
Spis treści

Słowo wstępne	11
Przedmowa	13
Podziękowania	17
Rozdział 1. Wprowadzenie	21
Rozdział 2. Tworzenie i usuwanie obiektów	25
Temat 1. Tworzenie statycznych metod factory zamiast konstruktorów	25
Temat 2. Zastosowanie budowniczego do obsługi wielu parametrów konstruktora	31
Temat 3. Wymuszanie właściwości singleton za pomocą prywatnego konstruktora	37
Temat 4. Wykorzystanie konstruktora prywatnego w celu uniemożliwienia utworzenia obiektu	39
Temat 5. Unikanie powielania obiektów	40
Temat 6. Usuwanie niepotrzebnych referencji do obiektów	44
Temat 7. Unikanie finalizatorów	47
Rozdział 3. Metody wspólne dla wszystkich obiektów	53
Temat 8. Zachowanie założeń w trakcie przeddefiniowywania metody equals	53
Temat 9. Przeddefiniowywanie metody hashCode wraz z equals	65
Temat 10. Przeddefiniowywanie metody toString	70
Temat 11. Rozsądne przeddefiniowywanie metody clone	73
Temat 12. Implementacja interfejsu Comparable	81

Rozdział 4. Klasy i interfejsy	87
Temat 13. Ograniczanie dostępności klas i ich składników	87
Temat 14. Stosowanie metod akcesorów zamiast pól publicznych w klasach publicznych	91
Temat 15. Zapewnianie niezmienności obiektu	93
Temat 16. Zastępowanie dziedziczenia kompozycją	101
Temat 17. Projektowanie i dokumentowanie klas przeznaczonych do dziedziczenia	107
Temat 18. Stosowanie interfejsów zamiast klas abstrakcyjnych	112
Temat 19. Wykorzystanie interfejsów jedynie do definiowania typów	117
Temat 20. Zastępowanie oznaczania klas hierarchią	119
Temat 21. Zastosowanie obiektów funkcyjnych do reprezentowania strategii	122
Temat 22. Zalety stosowania statycznych klas składowych	125
Rozdział 5. Typy ogólne	129
Temat 23. Nie korzystaj z typów surowych w nowym kodzie	129
Temat 24. Eliminowanie ostrzeżeń o braku kontroli	135
Temat 25. Korzystanie z list zamiast tablic	137
Temat 26. Stosowanie typów ogólnych	142
Temat 27. Stosowanie metod ogólnych	146
Temat 28. Zastosowanie związanych szablonów do zwiększania elastyczności API	151
Temat 29. Wykorzystanie heterogenicznych kontenerów bezpiecznych dla typów	158
Rozdział 6. Typy wyliczeniowe i adnotacje	165
Temat 30. Użycie typów wyliczeniowych zamiast stałych int	165
Temat 31. Użycie pól instancyjnych zamiast kolejności	176
Temat 32. Użycie EnumSet zamiast pól bitowych	177
Temat 33. Użycie EnumMap zamiast indeksowania kolejnością	178
Temat 34. Emulowanie rozszerzalnych typów wyliczeniowych za pomocą interfejsów	182
Temat 35. Korzystanie z adnotacji zamiast wzorców nazw	186
Temat 36. Spójne użycie adnotacji Override	192
Temat 37. Użycie interfejsów znacznikowych do definiowania typów	195
Rozdział 7. Metody	197
Temat 38. Sprawdzanie poprawności parametrów	197
Temat 39. Defensywne kopiowanie	200
Temat 40. Projektowanie sygnatur metod	204
Temat 41. Rozsądne korzystanie z przeciążania	206

Temat 42. Rozsądne korzystanie z metod varargs	212
Temat 43. Zwracanie pustych tablic lub kolekcji zamiast wartości null	215
Temat 44. Tworzenie komentarzy dokumentujących dla wszystkich udostępnianych elementów API	217
Rozdział 8. Programowanie	225
Temat 45. Ograniczanie zasięgu zmiennych lokalnych	225
Temat 46. Stosowanie pętli for-each zamiast tradycyjnych pętli for	228
Temat 47. Poznanie i wykorzystywanie bibliotek	231
Temat 48. Unikanie typów float i double, gdy potrzebne są dokładne wyniki	234
Temat 49. Stosowanie typów prostych zamiast opakowanych typów prostych	236
Temat 50. Unikanie typu String, gdy istnieją bardziej odpowiednie typy	239
Temat 51. Problemy z wydajnością przy łączeniu ciągów znaków	242
Temat 52. Odwoływanie się do obiektów poprzez interfejsy	243
Temat 53. Stosowanie interfejsów zamiast refleksyjności	245
Temat 54. Rozważne wykorzystywanie metod natywnych	248
Temat 55. Unikanie optymalizacji	249
Temat 56. Wykorzystanie ogólnie przyjętych konwencji nazewnictwa	252
Rozdział 9. Wyjątki	257
Temat 57. Wykorzystanie wyjątków w sytuacjach nadzwyczajnych	257
Temat 58. Stosowanie wyjątków przechwytywanych i wyjątków czasu wykonania	260
Temat 59. Unikanie niepotrzebnych wyjątków przechwytywanych	262
Temat 60. Wykorzystanie wyjątków standardowych	264
Temat 61. Zgłaszanie wyjątków właściwych dla abstrakcji	266
Temat 62. Dokumentowanie wyjątków zgłaszanych przez metodę	268
Temat 63. Udostępnianie danych o błędzie	270
Temat 64. Zachowanie atomowości w przypadku błędu	272
Temat 65. Nie ignoruj wyjątków	274

Rozdział 10. Współbieżność	275
Temat 66. Synchronizacja dostępu do wspólnych modyfikowalnych danych	275
Temat 67. Unikanie nadmiarowej synchronizacji	280
Temat 68. Stosowanie wykonawców i zadań zamiast wątków	286
Temat 69. Stosowanie narzędzi współbieżności zamiast wait i notify	288
Temat 70. Dokumentowanie bezpieczeństwa dla wątków	293
Temat 71. Rozsądne korzystanie z późnej inicjalizacji	296
Temat 72. Nie polegaj na harmonogramie wątków	300
Temat 73. Unikanie grup wątków	302
Rozdział 11. Serializacja	305
Temat 74. Rozsądne implementowanie interfejsu Serializable	305
Temat 75. Wykorzystanie własnej postaci serializowanej	310
Temat 76. Defensywne tworzenie metody readObject	317
Temat 77. Stosowanie typów wyliczeniowych zamiast readResolve do kontroli obiektów	323
Temat 78. Użycie pośrednika serializacji zamiast serializowanych obiektów	327
Dodatek A Tematy odpowiadające pierwszemu wydaniu	331
Dodatek B Zasoby	335
Skorowidz	339