输出流、集合类与泛型、图形用户界面以及 JDBC 的基本知识。本书结构合理,内容丰富,算法描述清晰,便于自学。

本书可作为高等院校计算机专业与其他相关专业的教材和参考书,也可供从事计算机软件开发的科技工作者参考。

图书在版编目(CIP)数据

Java面向对象程序设计 / 肖奎,姚竞主编. — 北京:
中国水利水电出版社, 2017.7
普通高等教育“十三五”规划教材. 软件工程专业
ISBN 978-7-5170-5572-3

I. ①J… II. ①肖… ②姚… III. ①JAVA语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆CIP数据核字(2017)第162290号

策划编辑: 祝智敏 责任编辑: 李 炎 加工编辑: 于杰琼 封面设计: 李 佳

书 名	普通高等教育“十三五”规划教材(软件工程专业) Java 面向对象程序设计 Java MIANXIANG DUIXIANG CHENGXU SHEJI
作 者	主 编 肖 奎 姚 竞 副主编 赵明俊 何 鹏 关培超 主 审 余敦辉
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
经 售	电话: (010) 68367658 (营销中心)、82562819 (万水) 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	三河市鑫金马印装有限公司
规 格	184mm×260mm 16 开本 11.5 印张 279 千字
版 次	2017 年 7 月第 1 版 2017 年 7 月第 1 次印刷
印 数	0001—3000 册
定 价	28.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换
版权所有·侵权必究

前 言

众所周知,“Java 程序设计”是计算机类相关专业的核心课程之一,它是一门集理论性、实践性于一体的课程。本书主要介绍 Java 语言的基本语法以及桌面应用程序开发的相关知识,在讲解知识的过程中,始终强调以面向对象的思想来分析与解决问题。

本书共 10 章,第 1 章介绍了 Java 语言的特性以及几种 Java 开发工具,第 2 章讲解了 Java 基本语法,第 3 章介绍了面向对象的基本概念,第 4 章至第 10 章分别介绍 Java 语言的异常处理机制、多线程与并发处理、输入输出流、集合类与泛型、图形用户界面以及 JDBC 的基本知识。本书的重要特色之一,是对那些与当前章节联系紧密的常用设计模式进行了介绍,比如工厂方法模式、迭代器模式、单例模式等。方便读者深刻地理解面向对象的思想与内涵。此外,书中的许多参考例子也可以直接作为模块用于开发,为学生实践练习提供了方便。

本书由肖奎、姚竞任主编,肖奎负责全书的统稿、修改、定稿工作,赵明俊、何鹏、关培超任副主编。主要编写人员分工如下:第 1、5、6 章由姚竞编写,第 2、3 章由何鹏编写,第 4、7 章由肖奎编写,第 8 章由关培超编写,第 9、10 章由赵明俊编写,全书由余敦辉主审。张龔、曾诚、孙斌、陈志雄、刘国君、李其锋、祝建平、戴瀚波、刘坤、汪波、李祥等老师也参与了部分章节的编写工作。本书的作者队伍均为本科院校计算机类专业一线教师,具有多年的教学经验。并且,编写队伍人员也一直参与企业项目开发,具有丰富的软件开发经验。

本书结构合理,内容丰富,算法描述清晰,便于自学,可作为高等院校计算机专业与其他相关专业的教材和参考书,也可供从事计算机软件开发的科技工作者参考。

由于编写时间紧张,本书难免存在疏漏,敬请读者批评与指正。

编 者

2017 年 5 月

目 录

前言

初级篇

第 1 章 初识 Java	1	2.4.1 赋值运算符和表达式	24
1.1 Java 语言概述	1	2.4.2 算术运算符和表达式	25
1.1.1 Java 语言发展简史	1	2.4.3 关系运算符和表达式	27
1.1.2 Java 语言特点	1	2.4.4 逻辑运算符和表达式	28
1.1.3 能用 Java 做什么	2	2.4.5 位运算符和表达式	29
1.2 使用 Java 开发程序	3	2.4.6 条件运算符和表达式	31
1.2.1 开发工具和运行环境	3	2.4.7 运算符的优先级	31
1.2.2 开发与运行流程	3	2.5 流程控制	32
1.3 编写和执行 HelloRandom 程序	4	2.5.1 顺序语句	32
1.3.1 以 JDK 方式开发 HelloRandom	4	2.5.2 条件分支语句	33
1.3.2 通过 BlueJ 集成开发环境开发 HelloRandom	5	2.5.3 循环结构语句	41
1.3.3 通过 Eclipse 集成开发环境开发 HelloRandom	7	2.6 数组	43
1.4 API 参考文档	10	2.6.1 多维数组	43
1.5 常用术语	11	2.6.2 二维数组	44
1.6 设计模式	11	本章小节	47
本章小结	12	习题	47
习题	12	第 3 章 面向对象	48
第 2 章 基础语法	13	3.1 面向对象基本概念	48
2.1 标识符与关键字	13	3.1.1 对象	48
2.1.1 标识符	13	3.1.2 类	48
2.1.2 Java 关键字	14	3.1.3 接口	49
2.2 基本数据类型	14	3.2 类	49
2.2.1 整型数据	15	3.3 继承	51
2.2.2 浮点型(实型)数据	16	3.4 接口	52
2.2.3 字符型数据	17	3.5 抽象类	53
2.2.4 布尔型数据	18	3.6 内部类	57
2.2.5 各类型数据间的相互转换	18	3.7 工厂方法模式	61
2.3 变量与常量	20	3.7.1 基本概念	61
2.4 运算符	24	3.7.2 工厂方法模式的优点	64
		3.7.3 工厂方法模式的使用场景	64
		本章小节	65

习题..... 65

第 4 章 异常处理..... 66

4.1 异常概述..... 66

4.1.1 异常的基本概念..... 66

4.1.2 异常类与对象..... 68

4.2 异常处理方法..... 71

4.2.1 try...catch...finally..... 71

4.2.2 使用 throws 语句声明抛出的异常..... 73

4.2.3 使用 throw 语句主动抛出异常..... 74

4.3 自定义异常..... 75

4.4 异常处理流程..... 76

本章小结..... 78

习题..... 78

第 5 章 并发处理..... 79

5.1 进程与线程..... 79

5.2 线程生命周期..... 79

5.3 线程的实现方式..... 80

5.3.1 产生 Runnable 对象..... 80

5.3.2 自定义线程类..... 81

5.4 线程调度..... 82

5.4.1 通过 sleep 方法暂停执行线程..... 82

5.4.2 通过 interrupt 方法中断线程..... 82

5.4.3 join 方法..... 83

5.5 线程通信..... 85

5.5.1 通过类属性共享数据..... 85

5.5.2 通过引用共同对象共享数据..... 86

5.6 线程同步 Synchronization..... 87

5.6.1 线程干扰和内存不一致性错误..... 88

5.6.2 同步方法 (synchronized methods) ... 88

5.6.3 同步语句 (synchronized statements) 89

5.7 单例模式 Singleton..... 89

本章小结..... 94

习题..... 94

第 6 章 输入/输出流和文件..... 95

6.1 输入/输出流..... 95

6.1.1 字节流 Byte Stream..... 96

6.1.2 字符流 Character Stream..... 97

6.1.3 基于行的输入输出操作..... 97

6.1.4 缓冲流..... 98

6.1.5 命令行输入输出操作..... 98

6.1.6 数据流..... 99

6.1.7 对象流..... 100

6.2 文件输入输出..... 102

6.2.1 路径 Path..... 102

6.2.2 Path 类..... 102

6.2.3 常用文件输入输出操作..... 104

6.2.4 随机文件存取..... 106

本章小结..... 107

习题..... 107

第 7 章 集合与泛型..... 108

7.1 Java 中的集合类..... 108

7.1.1 集合类概述..... 108

7.1.2 List 接口的常用方法..... 108

7.1.3 List 接口的实现类..... 109

7.1.4 Set 接口的常用方法..... 111

7.1.5 Set 接口的实现类..... 111

7.2 Map 接口..... 114

7.2.1 认识 Map..... 114

7.2.2 Map 接口常用的方法..... 114

7.2.3 Map 接口的实现类..... 114

7.3 泛型..... 116

7.3.1 什么是泛型..... 116

7.3.2 在集合类中使用泛型..... 118

7.3.3 类型通配符..... 119

本章小结..... 120

习题..... 120

高级篇

第 8 章 图形用户界面..... 121

8.1 AWT 组件和容器..... 121

8.1.1 Frame 和 Dialog 类..... 122

8.1.2 Label 类..... 124

8.1.3 Button 类..... 125

8.1.4 TextField 类..... 126

8.1.5 TextArea 类..... 127

8.1.6 MenuBar、Menu、MenuItem 类..... 128

8.1.7 布局管理器	130	9.3 UDP 网络通信程序	158
8.2 Java 图形事件处理机制	131	9.3.1 UDP 通信相关类	158
8.3 Swing 图形界面组件	134	9.3.2 UDP 通信示例	159
8.3.1 JFrame 和 JDialog	135	本章小结	161
8.3.2 中间容器	139	习题	161
8.3.3 常用组件	141	第 10 章 JDBC	163
8.4 Java 图形用户界面设计辅助工具	145	10.1 JDBC 概述	163
8.4.1 WindowBuilder 插件的获取	145	10.1.1 数据访问模型	163
8.4.2 WindowBuilder 插件的使用	147	10.1.2 JDBC 驱动程序与安装	164
本章小结	149	10.1.3 JDBC 数据库访问常用的类和接口	164
习题	149	10.2 数据库访问步骤	167
第 9 章 网络编程	150	10.2.1 加载驱动程序	168
9.1 网络基础知识	150	10.2.2 建立连接对象	168
9.1.1 网络基础概念	150	10.2.3 建立语句对象	169
9.1.2 网络协议	151	10.2.4 返回结果集合	170
9.1.3 端口及套接字	151	10.3 预处理语句	172
9.2 TCP 网络通信程序	152	10.4 数据库表常用操作示例	175
9.2.1 TCP 通信相关类	153	本章小结	176
9.2.2 服务器与客户机通信示例	155	习题	176

初级篇

第 1 章 初识 Java

教学要求:

1. Java 语言特点
2. Java 开发和运行流程
3. Java 开发工具
4. API 参考文档使用

开发计算机软件需要使用特定的程序设计语言,遵循特定的编程方法,借助相应的编程工具来完成。随着程序设计技术的更新,程序设计语言从早期的汇编语言,发展至今天的面向过程语言(如 C)、面向对象语言(如 Java)等。Java 是一种具有面向对象、跨平台等重要特征的编程语言,被广泛应用于设计基于桌面系统、移动应用平台、小型设备等环境的应用程序。Java 在过去的十年内一直都是被开发者使用最多和被应用于开发项目最多的编程语言之一。

1.1 Java 语言概述

1.1.1 Java 语言发展简史

1991 年,美国 Sun 公司的 James Gosling 和 Patrick Naughton 为在机顶盒、手机、PDA 等设备上开发基于有限资源的应用软件,开发了 Oak 语言, Oak 在 1995 年更名为 Java。

1996 年 1 月, Sun 公司发布了 Java 的第一个开发工具包(JDK 1.0),标志着 Java 成为一种独立的开发工具。

2005 年 6 月,在 Java One 大会上, Sun 公司发布了 Java Platform Standard Edition 即 Java SE 的第 6 版。Java 平台的其他版本也随之更名为 Java EE、Java ME 等。

2006 年 11 月 13 日, Java 技术的发明者 Sun 公司宣布,将 Java 技术作为免费软件对外发布。

2009 年,甲骨文 Oracle 公司宣布收购 Sun。

今天 Java SE 版本已经发展至第 8 版本,主要针对桌面系统、服务器和嵌入式系统提供了强有力的开发支持。

1.1.2 Java 语言特点

(1) 简单

Java 面向对象的特征使得其语法容易被理解,学习成本较低。Java 没有指针和运算符重

载等复杂语法。具有自动内存回收机制，使程序员能专注实现代码功能，一般不需要考虑内存管理的问题。语法严谨，编译和运行环境有严格的错误检查机制，能较早发现代码的错误。

(2) 面向对象

Java 程序中的数据除了少量基本数据类型外，通常都以对象的形式出现。程序的执行依赖对象之间的消息传递来进行，因此方便理解和构建复杂的应用程序。

(3) 平台无关

Java 程序编译后生成字节码 (byte-code)，而不是基于特定操作系统的机器码。Java 程序执行时，不同操作系统平台上的运行环境将负责对字节码进行解释和执行，因此 Java 程序可以在任何配备了 Java 解释程序和运行系统 (run-time system) 的系统上运行。

(4) 稳健和安全

Java 的编译和运行环境会对代码进行严格的检查，以消除错误。类型检查和异常处理机制能有效地发现早期错误，避免程序在运行期间出现严重问题。自动内存管理机制避免了内存非法引用和溢出等问题。运行时的沙箱环境使程序在相对独立的环境中执行，增强了代码健壮性的同时，也避免了恶意代码的破坏。安全管理器能防止程序的非法访问。

(5) 分布式

Java 丰富的标准函数库以及多种设计框架，能帮助程序员来处理不同层次，不同类型的分布式应用任务。从 socket 连接到 URL 访问，从 RMI 远程方法调用到 Web 服务，Java 都能很好地完成各项任务，以构建各式各样的应用。

(6) 多线程

和其他许多高级语言一样，Java 支持多线程程序的构建，并提供完善的并发处理机制，能对线程进行创建、调度、同步管理等操作。

1.1.3 能用 Java 做什么

Java 作为一种通用性的语言，适应于大多数的编程需要。常见的应用方向有如下几种：

(1) 桌面应用

Java 适用于创建基于完整计算机系统的应用，如基于 Windows、Mac OS、Linux 等平台的具有图形界面的应用程序。

(2) Web 应用

Java 适用于构建电子商务应用或独立网站等 Web 项目。开发人员可以运用许多不同的框架来创建 Web 项目，如 Spring、Struts、Hibernate 等。例如医疗救护、保险、教育、国防以及其他的不同部门网站都是以 Java 为基础来开发的。

(3) Android 应用

Java 是移动应用平台 Android 的基础，通过 Java 可以构建复杂的手机、平板电脑应用。虽然 Android 运用了不同的 JVM 以及不同的封装方式，但代码还是用 Java 语言编写的。相当一部分的手机中都支持 Java 游戏。

(4) 嵌入式领域

Java 在嵌入式应用领域中有很大的发展空间。具有 130KB 存储空间的物理设备就能够通过 Java 编写应用程序。典型的应用如智能卡、机顶盒、传感器和监控设备等。

1.2 使用 Java 开发程序

1.2.1 开发工具和运行环境

Java 开发工具 SDK (Software Development Kit) 是 Java 开发应用的核心, 包括了 Java 在不同应用方向上的运行环境 JRE (Java Runtime Environment)、Java 工具和 Java 的应用程序编程接口 (Java API, 也称基础类库)。SDK 具备开发、查错、监控 Java 应用程序等能力。实质上, 不论什么 Java 应用都基于某个版本的 SDK 来开发, 并基于特定应用环境的 JRE 来运行。典型的 SDK 版本基于不同的应用方向包括 SE (Standard Edition)、EE (Enterprise Edition) 等版本。JDK (Java Development Kit) 是 SDK 的扩展子集, Oracle 公司建议使用 JDK 来表示 Java SE 版本的 SDK。JRE 通常包含 JVM (Java Virtual Machine, Java 虚拟机) 标准实现和 Java 核心类库。

Java 的集成开发工具众多, 比较著名的包括 BlueJ、Eclipse、NetBeans 等。

BlueJ 集成开发环境主要用于 Java 教学以及小规模软件开发。BlueJ 依赖 JDK 运行, 由英国 Kent 大学开发和管理。网址为 <https://bluej.org>。

Eclipse 集成开发环境是最广泛使用的 Java IDE (Integrated Development Environment), 同时也支持其他编程语言的开发。Eclipse 具有较强的可扩展性。网址为 www.eclipse.org。

NetBeans 集成开发环境是 Oracle 公司 (最早由 Sun Microsystems 开发) 拥有并管理的 Java 集成开发环境, 可通过增加模块以支持多种编程语言。网址: <https://netbeans.org>。

1.2.2 开发与运行流程

在进行 Java 应用编程过程中, 可以选择直接使用 SDK 通过操作系统终端进行开发, 或者使用集成开发环境完成全部工作, 基本流程是一样的。如图 1-1 所示, Java 程序的开发与运行过程包括如下 3 个步骤:

步骤 1: 通过文本编辑工具创建扩展名为 .java 的源代码文件。

步骤 2: 使用编译工具 (如 JDK 中的 javac) 将 .java 文件编译成 .class 字节码文件。

步骤 3: 使用装载工具 (如 JDK 中的 java) 装载 .class 文件至 Java 运行环境 JRE 中运行。

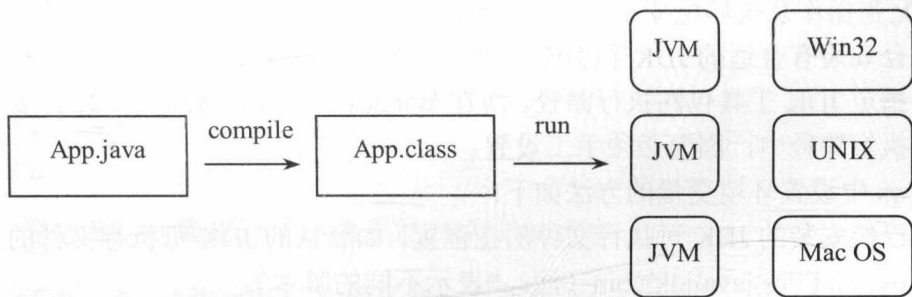


图 1-1 Java 程序开发及运行过程

特定的运行平台如 Windows、Mac OS 等可以预先安装特定版本的 Java 运行环境 JRE, 而

编译后的.class 文件一般无需再次编译，即可在各种 JVM 中执行。

1.3 编写和执行 HelloRandom 程序

HelloRandom 程序用于向操作系统终端或控制台输出一个字符串“Hello”，然后再输出一个随机数。程序需要存放在一个名字为 HelloRandom.java 的 Java 源文件中，其源代码如例 1-1 所示：

例 1-1 HelloRandom 程序。

```
//HelloRandom.java
public class HelloRandom{                                //声明类 HelloRandom
    public static void main(String [] args){              //主方法 main，是程序执行的起点
        System.out.print("Hello ");                      //向终端输出字符串 Hello
        System.out.println(Math.random());               //利用数学类产生随机数，然后输出至终端
    }
}
```

取决于产生的随机数，程序运行结果可能为：

Hello 97

1.3.1 以 JDK 方式开发 HelloRandom

下面介绍直接以 JDK 方式在 Windows 操作系统中开发一个 HelloRandom 程序的完整过程。具体步骤包括：

- (1) 在操作系统中安装开发工具 JDK 和运行环境 JRE。
- (2) 配置操作系统环境变量（可选项）。
- (3) 利用文本编辑器编写 HelloRandom.java 源代码。
- (4) 利用编译器将 HelloRandom.java 编译成 HelloRandom.class 字节码文件。
- (5) 通过 JRE 执行 HelloRandom.class。

步骤 1：在操作系统中安装开发工具 JDK 和运行环境 JRE。如图 1-2 所示，使用浏览器访问 <http://www.oracle.com/technetwork/java/javase/downloads/index.html> 页面下载并安装 java SE 标准版 JDK。通常 JDK 的安装包中同时包括有 JRE 的安装包，因此在安装 JDK 的同时也可以同时安装 JRE。Windows 中 JDK 和 JRE 的默认安装路径都是 C:\Program Files\Java

步骤 2：配置操作系统环境变量。要编译和执行 Java 程序，首先要求运行 Java 程序的特定操作系统平台安装有合适的 JDK 和 JRE。UNIX、Linux 等操作系统在安装完 JDK 之后，通常不需要额外指定 JDK 工具包的执行路径。而在 Windows 的早期系统中，JDK 安装包通常不会自动添加到执行路径中，因而必须手工设置。

在 Windows 中设置环境变量的方法如下：

- 找到已经安装的 JDK 可执行文件所在位置，如默认的 JDK 可执行文件的安装路径为 C:\Program Files\java\jdk*\bin（注：*表示不同的版本）。
- 右键点击“我的电脑”，然后依次选择“属性”>“高级系统设置”>“高级”>“环境变量”。
- 在系统变量中选择 Path，双击选择修改。在 Path 变量值后增加“;”，以保留原有变

量值，然后将新安装的 JDK 中 bin 目录的路径复制到 Path 变量默认值的最后。

- 在命令行窗口中输入 `java -version` 并回车，如果有提示 Java 版本信息，则说明环境变量设置正确。



图 1-2 JDK 下载页面

步骤 3: 利用文本编辑器编写 `HelloRandom.java` 源代码。打开记事本或其他文本编辑器新建一个名为 `HelloRandom.java` 的文件，在其中输入 `HelloRandom` 源代码。

注意，此步骤中一定要确保源代码的文件扩展名是 `.java`，同时要确保文件名 `HelloRandom` 和类的名字完全一致。

步骤 4: 利用编译器将 `HelloRandom.java` 编译成 `HelloRandom.class` 字节码文件。

在终端窗口中，使用 DOS 指令找到 `HelloRandom.java` 文件所在位置。

输入如下指令：

```
javac HelloRandom.java
```

如果编译没有错误，将没有任何警告信息，并生成如下文件：

```
HelloRandom.class
```

步骤 5: 通过 JRE 执行 `HelloRandom.class`。

在终端窗口中输入如下指令：

```
java HelloRandom
```

程序的运行结果可能为：

```
Hello 97
```

1.3.2 通过 BlueJ 集成开发环境开发 `HelloRandom`

下面介绍通过 BlueJ 集成开发工具开发一个 `HelloRandom` 程序的完整过程。具体步骤包括：

- (1) 在操作系统中安装 BlueJ。
- (2) 创建 Java 工程项目。

(3) 编写 HelloRandom.java 源代码。

(4) 编译、执行 HelloRandom。

步骤 1：在操作系统中安装 BlueJ。使用浏览器访问 <https://bluej.org/> 页面，下载并安装最新版 BlueJ Installer with Java 8 JDK，如图 1-3 所示。BlueJ 安装后的默认路径为 C:\Program Files (x86)\BlueJ。

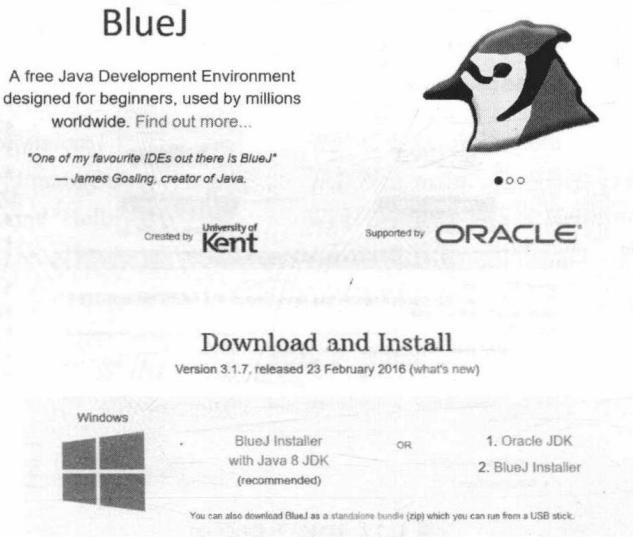


图 1-3 BlueJ 下载页面

步骤 2：创建 Java 工程项目。打开 BlueJ 程序，在主菜单中选择 Project>New Project 创建新项目，在设置好项目文件目录后，打开项目主界面，如图 1-4 所示。

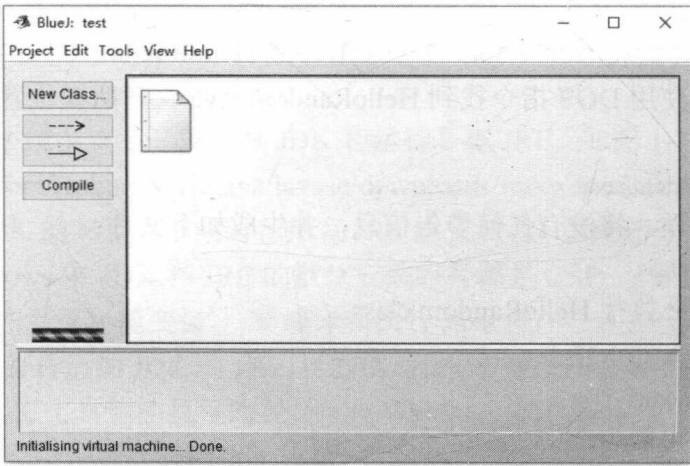


图 1-4 BlueJ 项目主界面

步骤 3：编写 HelloRandom.java 源代码。点击项目主界面中的 New Class 按钮生成 HelloRandom 源代码。BlueJ 会自动产生一个基本的源代码框架，并在项目主界面中生成 HelloRandom 程序图标，如图 1-5 所示。双击图标，将其内容替换为例 1-1 中的源代码。

步骤 4：编译和执行 HelloRandom。右键点击项目主界面中的 HelloRandom 图标，选择 Compile 编译（也可从左边菜单中选择 Compile 按钮），然后再次右键选择 HelloRandom 图标，

选择 `void main(String [] args)` 方法执行该程序，BlueJ 将弹出终端 Terminal 并显示执行结果。

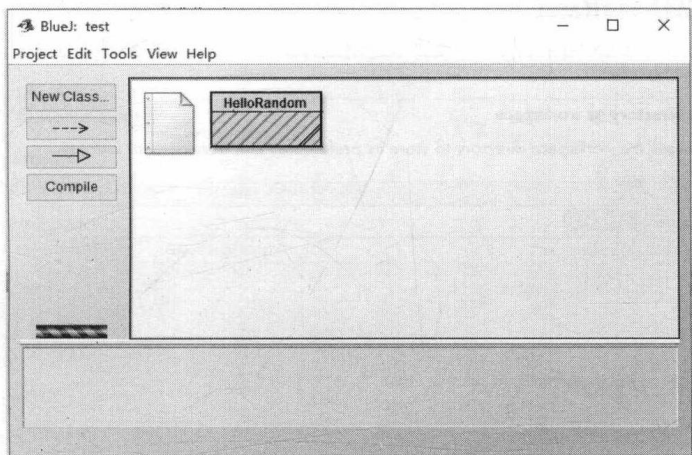


图 1-5 编写 HelloRandom 源代码

1.3.3 通过 Eclipse 集成开发环境开发 HelloRandom

下面介绍通过 Eclipse 集成开发工具开发一个 HelloRandom 程序的完整过程。具体步骤包括：

- (1) 在操作系统中安装 Eclipse。
- (2) 创建 Eclipse 工作区
- (3) 新建 Eclipse 项目 HelloRandom。
- (4) 新建 HelloRandom.java 文件。
- (5) 编译、执行 HelloRandom。

步骤 1：在操作系统中下载并安装 Eclipse。使用浏览器访问 <http://www.eclipse.org/downloads/> 页面下载并安装最新版 Eclipse。在本书撰写时，Eclipse 最新版为 Neon，如图 1-6 所示。

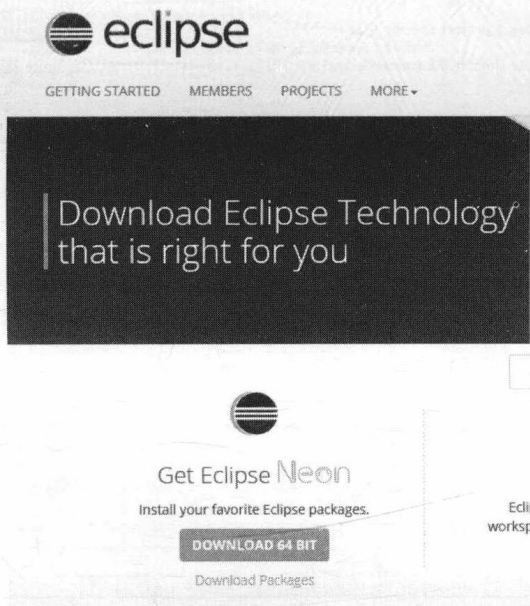


图 1-6 Eclipse Neon 下载页面

步骤 2: 启动 Eclipse 后，将提示创建 Eclipse 工作区，即创建 Eclipse 文件存放位置如图 1-7 所示，该位置可以任意指定。

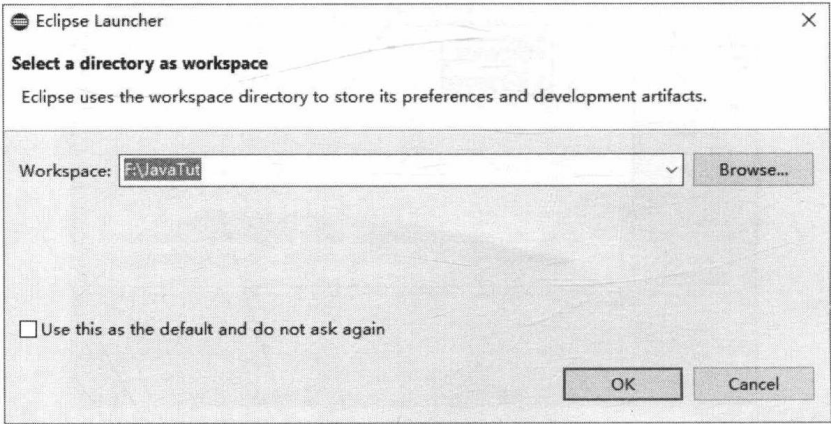


图 1-7 新建 Eclipse 工作区

步骤 3: 新建 Eclipse 项目 HelloRandom。在建立好工作区后，在出现的主窗口中依次选择 File>New>Java Project 新建工程项目 HelloRandom，如图 1-8 所示。

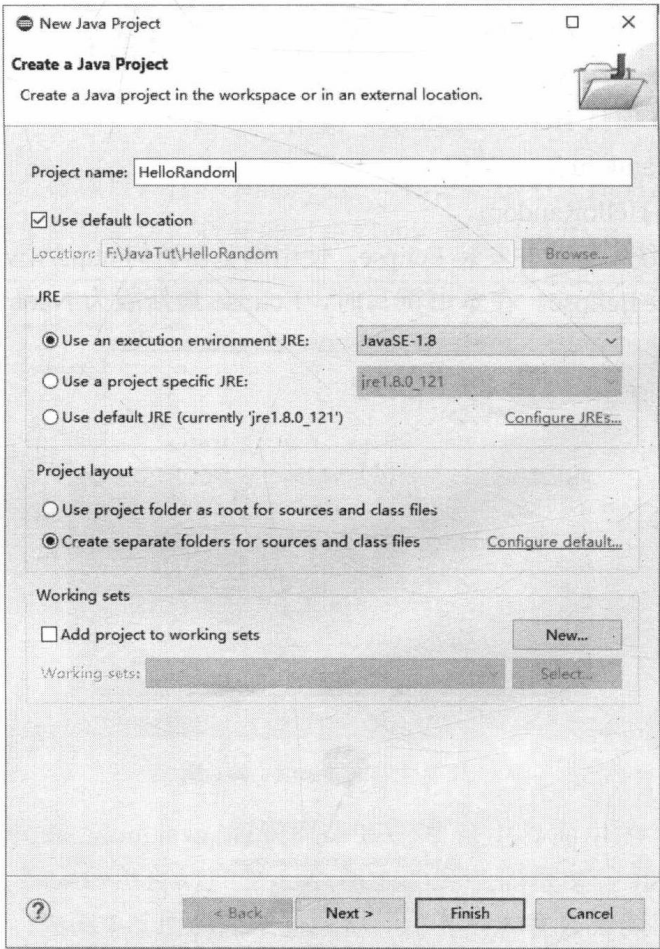


图 1-8 新建 Eclipse 项目 HelloRandom

步骤 4: 新建 HelloRandom.java。在主窗口中依次选择 File>New>Class 建立 Java 类 HelloRandom, 相应的源文件也随之建立, 如图 1-9 所示。在主窗口中选中 HelloRandom.java 进行编辑, 修改为例 1-1 中所示的内容, 如图 1-10 所示, 然后使用 Ctrl+S 组合键保存。



图 1-9 新建 HelloRandom.java

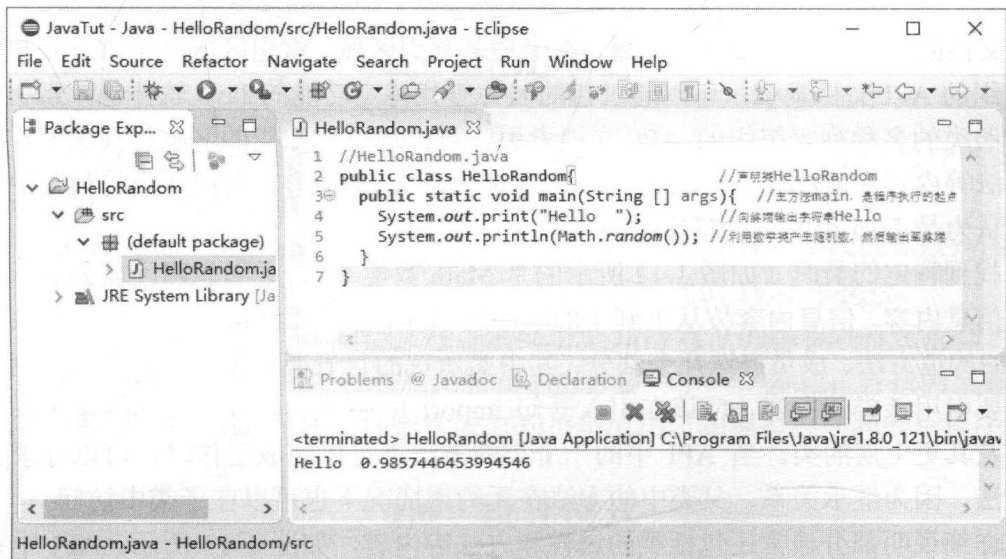


图 1-10 编辑 HelloRandom.java

步骤 5: 编译、执行 HelloRandom。在主窗口中依次选择 Run>Run HelloRandom 来执行

HelloRandom 程序。默认情况下 Eclipse 会进行自动编译，因此可直接执行程序，编译工作自动完成。

1.4 API 参考文档

API（Application Program Interface）本身是一组已经构建好的 Java 类库，可以被开发人员进行调用。而 API 文档则包含了有关 Java 类库的文字说明。从组织结构上看 Java API 中的功能通常以类 class 和接口 interface 的形式进行封装，功能类似的类和接口被分组到不同的包结构 package 中，以方便管理、分发、查询等操作。

在浏览器中输入地址 <http://docs.oracle.com/javase/8/docs/api/> 将打开 Java SE 标准版第 8 版本的 API 文档，如图 1-11 所示。

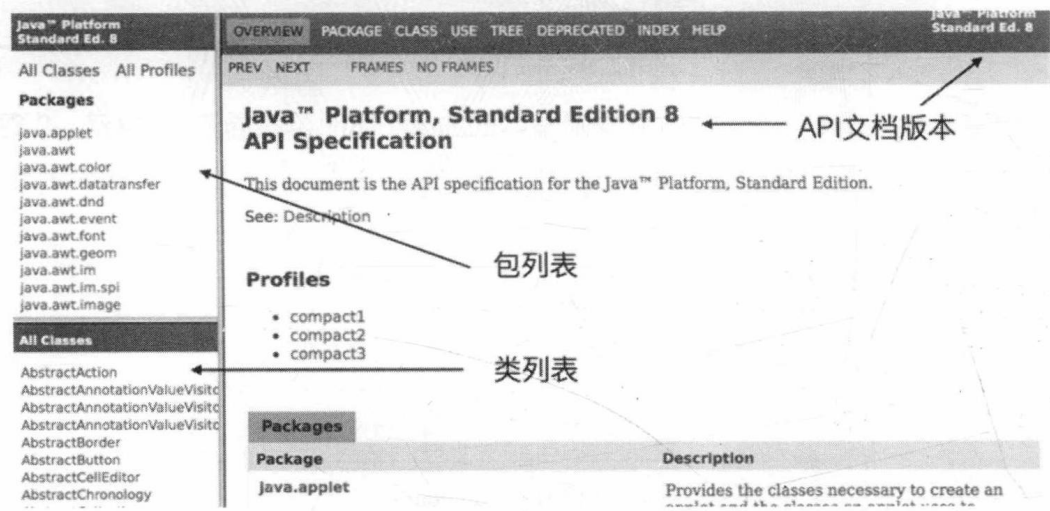


图 1-11 API 文档组成

API 文档的右边是详细说明区域，左边是索引区域。索引区域上半部分是按照包的名称列表组织的 API 文档索引，可以通过包的名称查找特定功能的 Java 软件包。索引区域下半部分是按照类的名称列表组织的 API 文档索引，可以通过已知类的名称查找类的详细信息、属性、方法等内容。请注意，在这里 Java 中的接口会以斜体形式出现在类的列表中间，接口有时也被认为是 Java 类的一种特定类型。

当查找到特定的类时（如图 1-12 所示的是 Math 数学类），详细说明区域会展示与该类相关的全部信息内容。信息内容按从上到下的顺序通常包括类名称和继承关系、类的说明、类的属性、类的构造方法、成员方法几个部分。其中类所在的包表明了类的从属关系，同时也说明了如果需要引用该类的内容，应该在引入语句 import 中添加的内容。类的继承关系展示了该类的父类及其更上层的类。当 API 中的一个类的方法不足以完成工作时，可以寻求其父类中定义的方法，因为继承关系，父类中的方法在无约束情况下也可以在子类中访问。

类的详细说明部分通常还包括类的属性和方法等内容，如图 1-13 所示。API 中定义的类的属性部分通常是某个类的常量，如数学类 Math 中定义的 PI 表示圆周率。类的方法部分通常定义有该类能完成的具体功能，类似 C 语言中的函数。