

Оглавление

От издательства	16
О научном редакторе русского издания.....	16
Об авторах	17
О техническом редакторе.....	17
О техническом корректоре	18
Благодарности.....	18
Предисловие	19
Как организована книга.....	19
Для кого эта книга	20
Помощь читателю	20
Файлы для самостоятельной работы	20
Как связаться с авторами.....	20
Глава 1. История Java в двух словах.....	21
Введение.....	21
Этапы развития Java	22
Различия ключевых версий Java	24
Кодирование дженериков в Java 5.....	25
Функциональное программирование в Java 8	25
Модульное программирование — начиная с Java 11.....	26
Программирование текстовых блоков и записей — начиная с Java 17.....	27
Виртуальные потоки в Java 21.....	29
Что делать с устаревающими элементами и как их удалять.....	29
Выявление переименований	30
Преобразование в Jakarta EE	30
Переименование сертификатов	31
Принципы внесения изменений	31
Дополнительные материалы	32
Резюме	32
Глава 2. Освоение IDE: секрет успеха.....	33
История IDE.....	34
Запуск нового проекта	36
Создание проекта с нуля.....	36

Создание проекта из существующих исходных кодов.....	37
Создание проекта из системы контроля версий.....	37
Добавление проекта в систему контроля версий	37
Добавление модуля в существующий проект	38
Создание конфигурации запуска	38
Горячие клавиши	41
Навигация по кодовой базе.....	42
Горячие клавиши копирования и вставки	44
Переупорядочивание кода	44
Другие полезные горячие клавиши	45
Отладка кода.....	46
Отладка программы.....	46
Оптимизация производительности отладчика.....	48
Удаленная отладка	48
Отладка с помощью горячей замены.....	50
Рефакторинг кода	51
Избегайте дублирования кода	51
Переименование членов.....	56
Встраивание	57
Изменение сигнатур: добавление и удаление параметров.....	58
Освоение возможностей редактора	58
Автоматическое переформатирование кода	59
Режим колонок (Column Mode).....	61
Расширение IDE.....	62
Обзор Eclipse	63
Обзор VS Code	65
Сравнение различных IDE	67
Дополнительные материалы	68
Резюме	68

Глава 3. Git, Jira и Confluence в совместной работе в масштабах предприятия69

Совместная работа с помощью Git	70
Основы Git.....	71
Ключевые концепции Git.....	71
Различия между коммитами	72
Отображение статуса Git.....	72
Ветвление	72
Тегирование.....	73
Слияние	73
Чтение логов Git.....	73
Индексация изменений.....	74
Установка Git	75

Рабочий процесс Git на примере	75
Разница между fetch, merge и pull.....	83
Игра с ветками.....	84
Разрешение конфликтов слияния	86
Использование пулл- и мердж-реквестов	88
Использование Git Log	88
Перебазирование (Rebasing).....	89
Cherry-picking.....	93
Возврат и сброс (reverting и resetting).....	93
Оптимизация благодаря IDE-поддержке.....	94
Создание файлов README с помощью языка Markdown	104
Использование Gitflow при совместной работе	106
Применение Jira для координации корпоративных процессов.....	108
Начало работы с Jira.....	109
Создание проекта	110
Создание задачи.....	111
Связь с эпиком	112
Работа с досками.....	113
Создание спринта.....	114
Добавление пользователей	116
Добавление колонок.....	117
Использование фильтров	117
Просмотр своих задач.....	117
Запросы на JQL.....	118
Внесение массовых изменений.....	120
Интеграция с Git	121
Работа с Confluence — корпоративной системой управления знаниями	121
Дополнительные материалы	124
Резюме	125

Глава 4. Автоматизация CI/CD-сборок с помощью Maven, Gradle и Jenkins 126

Сборка с помощью Maven.....	127
Сборка базового проекта Maven	128
Понятие репозитория Maven и зависимостей	130
Различия между фазами жизненного цикла.....	132
Знакомство с POM	133
Использование распространенных плагинов.....	135
Работа с родительским POM	139
Использование других функций Maven	144
Работа с Maven в IDE	145
Конфигурация Maven для корпоративной среды.....	148

Сборка с помощью Gradle	149
Сборка базового проекта Gradle	150
Локальный репозиторий и зависимости в Gradle.....	152
Определение переменных	153
Использование Java-плагинов	154
Работа с Gradle в IDE.....	155
Поиск зависимостей.....	156
Интеграция с Jenkins	158
Установка Jenkins	159
Знакомство с терминологией Jenkins.....	160
Создание заданий.....	161
Изучение основных плагинов.....	171
Прочие возможности Jenkins	172
Сканирование с помощью Sonar.....	173
Практики CI/CD.....	174
Дополнительные материалы	175
Резюме	175

Глава 5. Отслеживание состояния приложения при помощи

логирования	176
Почему не стоит ограничиваться методом System.out.println?	177
Использование Java Util Logging	178
Сравнение уровней логирования.....	179
Форматирование значений	180
Конфигурирование базовых параметров	181
Настройка места вывода логов	182
Ленивое (отложенное) логирование	184
Наследование логгеров	184
Использование LOG4J	187
Сравнение уровней логирования.....	189
Форматирование значений	189
Конфигурирование базовых параметров.....	190
Сравнение форматов конфигурационных файлов.....	191
Настройка места вывода логов	196
Выбор формата вывода лог-сообщений	198
Ленивое логирование в Log4j.....	199
Использование SLF4J.....	200
Пропуск фреймворка логирования.....	200
Настройка SLF4J Simple	201
Сравнение уровней логирования.....	201
Форматирование значений	202
Ленивое логирование в SLF4J	202

Конфигурирование базовых параметров	203
Использование SLF4J с другими фреймворками логирования.....	204
Использование Logback.....	207
Сравнение уровней логирования.....	209
Конфигурирование базовых параметров	209
Настройка места вывода логов	210
Выбор формата вывода лог-сообщений	211
Дополнительные сведения о логировании.....	212
Стандарты кодирования	212
Оптимизация производительности	212
Выбор языка для логов.....	213
Предотвращение фальсификации логов.....	213
Агрегация и пересылка логов.....	214
Сравнение фреймворков логирования.....	214
Дополнительные материалы	215
Резюме	215
Глава 6. Знакомство со Spring Framework	216
Конфигурирование Spring.....	218
Использование конфигурационных XML-файлов	219
Использование Java-классов конфигурации	221
Использование сканирования компонентов	223
Настройка приложений Spring с помощью свойств.....	225
Внедрение свойств	225
Конфигурирование свойств для разных окружений с помощью профилей Spring	226
Ускорение разработки с помощью Spring Boot	228
Инициализация проектов Spring Boot с помощью Spring Initializr	229
Интеграция Initializr в IntelliJ IDEA.....	230
Работа со Spring MVC	234
Обработка ошибок в Spring.....	241
Отслеживание состояния приложения с помощью Actuator	242
Защита приложения с помощью Spring Security	245
Знакомство с терминологией информационной безопасности	245
Принципы обработки в Spring Security.....	247
Обзор проектов Spring	251
Дополнительные материалы	252
Резюме	252
Глава 7. Автоматизация тестирования кода	253
Основы тестирования.....	254
Staging-окружение тестовой директории	254

Конфигурирование JUnit в сборке Maven	255
Конфигурирование JUnit в сборке Gradle	259
Тесты при разработке через тестирование.....	261
Знакомство с JUnit	265
Последовательность выполнения тестов	265
Пропуск теста	267
Утверждения	268
Проверка условий с использованием Assume	270
Параметризованные тесты	271
Лучшие библиотеки для тестирования.....	272
AssertJ	272
JUnit Pioneer	274
Мокирование объектов	277
Мокирование с помощью Mockito	278
Мокирование с помощью EasyMock	285
Мокирование с помощью Spring MockMvc	286
Измерение покрытия тестами	289
Расчет покрытия тестами	290
Просмотр покрытия кода в IDE.....	291
Выполнение покрытия кода во время сборки.....	292
Оптимизация тестирования с помощью IntelliJ IDEA.....	293
Просмотр Assert-сообщений	293
Запуски тестов.....	294
Переход к тесту	295
Генерация тестов на основе boilerplate-кода с помощью TestMe	295
Другие концепции тестирования	296
Использование BDD.....	296
Контрактное тестирование	297
Golden Master Pattern	297
Мутационное тестирование.....	298
Выбор между DRY и DAMP	299
Дополнительные материалы	300
Резюме	301

Глава 8. Project Lombok: улучшение кода с помощью аннотаций..... 302

Подготовка среды разработки для Lombok.....	303
Установка в IntelliJ IDEA	303
Установка в Eclipse.....	304
Установка в VS Code	305
Включение в Maven или Gradle	305
Компиляция с помощью javac	306

Использование Lombok	306
Переход на стиль Lombok	308
Использование универсальной аннотации @Data	308
Автоматическая генерация методов @Getter и @Setter	309
Использование @ToString	311
Использование @EqualsAndHashCode	312
Генерация конструкторов с аннотациями @AllArgsConstructor, @RequiredArgsConstructor и @NoArgsConstructor	313
Встраивание логгеров в код с помощью @Log и других аннотаций	314
Обработка кода с помощью delombok	316
Дополнительные материалы	316
Резюме	316

Глава 9. Параллелизация приложения с использованием конкурентности Java..... 318

Основы конкурентности	319
Решение конфликта при доступе потоков к общим ресурсам	322
Атомарные операции	325
Ключевое слово volatile	325
Переходы между состояниями потока	327
Синхронизация	328
Куча, стек и метaproстранство	330
Конкурентные коллекции	331
Встроенная поддержка конкурентности в Java Stream	332
Снижение сложности благодаря компонентам конкурентности	333
Блокировка с помощью ReentrantLock	333
Управление доступом потоков с помощью Phaser	334
Runnable, Callable и Future	335
Взаимодействие с ExecutorService	336
Создание реактивного кода с помощью CompletableFuture	338
Знакомство с виртуальными потоками	346
Межпотокое взаимодействие	352
Поддержка конкурентности в Spring Framework	354
Использование @Async и @EnableAsync	355
Планирование задач с помощью @Scheduled и @EnableScheduling	357
Запуск потоков с помощью ThreadPoolTaskExecutor	358
Понимание @Transactional	358
Дополнительные материалы	359
Резюме	359

Глава 10. Регулярные выражения 360

Введение в регулярные выражения.....	361
Основы синтаксиса регулярных выражений.....	363
Распространенные квантификаторы.....	364
Сопоставление базовых границ.....	366
Работа с основными классами символов.....	367
Выбор вариантов.....	369
Экранирование специальных символов.....	369
Использование регулярных выражений с Java API.....	370
Методы класса String.....	370
Работа с классами Pattern и Matcher.....	375
Управление поведением с помощью флагов.....	379
Расширенный синтаксис регулярных выражений.....	381
Анализ контекста (окружающего текста).....	381
Различия между типами квантификаторов.....	383
Выявление типичных ошибок.....	386
Программирование с использованием регулярных выражений.....	387
Использование с фреймворками и инструментами.....	389
Регулярные выражения для Apache Commons Validator.....	389
Регулярные выражения для JUnit.....	390
Регулярные выражения для Log4j.....	391
Регулярные выражения для валидации бинов.....	391
Регулярные выражения для Spring.....	391
Дополнительные материалы.....	392
Резюме.....	393

Глава 11. Аспектно-ориентированное программирование 394

Зачем нужны аспекты?.....	394
Создание первого примера.....	395
Исследуем язык pointcut-выражений.....	399
Подстановочные символы *.....	399
Подстановочные символы.....	400
Использование @AfterReturning.....	400
Использование @AfterThrowing.....	401
Использование @After.....	402
Использование @Around.....	402
Использование @Pointcut.....	404
Комбинирование срезов.....	406
Pointcut-выражения, основанные на аннотациях.....	406
Дополнительные материалы.....	409
Резюме.....	409

Глава 12. Мониторинг приложений: наблюдаемость в экосистеме Java 410

Введение в наблюдаемость	411
Начало работы с Prometheus.....	411
Запуск тестового приложения.....	412
Установка экспортера.....	412
Установка Prometheus.....	413
Настройка скрейпинга.....	414
Знакомство с PromQL — языком запросов Prometheus	416
Использование HTTP API Prometheus	423
Создание кастомных метрик	425
Добавляем управление оповещениями.....	430
Создаем дашборды с помощью Grafana	432
Логирование и трассировка.....	436
Основные системы распределенного логирования.....	436
Реализация трассировки в приложениях Spring Boot	439
Дополнительные материалы.....	439
Резюме	440

Глава 13. Оптимизация производительности сервисов 441

Основные понятия и терминология.....	442
Бенчмаркинг.....	443
Бенчмаркинг со встроенными Java API	443
Микробенчмаркинг	445
Настройка параметров JVM.....	449
Настройка параметров памяти	449
Сборка мусора	450
Тестирование с помощью JMeter.....	453
Создание плана тестирования	454
Запуск нагрузочного тестирования	457
Анализ результатов.....	458
Использование инструментов JDK.....	460
Использование Java Flight Recorder.....	461
Визуализация в Java Mission Control.....	464
Мониторинг с помощью JConsole	465
Применение VisualVM.....	466
Отчетность с помощью JStat.....	467
Оптимизация приложения	468
Разбор примера	468
Использование SonarQube.....	471
Что можно предпринять для повышения производительности.....	472
Дополнительные материалы.....	472
Резюме	473

Глава 14. Знакомство с экосистемой Java	474
Как писать Javadoc.....	475
Сравнение языков JVM	476
Kotlin	476
Groovy	477
Scala.....	478
Jakarta EE.....	479
Сравнение баз данных разного типа.....	480
Реляционные базы данных	480
Базы данных NoSQL	482
Интеграция.....	482
Развертывание Java-приложений.....	483
Различие между веб-серверами и серверами приложений	483
Использование контейнеров	484
Развертывание в облаке	486
Создание REST API	487
Создание веб-сервисов	487
Документирование API.....	488
Выбор виртуальной машины	490
Обзор библиотек	490
Защита приложения.....	491
Что такое OWASP Top 10	491
Различные типы инструментов безопасности	493
Отслеживание изменений в экосистеме	494
Дополнительные материалы	494
Резюме	495

Приложение. Чтение и запись XML, JSON и YAML

