

ОГЛАВЛЕНИЕ

Предисловие от автора	10
Предисловие от Friflex	11
От издательства	12
Глава 1. Краткая история Dart и встроенные типы данных	13
1.1. Краткая история и основные особенности Dart	13
1.2. Установка и настройка рабочей среды	15
1.3. Правила именования	20
1.4. Встроенные типы данных Dart	21
1.4.1. Числа (int, double)	22
1.4.2. Строки (String)	23
1.4.3. Логические значения (bool)	26
1.4.4. Списки (List)	26
1.4.5. Записи (Record)	34
1.4.6. Множества (Set)	36
1.4.7. Таблицы (Map)	39
1.4.8. Руны (Runes) и символы (Symbols)	41
1.5. Модификаторы final, const и late	42
1.6. Null-безопасность (null-safety)	43
1.7. Тип данных dynamic vs Object	45
1.8. Чтение данных с клавиатуры	47
Резюме	50
Вопросы для самопроверки	50
Глава 2. Операторы, сопоставление с шаблоном и управляющие конструкции	52
2.1. Основные операторы Dart	52
2.2. Что такое сопоставление с шаблоном и деструктурирование	56
2.2.1. Деструктурирование списка	57
2.2.2. Деструктурирование записи	58
2.2.3. Деструктурирование таблицы	59
2.2.4. Деструктурирование экземпляра класса	60
2.3. Управление потоком выполнения кода	61
2.3.1. Условный оператор if	61
2.3.2. Оператор if-case	64
2.3.3. Тернарный оператор ?	67

2.3.4. Оператор ??	67
2.3.5. Операторы циклов (for, for-in, while и do-while).....	68
2.3.6. Операторы потока выполнения (break, continue, return)	71
2.3.7. Оператор выбора потока выполнения (switch-case)	73
Проект: игра «Крестики-нолики» v.0.....	85
Резюме.....	89
Вопросы для самопроверки	90
Глава 3. Функции, библиотеки, пакеты и их тестирование	91
3.1. Функции в Dart	91
3.1.1. Объявление входных аргументов функции.....	93
3.1.2. Необязательные аргументы функции по умолчанию	96
3.1.3. Область видимости переменных	98
3.1.4. Обращение к функции через переменную	99
3.1.5. Функция как входной аргумент другой функции.....	99
3.1.6. Туре Alias	101
3.1.7. Анонимные и стрелочные функции	103
3.1.8. Замыкания.....	104
3.1.9. Рекурсия.....	106
3.1.10. Генераторные функции	107
3.2. Создание и импортирование библиотек.....	110
3.2.1. Импортирование кода из файла с расширением .dart	111
3.2.2. Импортирование части функциональности	114
3.2.3. Отложенная загрузка импортируемого файла или библиотеки	115
3.2.4. Создание и использование библиотеки	115
3.3. Тестирование функций.....	117
3.3.1. Установка пакета test в проект	118
3.3.2. Написание тестов	119
3.3.3. Запуск тестов.....	121
3.3.4. Конфигурация тестов	123
3.3.5. Тестирование пользовательского ввода.....	127
3.4. Создание пакета и его подключение к проекту	129
3.4.1. Создание пакета.....	129
3.4.2. Локальное подключение пакета.....	131
3.4.3. Удаленное подключение пакета	132
Проект: игра «Крестики-нолики» v.1.....	132
Резюме.....	137
Вопросы для самопроверки	137
Глава 4. Объектно-ориентированное программирование.....	139
4.1. Конструктор класса.....	143
4.1.1. Именованный конструктор	146
4.1.2. Константный конструктор.....	149
4.1.3. Фабричный конструктор.....	150
4.2. Статические переменные и методы класса	152

4.3. Перегрузка операторов	153
4.3.1. Перегрузка арифметических операторов	153
4.3.2. Перегрузка операторов сравнения	155
4.3.3. Перегрузка битовых операторов.....	156
4.3.4. Перегрузка индексных операторов.....	158
4.4. Методы расширения.....	159
4.5. Наследование и переопределение методов.....	161
4.6. Абстрактный класс и интерфейс	170
4.7. Продвижение приватных полей	176
4.8. Модификаторы класса.....	177
4.8.1. Отсутствие модификатора.....	178
4.8.2. Модификатор abstract.....	180
4.8.3. Модификатор base.....	180
4.8.4. Модификаторы interface и abstract interface	184
4.8.5. Модификатор final.....	190
4.8.6. Модификатор sealed.....	191
4.8.7. Миксины и модификатор класса mixin.....	201
4.9. Дженерики (generics)	206
4.10. Перечисления (enum)	208
4.11. Расширение типа (extension types)	216
4.12. Исключения (exceptions).....	224
4.12.1. Конструкция try...catch...finally	225
4.12.2. Генерация исключений и ошибок.....	227
4.12.3. Пользовательские исключения и ошибки	229
4.12.4. Трассировка стека.....	230
4.12.5. Утверждение.....	231
4.13. Тестирование классов.....	232
Проект: игра «Крестики-нолики» v.2.....	233
Резюме.....	239
Вопросы для самопроверки	240
Глава 5. Сборка приложения. Работа с файлами и каталогами	241
5.1. Сборка приложения	241
5.1.1. Флаг exe.....	242
5.1.2. Флаг aot-snapshot.....	243
5.1.3. Флаг jit-snapshot.....	243
5.1.4. Флаг kernel	244
5.2. Конфигурация запускаемого приложения	244
5.3. Работа с файлами.....	246
5.4. Работа с каталогами	251
5.5. База данных на основе файла и однонаправленного списка	254
5.6. Работа с JSON-файлами.....	267
5.6.1. Зачем нужен pull	273
5.6.2. Валидация данных.....	273
5.6.3. Пакет json_serializable для десериализации и сериализации объектов в JSON-формат.....	276

5.7. Простая БД по типу «ключ:значение» в формате JSON.....	280
Проект: игра «Крестики-нолики» v.3.....	285
Резюме.....	292
Вопросы для самопроверки.....	292
Глава 6. Асинхронное и сетевое программирование. Изолят.....	293
6.1. Архитектура цикла событий в Dart.....	294
6.1.1. Базовая концепция цикла и очереди событий.....	294
6.1.2. Очереди и цикл событий в Dart.....	295
6.2. Асинхронное программирование.....	297
6.2.1. Future API, async и await.....	297
6.2.2. Поток (stream).....	309
6.3. Изоляты (isolate).....	318
6.4. Тонкости работы с изоляционными группами.....	333
6.4.1. Уникальный идентификатор объекта.....	333
6.4.2. Объявление переменной или имени.....	335
6.4.3. Жизненный цикл объекта.....	341
6.4.4. Общая память изоляционных групп.....	343
6.4.5. Разница в передаче обычных и константных объектов между изолятами группы.....	345
6.4.6. Создание изолята из другого изолята.....	349
6.5. Async или Isolate.....	353
6.6. Зоны (zones).....	354
6.7. Сетевое программирование.....	360
6.7.1. Разработка пакета protocol.....	362
6.7.2. Клиент-серверное приложение на основе TCP.....	368
6.7.3. Передача данных между сервером и клиентом по протоколу UDP.....	375
6.7.4. Структура HTTP-сообщений и REST URL.....	379
6.7.5. HTTP-сервер и клиент.....	381
Проект: игра «Крестики-нолики» v.4.....	388
Разработка общего Dart-пакета common.....	389
Разработка серверной части.....	400
Разработка клиентской части.....	409
Запуск игры.....	418
Задания на изменение проекта.....	419
Резюме.....	420
Вопросы для самопроверки.....	420
Глава 7. Алгоритмы, структуры данных на Dart и встроенные коллекции.....	422
7.1. Структуры данных.....	423
7.1.1. Массив (array).....	424
7.1.2. Односвязный список (single-linked list).....	429
7.1.3. Двусвязный список (double-linked list).....	435
7.1.4. Хеш-таблица (hash table).....	440
7.1.5. Множество (set).....	445
7.1.6. Стек (stack).....	447

7.1.7. Очередь (queue).....	451
7.1.8. Двоичное дерево поиска (Binary Search Tree).....	455
7.1.9. AVL-дерево (AVL-tree).....	467
7.1.10. Куча (heap).....	474
7.2. Алгоритмы сортировки.....	484
7.2.1. Сортировка пузырьком (bubble sort).....	484
7.2.2. Шейкерная сортировка (cocktail sort).....	487
7.2.3. Сортировка подсчетом (counting sort).....	489
7.2.4. Сортировка слиянием (merge sort).....	490
7.2.5. Быстрая сортировка (quick sort).....	493
7.2.6. Пирамидальная сортировка (heap sort).....	495
7.3. Алгоритмы поиска.....	497
7.3.1. Линейный поиск (linear search).....	498
7.3.2. Бинарный поиск (binary search).....	499
7.3.3. Троичный поиск (ternary search).....	501
7.4. Встроенные коллекции Dart.....	502
7.4.1. Map.....	503
7.4.2. Set.....	508
7.4.3. List.....	510
7.4.4. Queue.....	511
7.4.5. LinkedList.....	513
Резюме.....	514
Вопросы для самопроверки.....	516
Глава 8. Интероперабельность в Dart.....	518
8.1. Интерфейс внешних функций.....	518
8.2. Нативные типы данных Си и их аналоги в Dart.....	519
8.3. Перенос функции на нативный уровень.....	521
8.4. Автогенерация кода для подключения библиотеки.....	525
8.5. Вызов функции обратного вызова Dart из Си.....	528
8.6. Связывание структур Си с Dart и их финализация.....	532
8.6.1. Объявление структуры и ее связывание с классом Dart.....	532
8.6.2. Обертка над структурой и ее финализация.....	536
Резюме.....	538
Вопросы для самопроверки.....	539
Курсы автора на stepik.org.....	540
Список используемых источников.....	541