

Inhaltsverzeichnis

Vorwort zur ersten Auflage	15
Über den Fachkorrektor der deutschen Ausgabe	17
Einleitung	19
Was ist Refactoring?	20
Worum geht es in diesem Buch?	21
Wer sollte dieses Buch lesen?	23
Von anderen geschaffene Grundlagen	24
Danksagungen	25
1 Refactoring: Ein erstes Beispiel	27
1.1 Der Ausgangspunkt	27
1.2 Anmerkungen zum ersten Programm	30
1.3 Der erste Schritt des Refactorings	31
1.4 Aufteilung der statement-Funktion	32
1.4.1 Entfernen der Variable play	38
1.4.2 Extrahieren der Berechnung von Treuepunkten	43
1.4.3 Entfernen der Variable format	44
1.4.4 Entfernen der Gesamtzahl der Treuepunkte	47
1.5 Status: Eine Menge verschachtelter Funktionen	52
1.6 Die Phasen Berechnung und Formatierung voneinander trennen	54
1.7 Status: Aufgeteilt in zwei Dateien (und Phasen)	63
1.8 Neuorganisation der Berechnungen nach Art	66
1.8.1 Berechnen der Kosten einer Vorstellung	68
1.8.2 Funktionen in die Klasse verschieben	69
1.8.3 Polymorphes Verhalten der Klasse PerformanceCalculator	71
1.9 Status: Erzeugen der Daten mit der polymorphen Klasse PerformanceCalculator	74
1.10 Abschließende Überlegungen	77
2 Prinzipien des Refactorings	79
2.1 Refactoring: Definition	79
2.2 Die beiden Hüte	80
2.3 Gründe für ein Refactoring	81
2.3.1 Refactoring verbessert das Design der Software	81
2.3.2 Refactoring macht Software besser verständlich	81
2.3.3 Refactoring hilft mir, Bugs aufzuspüren	82
2.3.4 Refactoring hilft mir, schneller zu programmieren	82

2.4	Der richtige Zeitpunkt für ein Refactoring	84
2.4.1	Vorbereitendes Refactoring: Hinzufügen eines Features vereinfachen	84
2.4.2	Refactoring zwecks Verbesserung der Verständlichkeit des Codes	85
2.4.3	Refactoring als Abfallentsorgung	86
2.4.4	Geplantes Refactoring und spontanes Refactoring	86
2.4.5	Langfristiges Refactoring	87
2.4.6	Refactoring beim Code-Review	88
2.4.7	Wie sag ich's meinem Chef?	89
2.4.8	Wann sollte man kein Refactoring durchführen?	89
2.5	Probleme beim Refactoring	90
2.5.1	Verlangsamung der Entwicklung neuer Features	90
2.5.2	Code-Eigentümerschaft	91
2.5.3	Entwicklungszweige	92
2.5.4	Testen	94
2.5.5	Legacy Code.	95
2.5.6	Datenbanken	96
2.6	Refactoring, Architektur und Yagni	97
2.7	Refactoring und der Softwareentwicklungsprozess	98
2.8	Refactoring und Performance	99
2.9	Ursprünge des Refactorings	102
2.10	Automatisiertes Refactoring	103
2.11	Weiterführende Literatur	105
3	Code-Smells: Schlechte Gerüche im Code	107
3.1	Rätselhafte Bezeichnung	108
3.2	Redundanter Code.	108
3.3	Lange Funktion	109
3.4	Lange Parameterliste.	110
3.5	Globale Daten	110
3.6	Veränderliche Daten	111
3.7	Divergierende Änderungen	112
3.8	Chirurgie mit der Schrotflinte	113
3.9	Feature-Neid	113
3.10	Datenklumpen.	114
3.11	Obsession für elementare Datentypen.	115
3.12	Wiederholte switch-Anweisungen	115
3.13	Schleifen.	116
3.14	Träges Element	116
3.15	Spekulative Generalisierung.	116
3.16	Temporäres Feld	117
3.17	Mitteilungsketten	117
3.18	Vermittler.	118
3.19	Insiderhandel.	118

3.20	Umfangreiche Klasse	118
3.21	Alternative Klassen mit unterschiedlichen Schnittstellen	119
3.22	Datenklasse	119
3.23	Ausgeschlagenes Erbe	120
3.24	Kommentare	120
4	Tests erstellen.	123
4.1	Der Nutzen selbsttestenden Codes	123
4.2	Beispielcode zum Testen	125
4.3	Ein erster Test	129
4.4	Hinzufügen eines weiteren Tests	132
4.5	Ändern des Test-Fixtures	134
4.6	Austesten der Grenzen	135
4.7	Und noch viel mehr	139
5	Der Katalog.	141
5.1	Format der Refactorings	141
5.2	Die Auswahl der Refactorings	142
6	Eine erste Zusammenstellung von Refactorings.	145
6.1	Funktion extrahieren (Extract Function)	145
6.1.1	Motivation	146
6.1.2	Vorgehen	147
6.1.3	Beispiel: Keine Variablen außerhalb des Gültigkeitsbereichs	148
6.1.4	Beispiel: Verwendung lokaler Variablen	151
6.1.5	Beispiel: Zuweisung zu einer lokalen Variable	152
6.2	Funktion inline platzieren (Inline Function)	155
6.2.1	Motivation	156
6.2.2	Vorgehen	156
6.2.3	Beispiel	157
6.3	Variable extrahieren (Extract Variable)	159
6.3.1	Motivation	159
6.3.2	Vorgehen	160
6.3.3	Beispiel	160
6.3.4	Beispiel mit einer Klasse	162
6.4	Variable inline platzieren (Inline Variable)	163
6.4.1	Motivation	164
6.4.2	Vorgehen	164
6.5	Funktionsdeklaration ändern (Change Function Declaration)	164
6.5.1	Motivation	165
6.5.2	Vorgehen	166
6.5.3	Beispiel: Umbenennen einer Funktion (einfaches Vorgehen)	167
6.5.4	Beispiel: Umbenennen einer Funktion (Migrationsansatz)	168
6.5.5	Beispiel: Hinzufügen eines Parameters	169
6.5.6	Beispiel: Einen Parameter durch eine seiner Eigenschaften ersetzen	170

6.6	Variable kapseln (Encapsulate Variable)	172
6.6.1	Motivation	173
6.6.2	Vorgehen	174
6.6.3	Beispiel	174
6.7	Variable umbenennen (Rename Variable)	177
6.7.1	Motivation	178
6.7.2	Vorgehen	178
6.7.3	Beispiel	178
6.8	Parameterobjekt einführen (Introduce Parameter Object)	180
6.8.1	Motivation	180
6.8.2	Vorgehen	181
6.8.3	Beispiel	181
6.9	Funktionen zu einer Klasse vereinen (Combine Functions into Class)	184
6.9.1	Motivation	185
6.9.2	Vorgehen	185
6.9.3	Beispiel	186
6.10	Funktionen zu einer Transformation vereinen (Combine Functions into Transform)	189
6.10.1	Motivation	190
6.10.2	Vorgehen	190
6.10.3	Beispiel	191
6.11	Phase aufteilen (Split Phase)	195
6.11.1	Motivation	196
6.11.2	Vorgehen	196
6.11.3	Beispiel	197
7	Kapselung	203
7.1	Datensatz kapseln (Encapsulate Record)	203
7.1.1	Motivation	204
7.1.2	Vorgehen	205
7.1.3	Beispiel	205
7.1.4	Beispiel: Kapselung eines verschachtelten Datensatzes	207
7.2	Collection kapseln (Encapsulate Collection)	212
7.2.1	Motivation	213
7.2.2	Vorgehen	214
7.2.3	Beispiel	214
7.3	Elementaren Wert durch Objekt ersetzen (Replace Primitive with Object)	217
7.3.1	Motivation	217
7.3.2	Vorgehen	218
7.3.3	Beispiel	218
7.4	Temporäre Variable durch Abfrage ersetzen (Replace Temp with Query)	221
7.4.1	Motivation	222
7.4.2	Vorgehen	223
7.4.3	Beispiel	223

7.5	Klasse extrahieren (Extract Class)	225
7.5.1	Motivation	226
7.5.2	Vorgehen	226
7.5.3	Beispiel.	227
7.6	Klasse inline platzieren (Inline Class)	229
7.6.1	Motivation	230
7.6.2	Vorgehen	230
7.6.3	Beispiel.	231
7.7	Delegation verbergen (Hide Delegate)	232
7.7.1	Motivation	233
7.7.2	Vorgehen	234
7.7.3	Beispiel.	234
7.8	Vermittler entfernen (Remove Middle Man)	235
7.8.1	Motivation	235
7.8.2	Vorgehen	236
7.8.3	Beispiel.	236
7.9	Algorithmus ersetzen (Substitute Algorithm)	237
7.9.1	Motivation	238
7.9.2	Vorgehen	239
8	Verschiebungen	241
8.1	Funktion verschieben (Move Function)	241
8.1.1	Motivation	242
8.1.2	Vorgehen	242
8.1.3	Beispiel: Verschieben einer verschachtelten Funktion in die oberste Ebene.	243
8.1.4	Beispiel: Verschiebungen zwischen Klassen.	248
8.2	Feld verschieben (Move Field)	251
8.2.1	Motivation	251
8.2.2	Vorgehen	253
8.2.3	Beispiel.	253
8.2.4	Beispiel: Verschiebung in ein gemeinsam genutztes Objekt.	255
8.3	Anweisungen in Funktion verschieben (Move Statements into Function)	257
8.3.1	Motivation	258
8.3.2	Vorgehen	258
8.3.3	Beispiel.	259
8.4	Anweisungen in Aufrufer verschieben (Move Statements to Caller)	261
8.4.1	Motivation	262
8.4.2	Vorgehen	262
8.4.3	Beispiel.	263
8.5	Inline-Code durch Funktionsaufruf ersetzen (Replace Inline Code with Function Call)	267
8.5.1	Motivation	267
8.5.2	Vorgehen	268

8.6	Anweisungen verschieben (Slide Statements)	268
8.6.1	Motivation	269
8.6.2	Vorgehen	269
8.6.3	Beispiel	269
8.6.4	Beispiel: Verschiebungen und bedingte Anweisungen	271
8.6.5	Literaturhinweis	272
8.7	Schleife aufteilen (Split Loop)	273
8.7.1	Motivation	273
8.7.2	Vorgehen	274
8.7.3	Beispiel	274
8.8	Schleife durch Pipeline ersetzen (Replace Loop with Pipeline)	277
8.8.1	Motivation	277
8.8.2	Vorgehen	277
8.8.3	Beispiel	278
8.8.4	Literaturhinweis	283
8.9	Toten Code entfernen (Remove Dead Code)	283
8.9.1	Motivation	283
8.9.2	Vorgehen	284
9	Daten organisieren	285
9.1	Variable aufteilen (Split Variable)	285
9.1.1	Motivation	286
9.1.2	Vorgehen	286
9.1.3	Beispiel	286
9.1.4	Beispiel: Zuweisung zu einem Eingabeparameter	288
9.2	Feld umbenennen (Rename Field)	289
9.2.1	Motivation	290
9.2.2	Vorgehen	290
9.2.3	Beispiel: Umbenennen eines Feldes	290
9.3	Abgeleitete Variable durch Abfrage ersetzen (Replace Derived Variable with Query)	293
9.3.1	Motivation	294
9.3.2	Vorgehen	294
9.3.3	Beispiel	295
9.3.4	Beispiel: Mehr als eine Quelle	296
9.4	Referenz durch Wert ersetzen (Change Reference to Value)	297
9.4.1	Motivation	298
9.4.2	Vorgehen	298
9.4.3	Beispiel	298
9.5	Wert durch Referenz ersetzen (Change Value to Reference)	301
9.5.1	Motivation	301
9.5.2	Vorgehen	302
9.5.3	Beispiel	302

10	Bedingungen vereinfachen	305
10.1	Bedingung zerlegen (Decompose Conditional)	305
10.1.1	Motivation	306
10.1.2	Vorgehen	306
10.1.3	Beispiel	306
10.2	Bedingten Ausdruck zusammenfassen (Consolidate Conditional Expression)	308
10.2.1	Motivation	309
10.2.2	Vorgehen	309
10.2.3	Beispiel	310
10.2.4	Beispiel: Verwendung von »and«	310
10.3	Verschachtelte Bedingung durch Wächterbedingung ersetzen (Replace Nested Conditional with Guard Clauses)	311
10.3.1	Motivation	312
10.3.2	Vorgehen	312
10.3.3	Beispiel	313
10.3.4	Beispiel: Umkehrung der Bedingungen	315
10.4	Bedingung durch Polymorphie ersetzen (Replace Conditional with Polymorphism)	317
10.4.1	Motivation	318
10.4.2	Vorgehen	318
10.4.3	Beispiel	319
10.4.4	Beispiel: Polymorphie für Variationen verwenden	324
10.5	Sonderfall einführen (Introduce Special Case)	336
10.5.1	Motivation	337
10.5.2	Vorgehen	337
10.5.3	Beispiel	338
10.5.4	Beispiel: Verwendung eines Objekts mit Literalen	343
10.5.5	Beispiel: Verwenden einer Transformation	346
10.6	Assertion einführen (Introduce Assertion)	351
10.6.1	Motivation	351
10.6.2	Vorgehen	352
10.6.3	Beispiel	352
11	Refactoring von APIs	355
11.1	Abfrage von Veränderung trennen (Separate Query from Modifier)	355
11.1.1	Motivation	356
11.1.2	Vorgehen	356
11.1.3	Beispiel	357
11.2	Funktion parametrisieren (Parameterize Function)	359
11.2.1	Motivation	360
11.2.2	Vorgehen	360
11.2.3	Beispiel	360

11.3	Steuerungs-Flag entfernen (Remove Flag Argument)	363
11.3.1	Motivation	363
11.3.2	Vorgehen	365
11.3.3	Beispiel	365
11.4	Vollständiges Objekt erhalten (Preserve Whole Object).	368
11.4.1	Motivation	369
11.4.2	Vorgehen	369
11.4.3	Beispiel	370
11.4.4	Beispiel: Eine Variante zum Erstellen der neuen Funktion	371
11.5	Parameter durch Abfrage ersetzen (Replace Parameter with Query)	373
11.5.1	Motivation	374
11.5.2	Vorgehen	375
11.5.3	Beispiel	375
11.6	Abfrage durch Parameter ersetzen (Replace Query with Parameter)	376
11.6.1	Motivation	377
11.6.2	Vorgehen	378
11.6.3	Beispiel	378
11.7	Setter entfernen (Remove Setting Method)	381
11.7.1	Motivation	381
11.7.2	Vorgehen	382
11.7.3	Beispiel	382
11.8	Konstruktor durch Fabrikfunktion ersetzen (Replace Constructor with Factory Function)	383
11.8.1	Motivation	384
11.8.2	Vorgehen	384
11.8.3	Beispiel	384
11.9	Funktion durch Befehl ersetzen (Replace Function with Command).	385
11.9.1	Motivation	386
11.9.2	Vorgehen	387
11.9.3	Beispiel	387
11.10	Befehl durch Funktion ersetzen (Replace Command with Function).	393
11.10.1	Motivation	393
11.10.2	Vorgehen	394
11.10.3	Beispiel	394
12	Der Umgang mit Vererbung	399
12.1	Methode nach oben verschieben (Pull Up Method)	399
12.1.1	Motivation	400
12.1.2	Vorgehen	401
12.1.3	Beispiel	401
12.2	Feld nach oben verschieben (Pull Up Field)	402
12.2.1	Motivation	403
12.2.2	Vorgehen	403
12.3	Konstruktorrumpf nach oben verschieben (Pull Up Constructor Body).	404
12.3.1	Motivation	405

12.3.2	Vorgehen	405
12.3.3	Beispiel	405
12.4	Methode nach unten verschieben (Push Down Method)	408
12.4.1	Motivation	409
12.4.2	Vorgehen	409
12.5	Feld nach unten verschieben (Push Down Field)	409
12.5.1	Motivation	410
12.5.2	Vorgehen	410
12.6	Typenschlüssel durch Unterklassen ersetzen (Replace Type Code with Subclasses)	410
12.6.1	Motivation	411
12.6.2	Vorgehen	411
12.6.3	Beispiel	412
12.6.4	Beispiel: Indirekte Vererbung	415
12.7	Unterklasse entfernen (Remove Subclass)	418
12.7.1	Motivation	419
12.7.2	Vorgehen	419
12.7.3	Beispiel	419
12.8	Basisklasse extrahieren (Extract Superclass)	424
12.8.1	Motivation	425
12.8.2	Vorgehen	425
12.8.3	Beispiel	425
12.9	Hierarchie abbauen (Collapse Hierarchy)	429
12.9.1	Motivation	429
12.9.2	Vorgehen	430
12.10	Unterklasse durch Delegation ersetzen (Replace Subclass with Delegate)	430
12.10.1	Motivation	431
12.10.2	Vorgehen	432
12.10.3	Beispiel	433
12.10.4	Beispiel: Ersetzen einer Hierarchie	440
12.11	Basisklasse durch Delegation ersetzen (Replace Superclass with Delegate)	450
12.11.1	Motivation	451
12.11.2	Vorgehen	452
12.11.3	Beispiel	452
A	Bibliografie	457
B	Liste der Refactorings	461
C	Code-Smells	463
	Stichwortverzeichnis	467