

Паттерны проектирования JavaScript

Создаем быстрые и эффективные приложения
любого масштаба

Уго Ди Франческо

SPRINT
BOOK

КРАТКОЕ СОДЕРЖАНИЕ

Об авторе	12
О научных редакторах.....	13
Предисловие	14

Часть I. Паттерны проектирования

Глава 1. Работа с порождающими паттернами проектирования	20
Глава 2. Реализация структурных паттернов проектирования.....	43
Глава 3. Использование поведенческих паттернов проектирования	71

Часть II. Архитектура и паттерны пользовательского интерфейса

Глава 4. Изучение паттернов библиотек реактивных представлений	98
Глава 5. Стратегии рендеринга и гидратации страниц	129
Глава 6. Микрофронтенды, зоны и островные архитектуры.....	165

Часть III. Паттерны производительности и безопасности

Глава 7. Паттерны производительности асинхронного программирования	204
Глава 8. Паттерны событийно-ориентированного программирования	231
Глава 9. Максимизация производительности — отложенная загрузка и разделение кода	256
Глава 10. Стратегии загрузки активов и выполнение кода вне основного потока.....	282

ОГЛАВЛЕНИЕ

Об авторе	12
О научных редакторах	13
Предисловие	14
Для кого эта книга.....	15
Структура издания.....	15
Как извлечь максимальную пользу из книги	17
Загрузка файлов кода примеров	17
Условные обозначения	18
От издательства.....	18

Часть I. Паттерны проектирования

Глава 1. Работа с порождающими паттернами проектирования	20
Что такое порождающие паттерны проектирования	21
Реализация паттерна Прототип в JavaScript	22
Реализация	22
Вариант применения.....	24
Паттерн Одиночка с немедленной и отложенной инициализацией в JavaScript.....	28
Реализация	29
Варианты использования.....	33
Преимущества паттерна «класс-одиночка»	33
Одиночка без полей класса, использующий поведение модуля ES	35
Паттерн Фабрика в JavaScript	37
Реализация	38
Варианты использования.....	40
Преимущества современного JavaScript.....	40
Резюме.....	41

Глава 2. Реализация структурных паттернов проектирования	43
Файлы примеров	44
Что такое структурные паттерны проектирования	44
Реализация паттерна Заместитель с помощью объектов Proxy и Reflect	44
Реализация заместителя для редактирования	45
Варианты использования	46
Улучшение паттерна Заместитель с помощью глобальных объектов Proxy и Reflect.....	46
Декоратор в JavaScript.....	52
Реализация	52
Варианты использования	53
Преимущества и недостатки	53
Приспособленец в JavaScript.....	55
Реализация	55
Варианты использования	59
Преимущества и недостатки	59
Адаптер в JavaScript	62
Реализация	62
Варианты использования	66
Преимущества и недостатки	66
Резюме.....	69
Глава 3. Использование поведенческих паттернов проектирования	71
Файлы примеров	72
Что такое поведенческие паттерны проектирования	72
Паттерн Наблюдатель в JavaScript.....	73
Реализация	73
Варианты использования паттерна Наблюдатель.....	76
Недостатки и преимущества	77
Состояние и Стратегия в JavaScript и упрощенный подход.....	79
Реализация	80
Варианты использования паттернов Состояние и Стратегия.....	87
Недостатки и преимущества	87
Посетитель в JavaScript.....	92
Реализация	93
Варианты использования паттерна Посетитель	95
Резюме.....	96

Часть II. Архитектура и паттерны пользовательского интерфейса

Глава 4. Изучение паттернов библиотек реактивных представлений	98
Файлы примеров	99
Что такое паттерны библиотеки реактивных представлений	99
Паттерн Рендер-проп	100
Варианты использования	100
Пример реализации	105
Недостатки	112
Компонент высшего порядка	113
Пример реализации	113
Варианты использования	114
Недостатки	114
Паттерн Хуки.....	115
Пример реализации.....	116
Варианты использования	120
Недостатки	120
Паттерн Провайдер.....	121
Пример использования: проблема сквозной передачи пропов.....	121
Пример реализации.....	122
Недостатки	127
Резюме.....	127
Глава 5. Стратегии рендеринга и гидратации страниц	129
Файлы примеров	130
Клиентский и серверный рендеринг с помощью React	130
Рендеринг на стороне клиента в React	130
Рендеринг на стороне сервера в React	133
Компромиссы между рендерингом на стороне клиента и сервера	136
Статический рендеринг с помощью Next.js.....	137
Автоматическая генерация статических сайтов	138
Генерация статических данных с помощью стороннего источника данных.....	140
Статическая генерация с динамическими путями	144

Стратегии гидратации страниц	152
Общие проблемы регидратации React.....	157
Потоковый рендеринг на стороне сервера в React	160
Резюме.....	164
Глава 6. Микрофронтенды, зоны и островные архитектуры	165
Файлы примеров	166
Обзор микрофронтендов	166
Основные преимущества.....	166
Классические паттерны микрофронтенда.....	167
Другие проблемы в мире микрофронтендов	171
Составление приложений с помощью зон Next.js.....	172
Корневое приложение	173
Добавление приложения /search	176
Добавление приложения /checkout.....	179
Преимущества/поддержка масштабирования команды	184
Недостатки зон Next.js	185
Масштабирование чувствительных к производительности страниц с помощью архитектуры островов.....	186
Настройка островов с помощью is-land.....	187
Остров товара	188
Остров корзины	190
Остров сопутствующих товаров	194
Масштабирование с помощью команды — сборка островов.....	200
Недостатки	201
Резюме.....	201

Часть III. Паттерны производительности и безопасности

Глава 7. Паттерны производительности асинхронного программирования	204
Файлы примеров	204
Управление последовательными асинхронными операциями с помощью синтаксиса async/await и промисов	205
Паттерны параллельных асинхронных операций.....	211
Асинхронная отмена и тайм-ауты с помощью класса AbortController	218
Замедление, откладывание и пакетирование асинхронных операций.....	222
Резюме.....	229

Глава 8. Паттерны событийно-ориентированного программирования	231
Файлы примеров	232
Оптимизация слушателей событий с помощью делегирования событий.....	232
Паттерны для безопасного обмена сообщениями между фреймом и нативным мостом WebView	241
Антипаттерны производительности слушателей событий.....	253
Резюме	254
Глава 9. Максимизация производительности — отложенная загрузка и разделение кода	256
Файлы примеров	257
Динамический импорт и разделение кода с помощью Vite.....	257
Разделение и объединение кода на основе маршрутов	261
Загрузка JavaScript при видимости и взаимодействии элементов	264
Резюме	280
Глава 10. Стратегии загрузки активов и выполнение кода вне основного потока	282
Файлы примеров	283
Оптимизация загрузки активов — async, defer, preconnect, preload и prefetch.....	283
Использование параметра strategy компонента Script в Next.js для оптимизации загрузки активов	292
Загрузка и запуск скриптов в потоке обработчиков	295
Резюме	299