

Przedmowa .....	10
<b>Część I. Podstawy programowania funkcyjnego</b> .....	<b>19</b>
<b>1. Wprowadzenie do programowania funkcyjnego</b> .....	<b>21</b>
Co sprawia, że język jest funkcyjny?	21
Koncepcje programowania funkcyjnego	23
Funkcje czyste i transparentność referencyjna	24
Niemutowalność	25
Rekurencja	25
Typy pierwszoklasowe i funkcje wyższego rzędu	26
Kompozycja funkcyjna	27
Rozwijanie funkcji	27
Częściowe zastosowanie funkcji	28
Ewaluacja leniwa	28
Zalety programowania funkcyjnego	29
Wady programowania funkcyjnego	30
Podsumowanie	31
<b>2. Funkcyjna Java</b> .....	<b>32</b>
Czym są wyrażenia lambda Javy?	32
Składnia wyrażeń lambda	32
Interfejsy funkcyjne	33
Wyrażenia lambda i zmienne zewnętrzne	36
A co z klasami anonimowymi?	39
Wyrażenia lambda w działaniu	42
Tworzenie wyrażeń lambda	42
Wywoływanie wyrażeń lambda	43
Odwoływanie się do metod	44

Koncepcje programowania funkcyjnego w Javie	47
Funkcje czyste i transparentność referencyjna	48
Niemutowalność	50
Typy pierwszoklasowe	51
Kompozycja funkcyjna	52
Ewaluacja leniwa	52
Podsumowanie	53
<b>3. Interfejsy funkcyjne pakietu JDK .....</b>	<b>54</b>
Cztery główne kategorie interfejsów funkcyjnych	54
Funkcje	54
Konsumenty	55
Dostawcy	56
Predykaty	56
Dlaczego jest tak wiele wariantów interfejsu funkcyjnego?	57
Argumentowość funkcji	57
Typy proste	59
Wypełnianie luki między interfejsami funkcyjnymi	61
Kompozycja funkcyjna	62
Rozszerzanie wsparcia funkcyjnego	63
Dodawanie metod domyślnych	64
Bezpośrednie implementowanie interfejsu funkcyjnego	65
Tworzenie statycznych metod pomocniczych	67
Podsumowanie	70

---

## **Część II. Podejście funkcyjne** **71**

<b>4. Niemutowalność .....</b>	<b>73</b>
Mutowalność i struktury danych w OOP	73
Niemutowalność (nie tylko) w programowaniu funkcyjnym	75
Stan niemutowalności Javy	77
java.lang.String	77
Niemutowalne kolekcje	79
Typy proste i obiektowe typy opakowujące	82
Niemutowalna matematyka	82
Java Time API (JSR-310)	83
Typy wyliczeniowe	84
Słowo kluczowe final	84
Rekordy	85
Jak osiągnąć niemutowalność?	86
Powszechne praktyki	87
Podsumowanie	88

<b>5. Praca z rekordami .....</b>	<b>89</b>
Typy agregacji danych	89
Krotki	89
Prosty POJO	90
Od POJO do niemutowalności	92
Od POJO do rekordu	93
Rekordy na ratunek	93
Mechanizmy wewnętrzne	95
Funkcjonalności rekordu	96
Brakujące funkcjonalności	100
Przypadki użycia i powszechne praktyki	106
Walidacja rekordów i oczyszczanie danych	106
Zwiększanie poziomu niemutowalności	107
Tworzenie zmodyfikowanych kopii	108
Rekordy jako lokalne krotki nominalne	111
Lepsza obsługa opcjonalnych danych	114
Serializowanie ewoluujących rekordów	115
Dopasowywanie wzorców rekordów (od Javy 19)	118
Jeszcze kilka słów na temat rekordów	119
Podsumowanie	120
<b>6. Przetwarzanie danych za pomocą strumieni .....</b>	<b>121</b>
Przetwarzanie danych za pomocą iteracji	121
Iteracja zewnętrzna	121
Iteracja wewnętrzna	124
Strumienie jako funkcyjne potoki danych	125
Funkcjonalności strumieni	127
Spliterator, kręgosłup strumieni	132
Budowanie potoków strumieniowych	134
Tworzenie strumienia	135
Wykonywanie pracy	136
Kończenie strumienia	143
Koszt operacji	155
Modyfikowanie zachowania strumienia	157
Używać strumienia czy nie?	158
Podsumowanie	160
<b>7. Praca ze strumieniami .....</b>	<b>161</b>
Strumieniowe typy proste	161
Strumienie iteracyjne	163
Strumienie nieskończone	165
Liczby losowe	166
Pamięć nie jest nieskończona	167

Od tablic do strumieni i z powrotem	168
Tablice typu obiektowego	168
Tablicowe typy proste	169
Niskopoziomowe tworzenie strumieni	169
Praca z operacjami we-wy plików	170
Odczytywanie zawartości katalogów	171
Przechodzenie katalogów w głąb	171
Przeszukiwanie systemu plików	173
Odczytywanie plików linia po linii	174
Zastrzeżenia dotyczące strumieni z operacjami we-wy plików	175
Obsługa daty i czasu	177
Kwerendowanie typów czasowych	177
Strumienie z zakresem <code>LocalDate</code>	178
Pomiar wydajności strumienia za pomocą narzędzia JMH	178
Kolektory	179
Kolektory downstreamowe	179
Tworzenie własnego kolektora	188
Jeszcze kilka słów na temat strumieni (sekwencyjnych)	190
Podsumowanie	190
<b>8. Równoległe przetwarzanie danych za pomocą strumieni .....</b>	<b>191</b>
Współbieżność a równoległość	191
Strumienie jako równoległe potoki funkcyjne	193
Strumienie równoległe w akcji	194
Kiedy używać, a kiedy unikać strumieni równoległych?	197
Wybór odpowiedniego źródła danych	198
Liczba elementów	199
Operacje strumieniowe	200
Koszty ogólne strumienia i dostępne zasoby	203
Przykład — jeszcze raz <i>Wojna i pokój</i>	204
Przykład — liczby losowe	205
Lista kontrolna dla strumieni równoległych	207
Podsumowanie	208
<b>9. Obsługa wartości null za pomocą typów <code>Optional</code> .....</b>	<b>210</b>
Problem z zerowymi referencjami	210
Jak obsługiwać wartość null w Javie (przed wprowadzeniem <code>Optional</code> )?	213
Najlepsze praktyki w zakresie obsługi wartości null	213
Kontrolę null z użyciem narzędzi	216
Typy wyspecjalizowane, takie jak <code>Optional</code>	217
Typy opcyjne na ratunek	217
Czym jest <code>Optional</code> ?	217
Tworzenie potoków opcyjnych	220

Opcje i strumienie	227
Typy opcyjne jako elementy strumienia	227
Operacje terminalne strumieni	228
Opcyjne typy proste	230
Zastrzeżenia	231
Opcje są zwykłymi typami	231
Metody uwzględniające tożsamość	232
Narzut związany z wydajnością	232
Szczególne kwestie dotyczące kolekcji	233
Opcje i serializacja	234
Jeszcze kilka słów na temat referencji null	234
Podsumowanie	235
<b>10. Obsługa wyjątków funkcyjnych .....</b>	<b>236</b>
Obsługa wyjątków Javy w pigułce	236
Blok try-catch	237
Różne rodzaje wyjątków i błędów	237
Wyjątki sprawdzane w wyrażeniach lambda	239
Wyodrębnianie kodu do postaci bezpiecznych metod	241
Anulowanie sprawdzania wyjątków	242
Sneaky throws	243
Funkcyjne podejście do wyjątków	245
Nierzucanie wyjątków	245
Błędy jako wartości	246
Wzorzec Próba, Powodzenie, Niepowodzenie	251
Jeszcze kilka słów na temat funkcyjnej obsługi wyjątków	256
Podsumowanie	257
<b>11. Ewaluacja leniwa .....</b>	<b>258</b>
Porównanie leniwości i gorliwości	258
Jak gorliwa jest Java?	259
Ewaluacja minimalna	260
Struktury sterowania	261
Leniwe typy pakietu JDK	261
Wyrażenia lambda i funkcje wyższego rzędu	263
Podejście gorliwe	263
Podejście bardziej leniwe	264
Podejście funkcyjne	265
Opóźnianie wykonywania za pomocą typu Thunk	266
Tworzenie prostego typu Thunk	266
Thunk bezpieczny dla wątków	268
Jeszcze kilka słów na temat leniwości	270
Podsumowanie	271

<b>12. Rekurencja .....</b>	<b>273</b>
Czym jest rekurencja?	273
Porównanie rekurencji głowowej i ogonowej	274
Rekurencja i stos wywołań	276
Bardziej złożony przykład	277
Iteracyjne przechodzenie drzewa	278
Rekurencyjne przechodzenie drzewa	279
Strumienie rekurencyjne	281
Jeszcze kilka słów na temat rekurencji	281
Podsumowanie	283
<b>13. Zadania asynchroniczne .....</b>	<b>284</b>
Porównanie zadań synchronicznych i asynchronicznych	284
Typy Future Javy	285
Projektowanie potoków asynchronicznych przy użyciu typu CompletableFuture	287
Obiecywanie wartości	287
Tworzenie instancji CompletableFuture	288
Komponowanie i łączenie zadań	289
Obsługa wyjątków	292
Operacje terminalne	294
Tworzenie metody pomocniczej CompletableFuture	295
Ręczne tworzenie i rozwiązywanie	299
Ręczne tworzenie	299
Ręczne rozwiązywanie	300
Przypadki użycia dla ręcznie utworzonych i ukończonych instancji	300
Pule wątków i limity czasu	304
Jeszcze kilka słów na temat zadań asynchronicznych	305
Podsumowanie	306
<b>14. Funkcyjne wzorce projektowe .....</b>	<b>308</b>
Czym są wzorce projektowe?	308
(Funkcyjne) wzorce projektowe	309
Wzorzec Metoda Wytwórcza	309
Wzorzec Dekorator	312
Wzorzec Strategia	317
Wzorzec Budowniczy	319
Jeszcze kilka słów na temat funkcyjnych wzorców projektowych	324
Podsumowanie	325

<b>15. Funkcyjne podejście do Javy .....</b>	<b>326</b>
Porównanie zasad OOP i FP	326
Funkcyjny sposób myślenia	327
Funkcje są typami pierwszoklasowymi	328
Unikanie skutków ubocznych	329
Funkcyjne przetwarzanie danych za pomocą mapowania, filtrowania i redukcji	336
Implementacje opierają się na abstrakcjach	336
Budowanie funkcyjnych pomostów	337
Ułatwienie równoległości i współbieżności	342
Uwaga na potencjalny narzut	343
Architektura funkcyjna w świecie imperatywnym	344
Od obiektów do wartości	346
Separacja zagadnień	346
Różne rozmiary FC/IS	348
Testowanie FC/IS	349
Jeszcze kilka słów na temat funkcyjnego podejścia do Javy	350
Podsumowanie	351