

Java 应用开发系列丛书

# Java SE 应用程序设计

张利国 刘伟 编著

 **北京理工大学出版社**  
BEIJING INSTITUTE OF TECHNOLOGY PRESS

# 前 言

Java 不仅仅是一门高级编程语言，更是一种完备的技术体系和开发平台。Java 拥有卓越的技术特性，丰富的编程接口（类库），多款功能强大的开发工具。全球 Java 程序员已超过 450 万，在企业级应用开发领域中占有过半的市场份额，超过 25 亿台设备正在使用 Java 技术并获得除微软外所有主流 IT 厂商的大力支持。从目前的应用现状和发展前景来看，Java 已经成为软件开发从业人员的首选技术。

遗憾的是，许多 Java 学习者因不得其门而入，无奈入宝山而空返。本书从最基础的内容开始，详细讲述了如何使用 Java 技术进行应用程序开发。全书共分为 19 章，内容由浅入深，全面覆盖了 Java 编程基础知识及高级性能。在保证知识体系完备，脉络清晰，论述精准深刻的同时，本书尤其注重培养读者的实际动手能力，并结合具体的知识点专门编写了大量的（近 300 个）实用程序作为例程，例程的源代码可以到网站 <http://www.v512.com> 免费下载。如果说 Java 知识像海洋一样茫无际涯，我们愿意通过本书，将之化作涓涓溪流，一路欢歌，流入读者心底。

IT 技术发展迅速，自 1995 年以来，Sun 公司已发布了 Java 开发工具集（JDK，Java Development Kit）的七个主要版本，每个版本中都会有新的功能或特性加入，当前的最新版本是 Java SE6.0，为跟上知识点的快速更新，实现本书的保值以回馈读者厚爱，我们将对本书内容进行持续更新升级，更新升级的内容将通过我们的网站 <http://www.v512.com> 免费发布。同时为帮助读者更好地使用本书，我们还亲自精心制作了与之配套的讲解视频，读者也可通过我们的网站免费下载。

本书篇幅较长、内容涵盖广泛，尽管我们尽了最大努力，但仍难免会有错误和疏漏之处，我们在 <http://www.v512.com> 上给出了专门的勘误页，并对其进行持续更新，勘误页中还还为读者提供了报告 bug 并提出改进意见的表单（建议您先阅读一遍勘误页，避免出现重复提交），读者也可以通过信箱 [javase@v512.com](mailto:javase@v512.com) 提交 bug 信息、关于本书的意见和建议等。我们真诚希望收到广大读者的反馈信息，并将认真阅读所有的 E-mail，感谢您的建议使本书更新更迅捷、更具实用价值！因信息量较大，我们可能无法及时回复每一个问题，请您谅解。

用心写作并不是一件轻松的事情，为不辜负读者厚爱，我们只能尽心竭力、精益求精，虽不敢比古人“二句三年得，一吟双泪流”的呕心沥血，但字斟句酌、诚惶诚恐确是做到了！感谢我们的家人和朋友们的全力支持！感谢我的女儿悠悠，是你给了我写作的灵感和无限快乐！

张利国

2007 年 8 月 于北京



# 目 录

第一章 Java 技术入门.....	1
1.1 Java 技术概述.....	1
1.1.1 Java 发展简史.....	1
1.1.2 Java 技术特性.....	2
1.2 Java 平台核心机制.....	4
1.2.1 Java 平台.....	4
1.2.2 Java 程序运行过程.....	5
1.3 Java 开发环境安装及配置.....	6
1.3.1 获取和安装 Java 开发工具集.....	6
1.3.2 JDK6.0 组成结构.....	7
1.3.3 配置环境变量 Path.....	7
1.4 第一个 Java 应用程序.....	9
1.4.1 创建源文件 HelloWorld.....	9
1.4.2 将源文件编译为字节码文件.....	10
1.4.3 运行程序.....	11
1.4.4 Java 应用程序结构.....	11
第二章 面向对象程序设计入门.....	13
2.1 面向对象程序设计概述.....	13
2.2 类和对象.....	14
2.2.1 类的声明.....	14
2.2.2 属性.....	15
2.2.3 方法.....	16
2.2.4 构造方法.....	16
2.2.5 对象创建和使用.....	17
2.3 信息封装和隐藏.....	18
2.4 Java API 文档.....	21
第三章 Java 基础语法.....	23
3.1 注释.....	23
3.2 分隔符、标识符和关键字.....	24
3.2.1 分隔符.....	24
3.2.2 标识符.....	25
3.2.3 关键字.....	26

3.3 数据类型.....	28
3.3.1 什么是数据类型.....	28
3.3.2 基本数据类型.....	29
3.3.3 引用数据类型.....	35
3.4 方法.....	40
3.4.1 什么是方法.....	40
3.4.2 形参、实参及返回值.....	41
3.4.3 方法的调用.....	42
3.5 变量.....	43
3.5.1 什么是变量.....	43
3.5.2 变量的分类、声明及初始化.....	43
3.5.3 变量的生存期、作用域和存储细节.....	44
3.6 值传递.....	48
3.7 关键字 this.....	50
3.8 标准输入/输出.....	52
3.8.1 什么是控制台.....	52
3.8.2 读取控制台输入.....	53
3.8.3 格式化输出.....	54
3.9 编码惯例.....	56
3.9.1 命名惯例.....	57
3.9.2 文件组织.....	57
3.9.3 分隔和缩进.....	58
3.9.4 声明语句.....	59
3.9.5 其他.....	60
<b>第四章 运算符、表达式和流程控制.....</b>	<b>61</b>
4.1 运算符.....	61
4.1.1 算术运算符.....	61
4.1.2 关系运算符.....	62
4.1.3 布尔运算符.....	63
4.1.4 位运算符.....	65
4.1.5 移位运算符.....	67
4.1.6 赋值运算符.....	68
4.1.7 其他运算符.....	69
4.2 表达式.....	72

---

4.2.1 什么是表达式.....	72
4.2.2 表达式中的数值类型转换.....	74
4.3 程序运行流程.....	76
4.3.1 顺序性结构.....	76
4.3.2 分支结构.....	77
4.3.3 循环结构.....	78
4.4 分支语句.....	78
4.4.1 if-else 语句.....	78
4.4.2 switch 语句.....	83
4.5 循环语句.....	87
4.5.1 for 循环.....	88
4.5.2 while 循环.....	91
4.5.3 do-while 循环.....	92
4.5.4 for-each 循环.....	93
4.5.5 循环的嵌套.....	94
4.5.6 流程控制语句 break 和 continue.....	95
4.6 递归.....	99
4.6.1 什么是递归.....	99
4.6.2 Fibnacy 数列问题.....	101
4.6.3 汉诺塔问题.....	103
<b>第五章 数组.....</b>	<b>105</b>
5.1 一维数组.....	105
5.1.1 数组的声明.....	105
5.1.2 数组对象的创建和使用.....	106
5.1.3 数组对象的 length 属性.....	109
5.1.4 数组的静态初始化.....	111
5.2 多维数组.....	112
5.3 数组拷贝.....	115
5.4 数组排序.....	118
<b>第六章 面向对象编程进阶.....</b>	<b>120</b>
6.1 包.....	120
6.1.1 什么是包.....	120
6.1.2 包的创建.....	121
6.1.3 导入包中的类.....	122

6.1.4 应用程序类打包.....	124
6.2 继承.....	125
6.2.1 什么是继承.....	125
6.2.2 单重继承和多重继承.....	128
6.2.3 类之间的关系.....	129
6.3 访问控制.....	130
6.4 方法重写.....	133
6.5 关键字 super.....	136
6.6 多态性.....	138
6.6.1 何为多态.....	138
6.6.2 虚方法调用.....	141
6.6.3 对象造型.....	143
6.6.4 instanceof 运算符.....	145
6.6.5 协变返回类型.....	146
6.7 方法重载.....	148
6.7.1 什么是方法重载.....	148
6.7.2 使用 this 调用其他重载构造方法.....	151
6.7.3 调用重载方法的多重匹配问题.....	153
6.8 深究对象构造和初始化.....	155
6.9 关键字 static.....	157
6.9.1 static 属性.....	157
6.9.2 static 方法.....	159
6.9.3 初始化块.....	162
6.9.4 静态导入.....	163
6.9.5 Singleton 设计模式.....	164
6.10 关键字 final.....	167
6.10.1 final 类和 final 方法.....	167
6.10.2 final 常量.....	169
<b>第七章 高级类特性.....</b>	<b>171</b>
7.1 抽象类.....	171
7.2 接口.....	174
7.2.1 什么是接口.....	174
7.2.2 接口的多重实现.....	176
7.2.3 接口的多重继承.....	179

---

7.3 嵌套类.....	180
7.3.1 嵌套类概述.....	180
7.3.2 内部类.....	181
7.3.3 局部内部类.....	185
7.3.4 匿名内部类.....	186
7.3.5 静态嵌套类.....	189
7.4 枚举类型.....	191
7.4.1 什么是枚举类型.....	191
7.4.2 组合使用枚举类型与 switch 语句.....	193
<b>第八章 异常处理.....</b>	<b>195</b>
8.1 异常的概念及分类.....	195
8.1.1 什么是异常.....	195
8.1.2 Java 异常分类及常见异常.....	196
8.2 Java 异常处理机制.....	200
8.2.1 捕获异常.....	200
8.2.2 声明抛弃异常.....	206
8.2.3 人工抛出异常.....	208
8.3 用户自定义异常.....	210
8.4 断言.....	211
8.4.1 启用和禁用断言.....	212
8.4.2 使用断言.....	212
<b>第九章 精通常用的 Java 类.....</b>	<b>215</b>
9.1 Java 类的共同父类 Object.....	215
9.1.1 hashCode() 方法.....	215
9.1.2 toString() 方法.....	216
9.1.3 equals() 方法.....	218
9.1.4 finalize() 方法.....	221
9.1.5 clone() 方法.....	223
9.2 字符串相关类型.....	228
9.2.1 String 类.....	228
9.2.2 StringBuffer 类.....	231
9.2.3 StringBuilder 类.....	232
9.2.4 StringTokenizer 类.....	232
9.3 封装类.....	234

---

9.3.1 封装类的概念和使用 .....	234
9.3.2 自动封装和拆封.....	237
9.4 日期相关类型 .....	239
9.4.1 Date 类.....	239
9.4.2 Calendar 类 .....	240
9.4.3 Locale 类.....	242
9.4.4 TimeZone 类.....	245
9.4.5 GregorianCalendar 类.....	247
9.4.6 DateFormat.....	248
9.4.7 SimpleDateFormat .....	250
9.5 数学相关类型 .....	253
9.5.1 基本数学功能类 Math.....	253
9.5.2 随机数生成器类 Random .....	254
9.5.3 大数值类型 BigInteger/BigDecimal .....	257
9.5.4 数据格式化工具类 NumberFormat/DecimalFormat .....	259
<b>第十章 控制台应用程序设计 .....</b>	<b>264</b>
10.1 命令行参数 .....	264
10.2 系统属性 .....	265
10.3 标准输入/输出.....	269
10.4 文件操作 .....	272
10.4.1 File 类 .....	272
10.4.2 文件 I/O.....	276
10.4.3 文件过滤.....	280
10.5 可变参数方法 .....	283
10.6 过时 API (Deprecation) .....	285
10.6.1 什么是过时 API.....	285
10.6.2 用户自定义过时 API.....	286
10.7 注解 (Annotation) .....	287
10.7.1 什么是注解.....	287
10.7.2 内置注解类型.....	288
10.8 归档工具 (jar) .....	290
10.8.1 jar 文件基本用法.....	290
10.8.2 发布 Java 应用程序.....	292
10.8.3 清单文件.....	294

---

第十一章 集合与映射 .....	296
11.1 集合框架概述 .....	296
11.2 Collection 及 Map 接口 .....	297
11.3 列表 (List) .....	299
11.3.1 ArrayList 类 .....	299
11.3.2 Vector 类 .....	301
11.3.3 Stack 类 .....	304
11.4 Iterator 接口 .....	305
11.5 集 (Set) .....	309
11.5.1 HashSet 类 .....	309
11.5.2 TreeSet 类 .....	310
11.5.3 Comparable 接口 .....	311
11.6 映射 (Map) .....	313
11.6.1 HashMap 类 .....	314
11.6.2 Hashtable 类 .....	316
11.6.3 TreeMap 类 .....	316
11.7 其他相关 API .....	319
11.7.1 Enumeration 接口 .....	319
11.7.2 Collections 类 .....	320
11.7.3 Arrays 类 .....	324
第十二章 GUI 程序设计 .....	326
12.1 Java GUI 设计 .....	326
12.1.1 抽象窗口工具集 AWT .....	326
12.1.2 组件和容器 .....	326
12.1.3 布局管理器 .....	331
12.2 GUI 事件处理 .....	341
12.2.1 Java 事件和事件处理机制 .....	341
12.2.2 多重监听器 .....	345
12.2.3 事件适配器 .....	349
12.2.4 内部类和匿名类在 GUI 事件处理中的应用 .....	352
12.3 AWT 常用组件和视觉控制 .....	356
12.3.1 按钮和标签 .....	356
12.3.2 文本组件 .....	356
12.3.3 菜单组件 .....	363

12.3.4 选择组件 .....	368
12.3.5 对话框组件 .....	372
12.3.6 滚动组件 .....	380
12.3.7 视觉控制类 .....	384
12.4 AWT 绘图 .....	386
12.4.1 绘制基本图形 .....	386
12.4.2 图像处理 .....	388
12.5 Applet .....	395
12.5.1 什么是 Applet .....	395
12.5.2 Applet 运行机制 .....	397
12.5.3 Applet 插件标记 .....	398
<b>第十三章 Swing 基础 .....</b>	<b>401</b>
13.1 Swing 概述 .....	401
13.2 Swing 典型组件 .....	403
13.2.1 JFrame .....	403
13.2.2 Swing 按钮、菜单和工具条 .....	406
13.2.3 标准对话框 .....	408
13.2.4 表格和树 .....	412
13.2.5 定时器 .....	416
<b>第十四章 高级 I/O 编程 .....</b>	<b>419</b>
14.1 Java I/O 原理 .....	419
14.2 基础 I/O 流类型 .....	420
14.2.1 InputStream .....	420
14.2.2 OutputStream .....	421
14.2.3 Reader .....	422
14.2.4 Writer .....	423
14.3 常用 I/O 流类型 .....	424
14.3.1 FileInputStream/FileOutputStream .....	424
14.3.2 FileReader/FileWriter .....	425
14.3.3 BufferedReader/BufferedWriter .....	426
14.3.4 InputStreamReader/OutputStreamWriter .....	427
14.3.5 PrintStream/PrintWriter .....	427
14.3.6 DataInputStream/DataOutputStream .....	428
14.3.7 CharArrayReader/CharArrayWriter .....	430

---

14.4 I/O 应用专题 .....	432
14.4.1 标准 I/O 重定向 .....	432
14.4.2 属性信息导入/导出 .....	436
14.4.3 随机存取文件 .....	437
14.4.4 临时文件 .....	440
14.5 对象序列化 .....	442
14.6 NIO .....	448
14.6.1 缓冲区 (Buffer) .....	448
14.6.2 通道 (Channel) .....	452
14.6.3 字符集转换 (Charset) .....	453
14.6.4 高级 NIO 技术 .....	458
<b>第十五章 泛 型 .....</b>	<b>467</b>
15.1 泛型简介 .....	467
15.2 使用泛型 .....	468
15.2.1 集合框架中的泛型 .....	468
15.2.2 泛型的向后兼容性 .....	470
15.3 泛型进阶 .....	471
15.3.1 类型参数 .....	471
15.3.2 类型通配符 .....	473
15.3.3 泛型方法 .....	477
15.3.4 受限制的类型参数 .....	479
<b>第十六章 线 程 .....</b>	<b>482</b>
16.1 线程基础 .....	482
16.1.1 什么是线程 .....	482
16.1.2 创建线程 .....	483
16.1.3 创建线程的第二种方式 .....	486
16.1.4 后台线程 .....	487
16.1.5 GUI 线程 .....	488
16.2 线程控制 .....	490
16.2.1 线程状态 .....	490
16.2.2 线程优先级 .....	491
16.2.3 线程串行化 .....	493
16.2.4 线程休眠 .....	494
16.2.5 线程让步 .....	497

---

16.2.6 线程挂起和恢复 .....	498
16.2.7 线程等待和通知 .....	502
16.3 线程的同步 .....	502
16.3.1 临界资源问题.....	502
16.3.2 互斥锁 .....	505
16.3.3 死锁 .....	506
16.3.4 线程同步通讯.....	508
16.3.5 生产者-消费者问题.....	509
16.4 多线程编程专题 .....	512
16.4.1 线程间数据传输 .....	512
16.4.2 类的同步性和线程安全.....	514
16.4.3 定时器 .....	517
<b>第十七章 Socket 网络编程.....</b>	<b>520</b>
17.1 网络基础 .....	520
17.1.1 什么是计算机网络 .....	520
17.1.2 网络通信协议.....	521
17.1.3 IP 地址、域名和端口号 .....	523
17.2 URL.....	525
17.3 Socket 编程 .....	527
17.4 NIO 非阻塞式 Socket 通信 .....	544
17.4.1 传统的阻塞式 I/O .....	544
17.4.2 非阻塞式 I/O .....	550
<b>第十八章 JDBC 编程 .....</b>	<b>561</b>
18.1 数据库简介 .....	561
18.1.1 关系型数据库.....	561
18.1.2 SQL 语言 .....	561
18.1.3 ODBC.....	561
18.2 JDBC 基本概念 .....	562
18.2.1 什么是 JDBC.....	562
18.2.2 JDBC 驱动程序 .....	563
18.2.3 JDBC URL.....	564
18.3 JDBC 编程.....	565
18.3.1 JDBC 编程基本步骤.....	565
18.3.2 执行 SQL 语句 .....	568

---

18.3.3 获取数据库元数据 .....	572
18.3.4 访问 SQL Server 数据库 .....	575
18.4 JDBC-ODBC 编程 .....	577
18.4.1 创建 ODBC 数据源 .....	577
18.4.2 连接并操作 ODBC 数据源 .....	579
18.4.3 访问 Access 数据库 .....	581
18.4.4 访问 SQL Server 数据库 .....	582
18.5 JDBC 编程进阶 .....	583
18.5.1 使用属性文件配置运行环境信息 .....	583
18.5.2 OCI 方式访问 Oracle 数据库 .....	585
18.5.3 可滚动和可更新结果集 .....	590
18.5.4 预处理语句 .....	599
18.5.5 调用存储过程 .....	602
18.5.6 事务处理 .....	603
18.5.7 批处理 .....	607
18.5.8 高级 SQL 类型 BLOB/CLOB .....	609
<b>第十九章 应用程序国际化 .....</b>	<b>615</b>
19.1 国际化和本地化简介 .....	615
19.2 数字、日期和时间国际化 .....	616
19.3 资源包 .....	618
19.3.1 属性文件 .....	618
19.3.2 资源绑定类 .....	622
19.4 消息格式化 .....	626
19.4.1 简单的消息格式化 .....	626
19.4.2 占位符类型和样式设置 .....	628
19.4.3 国际化程序中的消息格式化 .....	630
后记 .....	632



# 第一章 Java 技术入门

## 1.1 Java 技术概述

Java 语言自 1995 年 5 月由 Sun 公司发布以来，以极为迅猛的势头发展至今，现已不仅仅是一门高级编程语言，而是成为一种完备的技术体系和开发平台。Java 拥有卓越的技术特性、丰富的编程接口（类库）、多款功能强大的开发工具、全球 Java 程序员已超过 450 万、在企业级应用开发领域中占有过半的市场份额、超过 25 亿台设备正在使用 Java 技术并获得除微软外所有主流 IT 厂商的大力支持。从目前的应用现状和发展前景来看，Java 已经成为软件开发从业人员的首选技术。

### 1.1.1 Java 发展简史

Java 技术源于 Sun 公司早期一个名为“Green”的研究项目，该项目始于 1991 年，由 Patrick Naughton 及 James Gosling 主持（后者被称为“Java 语言之父”），目标是设计一种“轻型”的计算机语言以运行在联网的小型家电上，要求其运行开销足够小并能在不同厂商的处理器上运行（跨平台），以适应家电设备有限的处理速度和存储容量、尤其是处理器体系结构上的差异。

最初 Sun 公司将 Green 项目所设计出的语言命名为 Oak（后改名为 Java），并于 1992 年推出 Green 项目的第一款电子消费产品\*7——一种家庭娱乐设备，它带有触摸屏幕，并能够播放动画，其高度智能化的遥控功能就是采用 Oak 技术实现的。遗憾的是，家用电器厂商对这种跨平台的编程技术并不感兴趣，原因是消费者没有对家电联网控制这方面的需求。直到 20 世纪 90 年代中期，随着互联网技术的蓬勃发展，为数众多的联网计算机在硬件结构、软件性能方面存在着巨大的差异，能够同时在这些机器上运行的软件，即跨平台编程技术受到追捧，Java 终于守得云开见月明！

1995 年 5 月，Sun 公司在 SunWorld'95 大会上正式宣布 Java 技术诞生并发布了 HotJava 浏览器。“一次编写、随处运行”（write once, run anywhere, run anytime.）是当时人们对 Java 技术的第一印象。为数众多的开发者自此追随 Java 技术，IT 厂商则开始支持并申请 Java 技术使用许可，其中 Netscape 和 Oracle 公司分别于同年的 8 月和 10 月即获得 Sun 公司的 Java 许可证。随后，Sun 公司宣布成立新的业务部门 JavaSoft，该部门主要负责开发、销售和支持基于 Java 技术的产品。

1996 年 1 月，Sun 公司发布了 Java 的第 1 版。要使用一种编程语言进行应用开发，光有语言规范（Language Specification）是不够的，还必须有强大的开发工具和代码库，Sun 公司从 Java 的第 1 版开始提供并持续维护了完备的 Java 开发工具集（Java Development Kit, JDK）。Java 1.0 版本除了语言规范本身主要包含两部分——开发工具集（JDK）和运行环境（Java

Runtime Environment, JRE)。

1997年2月, Sun公司发布了Java 1.1版。相对于JDK1.0, JDK1.1引入了内部类和即时编译(Just-In-Time, JIT)技术。

1998年12月, Sun公司发布了一个里程碑式的版本Java 1.2。该版本中出现了许多革命性的变化, 这些变化一直沿用到现在并对Java发展产生了极为深远的影响。首先, 出于宣传推广的需要, 从1.2版本开始Sun公司将Java改名为有吸引力的Java 2(意为第二代的Java), 想来微软当年推出C++时也是出于类似的考虑; 其次, 原来的开发工具集(JDK)也更名为Java 2软件开发工具集(Java2 Software Development Kit, J2SDK), 后来5.0版以后又改回为JDK, 也许可算做返朴归真了吧。最主要的是, 从Java 1.2开始Sun公司将Java版本一分为三, 包括:

- ✧ 标准版(Java2 Standard Edition, J2SE)——开发的程序适用于PC机上运行;
- ✧ 微缩版(Java2 Micro Edition, J2ME)——适用于手机等嵌入式设备;
- ✧ 企业版(Java2 Enterprise Edition, J2EE)——适用于服务器端应用开发。

各版本都有自己的开发工具集, 本书只介绍其中的标准版。从JDK1.2开始, Sun公司大约每两年推出一个JDK的新版本。

除上述改进外, 标准版的J2SE1.2中还增加了功能强大的图形用户界面(Graphical User Interface, GUI)开发工具库Swing。另外Java 2还在多线程、集合类和非同步类上做了大量的改进。

2000年5月, Sun公司发布J2SE1.3版。此版本主要改进了类库和对本地资源的访问、支持XML、并采用了新的Hotspot虚拟机。

2002年2月, Sun公司发布J2SE1.4版。此版本进一步改进了Hotspot虚拟机的性能, 并引入了新的语言特性“断言”(Assert)。

2004年10月, Sun公司发布Java SE5.0版。此版本原被命名为1.5版, 后又将版本号改为5.0, 以说明其较以前版本的巨大改进。在Java SE5.0中增加了诸如泛型、for-each循环语句、可变数目参数、注解、自动装箱和拆箱等功能。

2006年11月, Sun公司发布Java SE6.0版(似乎Sun公司又打算将Java2重新称做Java)。Java SE6不仅在性能、易用性方面得到了前所未有的提高, 而且还提供了如脚本、全新的API(Swing和AWT等API已经被更新)的支持。而且Java SE6.0专为Vista做过针对性设计, 它在Vista上将会拥有更好的性能。在推出Java SE6.0的同时, Java SE7.0项目也已经启动。

### 1.1.2 Java 技术特性

#### 1. 简单性

Java技术功能强大但语法简洁, 它在基础语法规则的设计上尽量符合当前软件开发技术、主要是如C++的规范和惯例, 以使有任何编程基础的使用者易于上手, 同时剔除了其中不适用的部分, 如指针、人工分配和回收内存、无条件转移语句以及无节制的强制类型转换等。JDK中包含了丰富实用的应用程序编程接口(Application Programming Interface, API), 即预先准备好的、提供各种常用功能的代码模块, 在此基础上开发者可极大地提高开发效率。