

Java

Полное руководство

12-е издание

Всестороннее раскрытие языка Java

Полностью обновленное и расширенное издание



Герберт Шилдт



Mc
Graw
Hill
Education

D ДИДЛЕКЛИКА

**Полное
руководство**

Java
12-е издание

The Complete Reference

JavaTM
Twelfth Edition

Herbert Schildt



New York Chicago San Francisco
Athens London Madrid Mexico City
Milan New Delhi Singapore Sydney Toronto

Полное руководство

Java

12-е издание

Герберт Шилдт



Москва • Санкт-Петербург
2022

ББК 32.973.26-018.2.75

Ш57

УДК 004.432

ООО “Диалектика”

Перевод с английского и редакция *Ю.Н. Артеменко*

По общим вопросам обращайтесь в издательство “Диалектика” по адресу:
info.dialektika@gmail.com, <http://www.dialektika.com>

Шилдт, Герберт.

Ш57 Java. Полное руководство, 12-е изд. : Пер. с англ. — СПб. : ООО “Диалектика”, 2023. — 1344 с. : ил. — Парал. тит. англ.

ISBN 978-5-907458-86-4 (рус.)

ББК 32.973.26-018.2.75

Все права защищены.

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 McGraw Hill.

Copyright © 2022 by McGraw Hill.

All rights reserved.

All trademarks are trademarks of their respective owners.

Oracle Corporation does not make any representations or warranties as to the accuracy, adequacy, or completeness of any information contained in this Work, and is not responsible for any errors or omissions.

Authorized translation from the English language edition of the *Java: The Complete Reference*, 12th Edition (ISBN 978-1-26-046341-5), published by McGraw Hill.

Except as permitted under the United States Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

Научно-популярное издание

Герберт Шилдт
Java. Полное руководство
12-е издание

Подписано в печать 12.09.2022. Формат 70×100/16

Усл. печ. л. 108,36. Уч.-изд. л. 68,2

Тираж 500 экз. Заказ № 7327

Отпечатано в АО “Первая Образцовая типография”

Филиал “Чеховский Печатный Двор”

142300, Московская область, г. Чехов, ул. Полиграфистов, д. 1

Сайт: www.chpd.ru, E-mail: sales@chpd.ru, тел. 8 (499) 270-73-59

ООО “Диалектика”, 195027, Санкт-Петербург, Магнитогорская ул., д. 30, лит. А, пом. 848

ISBN 978-5-907458-86-4 (рус.)

© ООО “Диалектика”, 2022,

перевод, оформление, макетирование

ISBN 978-1-26-046341-5 (англ.)

© 2022 by McGraw Hill

Оглавление

Предисловие	29
Часть I. Язык Java	33
Глава 1. История и эволюция языка Java	34
Глава 2. Краткий обзор языка Java	58
Глава 3. Типы данных, переменные и массивы	80
Глава 4. Операции	110
Глава 5. Управляющие операторы	131
Глава 6. Введение в классы	162
Глава 7. Подробный анализ методов и классов	183
Глава 8. Наследование	217
Глава 9. Пакеты и интерфейсы	245
Глава 10. Обработка исключений	274
Глава 11. Многопоточное программирование	296
Глава 12. Перечисления, автоупаковка и аннотации	328
Глава 13. Ввод-вывод, оператор <code>try</code> с ресурсами и другие темы	367
Глава 14. Обобщения	401
Глава 15. Лямбда-выражения	444
Глава 16. Модули	473
Глава 17. Выражения <code>switch</code> , записи и прочие недавно добавленные средства	503

Часть II. Библиотека Java	539
Глава 18. Обработка строк	540
Глава 19. Исследование пакета <code>java.lang</code>	569
Глава 20. Пакет <code>java.util</code> , часть 1: Collections Framework	648
Глава 21. Пакет <code>java.util</code> , часть 2: дополнительные служебные классы	743
Глава 22. Ввод-вывод: исследование пакета <code>java.io</code>	813
Глава 23. Исследование системы NIO	868
Глава 24. Работа в сети	907
Глава 25. Обработка событий	934
Глава 26. Введение в AWT: работа с окнами, графикой и текстом	970
Глава 27. Использование элементов управления, диспетчеров компоновки и меню AWT	1001
Глава 28. Изображения	1053
Глава 29. Утилиты параллелизма	1079
Глава 30. Поточковый API-интерфейс	1134
Глава 31. Регулярные выражения и другие пакеты	1160
Часть III. Введение в программирование графических пользовательских интерфейсов с помощью Swing Java	1189
Глава 32. Введение в Swing	1190
Глава 33. Исследование Swing	1210
Глава 34. Введение в меню Swing	1240
Часть IV. Применение Java	1275
Глава 35. Архитектура JavaBeans	1276
Глава 36. Введение в сервлеты	1289
Часть V. Приложения	1315
Приложение А. Использование документирующих комментариев Java	1316
Приложение Б. Введение в JShell	1325
Приложение В. Компиляция и запуск простых однофайловых программ за один шаг	1336
Предметный указатель	1338

Содержание

Предисловие	29
Часть I. Язык Java	33
Глава 1. История и эволюция языка Java	34
Происхождение Java	34
Зарождение современного программирования: язык C	35
C++: следующий шаг	37
Условия для появления языка Java	38
Создание языка Java	38
Связь с языком C#	41
Влияние языка Java на Интернет	41
Апплеты Java	41
Безопасность	42
Переносимость	42
Магия Java: байт-код	43
Выход за рамки апплетов	44
Более быстрый график выпуска	45
Сервлеты: Java на серверной стороне	46
Терминология языка Java	47
Простота	47
Объектная ориентация	47
Надежность	48
Многопоточность	49
Нейтральность к архитектуре	49
Интерпретируемость и высокая производительность	49
Распределенность	49
Динамичность	50
Эволюция языка Java	50
Культура инноваций	57
Глава 2. Краткий обзор языка Java	58
Объектно-ориентированное программирование	58
Две парадигмы	58
Абстракция	59
Три принципа ООП	60
Первая простая программа	66
Ввод кода программы	66
Компиляция программы	67
Подробный анализ первого примера программы	68
Вторая простая программа	70
Два управляющих оператора	72
Оператор if	72
Цикл for	73

Использование блоков кода	75
Лексические вопросы	76
Пробельные символы	76
Идентификаторы	76
Литералы	77
Комментарии	77
Разделители	77
Ключевые слова Java	77
Библиотеки классов Java	79
Глава 3. Типы данных, переменные и массивы	80
Java — строго типизированный язык	80
Примитивные типы	80
Целые числа	81
Тип byte	82
Тип short	82
Тип int	82
Тип long	83
Типы с плавающей точкой	83
Тип float	84
Тип double	84
Символы	85
Булевские значения	86
Подробный анализ литералов	87
Целочисленные литералы	87
Литералы с плавающей точкой	88
Булевские литералы	89
Символьные литералы	89
Строковые литералы	90
Переменные	91
Объявление переменной	91
Динамическая инициализация	92
Область видимости и время жизни переменных	92
Преобразование и приведение типов	95
Автоматические преобразования в Java	95
Приведение несовместимых типов	95
Автоматическое повышение типов в выражениях	97
Правила повышения типов	98
Массивы	99
Одномерные массивы	99
Многомерные массивы	101
Альтернативный синтаксис объявления массивов	105
Знакомство с выводением типов локальных переменных	106
Некоторые ограничения var	108
Несколько слов о строках	109
Глава 4. Операции	110
Арифметические операции	110
Основные арифметические операции	111
Операция деления по модулю	112
Составные арифметические операции присваивания	112
Операции инкремента и декремента	113

Побитовые операции	115
Побитовые логические операции	116
Сдвиг влево	119
Сдвиг вправо	120
Беззнаковый сдвиг вправо	121
Составные побитовые операции присваивания	123
Операции отношения	124
Булевские логические операции	125
Короткозамкнутые логические операции	126
Операция присваивания	127
Операция ?	128
Старшинство операций	129
Использование круглых скобок	130
Глава 5. Управляющие операторы	131
Операторы выбора Java	131
Оператор <code>if</code>	131
Традиционный оператор <code>switch</code>	134
Операторы итерации	140
Цикл <code>while</code>	140
Цикл <code>do-while</code>	141
Цикл <code>for</code>	144
Версия цикла <code>for</code> в стиле “for-each”	148
Выведение типов локальных переменных в цикле <code>for</code>	153
Вложенные циклы	154
Операторы перехода	155
Использование оператора <code>break</code>	155
Использование оператора <code>continue</code>	159
Оператор <code>return</code>	161
Глава 6. Введение в классы	162
Основы классов	162
Общая форма класса	162
Простой класс	163
Объявление объектов	166
Подробный анализ операции <code>new</code>	166
Присваивание для переменных ссылок на объекты	168
Введение в методы	169
Добавление метода в класс <code>Box</code>	169
Возвращение значения	171
Добавление метода, принимающего параметры	173
Конструкторы	175
Параметризованные конструкторы	177
Ключевое слово <code>this</code>	178
Скрытие переменных экземпляра	178
Сборка мусора	179
Класс <code>Stack</code>	180
Глава 7. Подробный анализ методов и классов	183
Перегрузка методов	183
Перегрузка конструкторов	186
Использование объектов в качестве параметров	188

Подробный анализ передачи аргументов	190
Возвращение объектов	192
Рекурсия	193
Введение в управление доступом	195
Ключевое слово <code>static</code>	199
Ключевое слово <code>final</code>	201
Снова о массивах	201
Вложенные и внутренние классы	203
Исследование класса <code>String</code>	206
Использование аргументов командной строки	208
Аргументы переменной длины	209
Перегрузка методов с аргументами переменной длины	212
Аргументы переменной длины и неоднозначность	214
Выведение типов локальных переменных для ссылочных типов	215
Глава 8. Наследование	217
Основы наследования	217
Доступ к членам и наследование	219
Более реалистичный пример	220
Переменная типа суперкласса может ссылаться на объект подкласса	222
Использование ключевого слова <code>super</code>	223
Использование ключевого слова <code>super</code>	
для вызова конструкторов суперкласса	223
Использование второй формы ключевого слова <code>super</code>	226
Создание многоуровневой иерархии	227
Когда конструкторы выполняются	230
Переопределение методов	231
Динамическая диспетчеризация методов	233
Зачем нужны переопределенные методы?	235
Применение переопределения методов	236
Использование абстрактных классов	237
Использование ключевого слова <code>final</code>	
с наследованием	240
Использование ключевого слова <code>final</code>	
для предотвращения переопределения	240
Использование ключевого слова <code>final</code> для предотвращения наследования	241
Выведение типов локальных переменных и наследование	241
Класс <code>Object</code>	243
Глава 9. Пакеты и интерфейсы	245
Пакеты	245
Определение пакета	246
Поиск пакетов и <code>CLASSPATH</code>	247
Краткий пример пакета	247
Пакеты и доступ к членам классов	248
Пример, демонстрирующий использование модификаторов доступа	250
Импортирование пакетов	252
Интерфейсы	254
Определение интерфейса	255
Реализация интерфейсов	256
Вложенные интерфейсы	259

Применение интерфейсов	260
Переменные в интерфейсах	263
Интерфейсы можно расширять	265
Стандартные методы интерфейса	266
Основы стандартных методов	267
Более реалистичный пример	269
Проблемы множественного наследования	269
Использование статических методов в интерфейсе	271
Закрытые методы интерфейса	271
Заключительные соображения по поводу пакетов и интерфейсов	273
Глава 10. Обработка исключений	274
Основы обработки исключений	274
Типы исключений	275
Неперехваченные исключения	276
Использование try и catch	277
Отображение описания исключения	279
Использование нескольких конструкций catch	279
Вложенные операторы try	281
Оператор throw	283
Конструкция throws	284
Конструкция finally	286
Встроенные исключения Java	287
Создание собственных подклассов Exception	289
Сцепленные исключения	292
Три дополнительных средства в системе исключений	294
Использование исключений	295
Глава 11. Многопоточное программирование	296
Потоковая модель Java	297
Приоритеты потоков	298
Синхронизация	299
Обмен сообщениями	300
Класс Thread и интерфейс Runnable	300
Главный поток	301
Создание потока	303
Реализация интерфейса Runnable	303
Расширение класса Thread	305
Выбор подхода	306
Создание множества потоков	306
Использование isAlive() и join()	308
Приоритеты потоков	310
Синхронизация	311
Использование синхронизированных методов	312
Оператор synchronized	314
Взаимодействие между потоками	316
Взаимоблокировка	320
Приостановка, возобновление и останов потоков	322
Получение состояния потока	325
Использование фабричных методов для создания и запуска потока	326
Использование многопоточности	327

Глава 12. Перечисления, автоупаковка и аннотации	328
Перечисления	328
Основы перечислений	329
Методы <code>values()</code> и <code>valueOf()</code>	331
Перечисления Java являются типами классов	332
Перечисления унаследованы от <code>Enum</code>	334
Еще один пример перечисления	336
Оболочки типов	337
Класс <code>Character</code>	338
Класс <code>Boolean</code>	338
Оболочки числовых типов	339
Автоупаковка	341
Автоупаковка и методы	342
Автоупаковка/автораспаковка и выражения	342
Автоупаковка/автораспаковка типов <code>Boolean</code> и <code>Character</code>	344
Автоупаковка/автораспаковка помогает предотвратить ошибки	345
Предостережение	346
Аннотации	346
Основы аннотаций	346
Указание политики хранения	347
Получение аннотаций во время выполнения с использованием рефлексии	348
Интерфейс <code>AnnotatedElement</code>	353
Использование стандартных значений	354
Маркерные аннотации	355
Одноэлементные аннотации	356
Встроенные аннотации	357
Аннотации типов	360
Повторяющиеся аннотации	364
Некоторые ограничения	366
Глава 13. Ввод-вывод, оператор <code>true</code> с ресурсами и другие темы	367
Основы ввода-вывода	367
Потоки данных	368
Потоки байтовых и символьных данных	368
Предопределенные потоки данных	371
Чтение консольного ввода	372
Чтение символов	373
Чтение строк	374
Запись консольного вывода	376
Класс <code>PrintWriter</code>	376
Чтение файлов и запись в файлы	378
Автоматическое закрытие файла	384
Модификаторы <code>transient</code> и <code>volatile</code>	388
Введение в <code>instanceof</code>	388
Модификатор <code>strictfp</code>	391
Собственные методы	391
Использование <code>assert</code>	392
Параметры включения и отключения проверки утверждений	395
Статическое импортирование	395
Вызов перегруженных конструкторов через <code>this()</code>	398
Несколько слов о классах, основанных на значениях	400

Глава 14. Обобщения	401
Что такое обобщения?	402
Простой пример обобщения	402
Обобщения работают только со ссылочными типами	406
Обобщенные типы различаются на основе их аргументов типов	407
Каким образом обобщения улучшают безопасность в отношении типов?	407
Обобщенный класс с двумя параметрами типов	409
Общая форма обобщенного класса	411
Ограниченные типы	411
Использование аргументов с подстановочными знаками	413
Ограниченные аргументы с подстановочными знаками	416
Создание обобщенного метода	421
Обобщенные конструкторы	424
Обобщенные интерфейсы	424
Низкоуровневые типы и унаследованный код	427
Иерархии обобщенных классов	429
Использование обобщенного суперкласса	429
Обобщенный подкласс	431
Сравнение типов в обобщенной иерархии во время выполнения	432
Приведение	434
Переопределение методов в обобщенном классе	435
Выведение типов и обобщения	436
Выведение типов локальных переменных и обобщения	437
Стирание	438
Мостовые методы	438
Ошибки неоднозначности	440
Некоторые ограничения обобщений	441
Невозможность создать экземпляры параметров типов	441
Ограничения, касающиеся статических членов	442
Ограничения, касающиеся обобщенных массивов	442
Ограничения, касающиеся обобщенных исключений	443
Глава 15. Лямбда-выражения	444
Введение в лямбда-выражения	444
Основы лямбда-выражений	445
Функциональные интерфейсы	446
Примеры лямбда-выражений	447
Блочные лямбда-выражения	451
Обобщенные функциональные интерфейсы	453
Передача лямбда-выражений в качестве аргументов	454
Лямбда-выражения и исключения	457
Лямбда-выражения и захват переменных	458
Ссылки на методы	459
Ссылки на статические методы	459
Ссылки на методы экземпляра	461
Ссылки на методы и обобщения	464
Ссылки на конструкторы	467
Предопределенные функциональные интерфейсы	471

Глава 16. Модули	473
Основы модулей	473
Простой пример модуля	474
Компиляция и запуск первого примера модуля	478
Более подробный анализ операторов <code>requires</code> и <code>exports</code>	480
Модуль <code>java.base</code> и модули платформы	481
Унаследованный код и неименованные модули	482
Экспортирование в конкретный модуль	483
Использование <code>requires transitive</code>	485
Использование служб	489
Основы служб и поставщиков служб	489
Ключевые слова, связанные со службами	490
Пример службы, основанной на модулях	491
Графы модулей	497
Три специальных характерных черты модулей	498
Открытые модули	498
Оператор <code>opens</code>	499
Оператор <code>requires static</code>	499
Введение в <code>jlink</code> и файлы модулей JAR	499
Связывание файлов в развернутом каталоге	500
Связывание модульных файлов JAR	500
Файлы JMOD	501
Кратко об уровнях и автоматических модулях	502
Заключительные соображения по поводу модулей	502
Глава 17. Выражения <code>switch</code>, записи и прочие недавно добавленные средства	503
Расширения оператора <code>switch</code>	504
Использование списка констант <code>case</code>	505
Появление выражения <code>switch</code> и оператора <code>yield</code>	506
Появление стрелки в операторе <code>case</code>	509
Подробный анализ оператора <code>case</code> со стрелкой	510
Еще один пример выражения <code>switch</code>	514
Текстовые блоки	514
Основы текстовых блоков	515
Ведущие пробельные символы	516
Использование двойных кавычек в текстовом блоке	517
Управляющие последовательности в текстовых блоках	518
Записи	519
Основы записей	520
Создание конструкторов записи	522
Еще один пример конструктора записи	526
Создание методов получения для записи	528
Сопоставление с образцом	
в операции <code>instanceof</code>	530
Шаблонные переменные в логических выражениях “И”	531
Сопоставление с образцом в других операторах	532
Запечатанные классы и запечатанные интерфейсы	533
Запечатанные классы	533
Запечатанные интерфейсы	535
Будущие направления развития	537

Часть II. Библиотека Java	539
Глава 18. Обработка строк	540
Конструкторы класса String	541
Длина строки	543
Специальные строковые операции	543
Строковые литералы	543
Конкатенация строк	544
Конкатенация строк с другими типами данных	544
Преобразование в строку и toString()	545
Извлечение символов	546
charAt()	546
getChars()	547
getBytes()	547
toCharArray()	547
Сравнение строк	548
equals() и equalsIgnoreCase()	548
regionMatches()	549
startsWith() и endsWith()	549
equals() или ==	549
compareTo()	550
Поиск в строках	552
Модификация строк	553
substring()	553
concat()	554
replace()	554
trim() и strip()	555
Преобразование данных с использованием valueOf()	556
Изменение регистра символов внутри строк	556
Соединение строк	557
Дополнительные методы класса String	558
Класс StringBuffer	561
Конструкторы класса StringBuffer	561
length() и capacity()	561
ensureCapacity()	562
setLength()	562
charAt() и setCharAt()	562
getChars()	563
append()	563
insert()	564
reverse()	564
delete() и deleteCharAt()	565
replace()	566
substring()	566
Дополнительные методы класса StringBuffer	566
Класс StringBuilder	568
Глава 19. Исследование пакета java.lang	569
Оболочки примитивных типов	570
Number	570
Double и Float	570
Методы isInfinite() и isNaN()	576

Byte, Short, Integer и Long	576
Character	591
Дополнения класса Character	
для поддержки кодовых точек Unicode	594
Boolean	596
Void	598
Process	598
Runtime	599
Выполнение других программ	601
Runtime.Version	602
ProcessBuilder	604
System	608
Использование currentTimeMillis() для хронометража	
выполнения программы	610
Использование arraycopy()	611
Свойства среды	612
System.Logger и System.LoggerFinder	612
Object	612
Использование метода clone() и интерфейса Cloneable	613
Class	615
ClassLoader	621
Math	621
Тригонометрические функции	621
Экспоненциальные функции	622
Функции округления	623
Прочие методы Math	625
StrictMath	628
Compiler	628
Thread, ThreadGroup и Runnable	628
Интерфейс Runnable	628
Класс Thread	628
Класс ThreadGroup	632
ThreadLocal и InheritableThreadLocal	636
Package	636
Module	638
ModuleLayer	639
RuntimePermission	639
Throwable	639
SecurityManager	639
StackTraceElement	640
StackWalker и StackWalker.StackFrame	641
Enum	641
Record	642
ClassValue	643
Интерфейс CharSequence	643
Интерфейс Comparable	644
Интерфейс Appendable	644
Интерфейс Iterable	644
Интерфейс Readable	645
Интерфейс AutoCloseable	645
Интерфейс Thread.UncaughtExceptionHandler	646

Подпакеты <code>java.lang</code>	646
<code>java.lang.annotation</code>	646
<code>java.lang.constant</code>	646
<code>java.lang.instrument</code>	646
<code>java.lang.invoke</code>	647
<code>java.lang.management</code>	647
<code>java.lang.module</code>	647
<code>java.lang.ref</code>	647
<code>java.lang.reflect</code>	647
Глава 20. Пакет <code>java.util</code>, часть 1: Collections Framework	648
Обзор Collections Framework	649
Интерфейсы коллекций	651
Интерфейс <code>Collection</code>	652
Интерфейс <code>List</code>	655
Интерфейс <code>Set</code>	658
Интерфейс <code>SortedSet</code>	659
Интерфейс <code>NavigableSet</code>	660
Интерфейс <code>Queue</code>	662
Интерфейс <code>Deque</code>	663
Классы коллекций	666
Класс <code>ArrayList</code>	667
Класс <code>LinkedList</code>	671
Класс <code>HashSet</code>	672
Класс <code>LinkedHashSet</code>	674
Класс <code>TreeSet</code>	674
Класс <code>PriorityQueue</code>	675
Класс <code>ArrayDeque</code>	676
Класс <code>EnumSet</code>	677
Доступ в коллекцию через итератор	679
Использование итератора	680
Альтернатива итераторам в виде цикла <code>for</code> в стиле “for-each”	682
Сплитераторы	683
Хранение объектов пользовательских классов в коллекциях	686
Интерфейс <code>RandomAccess</code>	688
Работа с картами	688
Интерфейсы карт	688
Классы карт	697
Компараторы	703
Использование компаратора	705
Алгоритмы коллекций	711
Массивы	719
Унаследованные классы и интерфейсы	724
Интерфейс <code>Enumeration</code>	725
Класс <code>Vector</code>	725
Класс <code>Stack</code>	730
Класс <code>Dictionary</code>	732
Класс <code>Hashtable</code>	733
Класс <code>Properties</code>	737
Использование методов <code>store()</code> и <code>load()</code>	740
Заключительные соображения по поводу коллекций	742

Глава 21. Пакет java.util, часть 2:

Дополнительные служебные классы	743
Класс StringTokenizer	743
BitSet	745
Optional, OptionalDouble, OptionalInt и OptionalLong	749
Date	753
Calendar	755
GregorianCalendar	759
TimeZone	761
SimpleTimeZone	762
Locale	763
Random	764
Timer и TimerTask	767
Currency	770
Formatter	771
Конструкторы класса Formatter	772
Методы класса Formatter	772
Основы форматирования	773
Форматирование строк и символов	776
Форматирование чисел	776
Форматирование времени и даты	777
Спецификаторы %n и %%	779
Указание минимальной ширины поля	779
Указание точности	781
Использование флагов формата	782
Выравнивание выводимых данных	782
Флаги пробела, +, 0 и (	783
Флаг запятой	784
Флаг #	784
Версии в верхнем регистре	784
Использование индекса аргумента	785
Закрытие объекта Formatter	787
Альтернативный вариант: метод printf()	787
Scanner	787
Конструкторы класса Scanner	788
Основы сканирования	789
Примеры использования класса Scanner	794
Установка разделителей	798
Дополнительные средства класса Scanner	799
ResourceBundle, ListResourceBundle и PropertyResourceBundle	800
Смешанные служебные классы и интерфейсы	805
Подпакеты пакета java.util	807
java.util.concurrent, java.util.concurrent.atomic	
и java.util.concurrent.locks	807
java.util.function	807
java.util.jar	811
java.util.logging	812
java.util.prefs	812
java.util.random	812
java.util.regex	812
java.util.spi	812
java.util.stream	812
java.util.zip	812

Глава 22. Ввод-вывод: исследование пакета java.io	813
Классы и интерфейсы ввода-вывода	814
File	814
Каталоги	818
Использование интерфейса FilenameFilter	819
Альтернативные методы listFiles()	820
Создание каталогов	820
Интерфейсы AutoCloseable, Closeable и Flushable	821
Исключения ввода-вывода	821
Два способа закрытия потока данных	822
Классы потоков данных	824
Байтовые потоки	824
InputStream	824
OutputStream	826
FileInputStream	827
FileOutputStream	829
ByteArrayInputStream	831
ByteArrayOutputStream	832
Фильтрующие байтовые потоки	834
Буферизованные байтовые потоки	834
SequenceInputStream	838
PrintStream	840
DataOutputStream и DataInputStream	842
RandomAccessFile	844
Символьные потоки	845
Reader	845
Writer	847
FileReader	848
FileWriter	848
CharArrayReader	850
CharArrayWriter	851
BufferedReader	852
BufferedWriter	853
PushbackReader	854
PrintWriter	855
Класс Console	856
Сериализация	858
Serializable	859
Externalizable	859
ObjectOutput	859
ObjectOutputStream	860
ObjectInput	862
ObjectInputStream	862
Пример сериализации	864
Преимущества потоков	867
Глава 23. Исследование системы NIO	868
Классы NIO	868
Основы NIO	869
Буферы	869
Каналы	873
Наборы символов и селекторы	875

Усовершенствования, появившиеся в NIO.2	875
Интерфейс Path	875
Класс Files	878
Класс Paths	881
Интерфейсы для файловых атрибутов	882
Классы FileSystem, FileSystems и FileStore	885
Использование системы NIO	885
Использование системы NIO для ввода-вывода, основанного на каналах	886
Использование системы NIO для ввода-вывода, основанного на потоках	896
Использование системы NIO для операций с путями и файловой системой	898
Глава 24. Работа в сети	907
Основы работы в сети	907
Классы и интерфейсы пакета java.net для работы в сети	909
InetAddress	910
Фабричные методы	910
Методы экземпляра	911
Inet4Address и Inet6Address	912
Клиентские сокеты TCP/IP	912
URL	916
URLConnection	917
HttpURLConnection	920
Класс URI	922
Cookie-наборы	923
Серверные сокеты TCP/IP	923
Дейтаграммы	924
DatagramSocket	924
DatagramPacket	925
Пример использования дейтаграмм	926
Введение в пакет java.net.http	928
Три ключевых элемента	928
Простой пример клиента HTTP	931
Что еще рекомендуется изучить в java.net.http	933
Глава 25. Обработка событий	934
Два механизма обработки событий	935
Модель делегирования обработки событий	935
События	936
Источники событий	936
Прослушиватели событий	937
Классы событий	937
Класс ActionEvent	938
Класс AdjustmentEvent	940
Класс ComponentEvent	941
Класс ContainerEvent	942
Класс FocusEvent	942
Класс InputEvent	943
Класс ItemEvent	944
Класс KeyEvent	945
Класс MouseEvent	946
Класс MouseWheelEvent	947

Класс <code>TextEvent</code>	948
Класс <code>WindowEvent</code>	949
Источники событий	950
Интерфейсы прослушивателей событий	951
Интерфейс <code>ActionListener</code>	952
Интерфейс <code>AdjustmentListener</code>	952
Интерфейс <code>ComponentListener</code>	952
Интерфейс <code>ContainerListener</code>	952
Интерфейс <code>FocusListener</code>	953
Интерфейс <code>ItemListener</code>	953
Интерфейс <code>KeyListener</code>	953
Интерфейс <code>MouseListener</code>	953
Интерфейс <code>MouseMotionListener</code>	954
Интерфейс <code>MouseWheelListener</code>	954
Интерфейс <code>TextListener</code>	954
Интерфейс <code>WindowFocusListener</code>	954
Интерфейс <code>WindowListener</code>	954
Использование модели делегирования обработки событий	955
Основные концепции графических пользовательских интерфейсов AWT	955
Обработка событий мыши	956
Обработка событий клавиатуры	960
Классы адаптеров	963
Внутренние классы	966
Анонимные внутренние классы	968
Глава 26. Введение в AWT: работа с окнами, графикой и текстом	970
Классы AWT	971
Основы окон	974
<code>Component</code>	974
<code>Container</code>	975
<code>Panel</code>	975
<code>Window</code>	975
<code>Frame</code>	975
<code>Canvas</code>	975
Работа с окнами <code>Frame</code>	976
Установка размеров окна	976
Скрытие и отображение окна	976
Установка заголовка окна	976
Заккрытие фреймового окна	977
Метод <code>paint()</code>	977
Отображение строки	977
Установка цветов фона и переднего плана	978
Запрос перерисовки	978
Создание приложения на основе <code>Frame</code>	980
Введение в графику	980
Вычерчивание линий	981
Вычерчивание прямоугольников	981
Вычерчивание эллипсов и окружностей	981
Вычерчивание дуг	982
Вычерчивание многоугольников	982
Демонстрация работы методов вычерчивания	982
Установка размеров графики	983

Работа с цветом	985
Методы класса <code>Color</code>	986
Установка текущего цвета графики	987
Программа, демонстрирующая работу с цветом	987
Установка режима рисования	988
Работа со шрифтами	990
Выяснение доступных шрифтов	992
Создание и выбор шрифта	993
Получение информации о шрифте	995
Управление выводом текста с использованием <code>FontMetrics</code>	996
Глава 27. Использование элементов управления, диспетчеров компоновки и меню AWT	1001
Основы элементов управления AWT	1002
Добавление и удаление элементов управления	1002
Реагирование на события, генерируемые элементами управления	1003
Исключение <code>HeadlessException</code>	1003
Метки	1003
Использование кнопок	1005
Обработка событий для кнопок	1005
Использование флажков	1009
Обработка событий для флажков	1010
Группы флажков	1012
Элементы управления выбором	1014
Обработка событий для списков выбора	1015
Использование списков	1016
Обработка событий для списков	1018
Управление полосами прокрутки	1019
Обработка событий для полос прокрутки	1021
Использование текстовых полей	1023
Обработка событий для текстовых полей	1024
Использование текстовых областей	1026
Понятие диспетчеров компоновки	1028
<code>FlowLayout</code>	1029
<code>BorderLayout</code>	1030
Использование вставок	1031
<code>GridLayout</code>	1033
<code>CardLayout</code>	1034
<code>GridBagLayout</code>	1037
Меню и панели меню	1043
Диалоговые окна	1048
Несколько слов о переопределении метода <code>paint()</code>	1052
Глава 28. Изображения	1053
Форматы файлов	1053
Основы работы с изображениями: создание, загрузка и отображение	1054
Создание объекта изображения	1054
Загрузка изображения	1055
Отображение изображения	1055
Двойная буферизация	1057
<code>ImageProducer</code>	1060
<code>MemoryImageSource</code>	1060

ImageConsumer	1062
PixelGrabber	1062
ImageFilter	1065
CropImageFilter	1065
RGBImageFilter	1067
Дополнительные классы для обработки изображений	1078
Глава 29. Утилиты параллелизма	1079
Пакеты параллельного API	1080
java.util.concurrent	1081
java.util.concurrent.atomic	1082
java.util.concurrent.locks	1082
Использование объектов синхронизации	1082
Semaphore	1083
CountDownLatch	1088
CyclicBarrier	1090
Exchanger	1092
Phaser	1095
Использование исполнителя	1103
Простой пример использования исполнителя	1104
Использование интерфейсов Callable и Future	1105
Перечисление TimeUnit	1108
Параллельные коллекции	1109
Блокировки	1110
Атомарные операции	1113
Параллельное программирование с помощью Fork/Join Framework	1114
Главные классы Fork/Join Framework	1115
Стратегия “разделяй и властвуй”	1119
Простой пример использования Fork/Join Framework	1121
Влияние уровня параллелизма	1123
Пример использования RecursiveTask<V>	1126
Выполнение задачи асинхронным образом	1129
Отмена задачи	1129
Определение состояния завершения задачи	1130
Перезапуск задачи	1130
Дальнейшие исследования	1130
Советы по использованию Fork/Join Framework	1132
Сравнение утилит параллелизма и традиционного подхода к многопоточности в Java	1133
Глава 30. Поточковый API-интерфейс	1134
Основы потоков	1134
Потоковые интерфейсы	1135
Получение потока	1138
Простой пример использования потока	1139
Операции редукции	1142
Использование параллельных потоков	1145
Сопоставление	1147
Накопление	1151
Итераторы и потоки	1155
Использование итератора с потоком	1155
Использование сплитератора	1156
Дальнейшее исследование потокового API	1159

Глава 31. Регулярные выражения и другие пакеты	1160
Обработка регулярных выражений	1160
Класс Pattern	1161
Класс Matcher	1161
Синтаксис регулярных выражений	1162
Демонстрация сопоставления с шаблоном	1163
Два варианта сопоставления с шаблоном	1168
Дальнейшее исследование регулярных выражений	1169
Рефлексия	1169
Удаленный вызов методов	1174
Простое клиент-серверное приложение, использующее удаленный вызов методов	1174
Форматирование даты и времени с помощью пакета java.text	1178
Класс DateFormat	1178
Класс SimpleDateFormat	1180
Пакеты java.time, поддерживающие API даты и времени	1182
Фундаментальные классы для поддержки даты и времени	1182
Форматирование даты и времени	1184
Разбор строк с датой и временем	1187
Дальнейшее исследование пакета java.time	1188
 Часть III. Введение в программирование графических пользовательских интерфейсов с помощью Swing Java	1189
Глава 32. Введение в Swing	1190
Происхождение инфраструктуры Swing	1190
Инфраструктура Swing построена на основе AWT	1191
Две ключевые особенности Swing	1191
Компоненты Swing являются легковесными	1192
Инфраструктура Swing поддерживает подключаемый внешний вид	1192
Связь с архитектурой MVC	1192
Компоненты и контейнеры	1194
Компоненты	1194
Контейнеры	1195
Панели контейнеров верхнего уровня	1195
Пакеты Swing	1196
Простое приложение Swing	1196
Обработка событий	1201
Рисование в Swing	1204
Основы рисования	1205
Вычисление области рисования	1206
Пример программы рисования	1206
 Глава 33. Исследование Swing	1210
JLabel и ImageIcon	1210
JTextField	1212
Кнопки Swing	1214
JButton	1214
JToggleButton	1217
Флажки	1219
Взаимоисключающие переключатели	1221

JTabbedPane	1223
JScrollPane	1226
JList	1227
JComboBox	1231
Деревья	1233
JTable	1236
Глава 34. Введение в меню Swing	1240
Основы меню	1240
Обзор JMenuBar, JMenu и JMenuItem	1242
JMenuBar	1242
JMenu	1243
JMenuItem	1244
Создание главного меню	1245
Добавление мнемонических символов и клавиатурных сочетаний к пунктам меню	1249
Добавление изображений и всплывающих подсказок к пунктам меню	1252
Использование JRadioButtonMenuItem и JCheckBoxMenuItem	1253
Создание всплывающего меню	1255
Создание панели инструментов	1259
Использование действий	1261
Построение окончательной программы MenuDemo	1267
Продолжение исследования Swing	1273
Часть IV. Применение Java	1275
Глава 35. Архитектура JavaBeans	1276
Что собой представляет Bean-компонент	1276
Преимущества Bean-компонентов	1277
Самоанализ	1277
Паттерны проектирования для свойств	1278
Паттерны проектирования для событий	1279
Методы и паттерны проектирования	1280
Использование интерфейса BeanInfo	1280
Связанные и ограниченные свойства	1281
Постоянство	1281
Настройщики	1281
JavaBeans API	1282
Introspector	1285
PropertyDescriptor	1285
EventSetDescriptor	1285
MethodDescriptor	1285
Пример Bean-компонента	1286
Глава 36. Введение в сервлеты	1289
Происхождение сервлетов	1289
Жизненный цикл сервлета	1290
Варианты разработки сервлетов	1291
Использование Tomcat	1291
Простой сервлет	1293
Создание и компиляция исходного кода сервлета	1293

Запуск Tomcat	1294
Запуск веб-браузера и запрашивание сервлета	1294
Servlet API	1294
Пакет jakarta.servlet	1295
Интерфейс Servlet	1296
Интерфейс ServletConfig	1297
Интерфейс ServletContext	1297
Интерфейс ServletRequest	1298
Интерфейс ServletResponse	1299
Класс GenericServlet	1299
Класс ServletInputStream	1300
Класс ServletOutputStream	1300
Классы исключений сервлетов	1300
Чтение параметров сервлета	1300
Пакет jakarta.servlet.http	1302
Интерфейс HttpServletRequest	1302
Интерфейс HttpServletResponse	1304
Интерфейс HttpSession	1305
Класс Cookie	1306
Класс HttpServlet	1307
Обработка запросов и ответов HTTP	1308
Обработка HTTP-запросов GET	1309
Обработка HTTP-запросов POST	1310
Использование cookie-наборов	1311
Отслеживание сеансов	1313
Часть V. Приложения	1315
Приложение А. Использование документирующих комментариев Java	1316
Дескрипторы javadoc	1316
@author	1318
{@code}	1318
@deprecated	1318
{@docRoot}	1318
@exception	1319
@hidden	1319
{@index}	1319
{@inheritDoc}	1319
{@link}	1319
{@linkplain}	1320
{@literal}	1320
@param	1320
@provides	1320
@return	1320
@see	1321
@serial	1321
@serialData	1321
@serialField	1321
@since	1322
{@summary}	1322
{@systemProperty}	1322

@throws	1322
@uses	1322
{@value}	1323
@version	1323
Общая форма документирующего комментария	1323
Вывод утилиты javadoc	1323
Пример использования документирующих комментариев	1324
Приложение Б. Введение в JShell	1325
Основы JShell	1325
Просмотр, редактирование и повторного выполнение кода	1328
Добавление метода	1329
Создание класса	1330
Использование интерфейса	1331
Вычисление выражений и использование встроенных переменных	1332
Импортирование пакетов	1333
Исключения	1334
Другие команды JShell	1334
Дальнейшее исследование JShell	1335
Приложение В. Компиляция и запуск простых однофайловых программ за один шаг	1336
Предметный указатель	1338

Об авторе

Автор многочисленных бестселлеров **Герберт Шилдт** занимался написанием книг по программированию на протяжении более трех десятилетий и считается признанным авторитетом по языку Java. Журнал *International Developer* назвал его одним из ведущих авторов книг программированию в мире. Книги Герберта Шилдта продавались миллионными тиражами по всему миру и переведены на многие языки. Он является автором многочисленных книг по Java, в том числе *Java: руководство для начинающих*, *Java: методики программирования Шилдта*, *Introducing JavaFX 8 Programming* и *Swing: руководство для начинающих*. Герберт Шилдт также много писал о языках C, C++ и C#. В книге Эда Бернса *Secrets of the Rock Star Programmers: Riding the IT Crest* указано, что как один из звездных программистов Шилдт интересуется всеми аспектами вычислений, но его основное внимание сосредоточено на языках программирования. Герберт Шилдт получил степени бакалавра и магистра в Иллинойском университете. Его сайт доступен по адресу www.HerbSchildt.com.

О научном редакторе

Доктор **Дэнни Ковард** работал над всеми версиями платформы Java. Он руководил определением сервлетов Java в первой версии платформы Java EE и в более поздних версиях, внедрением веб-служб в Java ME, а также стратегией и планированием для Java SE 7. Он основал технологию JavaFX и совсем недавно создал крупнейшее дополнение к стандарту Java EE 7, Java WebSocket API. Доктор Дэнни Ковард обладает уникально широким взглядом на многие аспекты технологии Java. Его опыт простирается от написания кода на Java до разработки API с отраслевыми экспертами и работы в течение нескольких лет в качестве исполнительного директора в Java Community Process. Кроме того, он является автором двух книг по программированию на Java: *Java WebSocket Programming* и *Java EE 7: The Big Picture*. Совсем недавно он применил свои знания Java, помогая масштабировать крупные службы на основе Java для одной из самых успешных в мире компаний-разработчиков программного обеспечения. Доктор Дэнни Ковард получил степень бакалавра, магистра и доктора математики в Оксфордском университете.

Предисловие

Java — один из самых важных и широко используемых языков программирования в мире. На протяжении многих лет ему была присуща эта отличительная особенность. В отличие от ряда других языков программирования, влияние которых с течением времени ослабевало, влияние Java становилось только сильнее. С момента своего первого выпуска язык Java выдвинулся на передний край программирования для Интернета. Его позиции закреплялись с каждой последующей версией. На сегодняшний день Java по-прежнему является первым и лучшим выбором для разработки веб-приложений, а также мощным языком программирования общего назначения, подходящий для самых разных целей. Проще говоря, большая часть современного кода написана на Java. Язык Java действительно настолько важен.

Ключевая причина успеха языка Java кроется в его гибкости. С момента своего первоначального выпуска 1.0 он постоянно адаптировался к изменениям в среде программирования и к изменениям в способах написания кода программистами. Самое главное то, что язык Java не просто следовал тенденциям — он помогал их создавать. Способность языка Java приспосабливаться к быстрым изменениям в мире программирования является важной частью того, почему он был и остается настолько успешным.

С момента первой публикации этой книги в 1996 году она выдержала множество переизданий, в каждом из которых отражалась непрерывная эволюция Java. Текущее двенадцатое издание книги обновлено с учетом Java SE 17 (JDK 17). В итоге оно содержит значительный объем нового материала, обновлений и изменений. Особый интерес представляет обсуждение следующих ключевых возможностей, которые были добавлены в язык Java в сравнении с предыдущим изданием:

- усовершенствования оператора `switch`;
- записи;
- сопоставление с образцом в `instanceof`;
- запечатанные классы и интерфейсы;
- текстовые блоки.

В совокупности они составляют существенный набор новых функциональных средств, которые значительно расширяют диапазон охвата, область применимости и выразительность языка. Усовершенствования `switch` добавляют мощи и гибкости этому основополагающему оператору управления. Появившиеся записи предлагают эффективный способ агрегирования данных. Добавление сопоставления с образцом в `instanceof` обеспечивает более рациональный и устойчивый подход к решению обычной задачи программирования. Запечатанные классы и интерфейсы делают возможным детализированный контроль над наследованием. Текстовые блоки позволяют вводить многострочные строковые литералы, что значительно упрощает процесс вставки таких строк в исходный код. Все вместе новые функциональные средства существенно расширяют возможности разработки и внедрения решений.

Книга для всех программистов

Книга предназначена для всех программистов: от новичков до профессионалов. Новичок сочтет особенно полезными тщательно продуманные обсуждения и множество примеров. Профессионалам понравится подробное описание более сложных функциональных средств и библиотек Java. И те, и другие получат в свое распоряжение прочный информационный ресурс и удобный справочник.

Структура книги

Эта книга представляет собой исчерпывающее руководство по языку Java, в котором описаны его синтаксис, ключевые слова и базовые принципы программирования. Вдобавок исследуется значительное подмножество библиотеки Java API. Книга разделена на четыре части, каждая из которых посвящена отдельному аспекту среды программирования Java.

Часть I предлагает подробное учебное пособие по языку Java. Она начинается с основ, включая типы данных, операции, управляющие операторы и классы. Затем обсуждаются наследование, пакеты, интерфейсы, обработка исключений и многопоточность. Далее описаны аннотации, перечисления, автоупаковка, обобщения, модули и лямбда-выражения. Также предлагается введение в систему ввода-вывода. В последней главе части I рассматривается несколько недавно добавленных средств: записи, запечатанные классы и интерфейсы, расширенный оператор `switch`, сопоставление с образцом в `instanceof` и текстовые блоки.

В части II исследуются ключевые аспекты стандартной библиотеки API Java, включая строки, ввод-вывод, работу в сети, стандартные утилиты, Collections Framework, AWT, обработку событий, визуализацию, параллелизм (в том числе Fork/Join Framework), регулярные выражения и библиотеку потоков.

В трех главах части III дается введение в инфраструктуру Swing.

Часть IV состоит из двух глав, в которых демонстрируются примеры приложений Java. В одной главе обсуждаются компоненты Java Beans, а в другой представлено введение в сервлеты.

Благодарности

Я хочу выразить особую благодарность Патрику Нотону, Джо О'Нилу и Дэнни Коварду.

Патрик Нотон был одним из создателей языка Java. Он также помог написать первое издание этой книги. Например, помимо многих других вкладов большая часть материала в главах 22, 24 и 28 изначально была предоставлена Патриком. Его проницательность, опыт и энергия во многом способствовали успеху книги.

Во время подготовки второго и третьего изданий этой книги Джо О'Нил предоставил первоначальные наброски материала, который сейчас содержится в главах 31, 33, 35 и 36 текущего издания. Джо помогал мне при написании нескольких книг, и его вклад всегда был превосходным.

Дэнни Ковард занимался научным редактированием этого издания книги. Дэнни работал над несколькими моими книгами, и его советы, идеи и предложения всегда имели большую ценность и получали высокую оценку.

Я также хочу поблагодарить мою жену Шерри за весь ее вклад в эту и в другие мои книги. Ее вдумчивые советы, корректура и подготовка предметного указателя всегда в значительной степени способствовали успешному завершению каждого проекта.

Герберт Шилдт

Ждем ваших отзывов!

Вы, читатель этой книги, и есть главный ее критик. 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 интересны любые ваши замечания в наш адрес.

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

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

Наши электронные адреса:

E-mail: info.dialektika@gmail.com

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

