

*Библиотека профессионала*

# Java™ 2

Том 1. Основы

ВОСЬМОЕ ИЗДАНИЕ

# Core Java

## Volume I – Fundamentals

Eighth Edition

Cay S. Horstmann  
Gary Cornell



Sun Microsystems Press

Upper Saddle River, NJ • Boston • Indianapolis • San Francisco  
New York • Toronto • Montreal • London • Munich • Paris • Madrid  
Capetown • Sydney • Tokyo • Singapore • Mexico City

*Библиотека профессионала*

# Java™ 2

## Том 1. Основы

ВОСЬМОЕ ИЗДАНИЕ

Кей Хорстманн • Гари Корнелл



Москва • Санкт-Петербург • Киев  
2013

ББК 32.973.26-018.2.75

X82

УДК 681.3.07

Издательский дом “Вильямс”

Зав. редакцией *С.Н. Тригуб*

Перевод с английского *Н.А. Мухина*

Под редакцией *Ю.Н. Артеменко*

По общим вопросам обращайтесь в Издательский дом “Вильямс” по адресу:  
info@williamspublishing.com, http://www.williamspublishing.com

**Хорстманн, Кей С., Корнелл, Гари.**

X82 Java 2. Библиотека профессионала, том 1. Основы. 8-е издание. : Пер. с англ. — М. : ООО “И.Д. Вильямс”, 2013. — 816 с. : ил. — Парал. тит. англ.

ISBN 978-5-8459-1378-4 (рус.)

Книга ведущих специалистов по программированию на языке Java представляет собой обновленное издание фундаментального труда, учитывающее всю специфику новой версии платформы Java SE 6. Подробно рассматриваются такие темы, как организация и настройка среды программирования на Java, основные структуры данных, объектно-ориентированное программирование и его реализация в Java, интерфейсы, программирование графики, обработка событий, Swing, развертывание приложений и апплетов, отладка, обобщенное программирование, коллекции и построение многопоточных приложений. Книга изобилует множеством примеров, которые не только иллюстрируют концепции, но также демонстрируют способы правильной разработки, применяемые в реальных условиях.

Книга рассчитана на программистов разной квалификации, а также будет полезна студентам и преподавателям дисциплин, связанных с программированием на языке Java.

**ББК 32.973.26-018.2.75**

Все названия программных продуктов являются зарегистрированными торговыми марками соответствующих фирм.

Никакая часть настоящего издания ни в каких целях не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами, будь то электронные или механические, включая фотокопирование и запись на магнитный носитель, если на это нет письменного разрешения издательства Prentice Hall, Inc.

Authorized translation from the English language edition published by Prentice Hall, Ptr., Copyright © 2008 Sun Microsystems, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Russian language edition published by Williams Publishing House according to the Agreement with R&I Enterprises International, Copyright © 2013

ISBN 978-5-8459-1378-4 (рус.)

ISBN 978-0-1323-5476-9 (англ.)

© Издательский дом “Вильямс”, 2013

© Sun Microsystems, Inc., 2008

# Оглавление

<b>Предисловие</b>	16
<b>Глава 1. Введение в язык Java</b>	21
<b>Глава 2. Среда программирования Java</b>	36
<b>Глава 3. Основные конструкции языка Java</b>	56
<b>Глава 4. Объекты и классы</b>	125
<b>Глава 5. Наследование</b>	189
<b>Глава 6. Интерфейсы и вложенные классы</b>	259
<b>Глава 7. Программирование графики</b>	299
<b>Глава 8. Обработка событий</b>	339
<b>Глава 9. Компоненты пользовательского интерфейса Swing</b>	375
<b>Глава 10. Развертывание приложений и апплетов</b>	502
<b>Глава 11. Исключения, протоколирование, утверждения и отладка</b>	558
<b>Глава 12. Обобщенное программирование</b>	619
<b>Глава 13. Коллекции</b>	652
<b>Глава 14. Многопоточность</b>	716
<b>Приложение. Ключевые слова Java</b>	805
<b>Предметный указатель</b>	807

# Содержание

Об авторах	14
Благодарности	14
<b>Предисловие</b>	16
Краткий тур по книге	17
Соглашения	19
Коды примеров	20
От издательства	20
<b>Глава 1. Введение в язык Java</b>	21
Программная платформа Java	22
Характерные особенности языка Java	22
Простота	23
Поддержка объектов	24
Поддержка распределенных вычислений	24
Надежность	24
Безопасность	25
Независимость от архитектуры компьютера	26
Переносимость	26
Интерпретируемость	27
Производительность	27
Многопоточность	28
Динамичность	28
Апплеты Java и Internet	28
Краткая история языка Java	30
Распространенные заблуждения относительно языка Java	32
<b>Глава 2. Среда программирования Java</b>	36
Инсталляция Java Development Kit	37
Где найти JDK	37
Установка пути к исполняемым файлам	39
Инсталляция библиотек и документации	41
Установка примеров программ	42
Навигация по каталогам	42
Выбор среды разработки программ	43
Использование инструментов командной строки	43
Возможные ошибки	45
Использование интегрированной среды разработки	46
Сообщения об ошибках компиляции	48
Выполнение графического приложения	50
Создание и запуск апплетов	52
<b>Глава 3. Основные конструкции языка Java</b>	56
Простая программа на языке Java	57
Комментарии	60
Типы данных	61
Целые числа	61
Числа с плавающей точкой	62
Тип char	63

Тип <code>boolean</code>	65
Переменные	65
Инициализация переменных	66
Константы	67
Операции	68
Операции инкрементирования и декрементирования	69
Операции отношений и логические операции	69
Битовые операции	70
Математические функции и константы	71
Преобразование числовых типов	72
Приведение числовых типов	73
Скобки и иерархия операций	73
Перечислимые типы	74
Строки	75
Подстроки	75
Конкатенация	75
Неизменность строк	76
Проверка эквивалентности строк	77
Кодовые точки и кодовые единицы	78
API класса <code>String</code>	79
Интерактивная документация по API	81
Построение строк	84
Ввод и вывод	85
Чтение входных данных	85
Форматирование выходных данных	87
Файловый ввод и вывод	92
Поток управления	94
Блоки	94
Условные выражения	95
Неопределенные циклы	98
Определенные циклы	101
Многовариантное ветвление – оператор <code>switch</code>	104
Прерывание потока управления	106
Большие числа	109
Массивы	111
Цикл “ <code>for each</code> ”	112
Инициализация массивов и анонимные массивы	113
Копирование массивов	114
Параметры командной строки	115
Сортировка массива	116
Многомерные массивы	119
“Зубчатые” массивы	122
<b>Глава 4. Объекты и классы</b>	125
Введение в объектно-ориентированное программирование	126
Классы	127
Объекты	128
Идентификация классов	128
Отношения между классами	129
Использование готовых классов	131
Объекты и объектные переменные	131

## 8 Содержание

---

Класс <code>GregorianCalendar</code> из библиотеки <code>Java</code>	134
Модифицирующие методы и методы доступа	136
Определение собственных классов	142
Класс <code>Employee</code>	142
Использование нескольких исходных файлов	145
Анализ класса <code>Employee</code>	146
Первые шаги с конструкторами	146
Явные и неявные параметры	147
Преимущества инкапсуляции	148
Доступ к данным из различных экземпляров класса	150
Приватные методы	151
Неизменяемые поля экземпляра	152
Статические поля и методы	152
Статические поля	152
Статические константы	153
Статические методы	154
Методы-фабрики	155
Метод <code>main()</code>	155
Параметры методов	158
Конструирование объектов	163
Перегрузка	164
Инициализация полей по умолчанию	164
Конструктор по умолчанию	165
Явная инициализация полей	165
Имена параметров	166
Вызов одного конструктора из другого	167
Инициализационные блоки	167
Уничтожение объекта и метод <code>finalize()</code>	172
Пакеты	172
Импортирование классов	173
Статический импорт	174
Добавление классов в пакеты	175
Область видимости пакета	177
Путь поиска классов	179
Установка пути к классам	181
Комментарии и документирование	181
Включение комментариев	182
Комментарии к классу	182
Комментарии к методам	183
Комментарии к полям	184
Комментарии общего характера	184
Пакетные и обзорные комментарии	185
Извлечение комментариев	185
Рекомендации по проектированию классов	186
<b>Глава 5. Наследование</b>	189
Классы, суперклассы и подклассы	190
Иерархия наследования	196
Полиморфизм	197
Динамическое связывание	198



Предотвращение наследования: финальные классы и методы	201
Приведение типов	202
Абстрактные классы	204
Защищенный доступ	209
Object: глобальный суперкласс	210
Метод equals ()	210
Проверка эквивалентности объектов и наследование	211
Метод hashCode ()	214
Метод toString ()	216
Обобщенные массивы-списки	222
Доступ к элементам списочных массивов	224
Совместимость между типизированными и “сырыми” списочными массивами	228
Объектные оболочки и автоупаковка	229
Методы с переменным числом параметров	232
Классы-перечисления	233
Рефлексия	235
Класс Class	236
Основы обработки исключений	238
Использование рефлексии для анализа свойств классов	239
Использование рефлексии для анализа объектов во время выполнения	244
Использование рефлексии для написания кода обобщенного массива	249
Указатели на методы!	253
Рекомендации по применению наследования	256
<b>Глава 6. Интерфейсы и вложенные классы</b>	259
Интерфейсы	260
Свойства интерфейсов	265
Интерфейсы и абстрактные классы	267
Клонирование объектов	268
Интерфейсы и обратный вызов	274
Вложенные классы	276
Использование вложенных классов для доступа к состоянию объекта	278
Специальные синтаксические правила для работы с вложенными классами	281
Чем полезны вложенные классы, нужны ли они и насколько они безопасны	282
Локальные вложенные классы	285
Доступ к финальным переменным из внешних методов	285
Анонимные вложенные классы	288
Статические вложенные классы	290
Прокси-классы	293
Свойства прокси-классов	297
<b>Глава 7. Программирование графики</b>	299
Общие сведения о пакете Swing	300
Создание фрейма	304
Позиционирование фрейма	306
Определение подходящего размера фрейма	308
Отображение информации в компоненте	312
Двумерные фигуры	316
Работа с цветом	324
Использование шрифтов при выводе текста	327
Вывод графических изображений	335

<b>Глава 8. Обработка событий</b>	339
Общие сведения об обработке событий	340
Пример: обработка щелчка на кнопке	342
Удобство вложенных классов	346
Создание слушателей, содержащих единственный вызов метода	349
Пример: изменение визуального стиля	350
Классы-адаптеры	353
Действия	357
События мыши	364
Иерархия событий AWT	372
Семантические и низкоуровневые события в библиотеке AWT	373
<b>Глава 9. Компоненты пользовательского интерфейса Swing</b>	375
Swing и шаблон проектирования “Модель–представление–контроллер”	376
Шаблоны проектирования	376
Шаблон “Модель-представление-контроллер”	377
Кнопки Swing с точки зрения шаблона “Модель–представление–контроллер”	381
Общие сведения о компоновке элементов	382
Диспетчер компоновки BorderLayout	384
Диспетчер компоновки GridLayout	386
Ввод текста	390
Текстовые поля	391
Метки и маркирующие компоненты	393
Поля для ввода пароля	394
Текстовые области	395
Панели прокрутки	395
Компоненты для выбора вариантов	398
Флажки опций	398
Переключатели	401
Рамки	405
Выпадающие списки	409
Линейные регуляторы	412
Меню	418
Построение меню	419
Пиктограммы в пунктах меню	421
Флажки опций и переключатели в меню	422
Всплывающие меню	423
Клавиши быстрого доступа и клавиатурные сокращения	425
Блокирование и разблокирование пунктов меню	427
Панели инструментов	431
Всплывающие подсказки	433
Расширенные средства компоновки	435
Диспетчер компоновки GridBagLayout	437
Альтернативные способы указания параметров gridx, gridy, gridwidth и gridheight	440
Диспетчер компоновки GroupLayout	447
Компоновка без использования диспетчера	456
Диспетчер компоновки, определяемый разработчиком	457
Порядок обхода компонентов	461
Диалоговые окна	463
Диалоговые окна для выбора опций	463

Создание диалоговых окон	473
Передача данных	478
Файловые диалоги	485
Диалоговое окно для выбора цвета	496
<b>Глава 10. Развертывание приложений и апплетов</b>	<b>502</b>
Файлы JAR	503
Файл манифеста	504
Исполняемые JAR-файлы	505
Ресурсы	506
Герметизация пакета	509
Технология Java Web Start	510
“Песочница”	513
Подписанный код	514
JNLP API	516
Апплеты	524
Простой апплет	525
HTML-дескрипторы и атрибуты для работы с апплетами	529
Дескриптор <code>&lt;object&gt;</code>	532
Передача информации апплетам через параметры	532
Обращение к файлам изображений и звуковым файлам	538
Контекст апплета	539
Хранение параметров настройки	547
Карты свойств	547
Интерфейс Preferences API	552
<b>Глава 11. Исключения, протоколирование, утверждения и отладка</b>	<b>558</b>
Обработка ошибок	559
Классификация исключений	561
Объявление контролируемых исключений	562
Как сгенерировать исключение	564
Создание классов исключений	565
Перехват исключений	566
Перехват нескольких исключений	568
Повторная генерация исключения	569
Блок <code>finally</code>	570
Анализ элементов трассировки стека	573
Как использовать исключения	575
Использование диагностических утверждений	578
Включение и отключение диагностических утверждений	579
Использование диагностических утверждений для проверки параметров	580
Использование утверждений для документирования предположений	581
Протоколирование	582
Базовое протоколирование	583
Расширенное протоколирование	583
Настройка диспетчера протоколирования	585
Локализация	586
Обработчики	587
Фильтры	591
Средства форматирования	591
“Рецепт” протоколирования	591
Приемы отладки	599

## 12 Содержание

---

Использование консольного окна	605
Трассировка событий AWT	606
Позволим роботу AWT выполнять его работу	610
Применение отладчика	614
<b>Глава 12. Обобщенное программирование</b>	<b>619</b>
Назначение обобщенного программирования	620
Кто хочет быть обобщенным программистом?	621
Определение простого обобщенного класса	622
Обобщенные методы	624
Ограничения переменных типов	625
Обобщенный код и виртуальная машина	627
Трансляция обобщенных выражений	628
Трансляция обобщенных методов	629
Вызов унаследованного кода	630
Ограничения и лимиты	632
Параметры типа не могут принимать примитивные типы	632
Исследование типов времени выполнения работает только с “сырыми” типами	632
Генерировать и перехватывать исключения-экземпляры обобщенного класса нельзя	632
Массивы параметризованных типов не разрешены	633
Нельзя создавать экземпляры переменных типов	633
Переменные типа в статическом контексте обобщенных классов не разрешены	635
Остерегайтесь конфликтов после подчистки	635
Правила наследования обобщенных типов	636
Подстановочные типы	638
Ограничения супертипа для подстановок	639
Неограниченные подстановки	641
Захват подстановок	642
Рефлексия и обобщения	645
Использование параметров <code>Class&lt;T&gt;</code> для сравнения типов	646
Информация об обобщенных типах в виртуальной машине	647
<b>Глава 13. Коллекции</b>	<b>652</b>
Интерфейсы коллекций	653
Отделение интерфейса от реализации коллекций	653
Интерфейсы <code>Collection</code> и <code>Iterator</code> из библиотеки Java	656
Конкретные коллекции	661
Специализированные классы <code>Set</code> и <code>Map</code>	686
Каркас коллекций	690
Представления и “оболочки”	694
Пакетные операции	700
Преобразования между коллекциями и массивами	701
Алгоритмы	702
Сортировка и перетасовка	703
Бинарный поиск	705
Простые алгоритмы	706
Написание собственных алгоритмов	708
Унаследованные коллекции	709
Класс <code>Hashtable</code>	709
Интерфейс <code>Enumeration</code>	709
Карты свойств	710

Стеки	711
Битовые наборы	711
<b>Глава 14. Многопоточность</b>	<b>716</b>
Что такое потоки?	717
Использование потоков для предоставления шанса другим задачам	722
Прерывание потоков	727
Состояния потока	730
Новые потоки	730
Работоспособные потоки	730
Блокированные и ожидающие потоки	731
Завершенные потоки	731
Свойства потока	733
Приоритеты потоков	733
Потоки-демоны	734
Обработчики неперехваченных исключений	734
Синхронизация	736
Пример состояния гонок	736
Объяснение состояния гонок	740
Блокирующие объекты	742
Объекты условий	745
Ключевое слово <code>synchronized</code>	749
Синхронизированные блоки	753
Концепция монитора	754
Поля <code>volatile</code>	755
Взаимные блокировки	756
Тестирование блокировок и таймауты	759
Блокировки чтения/записи	760
Почему методы <code>stop()</code> и <code>suspend()</code> не рекомендованы к применению	761
Блокирующие очереди	763
Коллекции, безопасные к потокам	770
Эффективные карты, наборы и очереди	770
Копирующиеся при записи массивы	772
Устаревшие безопасные к потокам коллекции	772
Интерфейсы <code>Callable</code> и <code>Future</code>	773
Класс <code>Executors</code>	777
Пулы потоков	778
Плановое выполнение	782
Управление группами задач	782
Синхронизаторы	784
Семафоры	784
Защелки с обратным отсчетом	785
Барьеры	785
Обменники	786
Синхронные очереди	786
Пример: приостановка и возобновление анимации	786
Потоки и <code>Swing</code>	792
Выполнение задач, требующих много времени	793
Использование класса <code>SwingWorker</code>	797
Правило единственного потока	803
<b>Приложение. Ключевые слова Java</b>	<b>805</b>
<b>Предметный указатель</b>	<b>807</b>

## Об авторах

**Кей Хорстманн** — профессор факультета вычислительной техники в Университете Сан-Хосе, обладатель звания “Чемпион по Java” и частый докладчик на многих отраслевых конференциях. Является соавтором книги *JavaServer Faces. Библиотека профессионала* (ИД “Вильямс”, 2008 г.).

**Гари Корнелл** имеет более чем 20-летний опыт обучения и написания книг по программированию. Является одним из основателей издательства Apress. Он написал множество книг-бестселлеров для профессионалов, попал в финал конкурса Jolt Award, а также стал обладателем звания “Выбор читателей” в журнале *Visual Basic Magazine*.

## Благодарности

Написание книги всегда требует титанических усилий, причем переписывание не намного легче, особенно учитывая постоянные изменения технологии Java. Чтобы сделать книгу полезной, необходимы совместные усилия многих преданных делу людей, и мы с удовольствием выражаем признательность всей команде за их вклад в наше общее дело.

Огромное количество людей из Prentice Hall и Sun Microsystems Press оказали неоценимую помощь, хотя и остались в тени. Я хотел бы выразить им свою признательность за их усилия. Как всегда, мою самую горячую благодарность заслуживает редактор Prentice Hall Грег Доенч (Greg Doench) — за сопровождение книги на протяжении всего процесса ее написания и издания, а также за то, что позволил мне пребывать в блаженном неведении о многих деталях этого процесса, происходящих за сценой. Я благодарен Ванессе Мур (Vanessa Moore) за блестящую продуктивную поддержку. Также приношу свои благодарности моему соавтору по ранним изданиям Гари Корнеллу (Gary Cornell), который с тех пор перешел на другое предприятие.

Благодарю многих и многих читателей ранних изданий, которые сообщали о найденных ошибках и внесли массу ценных предложений по улучшению книги. Я особенно благодарен блестящей команде рецензентов, которые тщательно прошлись по рукописи и избавили ее от многих неприятных ошибок.

Среди рецензентов этого и предыдущих изданий хочу отметить Чака Аллисона (Chuck Allison, выпускающий редактор *C/C++ Users Journal*), Алека Битона (Alec Beaton, PointBase, Inc.), Клиффа Берга (Cliff Berg, iSavvix Corporation), Джошуа Блоха (Joshua Bloch, Sun Microsystems), Дэвида Брауна (David Brown), Корки Картрайта (Corky Cartwright), Френка Коена (Frank Cohen, PushToTest), Криса Крейна (Chris Crane, devXsolution), доктора Николаса Дж. Де Лилло (Dr. Nicholas J. De Lillo, Manhattan College), Ракеша Дхупара (Rakesh Dhoopar, Oracle), David Geary (David Geary, Sabreware), Брайана Гоеца (Brian Goetz, ведущий консультант в Quiotix Corp.), Анжелу Гордон (Angela Gordon, Sun Microsystems), Дэна Гордона (Dan Gordon, Sun Microsystems), Роба Гордона (Rob Gordon), Джона Грэя (John Gray, University of Hartford), Камерон Грегори (Cameron Gregory, olabs.com), Марти Холл (Marty Hall, The Johns Hopkins University Applied Physics Lab), Винсента Харди (Vincent Hardy, Sun Microsystems), Дэна Харки (Dan Harkey, San Jose State University), Вильяма Хиггинса (William Higgins, IBM), Владимира Ивановича (Vladimir Ivanovic, PointBase), Джерри Джексона (Jerry Jackson, ChannelPoint Software), Тима Киммета (Tim Kimmet, Preview

Systems), Криса Лаффра (Chris Laffra), Чарли Лаи (Charlie Lai, Sun Microsystems), Анжелику Лангер (Angelika Langer), Дуга Лэнгстона (Doug Langston), Ханг Лау (Hang Lau, McGill University), Марка Лоуренса (Mark Lawrence), Дуга Ли (Doug Lea, SUNY Oswego), Грегори Лонгшора (Gregory Longshore), Боба Линча (Bob Lynch, Lynch Associates), Филиппа Милна (Philip Milne, консультант), Марка Моррисси (Mark Morrissey, The Oregon Graduate Institute), Махеш Нилаканта (Mahesh Neelakanta, Florida Atlantic University), Хао Фам (Hao Pham), Пола Филиона (Paul Phillion), Блейка Рагсдейла (Blake Ragsdell), Стюарта Реджеса (Stuart Reges, University of Arizona), Рича Росена (Rich Rosen, Interactive Data Corporation), Питера Сандерса (Peter Sanders, ESSI University, Ницца, Франция), доктора Пола Сангеру (Dr. Paul Sanghera, San Jose State University and Brooks College), Поля Сэвинка (Paul Sevinc, Teamup AG), Деванг Ша (Devang Shah, Sun Microsystems), Бредли А. Смита (Bradley A. Smith), Стивена Стелтинга (Steven Stelting, Sun Microsystems), Кристофера Тэйлора (Christopher Taylor), Люка Тэйлора (Luke Taylor, Valtech), Джорджа Тхируватукала (George Thiruvathukal), Кима Топли (Kim Topley, автор Core JFC), Джанет Трауб (Janet Traub), Пола Тиму (Paul Тума констультант), Питера Ван Дер Линдена (Peter van der Linden, Sun Microsystems) и Берта Велша (Burt Walsh).

*Кей Хорстманн (Kay Horstmann)  
Сан-Франциско, 2007 г.*

# ПРЕДИСЛОВИЕ

**В** конце 1995 г. язык программирования Java вырвался на Internet-сцену и моментально завоевал популярность. Технология Java обещала стать *универсальным клеем*, соединяющим пользователей с информацией, откуда бы эта информация ни поступала — от Web-серверов, из баз данных, поставщиков информации, либо любого другого источника, который только можно было себе вообразить. И на самом деле, Java обладает всеми возможностями соответствовать этим обещаниям. Это исключительно основательно спроектированный язык, который получил признания всех основных участников рынка за исключением Microsoft. Его встроенные средства защиты и безопасности обнадежили как программистов, так и пользователей Java-программ. Java изначально обладала встроенной поддержкой таких сложных задач, как сетевое программирование, подключаемость к базам данных и многопоточность, обеспечивая простоту реализации всех этих вещей.

С 1995 г. Sun Microsystems выпустила семь главных версий Java Development Kit. За последние 11 лет интерфейс прикладного программирования (API) вырос с 200 до 3000 классов. Теперь API охватывает такие различные области, как конструирование пользовательских интерфейсов, управление базами данных, интернационализация, безопасность и обработка XML.

Книга, которую вы держите в руках — это первый том восьмого издания. С выходом каждого издания мы старались как можно быстрее следовать очередному выпуску Java Development Kit, каждый раз переписывая ее, чтобы вы могли воспользоваться преимуществами новейших средств Java. Настоящее издание отражает средства, имеющиеся в составе Java Standard Edition (SE) 6.

Как и все предыдущие издания этой книги, мы по-прежнему *ориентируем ее на аудиторию серьезных программистов*, которые хотят использовать Java для разработки реальных проектов. Мы воспринимаем вас, дорогой читатель, как программиста с солидным опытом в языках программирования, отличных от Java, и надеемся, что вы не любите книг, полных игрушечных примеров (вроде программ управления тостерами или животными в зоопарке либо “прыгающим текстом”). Ничего подобного в этой книге вы не найдете. Наша цель — достижение полного понимания языка Java и его библиотеки, а не создание иллюзии такого понимания.

В книге вы найдете массу примеров кода, демонстрирующих почти все обсуждаемые языковые и библиотечные средства. Эти примеры мы намеренно сделали насколько возможно простыми, чтобы сосредоточиться на основных моментах, но, тем не менее, в большинстве своем, они вовсе не игрушечные и не “срезают острых углов”. Все они могут стать хорошей отправной точкой для разработки вашего собственного кода.

Мы предполагаем, что вы хотите, а то и жаждете узнать обо всех расширенных средствах, которые предлагает Java вашему вниманию. Так, например, мы предлагаем детальное рассмотрение следующих тем.

- Объектно-ориентированное программирование.
- Рефлексия и прокси-классы.
- Интерфейсы и вложенные классы.



- Модель слушателей событий.
- Проектирование графического интерфейса пользователя с использованием инструментария Swing.
- Обработка исключений.
- Обобщенное программирование.
- Каркас коллекций.
- Параллелизм.

В связи со стремительным ростом библиотеки классов Java, одного тома оказалось недостаточно для описания всех средств Java, о которых следует знать серьезным программистам. Первый том, который вы держите в руках, сосредоточен на фундаментальных концепциях языка Java, наряду с основами программирования пользовательского интерфейса. Второй том посвящен средствам разработки приложений масштаба предприятия и усовершенствованному программированию пользовательских интерфейсов. В нем вы найдете детальное обсуждение следующих тем.

- Файлы и потоки.
- Распределенные объекты.
- Базы данных.
- Расширенные компоненты графического пользовательского интерфейса.
- “Родные” (native) методы.
- Обработка XML.
- Сетевое программирование.
- Расширенная графика.
- Интернационализация.
- JavaBeans.
- Аннотации.

В этом издании мы перегруппировали содержимое двух томов. В частности, многопоточность теперь рассматривается в первом томе, поскольку ее значение существенно возросло.

При написании книги ошибки и неточности неизбежны. Мы очень заинтересованы в том, чтобы знать о них. Но, конечно же, мы бы предпочли узнавать о каждой из них только однажды. Поэтому мы поместили список часто задаваемых вопросов, исправлений ошибок и “обходных путей” по адресу <http://horstmann.com/corejava>.

## Краткий тур по книге

**Глава 1** предоставляет обзор возможностей Java, которые отличают его от других языков программирования. Мы объясним, что было задумано проектировщиками языка, и в какой мере им удалось воплотить задуманное. Затем приведем краткую историю того, как развивался язык Java, и как он стал тем, чем он есть сегодня.

В **главе 2** мы расскажем о том, как загрузить и установить JDK, а также примеры программ для книги. Затем мы проведем вас через компиляцию и запуск трех типичных программ Java — консольного приложения, графического приложения и апплета — все это с использованием только JDK, специального Java-ориентированного текстового редактора и Java IDE.

В **главе 3** начинается дискуссия о языке программирования Java. В этой главе мы раскроем основы: переменные, циклы и простые функции. Если вы — программист на C или C++, для вас все должно пройти гладко, поскольку синтаксис этих языковых средств по существу очень похож. Если вы обладаете опытом в языке, не похожем на C, таком как Visual Basic, вам следует прочесть эту главу особенно внимательно.

Объектно-ориентированное программирование (ООП) в наши дни — доминирующая программистская практика, и Java — в полной мере объектно-ориентированный язык. В **главе 4** представлена инкапсуляция — первый из двух фундаментальных строительных блоков объектной ориентации, а также механизмы Java, реализующие ее — классы и методы. В дополнение к правилам языка Java мы также приводим советы по правильному объектно-ориентированному дизайну. И, наконец, мы опишем изумительный инструмент `javadoc`, который форматирует комментарии из вашего кода в набор Web-страниц с перекрестными ссылками. Если вы знакомы с C++, то можете лишь бегло просмотреть эту главу. Программисты, пришедшие из мира не объектно-ориентированных языков, должны потратить больше времени для овладения концепциями ООП, прежде чем дальше изучать Java.

Классы и инкапсуляция — это лишь часть идеологии ООП, и потому **глава 5** представляет еще один ее краеугольный камень — *наследование*. Наследование позволяет взять существующий класс и модифицировать его в соответствии с вашими потребностями. Это — фундаментальная техника в программировании на Java. Механизм наследования в Java довольно похож на тот же механизм в C++. Опять-таки, программисты на C++ могут сосредоточиться лишь на отличиях между языками.

В **главе 6** будет показано, как использовать в Java понятие *интерфейса*. Интерфейсы позволяют вам выйти за пределы простого наследования, описанного в главе 5. Овладение интерфейсами позволит вам получить полный доступ ко всей мощи объектно-ориентированного подхода Java к программированию. Мы также раскроем удобное техническое средство Java, называемое *встроенными классами*. Встроенные классы помогают сделать ваш код янее и более согласованным.

С **главы 7** начинается серьезное прикладное программирование. Каждый программист Java должен знать хоть немного о программировании графических пользовательских интерфейсов, и здесь вы найдете его основы. Мы покажем, как создаются окна, как выполняется рисование в них, как рисуются геометрические фигуры, как форматируется текст с применением множества шрифтов и как отображаются графические изображения.

**Глава 8** содержит детальную дискуссию о модели событий AWT — *абстрактного оконного инструментария*. Вы увидите, как писать код, реагирующий на такие события, как щелчки кнопками мыши или нажатия клавиш. По ходу дела вы ознакомитесь с обработкой базовых элементов графического пользовательского интерфейса, таких как кнопки и панели.

**Глава 9** посвящена более детальному обсуждению инструментария Swing. Пакет инструментов Swing позволяет строить межплатформенный графический интерфейс пользователя. Вы познакомитесь с различными типами кнопок, текстовых компонентов, рамок, ползунков, окон списков, меню и диалоговых окон. Однако некоторые из наиболее совершенных компонентов мы отложим до второго тома.

В **главе 10** будет показано, как разворачивать ваши программы — будь то приложения или апплеты. Мы опишем, как пакетировать программы в файлы JAR, и как доставлять приложения через Internet с применением Java Web Start и механизмов апплетов. И, наконец, мы объясним, как программы Java могут сохранять и извлекать конфигурационную информацию после своего разворачивания.

**Глава 11** посвящена *обработке исключений* — устойчивому механизму Java, предназначенному противостоять тому факту, что плохие вещи случаются и с хорошими программами. Исключения обеспечивают эффективный способ отделения кода нормальной обработки от кода обработки ошибок. Конечно, даже после укрепления вашей программы проверкой всех возможных исключительных ситуаций, может произойти сбой в ожидаемой работе. Во второй половине этой главы мы предоставим огромное количество полезных советов по организации отладки. И, наконец, приведем пример полного сценария отладки.

В **главе 12** представлен обзор *обобщенного программирования* — главного преимущества Java SE 5.0. Обобщенное программирование делает ваши программы легче читаемыми и более безопасными. Мы покажем, как используется строгая типизация, и исключаются необходимость в некрасивых и небезопасных приведениях типа, а также как справиться со сложностями, происходящими из-за необходимости сохранения совместимости с предыдущими версиями Java.

Тема **главы 13** — каркас коллекций платформы Java. Всякий раз, когда вы хотите собирать множество объектов и извлекать их позднее, вам понадобится использовать коллекцию, которая наилучшим образом подходит к вашим конкретным условиям, вместо того, чтобы сбрасывать их в обычный массив. В этой главе будет продемонстрировано, какие преимущества дают стандартные коллекции, которые заранее подготовлены для вас.

**Глава 14** завершает книгу обсуждением многопоточности, которая позволяет программировать выполняемые параллельно задачи. (Под потоком понимается поток управления внутри программы.) Мы покажем, как создаются потоки и как осуществляется их синхронизация. Средства многопоточности существенно изменились после Java SE 5.0, и мы расскажем вам о новых ее механизмах.

В **приложении** вы найдете зарезервированные слова языка Java.

## Соглашения

Как это принято во многих компьютерных книгах, моноширинный шрифт используется для представления исходного кода.

**На заметку!**

Такой пиктограммой выделяются замечания.

---

**Совет.**

Такой пиктограммой выделяются советы.

---

**Внимание!**

Такой пиктограммой выделяются предупреждения об опасности.

---



**На заметку!** В нашей книге есть множество замечаний, касающихся C++, в которых объясняются отличия между языками Java и C++. Вы можете пропустить их, если не имеете опыта работы с C++ или если склонны воспринимать этот опыт как страшный сон, который лучше забыть.

---

**API** **Интерфейс прикладного программирования**

Язык Java сопровождается огромной библиотекой интерфейса прикладного программирования (API). Используя вызов API впервые, в конце соответствующего раздела мы добавляем краткое описание, сопровождаемое такой пиктограммой. Эти описания не слишком информативны, но, как мы надеемся, более информативны, чем те, что представлены в официальной онлайн-документации по API. Мы снабжаем каждую врезку по API номером версии, в которой средство впервые было представлено, чтобы помочь тем читателям, которые не используют самую последнюю версию Java.

Программы с доступным исходным кодом организованы в виде примеров, как показано ниже.

**Листинг 1.1. Содержимое файла `WelcomeApplet.java`**

---

## Коды примеров

Коды примеров, рассмотренных в книге, доступны как на Web-сайте, посвященном книге (<http://horstmann.com/corejava>), так и на Web-сайте издательства.

## От издательства

Вы, читатель этой книги, и есть главный ее критик и комментатор. Мы ценим ваше мнение и хотим знать, что было сделано нами правильно, что можно было сделать лучше и что еще вы хотели бы увидеть изданным нами. Нам интересно услышать и любые другие замечания, которые вам хотелось бы высказать в наш адрес.

Мы ждем ваших комментариев и надеемся на них. Вы можете прислать нам бумажное или электронное письмо, либо просто посетить наш Web-сервер и оставить свои замечания там. Одним словом, любым удобным для вас способом дайте нам знать, нравится или нет вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас.

Посылая письмо или сообщение, не забудьте указать название книги и ее авторов, а также ваш обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию последующих книг.

Наши координаты:

E-mail: [info@williamspublishing.com](mailto:info@williamspublishing.com)

WWW: <http://www.williamspublishing.com>

Информация для писем из:

России: 127055, Москва, ул. Лесная, д. 43, стр. 1

Украины: 03150, Киев, а/я 152