

Предисловие	11
Почему мы написали эту книгу	11
Подход, ориентированный на разработчика	12
Что в этой книге?	12
Для кого эта книга?	13
Условные обозначения, используемые в книге	14
Использование примеров кода	15
Глава 1. Начало путешествия	16
Темы	16
Особенности Java	16
Разработка программного обеспечения и архитектура	17
SOLID	17
Тестирование	18
Структура глав	18
Самостоятельная работа	20
Глава 2. Анализатор банковских операций	21
Задача	21
Цель	21
Требования к анализатору банковских операций	22
Принцип KISS	22
Переменные final	25
Обслуживаемость кода и антишаблоны	25
Класс-бог	26
Дублирование кода	26
Принцип единственной ответственности	27
Связность	32
Внутриклассовая связность	35
Функциональная	36
Информационная	36

Служебная	37
Логическая	37
Последовательная	38
Временная	39
Связность методов	39
Связанность	40
Тестирование	42
Автоматизированное тестирование	43
Доверие	43
Устойчивость к изменениям	43
Понимание программы	44
Использование JUnit	44
Объявление метода теста	44
Операторы контроля	46
Покрытие кода	47
Выводы	48
Самостоятельная работа	49
В завершение	49
Глава 3. Расширяем анализатор банковских операций	50
Задача	50
Цель	50
Требования к расширенному анализатору банковских операций	51
Принцип открытости/закрытости	51
Создание экземпляра функционального интерфейса	55
Лямбда-выражения	55
Подводные камни интерфейсов	56
Интерфейс-бог	57
Слишком мизерный	58
Явный API против неявного	59
Доменный класс или примитив?	61
Множественный экспорт	62
Знакомство с доменным объектом	62
Объявление и реализация соответствующего интерфейса	64
Обработка исключений	65
Для чего нужны исключения?	66
Шаблоны и антишаблоны для исключений	68
Выбор между проверяемыми и непроверяемыми исключениями	68
Слишком специфические	69

Слишком однообразные	71
Шаблон уведомления	72
Методика применения исключений	74
Не игнорируйте исключение	74
Не перехватывайте «общие» исключения	74
Документируйте исключения	74
Будьте осторожны с исключениями, связанными с конкретной реализацией	75
Исключения против управляющего потока	75
Альтернативы исключениям	76
Использование null	76
Шаблон null-объекта	76
Optional<T>	77
Try<T>	77
Использование сборщиков	77
Зачем нужны сборщики	77
Работа с Maven	78
Структура проекта	79
Пример сборочного файла	80
Команды Maven	81
Использование Gradle	82
Пример сборочного файла	83
Команды Gradle	84
Выводы	84
Самостоятельная работа	85
В завершение	86
Глава 4. Система управления документами	87
Задача	87
Цель	87
Требования к системе управления документами	88
Воплощение идеи	89
Импортеры	90
Класс Document	91
Атрибуты и иерархия Documents	94
Реализация и регистрация импортеров	95
Принцип подстановки Лисков	97
Альтернативные подходы	99
Поместить импортер в класс	100
Область действия и инкапсуляция	100

Расширение и повторное использование кода	101
Гигиена тестов	107
Именованние тестов	108
Поведение, а не реализация	110
Не повторяйтесь	112
Хорошая диагностика	113
Тестирование ошибочных ситуаций	116
Константы	117
Выводы	118
Самостоятельная работа	118
В завершение	118
Глава 5. Движок бизнес-правил	119
Задача	119
Цель	119
Требования к движку бизнес-правил	120
Разработка через тестирование	121
Зачем нужен TDD?	122
Цикл TDD	123
Мокинг	125
Добавление условий	127
Моделирование состояния	127
Вывод типа локальной переменной	131
Switch-выражения	132
Принцип разделения интерфейса	135
Разработка текучего интерфейса (Fluent API)	139
Что такое Fluent API?	139
Моделирование домена	139
Шаблон Builder	141
Выводы	144
Самостоятельная работа	145
В завершение	145
Глава 6. Tootr	146
Задача	146
Цель	146
Требования к Tootr	147
Обзор разработки	148
Технология Pull	149
Технология Push	149

От событий к разработке	150
Связь	151
GUI — графический интерфейс пользователя	152
Продолжаем	153
Гексагональная архитектура	153
С чего начать	155
Пароли и безопасность	160
Подписчики и твуты	162
Моделирование ошибок	163
Твутинг	166
Создание моков	167
Верификация при помощи моков	168
Библиотеки для мокинга	169
SenderEndPoint	170
Позиции	172
Методы equals и hashCode	176
Взаимосвязь между equals и hashCode	177
Выводы	179
Самостоятельная работа	179
В завершение	179
Глава 7. Расширение Tootr	180
Задача	180
Цель	180
Резюме	181
Живучесть и шаблон «Репозиторий»	181
Проектирование репозитория	182
Объекты-запросы	184
Функциональное программирование	189
Лямбда-выражения	190
Ссылки на методы	192
Execute Around	193
Потоки	195
map ()	195
forEach ()	196
filter ()	196
reduce ()	198
Optional	200
Пользовательский интерфейс	203
Инверсия зависимости и внедрение зависимости	204

Пакеты и сборочные системы	207
Ограничения и упрощения	209
Выводы	210
Самостоятельная работа	210
В завершение	211
Глава 8. Заключение	212
Проектно-ориентированная структура	212
Самостоятельная работа	212
Сознательная практика	213
Следующие шаги и дополнительные ресурсы	214
Об авторах	216
В завершение	217
Предметный указатель	218