
Оглавление

Предисловие	12
Почему я написал эту книгу	12
Для кого эта книга.....	13
Что представляет собой эта книга	14
Структура издания.....	15
Как читать эту книгу	17
Использование примеров кода	17
Условные обозначения.....	18
Благодарности.....	19
От издательства	20
О научном редакторе русскоязычного издания	20
Глава 1. Введение в конкурентность.....	21
Краткая история потоков в Java.....	21
Язык Java построен на основе потоков	22
Потоки — основа программной платформы Java	24
Зарождение потоков в Java 1.0.....	27
Запуск потоков	28
Скрытые затраты на потоки	29
Сколько потоков можно создать	31
Эффективность использования ресурсов в высоконагруженных приложениях.....	32
Стратегия параллельного выполнения	33
Знакомство с фреймворком Executor.....	38
Оставшиеся проблемы.....	39
Выход за пределы базового пула потоков	41
Привязка к кэшу и распределение задач	41

Алгоритм кражи задач.....	41
Компоновка с помощью CompletableFuture	43
Другая парадигма асинхронного программирования.....	45
Отрицательная сторона использования реактивных фреймворков	47
Революция в конкурентности Java	49
Виртуальные потоки: новый подход.....	49
Бесшовная интеграция с существующими кодовыми базами.....	49
Виртуальные и платформенные потоки.....	50
Интеллектуальная обработка блокирующих операций.....	50
Преимущества использования виртуальных потоков	50
В заключение	51
Глава 2. Виртуальные потоки.....	52
Что такое виртуальные потоки.....	52
Два вида потоков в Java	53
Ключевые отличия виртуальных потоков от платформенных	55
Настройка окружения для работы с виртуальными потоками	56
Создание виртуальных потоков в Java	57
Адаптация к виртуальным потокам.....	58
Пример создания виртуальных потоков в Java	62
Пропускная способность и масштабируемость	63
Фундаментальный принцип масштабируемости виртуальных потоков	64
Практические выводы	67
Виртуальные потоки: внутреннее устройство	68
Стековые фреймы и управление памятью.....	68
Потоки-носители и взаимодействие с ОС	68
Обработка блокирующих операций.....	69
Прозрачность и невидимость.....	69
Упрощение асинхронных операций.....	69
Потенциал структурированной конкурентности	72
Управление ограничением ресурсов с помощью Rate Limiting.....	73
Семафоры в Java: принцип работы.....	76
В каких случаях использовать семафор.....	81
Ограничения виртуальных потоков.....	83
Пиннинг	85

Решение проблемы пиннинга с помощью ReentrantLock.....	87
Вызов нативных методов и пиннинг	91
Дилемма применения переменных ThreadLocal в виртуальных потоках	94
Проблемы виртуальных потоков	95
Мониторинг	98
Мониторинг ThreadLocals.....	98
Мониторинг пиннинга.....	99
Просмотр виртуальных потоков в дампах потоков с помощью jcmd.....	105
Генерация дампов потоков с помощью HotSpotDiagnosticsMXBean	110
Практические советы по миграции на виртуальные потоки.....	111
Еще раз о преимуществах виртуальных потоков	112
Масштабируемость	113
В заключение	114
Глава 3. Механизм современной конкурентности в Java	115
Пул потоков	115
Зачем нужен пул потоков.....	116
Создание простого пула потоков на Java	116
Фреймворк Executor	120
Интерфейсы Callable и Future: обработка результатов задач.....	125
ForkJoinPool	127
Почему ForkJoinPool используется для виртуальных потоков	132
Континуация.....	137
Создание виртуальных потоков с нуля.....	140
Виртуальные потоки и опрос ввода-вывода.....	146
В заключение	147
Глава 4. Структурированная конкурентность	148
Проблемы неструктурированной конкурентности.....	149
Потенциал структурированной конкурентности	155
Основы API	156
StructuredTaskScope	157
Области и задачи: связи и жизненный цикл.....	161
Политики объединения через Joiner.....	162

Стандартные политики объединения подзадач в одной области	163
Обработка исключений в StructuredTaskScope.....	189
Конфигурация	202
Кастомные джоинеры.....	208
Согласованность памяти	225
Вложенные области	228
Наблюдаемость.....	232
В заключение	237
Глава 5. Scoped values	239
В чем проблема передачи контекста	239
Загрязнение параметрами	242
Уязвимость интерфейсов	242
Стыковка и тестируемость.....	242
Использование ThreadLocal	242
Ограничения переменных ThreadLocal	244
Облегченная модель совместного использования данных.....	247
Основные компоненты ScopedValue	248
Запуск ScopedValue.....	250
ScopedValue и структурированная конкурентность.....	257
Вопросы производительности	258
Удобство использования и проектирование API.....	258
В заключение	267
Глава 6. Актуальность реактивного программирования на Java в контексте виртуальных потоков.....	268
Основы реактивного программирования в Java	269
Блокирующий и неблокирующий ввод-вывод	270
Событийно-ориентированная архитектура.....	293
Асинхронные API	298
Реактивное программирование в Java	300
Понятие реактивных потоков	300
Backpressure.....	308
Преимущества и недостатки реактивного программирования	314
В заключение	316

Глава 7. Современные фреймворки, использующие виртуальные потоки	318
Spring Boot.....	318
Ручное конфигурирование	321
Quarkus.....	323
Jakarta EE.....	327
В заключение	330
Глава 8. Заключение и основные выводы	331
Об авторе	334
Иллюстрация на обложке	335