

Spis treści

O autorze	25
Przedmowa	27

CZĘŚĆ I Język Java

Historia i ewolucja języka Java	33
Rodowód Javy	33
Narodziny nowoczesnego języka — C	33
Język C++ — następny krok	35
Podwaliny języka Java	35
Powstanie języka Java	35
Powiązanie z językiem C#	37
Jak Java wywarła wpływ na internet	37
Aplety Javy	37
Bezpieczeństwo	38
Przenośność	38
Magia języka Java — kod bajtowy	39
Wychodząc poza aplety	40
Serwlety — Java po stronie serwera	40
Hasła języka Java	40
Prostota	41
Obiektowość	41
Niezawodność	41
Wielowątkowość	42
Neutralność architektury	42
Interpretowalność i wysoka wydajność	42
Rozproszenie	42
Dynamika	42
Ewolucja Javy	43
Java SE 9	46
Kultura innowacji	46

2	Podstawy języka Java	47
	Programowanie obiektowe	47
	Dwa paradygmaty	47
	Abstrakcja	48
	Trzy zasady programowania obiektowego	48
	Pierwszy przykładowy program	52
	Wpisanie kodu programu	52
	Kompilacja programów	53
	Bliższe spojrzenie na pierwszy przykładowy program	53
	Drugi prosty program	55
	Dwie instrukcje sterujące	56
	Instrukcja if	56
	Pętla for	58
	Bloki kodu	59
	Kwestie składniowe	60
	Znaki białe	60
	Identyfikatory	60
	Stałe	60
	Komentarze	61
	Separatory	61
	Słowa kluczowe języka Java	61
	Biblioteki klas Javy	62
3	Typy danych, zmienne i tablice	63
	Java to język ze ścisłą kontrolą typów	63
	Typy proste	63
	Typy całkowitoliczbowe	64
	Typ byte	64
	Typ short	65
	Typ int	65
	Typ long	65
	Typy zmiennoprzecinkowe	65
	Typ float	66
	Typ double	66
	Typ znakowy	66
	Typ logiczny	68
	Bliższe spojrzenie na stałe	68
	Stałe całkowitoliczbowe	68
	Stałe zmiennoprzecinkowe	69
	Stałe logiczne	70
	Stałe znakowe	70
	Stałe łańcuchowe	71
	Zmienne	71
	Deklaracja zmiennej	71
	Inicjalizacja dynamiczna	72
	Zasięg i czas życia zmiennych	72
	Konwersja typów i rzutowanie	74
	Automatyczna konwersja typów	74
	Rzutowanie niezgodnych typów	75
	Automatyczne rozszerzanie typów w wyrażeniach	76
	Zasady rozszerzania typu	76
	Tablice	77
	Tablice jednowymiarowe	77
	Tablice wielowymiarowe	79
	Alternatywna składnia deklaracji tablicy	82
	Kilka słów o łańcuchach	83

4	Operatory	85
	Operatory arytmetyczne	85
	Podstawowe operatory arytmetyczne	86
	Operator reszty z dzielenia	86
	Operatory arytmetyczne z przypisaniem	87
	Inkrementacja i dekrementacja	88
	Operatory bitowe	89
	Logiczne operatory bitowe	90
	Przesunięcie w lewo	92
	Przesunięcie w prawo	93
	Przesunięcie w prawo bez znaku	94
	Operatory bitowe z przypisaniem	95
	Operatory relacji	96
	Operatory logiczne	97
	Operatory logiczne ze skracaniem	98
	Operator przypisania	99
	Operator ?	99
	Kolejność wykonywania operatorów	100
	Stosowanie nawiasów okrągłych	100
5	Instrukcje sterujące	103
	Instrukcje wyboru	103
	Instrukcja if	103
	Instrukcja switch	106
	Instrukcje iteracyjne	109
	Pętla while	110
	Pętla do-while	111
	Pętla for	113
	Wersja for-each pętli for	116
	Pętle zagnieżdżone	120
	Instrukcje skoku	121
	Instrukcja break	121
	Instrukcja continue	124
	Instrukcja return	125
6	Wprowadzenie do klas	127
	Klasy	127
	Ogólna postać klasy	127
	Prosta klasa	128
	Deklarowanie obiektów	130
	Bliższe spojrzenie na operator new	131
	Przypisywanie zmiennych referencyjnych do obiektów	131
	Wprowadzenie do metod	132
	Dodanie metody do klasy Box	132
	Zwracanie wartości	134
	Dodanie metody przyjmującej parametry	135
	Konstruktor	137
	Konstruktor sparametryzowany	138
	Słowo kluczowe this	139
	Ukrywanie zmiennych składowych	139
	Mechanizm odzyskiwania pamięci	140
	Klasa stosu	140

7	Dokładniejsze omówienie metod i klas	143
	Przeciążanie metod	143
	Przeciążanie konstruktorów	145
	Obiekty jako parametry	147
	Dokładniejsze omówienie przekazywania argumentów	149
	Zwracanie obiektów	150
	Rekurencja	151
	Wprowadzenie do kontroli dostępu	153
	Składowe statyczne	156
	Słowo kluczowe final	157
	Powtórka z tablic	158
	Klasy zagnieżdżone i klasy wewnętrzne	159
	Omówienie klasy String	162
	Wykorzystanie argumentów wiersza poleceń	163
	Zmienna liczba argumentów	164
	Przeciążanie metod o zmiennej liczbie argumentów	167
	Zmienna liczba argumentów i niejednoznaczności	168
8	Dziedziczenie	171
	Podstawy dziedziczenia	171
	Dostęp do składowych a dziedziczenie	172
	Bardziej praktyczny przykład	173
	Zmienna klasy bazowej może zawierać referencję do obiektu klasy pochodnej	175
	Słowo kluczowe super	176
	Wykorzystanie słowa kluczowego super do wywołania konstruktora klasy bazowej	176
	Drugie zastosowanie słowa kluczowego super	179
	Tworzenie hierarchii wielopoziomowej	180
	Kiedy są wykonywane konstruktory?	182
	Przesłanianie metod	183
	Dynamiczne przydzielanie metod	185
	Dlaczego warto przesłaniać metody?	186
	Zastosowanie przesłaniania metod	186
	Klasy abstrakcyjne	188
	Słowo kluczowe final i dziedziczenie	190
	Słowo kluczowe final zapobiega przesłanianiu	190
	Słowo kluczowe final zapobiega dziedziczeniu	191
	Klasa Object	191
9	Pakiety i interfejsy	193
	Pakiety	193
	Definiowanie pakietu	193
	Znajdowanie pakietów i ścieżka CLASSPATH	194
	Prosty przykład pakietu	195
	Dostęp do pakietów i składowych	195
	Przykład dostępu	196
	Import pakietów	199
	Interfejsy	200
	Definiowanie interfejsu	201
	Implementacja interfejsu	202
	Interfejsy zagnieżdżone	204
	Stosowanie interfejsów	205
	Zmienne w interfejsach	207
	Interfejsy można rozszerzać	209

Metody domyślne	210
Podstawy metod domyślnych	211
Bardziej praktyczny przykład	212
Problemy wielokrotnego dziedziczenia	213
Metody statyczne w interfejsach	213
Stosowanie metod prywatnych w interfejsach	214
Ostatnie uwagi dotyczące pakietów i interfejsów	215
10 Obsługa wyjątków	217
Podstawy obsługi wyjątków	217
Typy wyjątków	218
Nieprzechwycone wyjątki	218
Stosowanie instrukcji try i catch	219
Wyświetlenie opisu wyjątku	220
Wiele klauzul catch	221
Zagnieżdżone instrukcje try	222
Instrukcja throw	224
Klauzula throws	225
Słowo kluczowe finally	225
Wyjątki wbudowane w język Java	227
Tworzenie własnej klasy pochodnej wyjątków	227
Łańcuch wyjątków	230
Trzy dodatkowe cechy wyjątków	231
Wykorzystanie wyjątków	232
11 Programowanie wielowątkowe	233
Model wątków języka Java	234
Priorytety wątków	235
Synchronizacja	235
Przekazywanie komunikatów	236
Klasa Thread i interfejs Runnable	236
Wątek główny	236
Tworzenie wątku	238
Implementacja interfejsu Runnable	238
Rozszerzanie klasy Thread	240
Wybór odpowiedniego podejścia	240
Tworzenie wielu wątków	241
Stosowanie metod isAlive() i join()	242
Priorytety wątków	244
Synchronizacja	245
Synchronizacja metod	245
Instrukcja synchronized	247
Komunikacja międzywątkowa	248
Zakleszczenie	252
Zawieszanie, wznawianie i zatrzymywanie wątków	254
Uzyskiwanie stanu wątku	256
Stosowanie metody wytwórczej do tworzenia i uruchamiania wątku	257
Korzystanie z wielowątkowości	258
12 Wyliczenia, automatyczne opakowywanie typów prostych i adnotacje	259
Typy wyliczeniowe	259
Podstawy wyliczeń	259
Metody values() i valueOf()	261
Wyliczenia Javy jako typy klasowe	262

Wyliczenia dziedziczą po klasie Enum	264
Inny przykład wyliczenia	265
Opakowania typów	267
Klasa Character	267
Klasa Boolean	267
Opakowania typów numerycznych	268
Automatyczne opakowywanie typów prostych	269
Automatyczne opakowywanie i metody	270
Automatyczne opakowywanie i rozpakowywanie w wyrażeniach	270
Automatyczne opakowywanie typów znakowych i logicznych	272
Automatyczne opakowywanie pomaga zapobiegać błędom	272
Słowo ostrzeżenia	273
Adnotacje	273
Podstawy tworzenia adnotacji	274
Określanie strategii zachowywania adnotacji	274
Odczytywanie adnotacji w trakcie działania programu za pomocą refleksji	275
Interfejs AnnotatedElement	279
Wartości domyślne	279
Adnotacje znacznikowe	281
Adnotacje jednoelementowe	281
Wbudowane adnotacje	283
Adnotacje typów	284
Adnotacje powtarzalne	288
Ograniczenia	290
13 Wejście-wyjście, instrukcja try z zasobami i inne tematy	291
Podstawowa obsługa wejścia i wyjścia	291
Strumienie	292
Strumienie znakowe i bajtowe	292
Predefiniowane strumienie	294
Odczyt danych z konsoli	294
Odczyt znaków	294
Odczyt łańcuchów	295
Wyświetlanie informacji na konsoli	297
Klasa PrintWriter	297
Odczyt i zapis plików	298
Automatyczne zamykanie pliku	303
Modyfikatory transient i volatile	306
Operator instanceof	307
Modyfikator strictfp	309
Metody napisane w kodzie rdzennym	309
Stosowanie asercji	309
Opcje włączania i wyłączania asercji	311
Import statyczny	312
Wywoływanie przeciążonych konstruktorów za pomocą this()	314
Kilka słów o kompaktowych profilach API	316
14 Typy sparametryzowane	317
Czym są typy sparametryzowane?	317
Prosty przykład zastosowania typów sparametryzowanych	318
Typy sparametryzowane działają tylko dla typów referencyjnych	321
Typy sparametryzowane różnią się, jeśli mają inny argument typu	321
W jaki sposób typy sparametryzowane zwiększają bezpieczeństwo?	321

Klasa sparametryzowana z dwoma parametrami typu	323
Ogólna postać klasy sparametryzowanej	324
Typy ograniczone	324
Zastosowanie argumentów wieloznacznych	326
Ograniczony argument wieloznaczny	329
Tworzenie metody sparametryzowanej	333
Konstruktory sparametryzowane	334
Interfejsy sparametryzowane	335
Typy surowe i starszy kod	337
Hierarchia klas sparametryzowanych	339
Zastosowanie sparametryzowanej klasy bazowej	339
Podklasa sparametryzowana	341
Porównywanie typów w hierarchii klas sparametryzowanych w czasie wykonywania	342
Rzutowanie	344
Przesłanianie metod w klasach sparametryzowanych	344
Wnioskowanie typów a typy sparametryzowane	345
Znoszenie	346
Metody mostu	346
Błędy niejednoznaczności	348
Pewne ograniczenia typów sparametryzowanych	349
Nie można tworzyć egzemplarza parametru typu	349
Ograniczenia dla składowych statycznych	349
Ograniczenia tablic typów sparametryzowanych	349
Ograniczenia wyjątków typów sparametryzowanych	350
15 Wyrażenia lambda	351
Wprowadzenie do wyrażeń lambda	351
Podstawowe informacje o wyrażeniach lambda	352
Interfejsy funkcyjne	352
Kilka przykładów wyrażeń lambda	353
Blokowe wyrażenia lambda	356
Sparametryzowane interfejsy funkcyjne	358
Przekazywanie wyrażeń lambda jako argumentów	359
Wyrażenia lambda i wyjątki	362
Wyrażenia lambda i przechwytywanie zmiennych	363
Referencje do metod	364
Referencje do metod statycznych	364
Referencje do metod instancyjnych	365
Referencje do metod a typy sparametryzowane	368
Referencje do konstruktorów	370
Predefiniowane interfejsy funkcyjne	374
16 Moduły	377
Podstawowe informacje o modułach	377
Przykład prostego modułu	378
Kompilowanie i uruchamianie przykładowej aplikacji	382
Dokładniejsze informacje o instrukcjach requires i exports	383
java.base i moduły platformy	384
Stary kod i moduł nienazwany	384
Eksportowanie do konkretnego modułu	385
Wymagania przechodnie	386
Stosowanie usług	390
Podstawowe informacje o usługach i dostawcach usług	390
Słowa kluczowe związane z usługami	391
Przykład stosowania usług i modułów	391

Grafy modułów	397
Trzy wyspecjalizowane cechy modułów	398
Moduły otwarte	398
Instrukcja opens	398
Instrukcja requires static	398
Wprowadzenie do jlink i plików JAR modułów	399
Dołączanie plików dostarczonych jako struktura katalogów	399
Konsolidacja modularnych plików JAR	399
Pliki JMOD	400
Kilka słów o warstwach i modułach automatycznych	400
Końcowe uwagi dotyczące modułów	401

CZĘŚĆ II

Biblioteka języka Java

17 Obsługa łańcuchów	405
Konstruktory klasy String	405
Długość łańcucha	407
Specjalne operacje na łańcuchach	407
Literały tekstowe	407
Konkatenacja łańcuchów	408
Konkatenacja łańcuchów z innymi typami danych	408
Konwersja łańcuchów i metoda toString()	409
Wyodrębnianie znaków	410
Metoda charAt()	410
Metoda getChars()	410
Metoda getBytes()	410
Metoda toCharArray()	411
Porównywanie łańcuchów	411
Metody equals() i equalsIgnoreCase()	411
Metoda regionMatches()	412
Metody startsWith() i endsWith()	412
Metoda equals() kontra operator ==	412
Metoda compareTo()	413
Przeszukiwanie łańcuchów	414
Modyfikowanie łańcucha	415
Metoda substring()	415
Metoda concat()	416
Metoda replace()	416
Metoda trim()	417
Konwersja danych za pomocą metody valueOf()	417
Zmiana wielkości liter w łańcuchu	418
Łączenie łańcuchów	419
Dodatkowe metody klasy String	419
Klasa StringBuffer	419
Konstruktory klasy StringBuffer	420
Metody length() i capacity()	421
Metoda ensureCapacity()	421
Metoda setLength()	421
Metody charAt() i setCharAt()	422
Metoda getChars()	422
Metoda append()	422
Metoda insert()	423

Metoda reverse()	423
Metody delete() i deleteCharAt()	424
Metoda replace()	424
Metoda substring()	425
Dodatkowe metody klasy StringBuffer	425
Klasa StringBuilder	426
18 Pakiet java.lang	427
Opakowania typów prostych	427
Klasa Number	428
Klasy Double i Float	428
Metody isInfinite() i isNaN()	431
Klasy Byte, Short, Integer i Long	431
Klasa Character	439
Dodatki wprowadzone w celu obsługi punktów kodowych Unicode	440
Klasa Boolean	443
Klasa Void	443
Klasa Process	444
Klasa Runtime	445
Zarządzanie pamięcią	446
Wykonywanie innych programów	447
Runtime.Version	447
Klasa ProcessBuilder	448
Klasa System	450
Wykorzystanie metody currentTimeMillis()	
do obliczania czasu wykonywania programu	452
Użycie metody arraycopy()	452
Właściwości środowiska	453
Interfejs System.Logger i klasa System.LoggerFinder	453
Klasa Object	453
Wykorzystanie metody clone() i interfejsu Cloneable	454
Klasa Class	456
Klasa ClassLoader	458
Klasa Math	458
Funkcje trygonometryczne	458
Funkcje wykładnicze	459
Funkcje zaokrąglenia	459
Inne metody klasy Math	461
Klasa StrictMath	462
Klasa Compiler	463
Klasa Thread i ThreadGroup oraz interfejs Runnable	463
Interfejs Runnable	463
Klasa Thread	463
Klasa ThreadGroup	465
Klasy ThreadLocal i InheritableThreadLocal	469
Klasa Package	469
Klasa Module	469
Klasa ModuleLayer	471
Klasa RuntimePermission	471
Klasa Throwable	471
Klasa SecurityManager	471
Klasa StackTraceElement	471
Klasa StackWalker i interfejs StackWalker.StackFrame	471
Klasa Enum	471

Klasa ClassValue	473
Interfejs CharSequence	473
Interfejs Comparable	473
Interfejs Appendable	473
Interfejs Iterable	474
Interfejs Readable	474
Interfejs AutoCloseable	474
Interfejs Thread.UncaughtExceptionHandler	475
Podpakiety pakietu java.lang	475
Podpakiet java.lang.annotation	475
Podpakiet java.lang.instrument	475
Podpakiet java.lang.invoke	475
Podpakiet java.lang.module	475
Podpakiet java.lang.management	476
Podpakiet java.lang.ref	476
Podpakiet java.lang.reflect	476
19 Pakiet java.util, część 1. — kolekcje	477
Wprowadzenie do kolekcji	478
Interfejsy kolekcji	479
Interfejs Collection	480
Interfejs List	482
Interfejs Set	483
Interfejs SortedSet	484
Interfejs NavigableSet	484
Interfejs Queue	485
Interfejs Deque	486
Klasy kolekcji	487
Klasa ArrayList	488
Klasa LinkedList	491
Klasa HashSet	492
Klasa LinkedHashSet	493
Klasa TreeSet	493
Klasa PriorityQueue	495
Klasa ArrayDeque	495
Klasa EnumSet	496
Dostęp do kolekcji za pomocą iteratora	496
Korzystanie z iteratora Iterator	498
Pętla typu for-each jako alternatywa dla iteratora	499
Spliteratorzy	500
Przechowywanie w kolekcjach własnych klas	503
Interfejs RandomAccess	504
Korzystanie z map	504
Interfejsy map	504
Klasy map	510
Komparatory	514
Wykorzystanie komparatora	516
Algorytmy kolekcji	520
Klasa Arrays	525
Starsze klasy i interfejsy	529
Interfejs Enumeration	529
Klasa Vector	530
Klasa Stack	533
Klasa Dictionary	534

Klasa Hashtable	535
Klasa Properties	538
Wykorzystanie metod store() i load()	541
Ostatnie uwagi na temat kolekcji	542
20 Pakiet java.util, część 2. — pozostałe klasy użytkowe	543
Klasa StringTokenizer	543
Klasa BitSet	545
Klasy Optional, OptionalDouble, OptionalInt oraz OptionalLong	547
Klasa Date	550
Klasa Calendar	551
Klasa GregorianCalendar	554
Klasa TimeZone	555
Klasa SimpleTimeZone	556
Klasa Locale	557
Klasa Random	558
Klasy Timer i TimerTask	560
Klasa Currency	562
Klasa Formatter	563
Konstruktory klasy Formatter	564
Metody klasy Formatter	564
Podstawy formatowania	564
Formatowanie łańcuchów i znaków	567
Formatowanie liczb	567
Formatowanie daty i godziny	568
Specyfikatory %n i %%	569
Określanie minimalnej szerokości pola	570
Określanie precyzji	571
Używanie znaczników (flag) formatów	572
Wyrównywanie danych wyjściowych	572
Znaczniki spacji, plusa, zera i nawiasów	573
Znacznik przecinka	574
Znacznik #	574
Opcja wielkich liter	574
Stosowanie indeksu argumentu	575
Zamykanie obiektu klasy Formatter	576
Metoda printf() w Javie	576
Klasa Scanner	576
Konstruktory klasy Scanner	577
Podstawy skanowania	578
Kilka przykładów użycia klasy Scanner	581
Ustawianie separatorów	584
Pozostałe elementy klasy Scanner	585
Klasy ResourceBundle, ListResourceBundle i PropertyResourceBundle	586
Dodatkowe klasy i interfejsy użytkowe	590
Podpakiety pakietu java.util	591
java.util.concurrent, java.util.concurrent.atomic oraz java.util.concurrent.locks	592
java.util.function	592
java.util.jar	594
java.util.logging	594
java.util.prefs	594
java.util.regex	594
java.util.spi	595
java.util.stream	595
java.util.zip	595

21	Operacje wejścia-wyjścia: analiza pakietu java.io	597
	Klasy i interfejsy obsługujące operacje wejścia-wyjścia	598
	Klasa File	598
	Katalogi	601
	Stosowanie interfejsu FilenameFilter	602
	Alternatywna metoda listFiles()	603
	Tworzenie katalogów	603
	Interfejsy AutoCloseable, Closeable i Flushable	603
	Wyjątki operacji wejścia-wyjścia	604
	Dwa sposoby zamykania strumieni	604
	Klasy strumieni	605
	Strumienie bajtów	606
	Klasa InputStream	606
	Klasa OutputStream	607
	Klasa FileInputStream	607
	Klasa FileOutputStream	609
	Klasa ByteArrayInputStream	611
	Klasa ByteArrayOutputStream	612
	Filtrowane strumienie bajtów	614
	Buforowane strumienie bajtów	614
	Klasa SequenceInputStream	617
	Klasa PrintStream	619
	Klasy DataOutputStream i DataInputStream	621
	Klasa RandomAccessFile	622
	Strumienie znaków	623
	Klasa Reader	623
	Klasa Writer	623
	Klasa FileReader	623
	Klasa FileWriter	625
	Klasa CharArrayReader	626
	Klasa CharArrayWriter	627
	Klasa BufferedReader	628
	Klasa BufferedWriter	630
	Klasa PushbackReader	630
	Klasa PrintWriter	631
	Klasa Console	632
	Serializacja	633
	Interfejs Serializable	634
	Interfejs Externalizable	634
	Interfejs ObjectOutput	634
	Klasa ObjectOutputStream	635
	Interfejs ObjectInput	636
	Klasa ObjectInputStream	636
	Przykład serializacji	637
	Korzyści wynikające ze stosowania strumieni	639
22	System NIO	641
	Klasy systemu NIO	641
	Podstawy systemu NIO	642
	Bufory	642
	Kanały	642
	Zestawy znaków i selektory	645

Udoskonalenia dodane w systemie NIO.2	645
Interfejs Path	645
Klasa Files	645
Klasa Paths	648
Interfejsy atrybutów plików	649
Klasy FileSystem, FileSystems i FileStore	650
Stosowanie systemu NIO	651
Stosowanie systemu NIO dla operacji wejścia-wyjścia na kanałach	651
Stosowanie systemu NIO dla operacji wejścia-wyjścia na strumieniach	659
Stosowanie systemu NIO dla operacji na ścieżkach i systemie plików	661
23 Obsługa sieci	669
Podstawy działania sieci	669
Klasy i interfejsy obsługujące komunikację sieciową	670
Klasa InetAddress	671
Metody wytwórcze	671
Metody klasy	672
Klasy InetAddress oraz Inet6Address	673
Gniazda klientów TCP/IP	673
URL	676
Klasa URLConnection	678
Klasa HttpURLConnection	680
Klasa URI	682
Pliki cookie	682
Gniazda serwerów TCP/IP	682
Datagramy	683
Klasa DatagramSocket	683
Klasa DatagramPacket	684
Przykład użycia datagramów	685
24 Obsługa zdarzeń	687
Dwa mechanizmy obsługi zdarzeń	687
Model obsługi zdarzeń oparty na ich delegowaniu	688
Zdarzenia	688
Źródła zdarzeń	688
Obiekty nasłuchujące zdarzeń	689
Klasy zdarzeń	689
Klasa ActionEvent	691
Klasa AdjustmentEvent	691
Klasa ComponentEvent	692
Klasa ContainerEvent	692
Klasa FocusEvent	693
Klasa InputEvent	694
Klasa ItemEvent	694
Klasa KeyEvent	695
Klasa MouseEvent	696
Klasa MouseWheelEvent	697
Klasa TextEvent	698
Klasa WindowEvent	698
Źródła zdarzeń	699
Interfejsy nasłuchujące zdarzeń	700
Interfejs ActionListener	701
Interfejs AdjustmentListener	701
Interfejs ComponentListener	701

Interfejs ContainerListener	701
Interfejs FocusListener	701
Interfejs ItemListener	701
Interfejs KeyListener	701
Interfejs MouseListener	702
Interfejs MouseMotionListener	702
Interfejs MouseWheelListener	702
Interfejs TextListener	702
Interfejs WindowFocusListener	702
Interfejs WindowListener	702
Stosowanie modelu delegowania zdarzeń	703
Kluczowe zagadnienia tworzenia aplikacji graficznych z użyciem AWT	703
Obsługa zdarzeń generowanych przez mysz	704
Obsługa zdarzeń generowanych przez klawiaturę	707
Klasy adapterów	710
Klasy wewnętrzne	712
Anonimowa klasa wewnętrzna	714
25 Wprowadzenie do AWT: praca z oknami, grafiką i tekstem	717
Klasy AWT	718
Podstawy okien	720
Klasa Component	720
Klasa Container	720
Klasa Panel	720
Klasa Window	721
Klasa Frame	721
Klasa Canvas	721
Praca z oknami typu Frame	721
Ustawianie wymiarów okna	721
Ukrywanie i wyświetlanie okna	722
Ustawianie tytułu okna	722
Zamykanie okna typu Frame	722
Metoda paint()	722
Wyświetlanie łańcuchów znaków	722
Określanie koloru tekstu i tła	723
Żądanie ponownego wyświetlenia zawartości okna	723
Tworzenie aplikacji korzystających z klasy Frame	724
Wprowadzenie do stosowania grafiki	725
Rysowanie odcinków	725
Rysowanie prostokątów	725
Rysowanie elips, kół i okręgów	726
Rysowanie łuków	726
Rysowanie wielokątów	726
Prezentacja metod rysujących	727
Dostosowywanie rozmiarów obiektów graficznych	728
Praca z klasą Color	729
Metody klasy Color	730
Ustawianie bieżącego koloru kontekstu graficznego	731
Program demonstrujący zastosowanie klasy Color	731
Ustawianie trybu rysowania	732
Praca z czcionkami	733
Określanie dostępnych czcionek	735
Tworzenie i wybieranie czcionek	736
Uzyskiwanie informacji o czcionkach	738
Zarządzanie tekstowymi danymi wyjściowymi z wykorzystaniem klasy FontMetrics	739

26	Stosowanie kontrolek AWT, menedżerów układu graficznego oraz menu	743
	Podstawy kontrolek AWT	744
	Dodawanie i usuwanie kontrolek	744
	Odpowiadanie na zdarzenia kontrolek	744
	Wyjątek HeadlessException	745
	Etykiety	745
	Stosowanie przycisków	746
	Obsługa zdarzeń przycisków	747
	Stosowanie pól wyboru	750
	Obsługa zdarzeń pól wyboru	751
	Klasa CheckboxGroup	752
	Kontrolki list rozwijanych	754
	Obsługa zdarzeń list rozwijanych	755
	Stosowanie list	756
	Obsługa zdarzeń generowanych przez listy	757
	Zarządzanie paskami przewijania	759
	Obsługa zdarzeń generowanych przez paski przewijania	760
	Stosowanie kontrolek typu TextField	762
	Obsługa zdarzeń generowanych przez kontrolkę TextField	763
	Stosowanie kontrolek typu TextArea	764
	Wprowadzenie do menedżerów układu graficznego komponentów	766
	FlowLayout	767
	BorderLayout	768
	Stosowanie obramowań	769
	GridLayout	771
	Klasa CardLayout	772
	Klasa GridBagLayout	775
	Menu i paski menu	779
	Okna dialogowe	784
	Przestanianie metody paint()	787
27	Obrazy	789
	Formaty plików	789
	Podstawy przetwarzania obrazów: tworzenie, wczytywanie i wyświetlanie	790
	Tworzenie obiektu obrazu	790
	Ładowanie obrazu	790
	Wyświetlanie obrazu	791
	Podwójne buforowanie	792
	Interfejs ImageProducer	794
	Klasa MemoryImageSource	795
	Interfejs ImageConsumer	796
	Klasa PixelGrabber	796
	Klasa ImageFilter	799
	Klasa CropImageFilter	799
	Klasa RGBImageFilter	800
	Dodatkowe klasy obsługujące obrazy	811
28	Narzędzia współbieżności	813
	Pakiety interfejsu Concurrent API	814
	Pakiet java.util.concurrent	814
	Pakiet java.util.concurrent.atomic	815
	Pakiet java.util.concurrent.locks	815
	Korzystanie z obiektów służących do synchronizacji	815
	Klasa Semaphore	815
	Klasa CountDownLatch	820
	CyclicBarrier	822

Klasa Exchanger	824
Klasa Phaser	825
Korzystanie z egzekutorów	832
Przykład prostego egzekutora	832
Korzystanie z interfejsów Callable i Future	834
Typ wyliczeniowy TimeUnit	836
Kolekcje współbieżne	837
Blokady	837
Operacje atomowe	840
Programowanie równoległe przy użyciu frameworku Fork/Join	841
Najważniejsze klasy frameworku Fork/Join	842
Strategia dziel i zwyciężaj	845
Prosty przykład użycia frameworku Fork/Join	845
Znaczenie poziomu równoległości	848
Przykład użycia klasy RecursiveTask<V>	850
Asynchroniczne wykonywanie zadań	852
Anulowanie zadania	853
Określanie statusu wykonania zadania	853
Ponowne uruchamianie zadania	853
Pozostałe zagadnienia	853
Wskazówki dotyczące stosowania frameworku Fork/Join	855
Pakiet Concurrency Utilities a tradycyjne metody języka Java	856
29 API strumieni	857
Podstawowe informacje o strumieniach	857
Interfejsy strumieni	858
Jak można uzyskać strumień?	860
Prosty przykład stosowania strumieni	861
Operacje redukcji	864
Stosowanie strumieni równoległych	866
Odwzorowywanie	868
Tworzenie kolekcji	872
Iteratory i strumień	875
Stosowanie typu Iterator i strumieni	875
Stosowanie spliteratorów	876
Inne możliwości API strumieni	879
30 Wyrażenia regularne i inne pakiety	881
Przetwarzanie wyrażeń regularnych	881
Klasa Pattern	882
Klasa Matcher	882
Składnia wyrażeń regularnych	883
Przykład dopasowywania do wzorca	883
Dwie opcje dopasowywania do wzorca	888
Przegląd wyrażeń regularnych	888
Refleksje	888
Zdalne wywoływanie metod (RMI)	891
Prosta aplikacja typu klient-serwer wykorzystująca RMI	892
Formatowanie dat i czasu przy użyciu pakietu java.text	895
Klasa DateFormat	895
Klasa SimpleDateFormat	896
Interfejs API dat i czasu — java.time	898
Podstawowe klasy do obsługi dat i czasu	898
Formatowanie dat i godzin	900
Analiza łańcuchów zawierających daty i godziny	902
Inne możliwości pakietu java.time	903

CZĘŚĆ III

Wprowadzenie do programowania GUI przy użyciu pakietu Swing

31	Wprowadzenie do pakietu Swing	907
	Geneza powstania biblioteki Swing	907
	Bibliotekę Swing zbudowano na bazie zestawu narzędzi AWT	908
	Podstawowe cechy biblioteki Swing	908
	Komponenty biblioteki Swing są lekkie	908
	Biblioteka Swing obsługuje dołączany wygląd i sposób obsługi	909
	Podobieństwo do architektury MVC	909
	Komponenty i kontenery	910
	Komponenty	910
	Kontenery	911
	Panele kontenerów najwyższego poziomu	911
	Pakiety biblioteki Swing	912
	Prosta aplikacja na bazie biblioteki Swing	912
	Obsługa zdarzeń	916
	Rysowanie w bibliotece Swing	919
	Podstawy rysowania	919
	Wyznaczanie obszaru rysowania	920
	Przykład rysowania	920
32	Przewodnik po pakiecie Swing	923
	Klasy JLabel i ImageIcon	923
	Klasa JTextField	925
	Przyciski biblioteki Swing	926
	Klasa JButton	927
	Klasa JToggleButton	929
	Pola wyboru	930
	Przyciski opcji	932
	Klasa JTabbedPane	934
	Klasa JScrollPane	936
	Klasa JList	938
	Klasa JComboBox	941
	Drzewa	943
	Klasa JTable	945
33	Wprowadzenie do systemu menu pakietu Swing	949
	Podstawy systemu menu	949
	Przegląd klas JMenuBar, JMenu oraz JMenuItem	951
	Klasa JMenuBar	951
	Klasa JMenu	952
	Klasa JMenuItem	953
	Tworzenie menu głównego	953
	Dodawanie mnemonik i kombinacji klawiszy do opcji menu	957
	Dodawanie obrazów i etykiet ekranowych do menu	959
	Stosowanie klas JRadioButtonMenuItem i JCheckBoxMenuItem	960
	Tworzenie menu podręcznych	962
	Tworzenie paska narzędzi	964
	Stosowanie akcji	967
	Finalna postać programu MenuDemo	971
	Dalsze poznawanie pakietu Swing	977

CZĘŚĆ IV**Wprowadzenie do programowania GUI przy użyciu platformy JavaFX**

34	Wprowadzenie do tworzenia interfejsów graficznych z użyciem JavaFX	981
	Podstawowe pojęcia z zakresu JavaFX	982
	Pakiety JavaFX	982
	Klasy Stage i Scene	982
	Węzły i graf sceny	983
	Układy	983
	Klasa Application i metody cyklu życia	983
	Uruchamianie aplikacji JavaFX	984
	Szkielet aplikacji JavaFX	984
	Kompilacja i uruchamianie programów JavaFX	987
	Wątek aplikacji	988
	Prosta kontrolka JavaFX: Label	988
	Stosowanie przycisków i zdarzeń	990
	Podstawowe informacje o zdarzeniach	990
	Prezentacja kontrolki Button	991
	Przedstawienie obsługi zdarzeń i kontrolki Button	991
	Bezpośrednie rysowanie na płótnie	994
35	Prezentacja kontrolki JavaFX	999
	Stosowanie klas Image i ImageView	999
	Dodawanie obrazów do etykiet	1001
	Stosowanie obrazów w przyciskach	1003
	Kontrolka ToggleButton	1005
	Kontrolka RadioButton	1008
	Obsługa zdarzeń w grupie	1010
	Alternatywne sposoby obsługi przycisków opcji	1011
	Kontrolka CheckBox	1014
	Kontrolka ListView	1017
	Paski przewijania w kontrolkach ListView	1020
	Włączanie możliwości wielokrotnego wyboru	1021
	Kontrolka ComboBox	1022
	Kontrolka TextField	1024
	Kontrolka ScrollPane	1026
	Kontrolka TreeView	1029
	Prezentacja efektów i transformacji	1033
	Efekty	1034
	Transformacje	1035
	Prezentacja zastosowania efektów i transformacji	1036
	Dodawanie etykiet ekranowych	1038
	Dezaktywacja kontrolki	1039
36	Prezentacja systemu menu platformy JavaFX	1041
	Podstawowe informacje o menu	1041
	Prezentacja klas MenuBar, Menu oraz MenuItem	1043
	Klasa MenuBar	1043
	Klasa Menu	1043
	Klasa MenuItem	1044
	Tworzenie menu głównego	1045
	Dodawanie mnemonik i akceleratorów do elementów menu	1050
	Dodawanie obrazów do elementów menu	1051
	Stosowanie klas RadioMenuItem i CheckMenuItem	1052

Tworzenie menu podręcznego	1054
Tworzenie paska narzędzi	1057
Kompletna nowa wersja programu demonstracyjnego	1059
Dalsze poznawanie platformy JavaFX	1064

CZĘŚĆ V

Stosowanie Javy w praktyce

37	Java Beans	1067
	Czym jest komponent typu Java Bean?	1067
	Zalety komponentów Java Beans	1068
	Introspekcja	1068
	Wzorce właściwości	1068
	Wzorce projektowe dla zdarzeń	1069
	Metody i wzorce projektowe	1070
	Korzystanie z interfejsu BeanInfo	1070
	Właściwości ograniczone	1070
	Trwałość	1071
	Interfejs Customizer	1071
	Interfejs Java Beans API	1071
	Klasa Introspector	1073
	Klasa PropertyDescriptor	1073
	Klasa EventSetDescriptor	1073
	Klasa MethodDescriptor	1073
	Przykład komponentu Java Bean	1073
38	Serwlety	1077
	Podstawy	1077
	Cykl życia serwletu	1078
	Sposoby tworzenia serwletów	1078
	Korzystanie z serwera Tomcat	1079
	Przykład prostego serwletu	1080
	Tworzenie i kompilacja kodu źródłowego serwletu	1080
	Uruchamianie serwera Tomcat	1081
	Uruchamianie przeglądarki i generowanie żądania	1081
	Interfejs Servlet API	1081
	Pakiet javax.servlet	1081
	Interfejs Servlet	1082
	Interfejs ServletConfig	1082
	Interfejs ServletContext	1083
	Interfejs ServletRequest	1083
	Interfejs ServletResponse	1083
	Klasa GenericServlet	1083
	Klasa ServletInputStream	1083
	Klasa ServletOutputStream	1085
	Klasy wyjątków związanych z serwletami	1085
	Odczytywanie parametrów serwletu	1085
	Pakiet javax.servlet.http	1086
	Interfejs HttpServletRequest	1087
	Interfejs HttpServletResponse	1087
	Interfejs HttpSession	1087
	Klasa Cookie	1089
	Klasa HttpServlet	1090

Obsługa żądań i odpowiedzi HTTP	1090
Obsługa żądań GET protokołu HTTP	1090
Obsługa żądań POST protokołu HTTP	1092
Korzystanie ze znaczników kontekstu użytkownika	1093
Śledzenie sesji	1095

DODATKI

A Komentarze dokumentujące	1099
Znaczniki narzędzia javadoc	1099
Znacznik @author	1100
Znacznik {@code}	1100
Znacznik @deprecated	1101
Znacznik {@docRoot}	1101
Znacznik @exception	1101
Znacznik @hidden	1101
Znacznik {@index}	1101
Znacznik {@inheritDoc}	1101
Znacznik {@link}	1101
Znacznik {@linkplain}	1102
Znacznik {@literal}	1102
Znacznik @param	1102
Znacznik @provides	1102
Znacznik @return	1102
Znacznik @see	1102
Znacznik @serial	1102
Znacznik @serialData	1103
Znacznik @serialField	1103
Znacznik @since	1103
Znacznik @throws	1103
Znacznik @uses	1103
Znacznik {@value}	1103
Znacznik @version	1103
Ogólna postać komentarzy dokumentacyjnych	1104
Wynik działania narzędzia javadoc	1104
Przykład korzystający z komentarzy dokumentacyjnych	1104
B Przegląd technologii Java Web Start	1107
Czym jest Java Web Start?	1107
Cztery kluczowe aspekty Java Web Start	1108
Aplikacje Java Web Start wymagają pliku JAR	1108
Aplikacje Java Web Start są podpisywane cyfrowo	1108
Java Web Start bazuje na JNLP	1109
Tworzenie odnośnika do pliku JNLP	1110
Eksperymenty z Java Web Start z wykorzystaniem lokalnego systemu plików	1111
Utworzenie pliku JAR aplikacji ToggleButtonDemo	1112
Utworzenie magazynu kluczy i podpisanie pliku ButtonDemo.jar	1112
Utworzenie pliku JNLP dla aplikacji ToggleButtonDemo	1113
Utworzenie pliku HTML o nazwie StartTBD.html	1114
Dodanie pliku ToggleButtonDemo.jnlp	1114
do listy wyjątków w aplikacji Java Control Panel	1114
Wykonanie aplikacji ButtonDemo z poziomu przeglądarki	1114
Uruchamianie aplikacji Java Web Start przy użyciu programu javaws	1115
Stosowanie technologii Java Web Start z apletami	1115

C	Wprowadzenie do JShell	1117
	Podstawy JShell	1117
	Wyświetlanie, edytowanie i ponowne wykonywanie kodu	1119
	Dodanie metody	1120
	Utworzenie klasy	1121
	Stosowanie interfejsu	1121
	Przetwarzanie wyrażeń i wbudowane zmienne	1122
	Importowanie pakietów	1123
	Wyjątki	1124
	Inne polecenia JShell	1124
	Dalsze poznawanie możliwości JShell	1125
D	Podstawy apletów	1127
	Dwa rodzaje apletów	1127
	Podstawy apletów	1128
	Klasa Applet	1129
	Architektura apletów	1129
	Szkielet apletu	1129
	Inicjalizacja i zakończenie działania apletu	1131
	Aplety Swinga	1132
	Skorowidz	1135