

<<Java编程思想>>

图书基本信息

书名：<<Java编程思想>>

13位ISBN编号：9787111104414

10位ISBN编号：7111104412

出版时间：2002-9

出版时间：机械工业出版社

作者：[美] Bruce Eckel

页数：809

译者：侯捷

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

内容概要

本书的内容组织、讲授方法、选用示例和附带练习都别具特色。

作者根据多年教学实践中发现的问题，通过简练的示例和叙述，阐明了在学习Java中特别容易混淆的诸多概念。

与前一版相比，本书不但新增了Java2的语言特性，还根据语言的演变作出了彻底的更新，其中最主要的改变是第9章的群集。

本书被多家权威机构评为最佳Java书籍，适合作为各个层次的Java程序员阅读，也可以作为高等院校面向对象程序设计语言及Java语言的教材。

<<Java编程思想>>

作者简介

Bruce Eckel 《Thinking in C++》（C++编程思想）作者，该书曾获1995年Software Development Jolt Award最佳书籍大奖。

Eckel有20年专业编程经验，并自1986年起教育人们如何撰写面向对象程序，足迹遍及全球，成为一位知名的C++教师和顾问，如今兼涉Java。

他是C++标准委

<<Java编程思想>>

书籍目录

| |
|--|
| 出版者的话 III |
| 专家指导委员会 V |
| 读者回应 VI |
| 关于《Thinking in C++》 VIII |
| 题献 XIV |
| 译序 XV |
| Java环境设定 XIX |
| 综览 XXI |
| 序言 1 |
| 第2版序 3 |
| Java 2 4 |
| 书附光盘 5 |
| 简介 6 |
| 阅读门槛 6 |
| 学习Java 6 |
| 目标 7 |
| 在线说明文档 (Online documentation) 8 |
| 章节组织 8 |
| 习题 13 |
| 多媒体光盘 (Multimedia CD ROM) 13 |
| 源代码 (Source code) 14 |
| 编码标准 (Coding standards) 15 |
| Java版本 15 |
| 培训课程与顾问指导 16 |
| 关于错误 16 |
| 封面故事 16 |
| 致谢 17 |
| Internet上的贡献者 19 |
| 第1章 对象导论 20 |
| 抽象化的过程 20 |
| 每个对象都有接口 22 |
| 被隐藏的实现细节 24 |
| 重复运用实现码 25 |
| 继承：重复运用接口 26 |
| 是一个 (is-a) vs. 像是一个 (is-like-a) 29 |
| 随多态而生的可互换对象 30 |
| 抽象基类与接口 33 |
| 对象的形貌与寿命 33 |
| 群集和迭代器 34 |
| 单根继承体系 36 |
| 群集类库及其易用性支持 37 |
| 管家面临的两难：谁该负责清理？ 38 |
| 异常处理：面对错误的发生 39 |
| 多线程 39 |

<<Java编程思想>>

持久性 40
Java与Internet（网际网、互联网） 41
Web是什么？
41
客户端程序开发 42
服务端程序开发 47
另一个截然不同的战场：
应用系统 48
分析与设计 48
阶段0：策划 50
阶段1：建立什么？
51
阶段2：如何建立？
53
阶段3：打造核心 56
阶段4：use cases的迭代 57
阶段5：演化 58
取得成功 59
Extreme programming（XP） 59
测试优先 60
搭档设计 61
Java为什么成功 62
易于表达、易于理解的系统 62
通过程序库（libraries）发挥
最大杠杆效应 62
错误处理 62
大型程序设计 62
过渡策略 63
实践准则 63
管理上的障碍 64
Java vs. C++? 66
摘要 67
第2章 万事万物皆对象 69
Reference是操控对象之钥匙 69
所有对象都必须由你建立 70
存储在哪里 70
特例：基本型别（primitive types） 71
Java中的数组（array） 73
你再也不需要摧毁对象 73
生存空间（Scoping） 74
对象的生存空间 74
建立新的数据型别：class 75
数据成员（fields）和
函数（methods） 76
函数（methods），引数（arguments），
返回值（return values） 77
引数列（The argument list） 78

<<Java编程思想>>

打造一个 Java 程序 79
名称的可视性 (Name visibility) 79
使用其他组件 (Component) 80
关键字static 81
初试啼声：你的第一个 Java程序 82
编译与执行
 (Compiling and running) 84
注解及内嵌式文档 85
寓文档于注解 85
语法 86
内嵌的 HTML 86
@see：参考其他classes 87
Class (类) 文档所用的标签 87
Variable (变量) 文档所用的标签 88
Method (函数) 文档所用的标签 88
文档制作示例 89
编码风格 (Coding style) 90
摘要 90
练习 90
第3章 控制程序流程 92
使用Java运算符 (operators) 92
优先序 (Precedence) 92
赋值、指派 (Assignment) 93
数学运算符
 (Mathematical operators) 95
递增 (increment) 和递减
 (decrement) 97
关系运算符 (Relational operators) 98
逻辑运算符 (Logical operators) 99
位运算符 (Bitwise operators) 102
位移运算符 (Shift operators) 102
if-else 三元运算符 (Ternary if-else operator) 106
逗号运算符 (comma operator) 107
应用于String身上的operator + 107
使用运算符时的常犯错误 107
转型运算符 (Casting operators) 108
Java 没有 “ sizeof ” 运算符 110
再探优先序 (Precedence) 111
运算符综合说明 111
流程控制 120
true 和 false 120
if-else 120
return 121
迭代 (iteration) 121
do-while 122
for 122
break 和 continue 124

<<Java编程思想>>

switch 129
摘要 133
练习 133
第4章 初始化和清理 134
以构造函数 (constructor) 确保
初始化的进行 134
函数重载 (method overloading) 136
区分重载函数 138
搭配基本型别 (Primitives)
进行重载 138
以返回值 (return value) 作为重载
的基准 142
Default构造函数 142
关键字this 143
清理 (Cleanup) : 终结 (finalization)
与垃圾回收 (garbage collection) 146
finalize() 存在是为了什么 ?
147
你必须执行清理 (cleanup) 动作 147
死亡条件 (The death condition) 150
垃圾回收器 (garbage collector)
的运作方式 152
成员初始化 (Member initialization) 154
指定初值 156
以构造函数 (Constructor) 进行
初始化动作 157
Array 的初始化 163
多维 (Multidimensional) arrays 167
摘要 169
练习 170
第5章 隐藏实现细目 172
package: 程序库单元 (Library unit) 172
独一无二的package 命名 174
自订一个程序库 177
利用 imports 来改变行为 178
使用package时的一些忠告 180
Java访问权限饰词 (access specifiers) 180
“ Friendly ” (友善的) 180
public: 接口访问 (interface access) 181
private: 不要碰我 !
183
protected: 几分友善 184
Interface (接口) 与implementation
(实现) 185
Class的访问权限 186
摘要 189
练习 190

<<Java编程思想>>

第6章 重复运用Classes 192
组合 (Composition) 语法 192
继承 (Inheritance) 语法 195
base class的初始化 197
兼容组合 (composition) 及继承 (inheritance) 199
保证适当清理 200
名称遮蔽 (Name hiding) 203
组合与继承之间的抉择 204
protected (受保护的) 205
渐进式开发
(Incremental derelopment) 206
向上转型 (Upcasting) 207
为什么需要向上转型
(Why "upcasting") ?
207
关键字final 208
Final data 209
Final methods 212
Final classes 214
最后的告诫 215
初始化以及class的装载 215
继承与初始化 216
摘要 217
练习 218
第7章 多态 220
再探向上转型 (Upcasting) 220
将对象的型别忘掉 221
窍门 223
Method-call (函数调用) 绑定方式 223
产生正确的行为 224
扩充性 (Extensibility) 226
覆写 (overriding) vs. 重载
(overloading) 229
Abstract classes (抽象类) 和abstract methods (抽象函数) 230
构造函数 (Constructors) 和
多态 (polymorphism) 234
构造函数调用顺序
(order of constructor calls) 234
继承与finalize() 236
polymorphic methods
在构造函数中的行为 239
将继承 (inheritance) 运用于设计 241
纯粹继承 (Pure inheritance)
vs. 扩充 (extension) 242
向下转型 (downcasting) 与执行期型别
辨识 (run-time type identification) 243
摘要 245

<<Java编程思想>>

练习 246

第8章 接口与内隐类 248

Interfaces (接口) 248

Java的多重继承

(multiple inheritance in Java) 251

通过继承来扩充interface 254

产生常量群 (grouping constants) 255

将interfaces内的数据成员初始化 256

嵌套的 (nesting) interfaces 257

Inner classes (内隐类) 260

Inner classes (内隐类) 与

upcasting (向上转型) 261

位于methods 和scopes之内的

inner classes (内隐类) 263

匿名的

(anonymous) inner classes 265

与外围 (outer) class的连接关系 268

static inner classes (静态内隐类) 270

取用 (referring) outer class的对象 272

从多层嵌套class向外伸展触角 273

继承inner classes 273

inner classes可被覆写

(overridden) 吗 ?

274

Inner class的标识符 (identifiers) 276

为什么需要inner classes ?

277

Inner classes和control frameworks 281

摘要 287

练习 287

第9章 持有你的对象 291

Arrays (数组) 291

Arrays是第一级对象

(first-class objects) 292

返回一个array 295

Arrays class 297

array的充填 (filling) 306

array的复制 307

arrays的比较 308

array元素的比较 309

array的排序 312

在已排序的array中进行查找 313

array总结 315

容器 (containers) 简介 315

容器的打印 316

容器的充填 317

容器的缺点 : 元素型别未定 323

<<Java编程思想>>

有时候它总是可以运作 325
制作一个具有型别意识 (type-conscious)
的ArrayList 326
迭代器 (Iterators) 327
容器分类学 (Container taxonomy) 330
Collection的机能 333
List的机能 335
根据LinkedList制作一个stack 338
根据LinkedList制作一个queue 339
Set的机能 340
SortedSet 342
Map的机能 343
SortedMap 346
Hashing和hash codes 347
覆写hashCode() 354
持有references 356
WeakHashMap 358
再论Iterators (迭代器) 360
选择一份适当的实现品 361
在各种Lists之间抉择 361
在各种Sets之间抉择 364
在各种Maps之间抉择 366
Lists的排序和查找 368
公用函数 (Utilities) 369
让Collection或Map无法被更改 369
Collection或Map的同步控制 370
未获支持的操作
(Unsupported operations) 371
Java 1.0/1.1的容器 373
Vector和Enumeration 374
Hashtable 375
Stack 375
BitSet 376
摘要 377
练习 378
第10章 通过异常处理错误 382
基本异常 383
异常引数 (Exception arguments) 383
异常的捕捉 (Catching) 384
try block 384
异常处理函数 (Exception handlers) 385
编写你自己的异常类 386
异常规格 (exception specification) 389
捕捉所有异常 390
重掷 (Rethrowing) 异常 392
Java标准异常 395
RuntimeException的特殊情况 395

<<Java编程思想>>

以finally进行清理 (cleanup) 397
为什么需要finally ?
398
缺憾：异常遗失 (the lose exception) 400
异常的局限性 (Exception restrictions) 401
构造函数 (Constructor) 404
异常的比对 (matching) 407
异常的使用原则
(Exception guidelines) 408
摘要 409
练习 409
第11章 Java I/O 系统 412
File class 412
目录列示器 (a directory lister) 412
目录的检查和建立 416
输入 (Input) 和输出 (Output) 418
InputStream的类型 418
OutputStream的类型 419
附加属性 (attributes) 和有用接口 420
通过FilterInputStream自InputStream
读取数据 421
通过FilterOutputStream将数据写入OutputStream 421
Readers和Writers 422
数据的来源 (sources) 和
去处 (sinks) 423
改变stream的行为 424
未曾有任何变化的classes 425
RandomAccessFile 425
I/O streams的典型运用 426
Input streams 428
Output streams 430
这是个臭虫吗 ?
431
管线化的 (piped) streams 432
标准I/O 432
读取标准输入 432
将System.out转换为PrintWriter 433
标准I/O重导向 (Redirecting) 434
压缩 435
运用GZIP进行单纯压缩 435
运用Zip存储多份文件数据 436
Java ARchives (JARs) 438
对象次第读写 (Object serialization) 440
找出class 444
控制次第读写 (serialization) 445
使用对象持久机制 (persistence) 453
被语汇单元化 (Tokenizing) 的

<<Java编程思想>>

输入动作 459
StreamTokenizer 459
StringTokenizer 462
检验大小写 464
摘要 471
练习 472
第12章 执行期型别辨识 474
为什么需要RTTI 474
Class 对象 476
转型之前先检查 478
RTTI语法 485
Reflection (映像) : 执行期的类信息
(run-time class information) 487
实现一个函数提取器 (extractor) 489
摘要 493
练习 494
第13章 制作窗口和applet 496
基本的applet 498
applet的束缚 (restrictions) 498
applet的优点 498
应用程序框架
(Application frameworks) 499
在Web浏览器上执行applets 500
运用Appletviewer 501
测试applets 502
从命令行 (Command line)
执行applets 503
一个显示框架
(Adisplay framework) 504
运用Windows Explorer 507
制作一个按钮 507
捕捉一个事件 (Capturing an event) 508
文本区 (Text areas) 511
控制版面布局 (Controlling layout) 512
BorderLayout 513
FlowLayout 514
GridLayout 514
GridBagLayout 515
绝对定位 515
BoxLayout 515
最好的方法是什么?
519
Swing的事件模型 (event model) 519
事件 (event) 及监听器 (listener)
的种类 520
追踪多种事件 525
Swing组件一览 528

<<Java编程思想>>

按钮 (Buttons) 528
图示 (Icons) 531
工具提示 (Tool tips) 533
文本块 (Text fields) 533
边框 (Borders) 535
JScrollPane 536
迷你文本编辑器 (mini-editor) 538
复选框 (Check boxes) 539
单选按钮 (Radio buttons) 540
组合框 Combo boxes (下拉式列表
drop-down lists) 541
列表框 (List boxes) 542
页签式嵌板 (Tabbed panes) 544
消息框 (Message boxes) 545
菜单 (Menus) 547
弹出式菜单 (Pop-up menus) 552
绘图 (Drawing) 553
对话框 (Dialog Boxes) 556
文件对话框 (File dialogs) 560
Swing 组件上的HTML 562
滑块 (Sliders) 和进度指示器
(progress bars) 562
树状组件 (Trees) 563
表格 (Tables) 566
选择外观风格
(Selecting Look & Feel) 567
剪贴板 (clipboard) 569
将applet封装于JAR文件 572
编程技术 572
动态绑定事件
(Binding events dynamically) 573
将business logic和用户界面
logic隔离 574
标准型式 577
视觉化程序设计 (Visual programming)
与Beans 577
什么是Bean?
578
运用Introspector提取出BeanInfo 580
一个更为复杂精巧的Bean 585
Bean的包装 (Packaging a Bean) 588
Bean所支持的更复杂功能 590
其他 590
摘要 591
练习 591
第14章 多线程 594
反应灵敏的用户界面 594

<<Java编程思想>>

继承自Thread 596
运用线程打造出反应灵敏的
用户界面 598
结合线程和程序主类 600
产生多个线程 602
Daemon线程 604
共享有限资源 606
不当的资源访问 606
Java如何共享资源 610
再探JavaBeans 614
停滞（阻塞，Blocking） 618
转为停滞状态（Becoming blocked） 619
死锁（Deadlock） 628
优先权（Priorities） 632
优先权的读取和设定 632
线程群组（Thread groups） 636
再探Runnable 642
过多的线程 645
摘要 647
练习 649
第15章 分布式计算 651
网络编程 652
机器的识别 652
Sockets（插座） 655
服务多个客户 660
数据元（Datagrams） 665
在applet中使用URL 665
更多的网络相关信息 667
JDBC，Java数据库连接机制 668
让本节示例正常运作 670
一个GUI版的查询程序 673
为什么JDBC API看起来如此复杂 676
一个更复杂的示例 676
Servlets 683
Servlet基本教练 684
Servlets和多线程（multithreading） 687
以servlets处理sessions 688
执行servlet示例程序 691
Java Server Pages（JSP） 692
隐式对象（Implicit objects） 693
JSP指令（directives） 693
JSP的脚本描述成分
（scripting elements） 694
取出字段（fields）和数值（values） 696
JSP的页面属性（page attributes）和
有效范围（scope） 697
处理JSP中的sessions 698

<<Java编程思想>>

产生并修改cookies 699
JSP摘要 700
RMI (Remote Method Invocation ,
远端函数调用) 701
远端接口 (Remote interfaces) 701
实现出远端接口 (remote interface) 702
产生stubs和skeletons 704
使用远端对象 (remote object) 705
CORBA 706
CORBA的基本原理 706
一个示例 708
Java Applets和CORBA 712
CORBA vs. RMI 712
Enterprise (企业级) JavaBeans 713
JavaBeans vs. EJBs 714
EJB规格 714
EJB组件 714
EJB组件成分 715
EJB的各项操作 716
EJBs的类型 717
分发 (Developing) EJB 718
EJB 摘要 722
Jini: 分布式服务 (distributed serrices) 722
Jini的来龙去脉 722
什么是Jini ?
723
Jini如何运作 723
discovery动作 724
join动作 724
lookup动作 725
接口和实现的分离 725
将分布式系统抽象化 726
摘要 726
练习 727
附录A 对象的传递和返回 729
reference的传递 729
别名 (Aliasing) 730
制作一个区域性副本 (local copies) 731
Pass by value (传值) 732
对象的克隆 (Cloning objects) 733
赋予class克隆能力 (cloneability) 734
成功的克隆 736
Object.clone() 的效应 738
克隆一个组合对象
(composed object) 739
对ArrayList进行深层拷贝 741
通过serialization进行深层拷贝 743

<<Java编程思想>>

将克隆能力加到继承体系的更下层 745

为什么要有这种奇怪的设计？

746

克隆能力 (cloneability) 的控制 746

copy构造函数 750

唯读类 (Read-only classes) 754

编写一个唯读类 (read-only classes) 755

恒常性 (immutability) 的缺点 756

恒常不变的Strings 758

String和StringBuffer 760

Strings是特殊的东西 763

摘要 763

练习 764

附录B Java原生接口 765

原生函数 (Native method) 的调用 765

头文件产生器 : javah 766

名称重整 (Name mangling) 与函数

标记 (function signatures) 767

实现出你自己的DLL 767

取用JNI函数: 通过JNIEnv引数 768

访问Java Strings 769

传递和运用Java对象 769

JNI和Java异常 771

JNI和多线程 (threading) 772

使用既有的程序代码 772

补充信息 773

附录C Java编程准则 774

设计 774

实现 (Implementation) 778

附录D 资源 783

软件 783

书籍 783

分析& 设计 784

Python 786

我的著作 786

索引 788

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>