

[opis]

Java jest ceniona przez programistów za dojrzałość i nowoczesność. Jako jeden z nielicznych języków miała zasadniczy wpływ na ewolucję programowania i określenie standardów tworzenia kodu. Cały czas jest dynamicznie rozwijana i od lat pozostaje w awangardzie innowacyjności. Na uwagę zasługuje wszechstronność Javy: to język, który pozwala na tworzenie stron WWW, aplikacji webowych, mobilnych oraz wysoko specjalistycznego oprogramowania dla przeróżnych urządzeń technicznych. Jest to możliwe między innymi dzięki licznym bibliotekom i narzędziom wspierającym proces kodowania. Mimo upływu lat Java wciąż pozostaje bardzo mądrym wyborem dla przyszłych deweloperów.

Ta książka jest ósmym wydaniem znakomitego podręcznika programowania dla początkujących, starannie zaktualizowanym i uzupełnionym o informacje dotyczące edycji 11. Zawiera podstawową wiedzę o kompilacji i uruchamianiu programu w Javie oraz o słowach kluczowych i istotnych konstrukcjach w tym języku - to na początek. Potem krok po kroku przedstawia kolejne, coraz bardziej zaawansowane zagadnienia dotyczące obiektów, dziedziczenia czy wyjątków, a także współbieżności, typów sparametryzowanych, wyrażeń lambda oraz modułów. Poszczególne partie materiału zostały bogato uzupełnione przykładami kodu z komentarzami, jak również praktycznymi ćwiczeniami, testami sprawdzającymi, wskazówkami i dodatkowymi informacjami. Podręcznik jest przejrzysty, napisany jasnym i zrozumiałym językiem, co zdecydowanie ułatwia naukę.

W tej książce między innymi:

- solidne podstawy Javy i programowania obiektowego
- operacje wejścia-wyjścia oraz obsługa wyjątków
- programowanie wielowątkowe
- zagadnienia zaawansowane: wyrażenia lambda i referencje metod
- moduły, usługi i biblioteka Swing

Java: zdobądź kluczowe umiejętności i utrwal najlepsze nawyki programisty!

[spis]

O autorze 13

O redaktorze merytorycznym 14

Wstęp 15

1. Podstawy Javy 21

- Historia i filozofia Javy 22
 - Pochodzenie Javy 22
 - Java a języki C i C++ 23
 - Wpływ Javy na Internet 23
 - Magiczny kod bajtowy 25
 - Coś więcej niż aplety 26

- Szybszy harmonogram udostępniania 27
 - Terminologia Javy 27
- Programowanie obiektowe 27
 - Hermetyzacja 29
 - Polimorfizm 29
 - Dziedziczenie 29
- Java Development Kit 30
- Pierwszy prosty program 31
 - Wprowadzenie tekstu programu 31
 - Kompilowanie programu 32
 - Pierwszy program wiersz po wierszu 33
- Obsługa błędów składni 34
- Drugi prosty program 35
- Inne typy danych 37
- Przykład 1.1. Zamiana galonów na litry 38
- Dwie instrukcje sterujące 39
 - Instrukcja if 39
 - Pętla for 40
- Bloki kodu 41
- Średnik i pozycja kodu w wierszu 42
- Wcięcia 43
- Przykład 1.2. Ulepszony konwerter galonów na litry 43
- Słowa kluczowe języka Java 44
- Identyfikatory 45
- Biblioteki klas 45
- Test sprawdzający 46

2. Typy danych i operatory 47

- Dlaczego typy danych są tak ważne 47
- Typy proste 48
 - Typy całkowite 48
 - Typy zmiennoprzecinkowe 49
 - Znaki 50
- Typ logiczny 51
- Przykład 2.1. Jak daleko uderzył piorun? 52
- Literały 53
 - Literały szesnastkowe, ósemkowe i binarne 53
 - Specjalne sekwencje znaków 53
 - Literały łańcuchowe 54
- Zmienne 55
 - Inicjalizacja zmiennej 55
 - Dynamiczna inicjalizacja 55
- Zasięg deklaracji i czas istnienia zmiennych 56
- Operatory 58
- Operatory arytmetyczne 58
 - Inkrementacja i dekrementacja 59
- Operatory relacyjne i logiczne 60
- Warunkowe operatory logiczne 62
- Operator przypisania 63

- Skrótowe operatory przypisania 63
- Konwersje typów w instrukcjach przypisania 64
- Rzutowanie typów niezgodnych 65
- Priorytet operatorów 67
- Przykład 2.2. Tabela prawdy dla operatorów logicznych 67
- Wyrażenia 68
 - Konwersja typów w wyrażeniach 68
 - Odstępy i nawiasy 70
- Test sprawdzający 70

3. Instrukcje sterujące 71

- Wprowadzanie znaków z klawiatury 71
- Instrukcja if 72
- Zagnieżdżanie instrukcji if 73
- Drabinka if-else-if 74
- Instrukcja switch 75
- Zagnieżdżanie instrukcji switch 78
- Przykład 3.1. Rozpoczynamy budowę systemu pomocy 78
- Pętla for 80
- Wariacje na temat pętli for 81
- Brakujące elementy 82
 - Pętla nieskończona 83
- Pętle bez ciała 83
- Deklaracja zmiennych sterujących wewnątrz pętli 84
- Rozszerzona pętla for 85
- Pętla while 85
- Pętla do-while 86
- Przykład 3.2. Ulepszamy system pomocy 88
- Przerwanie pętli instrukcją break 90
- Zastosowanie break jako formy goto 91
- Zastosowanie instrukcji continue 95
- Przykład 3.3. Końcowa wersja systemu pomocy 96
- Pętle zagnieżdżone 99
- Test sprawdzający 99

4. Wprowadzenie do klas, obiektów i metod 101

- Podstawy klas 101
 - Ogólna postać klasy 102
 - Definiowanie klasy 102
- Jak powstają obiekty 105
- Referencje obiektów i operacje przypisania 105
- Metody 106
 - Dodajemy metodę do klasy Vehicle 106
- Powrót z metody 108
- Zwracanie wartości 109
- Stosowanie parametrów 110
 - Dodajemy sparametryzowaną metodę do klasy Vehicle 112
- Przykład 4.1. System pomocy jako klasa 113

- Konstruktory 117
- Konstruktory z parametrami 118
- Dodajemy konstruktor do klasy Vehicle 119
- Operator new 120
- Odzyskiwanie pamięci 120
- Słowo kluczowe this 121
- Test sprawdzający 122

5. Więcej typów danych i operatorów 123

- Tablice 123
 - Tablice jednowymiarowe 124
- Przykład 5.1. Sortowanie tablicy 126
- Tablice wielowymiarowe 128
 - Tablice dwuwymiarowe 128
 - Tablice nieregularne 129
 - Tablice o trzech i więcej wymiarach 130
 - Inicjalizacja tablic wielowymiarowych 130
- Alternatywna składnia deklaracji tablic 131
- Przypisywanie referencji tablic 131
- Wykorzystanie składowej length 132
- Przykład 5.2. Klasa Queue 134
- Styl for-each pętli for 137
 - Iteracje w tablicach wielowymiarowych 139
 - Zastosowania rozszerzonej pętli for 140
- Łańcuchy znaków 141
 - Tworzenie łańcuchów 141
 - Operacje na łańcuchach 142
 - Tablice łańcuchów 144
 - Łańcuchy są niezmiennie 144
 - Stosowanie łańcuchów do sterowania instrukcją switch 145
- Wykorzystanie argumentów wywołania programu 146
- Stosowanie wnioskowania typów w zmiennych lokalnych 147
 - Wnioskowanie typów zmiennych lokalnych w przypadku typów referencyjnych 149
 - Stosowanie wnioskowania typów zmiennych lokalnych w pętlach 150
 - Ograniczenia var 151
- Operatory bitowe 152
 - Operatory bitowe AND, OR, XOR i NOT 152
 - Operatory przesunięcia 156
 - Skrótowe bitowe operatory przypisania 157
- Przykład 5.3. Klasa ShowBits 158
- Operator ? 160
- Test sprawdzający 161

6. Więcej o metodach i klasach 163

- Kontrola dostępu do składowych klasy 163
 - Modyfikatory dostępu w Javie 164
- Przykład 6.1. Ulepszamy klasę Queue 167

- Przekazywanie obiektów do metod 168
 - Sposób przekazywania argumentów 169
- Zwracanie obiektów 171
- Przeciążanie metod 173
- Przeciążanie konstruktorów 177
- Przykład 6.2. Przeciążamy konstruktor klasy Queue 178
- Rekurencja 181
- Słowo kluczowe static 182
 - Bloki static 185
- Przykład 6.3. Algorytm Quicksort 186
- Klasy zagnieżdżone i klasy wewnętrzne 188
- Zmienne liczby argumentów 190
 - Metody o zmiennej liczbie argumentów 191
 - Przeciążanie metod o zmiennej liczbie argumentów 193
 - Zmienna liczba argumentów i niejednoznaczność 194
- Test sprawdzający 195

7. Dziedziczenie 197

- Podstawy dziedziczenia 197
- Dostęp do składowych a dziedziczenie 200
- Konstruktory i dziedziczenie 202
- Użycie słowa kluczowego super do wywołania konstruktora klasy bazowej 203
- Użycie słowa kluczowego super do dostępu do składowych klasy bazowej 207
- Przykład 7.1. Tworzymy hierarchię klas Vehicle 207
- Wielopoziomowe hierarchie klas 210
- Kiedy wywoływane są konstruktory? 212
- Referencje klasy bazowej i obiekty klasy pochodnej 213
- Przesłanianie metod 217
- Przesłanianie metod i polimorfizm 219
- Po co przesłaniać metody? 220
 - Zastosowanie przesłaniania metod w klasie TwoDShape 221
- Klasy abstrakcyjne 224
- Słowo kluczowe final 227
 - final zapobiega przesłanianiu 227
 - final zapobiega dziedziczeniu 227
 - Użycie final dla zmiennych składowych 228
- Klasa Object 229
- Test sprawdzający 230

8. Pakiety i interfejsy 231

- Pakiety 231
 - Definiowanie pakietu 232
 - Wyszukiwanie pakietów i zmienna CLASSPATH 233
 - Prosty przykład pakietu 233
- Pakiety i dostęp do składowych 234
 - Przykład dostępu do pakietu 235
- Składowe chronione 236
- Import pakietów 238

- Biblioteka klas Java używa pakietów 239
- Interfejsy 239
- Implementacje interfejsów 241
- Referencje interfejsu 244
- Przykład 8.1. Tworzymy interfejs Queue 245
- Zmienne w interfejsach 249
- Interfejsy mogą dziedziczyć 250
- Domyślne metody interfejsów 251
 - Podstawowe informacje o metodach domyślnych 252
 - Praktyczny przykład metody domyślnej 253
 - Problemy wielokrotnego dziedziczenia 254
- Stosowanie metod statycznych w interfejsach 255
- Stosowanie metod prywatnych w interfejsach 256
- Ostatnie uwagi o pakietach i interfejsach 257
- Test sprawdzający 257

9. Obsługa wyjątków 259

- Hierarchia wyjątków 260
- Podstawy obsługi wyjątków 260
 - Słowa kluczowe try i catch 260
 - Prosty przykład wyjątku 261
- Konsekwencje nieprzechwycenia wyjątku 263
 - Wyjątki umożliwiają obsługę błędów 264
- Użycie wielu klauzul catch 265
- Przechwytywanie wyjątków klas pochodnych 265
- Zagnieżdżanie bloków try 267
- Generowanie wyjątku 268
 - Powtórne generowanie wyjątku 268
- Klasa Throwable 269
- Klauzula finally 271
- Użycie klauzuli throws 272
- Trzy dodatkowe możliwości wyjątków 273
- Wyjątki wbudowane w Javę 274
- Tworzenie klas pochodnych wyjątków 276
- Przykład 9.1. Wprowadzamy wyjątki w klasie Queue 278
- Test sprawdzający 280

10. Obsługa wejścia i wyjścia 283

- Strumień wejścia i wyjścia 284
- Strumień bajtowe i strumień znakowe 284
- Klasy strumieni bajtowych 284
- Klasy strumieni znakowych 284
- Strumień predefiniowane 285
- Używanie strumieni bajtowych 286
 - Odczyt wejścia konsoli 286
 - Zapis do wyjścia konsoli 288
- Odczyt i zapis plików za pomocą strumieni bajtowych 289
 - Odczyt z pliku 289

- Zapis w pliku 292
- Automatyczne zamykanie pliku 294
- Odczyt i zapis danych binarnych 297
- Przykład 10.1. Narzędzie do porównywania plików 299
- Pliki o dostępie swobodnym 300
- Strumienie znakowe 302
 - Odczyt konsoli za pomocą strumieni znakowych 303
 - Obsługa wyjścia konsoli za pomocą strumieni znakowych 305
- Obsługa plików za pomocą strumieni znakowych 306
 - Klasa FileWriter 307
 - Klasa FileReader 307
- Zastosowanie klas opakowujących do konwersji łańcuchów numerycznych 309
- Przykład 10.2. System pomocy wykorzystujący pliki 310
- Test sprawdzający 316

11. Programowanie wielowątkowe 317

- Podstawy wielowątkowości 317
- Klasa Thread i interfejs Runnable 318
- Tworzenie wątku 319
 - Usprawnienie i dwie modyfikacje 322
- Przykład 11.1. Tworzymy klasę pochodną klasy Thread 325
- Tworzenie wielu wątków 327
- Jak ustalić, kiedy wątek zakończył działanie? 330
- Priorytety wątków 332
- Synchronizacja 335
- Synchronizacja metod 335
- Synchronizacja instrukcji 338
- Komunikacja międzywątkowa 340
 - Przykład użycia metod wait() i notify() 340
- Wstrzymywanie, wznowianie i kończenie działania wątków 345
- Przykład 11.2. Wykorzystanie głównego wątku 348
- Test sprawdzający 349

12. Typy wyliczeniowe, automatyczne opakowywanie, import składowych statycznych i adnotacje 351

- Wyliczenia 352
 - Podstawy wyliczeń 352
- Wyliczenia są klasami 354
- Metody values() i valueOf() 354
- Konstruktory, metody, zmienne instancji a wyliczenia 355
 - Dwa ważne ograniczenia 357
- Typy wyliczeniowe dziedziczą po klasie Enum 357
- Przykład 12.1. Komputerowo sterowana sygnalizacja świetlna 358
- Automatyczne opakowywanie 362
- Typy opakowujące 363
- Podstawy automatycznego opakowywania 364
- Automatyczne opakowywanie i metody 365
- Automatyczne opakowywanie i wyrażenia 366

- Przestroga 368
- Import składowych statycznych 368
- Adnotacje (metadane) 370
- Test sprawdzający 373

13. Typy sparametryzowane 375

- Podstawy typów sparametryzowanych 376
- Prosty przykład typów sparametryzowanych 376
 - Parametryzacja dotyczy tylko typów obiektowych 379
 - Typy sparametryzowane różnią się dla różnych argumentów 379
 - Klasa sparametryzowana o dwóch parametrach 380
 - Ogólna postać klasy sparametryzowanej 381
- Ograniczanie typów 381
- Stosowanie argumentów wieloznacznych 384
- Ograniczanie argumentów wieloznacznych 386
- Metody sparametryzowane 388
- Konstruktory sparametryzowane 390
- Interfejsy sparametryzowane 390
- Przykład 13.1. Sparametryzowana klasa Queue 393
- Typy surowe i tradycyjny kod 396
- Wnioskowanie typów i operator diamentowy 399
- Wnioskowanie typów zmiennych lokalnych a typy sparametryzowane 400
- Wymazywanie 400
- Błędy niejednoznaczności 401
- Ograniczenia związane z typami sparametryzowanymi 401
 - Zakaz tworzenia instancji parametru typu 402
 - Ograniczenia dla składowych statycznych 402
 - Ograniczenia tablic sparametryzowanych 402
 - Ograniczenia związane z wyjątkami 403
- Dalsze studiowanie typów sparametryzowanych 403
- Test sprawdzający 403

14. Wyrażenia lambda i referencje metod 405

- Przedstawienie wyrażeń lambda 406
 - Podstawowe informacje o wyrażeniach lambda 406
 - Interfejsy funkcyjne 407
 - Wyrażenia lambda w działaniu 409
- Blokowe wyrażenia lambda 412
- Sparametryzowane interfejsy funkcyjne 414
- Przykład 14.1. Przekazywanie wyrażenia lambda jako argumentu 415
- Wyrażenia lambda i przechwytywanie zmiennych 419
- Zgłaszanie wyjątków w wyrażeniach lambda 420
- Referencje metod 421
 - Referencje metod statycznych 422
 - Referencje metod instancyjnych 423
- Referencje konstruktorów 427
- Predefiniowane interfejsy funkcyjne 429
- Test sprawdzający 430

15. Moduły 433

- Podstawowe informacje o modułach 434
 - Przykład prostego modułu 435
 - Kompilowanie i uruchamianie przykładowej aplikacji 438
 - Dokładniejsze informacje o instrukcjach requires i exports 439
- java.base i moduły platformy 440
- Stary kod i moduł nienazwany 441
- Eksportowanie do konkretnego modułu 442
- Wymagania przechodnie 443
- Przykład 15.1. Eksperymenty z instrukcją requires transitive 444
- Stosowanie usług 447
 - Podstawowe informacje o usługach i dostawcach usług 447
 - Słowa kluczowe związane z usługami 448
 - Przykład stosowania usług i modułów 448
- Dodatkowe cechy modułów 454
 - Moduły otwarte 455
 - Instrukcja opens 455
 - requires static 455
- Dalsze samodzielne poznawanie modułów 456
- Test sprawdzający 456

16. Wprowadzenie do biblioteki Swing 459

- Pochodzenie i filozofia Swing 460
- Komponenty i kontenery 461
 - Komponenty 461
 - Kontenery 462
 - Panele kontenerów szczytowych 462
- Menedżery układu 463
- Pierwszy program wykorzystujący Swing 463
 - Pierwszy program Swing wiersz po wierszu 465
- Obsługa zdarzeń w Swing 467
 - Zdarzenia 468
 - Źródła zdarzeń 468
 - Obiekty nasłuchujące 468
 - Klasy zdarzeń i interfejsy obiektów nasłuchujących 469
- Komponent JButton 469
- Komponent JTextField 472
- Komponent JCheckBox 475
- Komponent JList 477
- Przykład 16.1. Porównywanie plików - aplikacja Swing 480
- Wykorzystanie anonimowych klas wewnętrznych lub wyrażeń lambda do obsługi zdarzeń 485
- Test sprawdzający 486

A. Rozwiązania testów sprawdzających 489

- Rozdział 1. Podstawy Javy 489
- Rozdział 2. Typy danych i operatory 491

- Rozdział 3. Instrukcje sterujące 492
- Rozdział 4. Wprowadzenie do klas, obiektów i metod 494
- Rozdział 5. Więcej typów danych i operatorów 495
- Rozdział 6. Więcej o metodach i klasach 499
- Rozdział 7. Dziedziczenie 502
- Rozdział 8. Pakiety i interfejsy 504
- Rozdział 9. Obsługa wyjątków 506
- Rozdział 10. Obsługa wejścia i wyjścia 508
- Rozdział 11. Programowanie wielowątkowe 511
- Rozdział 12. Typy wyliczeniowe, automatyczne opakowywanie, import składowych statycznych i adnotacje 513
- Rozdział 13. Typy sparametryzowane 516
- Rozdział 14. Wyrażenia lambda i referencje metod 519
- Rozdział 15. Moduły 522
- Rozdział 16. Wprowadzenie do Swing 523

B. Komentarze dokumentacyjne 529

- Znaczniki javadoc 529
 - @author 530
 - {@code} 530
 - @deprecated 531
 - {@docRoot} 531
 - @exception 531
 - @hidden 531
 - {@index} 531
 - {@inheritDoc} 531
 - {@link} 531
 - {@linkplain} 532
 - {@literal} 532
 - @param 532
 - @provides 532
 - @return 532
 - @see 532
 - @since 533
 - {@summary} 533
 - @throws 533
 - @uses 533
 - {@value} 533
 - @version 533
- Ogólna postać komentarza dokumentacyjnego 533
- Wynik działania programu javadoc 534
- Przykład użycia komentarzy dokumentacyjnych 534

C. Kompiluj i uruchamiaj proste programy w jednym kroku 537

D. Wprowadzenie do JShell 539

- Podstawy JShell 539
- Wyświetlanie, edycja i ponowne wykonywanie kodu 541

- Dodanie metody 542
- Utworzenie klasy 543
- Stosowanie interfejsu 543
- Przetwarzanie wyrażeń i wbudowane zmienne 544
- Importowanie pakietów 545
- Wyjątki 546
- Inne polecenia JShell 546
- Dalsze poznawanie możliwości JShell 547

E. Więcej słów kluczowych języka Java 549

- Modyfikatory transient i volatile 549
- instanceof 550
- strictfp 550
- assert 550
- Metody rodzime 551
- Inna postać this 551