

# Stručný obsah

Předmluva .....	17
Úvod .....	21
Čistý kód .....	25
Smysluplná jména .....	39
Funkce .....	53
Komentáře .....	73
Formátování .....	95
Objekty a datové struktury .....	111
Zpracování chyb .....	121
Hranice .....	131
Jednotkové testy .....	139
Třídy .....	151
Systemy .....	167
Vývoj .....	183
Souběžnost .....	189
Postupné vylepšování .....	203
Vnitřní části šablony JUnit .....	259
Refaktorování třídy SerialDate .....	273
Skryté problémy a heuristika .....	291
Souběžnost II .....	321
Doslov .....	415

# Obsah

Předmluva .....	17
Úvod .....	21
Poděkování.....	23
Poznámka redakce českého vydání .....	23

## Kapitola 1

### **Čistý kód..... 25**

Kód nezanikne .....	26
Špatný kód .....	26
Celková cena za nepořádek .....	27
Celková rekonstrukce na zelené louce .....	28
Přístup.....	28
Prvotní paradox.....	29
Umění čistého kódu?.....	29
Co je to čistý kód?.....	30
Myšlenkový směr .....	34
Autoři.....	35
Skautské pravidlo .....	36
Úvod a principy .....	37
Závěr.....	37
Použitá literatura.....	37

## Kapitola 2

### **Smysluplná jména ..... 39**

Úvod .....	40
Používejte jména vysvětlující význam .....	40
Vyhňte se dezinformacím .....	41
Dělejte smysluplné rozdíly .....	42
Používejte vyslovitelná jména .....	43
Používejte jména, která lze vyhledat.....	44
Vyhňte se kódování jmen.....	45
Maďarská notace .....	45
Členské předpony .....	45

Testy .....	318
T1: Nedostatečné testy .....	318
T2: Používejte nástroje pro pokrytí .....	318
T3: Nepřeskakujte triviální testy .....	318
T4: Opomenutý test je otázkou ohledně nejednoznačnosti .....	318
T5: Testujte hraniční podmínky .....	318
T6: V okolí programových chyb provádějte důkladné testy .....	318
T7: Zákonitosti v selhávání odhalují chyby .....	318
T8: Pokrytí kódu testy může odhalit chyby .....	319
T9: Testy by měly být rychlé .....	319
Závěr .....	319
Použitá literatura .....	319

## Dodatek A

### **Souběžnost II ..... 321**

Příklad klient/server .....	322
Server .....	322
Přidání podprocesů .....	323
Sledování serveru .....	324
Závěr .....	325
Počet cest při provádění kódu .....	325
Počet cest .....	326
Hlubší pohled .....	327
Závěr .....	329
Vyznejte se ve své knihovně .....	330
Běhový rámec .....	330
Řešení bez blokace .....	330
Bezpečné třídy bez podprocesů .....	332
Závislost mezi metodami může souběžný kód porušit .....	333
Tolerovat selhání .....	334
Zamykání na straně klienta .....	334
Zamykání na straně serveru .....	335
Zvyšování propustnosti .....	336
Kalkulace propustnosti jednoho podprocesu .....	337
Kalkulace propustnosti při více podprocesech .....	338
Zablokování .....	338
Vzájemné vyloučení .....	339
Zamkni a čekej .....	340
Zdroj nelze získat nucenou výměnou .....	340

Cyklické čekání .....	340
Řešení vzájemného vyloučení .....	340
Řešení problému „Zamkni a čekej“ .....	340
Řešení získávání zdrojů nucenou výměnou .....	341
Řešení cyklického čekání .....	341
Testování kódu s více podprocesy.....	342
Podpora nástrojů pro testování kódu s podprocesy .....	344
Závěr .....	345
Výukový program .....	345
Klient/server bez podprocesů.....	345
Klient/server s podprocesy .....	348

## Dodatek B

### **org.jfree.date.SerialDate ..... 351**

Doslov .....	415
Rejstřík .....	417



Rozhraní a implementace.....	46
Vyhňte se skrytému překládání jmen.....	46
Jména tříd.....	47
Jména metod.....	47
Nesnažte se být strojení.....	47
Volte jedno slovo pro jeden pojem.....	47
Nepoužívejte slovní hříčky.....	48
Používejte jména z domény řešení.....	48
Používejte jména domén problému.....	49
Přidejte smysluplné souvislosti.....	49
Nepřidávejte kontext bezdůvodně.....	50
Slovo na závěr.....	51

## Kapitola 3

### **Funkce ..... 53**

Malá!.....	56
Bloky a odsazování.....	57
Dělejte jen jednu věc.....	57
Sekce uvnitř funkcí.....	58
Jedna úroveň abstrakce na funkci.....	58
Čtení kódu odshora dolů: metoda sestupu.....	58
Příkazy Switch.....	59
Používejte popisná jména.....	61
Argumenty funkcí.....	61
Běžné tvary funkce s jedním argumentem.....	62
Logické argumenty.....	62
Funkce se dvěma argumenty.....	63
Funkce se třemi argumenty.....	63
Objekty jako argumenty.....	64
Seznamy argumentů.....	64
Slovesa a klíčová slova.....	64
Žádné vedlejší efekty.....	65
Výstupní argumenty.....	65
Oddělování příkazů a dotazů.....	66
Dejte přednost výjimkám před vrácením chybových kódů.....	66
Extrahujte bloky Try/Catch.....	67
Zpracování chyb je jedna věc.....	68
Magnet závislosti Error.java.....	68
Neopakujte se.....	68

Strukturované programování.....	69
Jak napíšete funkci, jako je tato? .....	69
Závěr.....	70
Použitá literatura.....	72

## Kapitola 4

### **Komentáře..... 73**

Komentáře nevyváží špatný kód.....	75
Vyjádřete se kódem .....	75
Dobré komentáře .....	75
Komentáře právníckého charakteru .....	75
Informativní komentáře .....	76
Vysvětlení záměru .....	76
Objasnění.....	77
Varování před důsledky.....	78
Komentáře TODO (co dělat).....	78
Zvýraznění.....	79
Javadoc ve veřejných API .....	79
Špatné komentáře .....	79
Huhňání.....	79
Nadbytečné komentáře .....	80
Matoucí komentáře .....	82
Závazné komentáře .....	82
Deníkové komentáře.....	82
Komentáře obsahující šum .....	83
Rušení nahánějící hrůzu .....	85
Nepoužívejte komentář, když můžete použít funkci nebo proměnnou.....	86
Označení pozice .....	86
Komentáře na konci složených závorek.....	86
Připisování a podtitulky se jmény .....	87
Zakomentované řádky kódu .....	87
Komentáře ve formátu HTML.....	88
Nelokální informace .....	89
Příliš mnoho informací.....	89
Nejasná spojitost .....	89
Záhlaví funkcí.....	90
Javadoc v neveřejném kódu.....	90
Příklad .....	90
Použitá literatura.....	93

## Kapitola 5

<b>Formátování.....</b>	<b>95</b>
Důvody formátování.....	96
Vertikální formátování .....	96
Přirovnání k novinám .....	97
Vertikální oddělování pojmů.....	98
Vertikální hustota .....	99
Vertikální vzdálenost.....	99
Vertikální uspořádání .....	103
Horizontální formátování.....	104
Horizontální oddělování a hustota .....	104
Horizontální zarovnání .....	105
Odsazování.....	106
Prázdné obory.....	108
Týmová pravidla .....	108
Formátovací pravidla strýčka Boba.....	109

## Kapitola 6

<b>Objekty a datové struktury .....</b>	<b>111</b>
Datové abstrakce .....	112
Datová a objektová antisymetrie .....	113
Démétrin zákon.....	115
Vykolejený vlak.....	116
Hybridy.....	116
Skrytá struktura.....	117
Objekty pro přenos dat .....	117
Aktivní záznam .....	118
Shrnutí .....	119
Použitá literatura.....	119

## Kapitola 7

<b>Zpracování chyb .....</b>	<b>121</b>
Používejte výjimky raději než návratové kódy.....	122
Pište nejdříve příkazy Try-Catch-Finally .....	123
Používejte nekontrolované výjimky .....	124
Poskytujte kontext s výjimkami.....	125
Definujte třídy výjimek z hlediska potřeb volajícího.....	125
Definujte normální tok .....	127



Nevracejte hodnotu null .....	128
Nepředávejte hodnotu null .....	129
Závěr .....	130
Použitá literatura .....	130

## Kapitola 8

### Hranice ..... 131

Použití kódu třetí strany .....	132
Zkoumání a studium hranic .....	134
Studium log4j .....	134
Poznávací testy se vyplatí .....	136
Používání kódu, který zatím neexistuje .....	136
Čisté hranice .....	137
Použitá literatura .....	138

## Kapitola 9

### Jednotkové testy ..... 139

Tři zákony vývoje řízeného testy (TDD) .....	140
Mějte testy čisté .....	141
Testy otevírají další možnosti .....	142
Čisté testy .....	142
Doménově specifický testovací jazyk .....	145
Dvojitý standard .....	145
Jedna aserce na jeden test .....	147
Jedna myšlenka pro jeden test .....	148
F.I.R.S.T. ....	149
Závěr .....	149
Použitá literatura .....	150

## Kapitola 10

### Třídy ..... 151

Organizace třídy .....	152
Zapouzdření .....	152
Třídy by měly být malé! .....	152
Princip jediné odpovědnosti .....	154
Soudržnost .....	156
Soudržnost vede k mnoho malým třídám .....	156
Organizace podporující změny .....	162



Izolování od změn .....	164
Použitá literatura .....	165

## Kapitola 11

### **Systemy ..... 167**

Jak byste postavili město? .....	168
Oddělujte tvorbu systému od jeho používání .....	168
Separování modulu Main .....	169
Továrny .....	170
Vkládání závislostí .....	170
Škálování .....	171
Průnik zájmů .....	173
Zprostředkovatel v Javě .....	174
Čisté rámce Java AOP .....	176
Aspekty AspectJ .....	179
Testování systémové architektury .....	179
Optimalizujte rozhodování .....	180
Používejte rozumně standardy, pokud přinášejí prokazatelnou hodnotu .....	180
Systemy potřebují doménově specifické jazyky .....	181
Závěr .....	181
Použitá literatura .....	182

## Kapitola 12

### **Vývoj ..... 183**

Čistota pomocí vyvíjejícího se návrhu .....	184
První pravidlo jednoduchého návrhu: Projde všemi testy .....	184
Zásady jednoduchého návrhu 2–4: refaktorování .....	185
Žádný zdvojený kód .....	185
Expresivita .....	187
Minimální třídy a metody .....	188
Závěr .....	188
Použitá literatura .....	188

## Kapitola 13

### **Souběžnost ..... 189**

Proč souběžnost? .....	190
Mýty a mylné názory .....	191
Problémy .....	192

Principy ochrany souběžnosti.....	192
Princip jedné odpovědnosti.....	192
Důsledek: omezujte rozsah dat.....	193
Důsledek: používejte kopie dat.....	193
Důsledek: Podprocesy by měly být co nejméně závislé.....	193
Vyznejte se ve své knihovně.....	194
Kolekce, které jsou z hlediska souběžného kódu bezpečné.....	194
Poznejte své běhové modely.....	194
Producent-spotřebitel.....	195
Čtenáři-zapisovatelé.....	195
Stolující filozofové.....	196
Pozor na závislosti mezi synchronizovanými metodami.....	196
Mějte synchronizované sekce malé.....	196
Je obtížné napsat korektní kód pro vypínání.....	197
Testování kódu podprocesů.....	197
Berte nejasná selhání jako budoucí možné problémy podprocesů.....	198
Uvedte nejdříve do provozu kód bez podprocesů.....	198
Vytvářejte souběžný kód jako zásuvný modul.....	198
Vytvořte souběžný kód nastavitelný.....	198
Spouštějte více procesů, než máte procesorů.....	199
Spouštějte kód na různých platformách.....	199
Upravte kód tak, aby vyzkoušel a navodil selhání.....	199
Ruční kódování.....	200
Automatizované kódování.....	200
Závěr.....	201
Použitá literatura.....	202

## Kapitola 14

### **Postupné vylepšování..... 203**

Rozbor analyzátoru argumentů příkazového řádku.....	204
Implementace třídy Args.....	204
Jak jsem to udělal?.....	210
Třída Args: nanečisto.....	210
Tak jsem se zastavil.....	220
O postupných změnách.....	220
Řetězcové argumenty.....	222
Závěr.....	257

## Kapitola 15

### Vnitřní části šablony JUnit..... 259

Šablona JUnit .....	260
Závěr .....	272

## Kapitola 16

### Refaktorování třídy SerialDate ..... 273

Nejdřív ať to funguje .....	274
Pak to sprav.....	276
Závěr .....	289
Použitá literatura .....	289

## Kapitola 17

### Skryté problémy a heuristika ..... 291

Komentáře .....	292
K1: Nevhodné komentáře.....	292
K2: Zastaralé komentáře .....	292
K3: Nadbytečný komentář.....	292
K4: Špatně napsaný komentář.....	293
K5: Zakomentovaný kód .....	293
Prostředí .....	293
P1: Sestavení vyžaduje více než jeden krok.....	293
P2: Testy vyžadují více než jeden krok.....	293
Funkce .....	293
F1: Příliš mnoho argumentů .....	293
F2: Výstupní argumenty .....	294
F3: Logické argumenty .....	294
F4: Mrtvé funkce .....	294
Obecné .....	294
O1: Více jazyků v jednom zdrojovém souboru .....	294
O2: Není implementováno to, co je samozřejmé .....	294
O3: Nekorektní funkčnost na hranicích.....	295
O4: Zrušená zabezpečení .....	295
O5: Zdvojení .....	295
O6: Kód na špatné úrovni abstrakce.....	296
O7: Základní třídy, které závisejí na odvozených třídách .....	297
O8: Příliš mnoho informací.....	297
O9: Mrtvý kód .....	297



O10: Vertikální oddělování.....	298
O11: Nekonzistentnost.....	298
O12: Zaneřáděnost.....	298
O13: Umělé vazby.....	298
O14: Chybějící schopnosti.....	298
O15: Přepínací argumenty.....	299
O16: Nejasný záměr.....	300
O17: Špatně umístěná odpovědnost.....	301
O18: Nevhodný modifikátor static.....	301
O19: Používejte vysvětlující proměnné.....	301
O20: Názvy funkcí by měly sdělovat, co dělají.....	302
O21: Pochopte algoritmus.....	302
O22: Udělejte z logických závislostí fyzické.....	303
O23: Volte raději polymorfismus než příkazy if/else nebo switch/case.....	304
O24: Dodržujte standardní konvence.....	304
O25: Nahradte magická čísla pojmenovanými konstantami.....	305
O26: Buďte přesní.....	306
O27: Struktura je více než konvence.....	306
O28: Zapouzdřete podmínky.....	306
O29: Vyhybejte se negativním podmíněným výrazům.....	306
O30: Funkce by měly provádět jen jednu věc.....	307
O31: Skrytá časová vazba.....	307
O32: Nepodléhejte libovůli.....	308
O33: Zapouzdřete hraniční podmínky.....	309
O34: Funkce by měly sestupovat jen o jednu úroveň abstrakce níže.....	309
O35: Mějte konfigurační data na vysokých úrovních.....	310
O36: Vyhněte se tranzitivním odkazům.....	311
<b>Java.....</b>	<b>311</b>
J1: Vyhněte se dlouhým seznamům a používejte zástupné znaky.....	311
J2: Vyhněte se dědění konstant.....	312
J3: Konstanty versus výčty.....	313
<b>Jména.....</b>	<b>314</b>
Jm1: Vybírejte popisná jména.....	314
Jm2: Vybírejte jména na adekvátní úrovni abstrakce.....	315
Jm3: Používejte standardní názvosloví všude, kde je to možné.....	316
Jm4: Jednoznačná jména.....	316
Jm5: Pro velké rozsahy používejte dlouhá jména.....	317
Jm6: Vyhněte se kódování jmen.....	317
Jm7: Jména by měla popisovat vedlejší efekty.....	317