

УДК 004.438С#
ББК 32.973.26-018.2
Д94

Дэвис, Алекс.

Д94 Асинхронное программирование в C# 5.0 / А. Дэвис. — 2-е изд., эл. — 1 файл pdf: 122 с. — Москва : ДМК Пресс, 2023. — Систем. требования: Adobe Reader XI либо Adobe Digital Editions 4.5 ; экран 10". — Текст : электронный.

ISBN 978-5-89818-335-6

Из этого краткого руководства вы узнаете, как механизм `async` в C# 5.0 позволяет упростить написание асинхронного кода. Помимо ясного введения в асинхронное программирование вообще, вы найдете углубленное описание работы этого конкретного механизма и ответ на вопрос, когда и зачем использовать его в собственных приложениях.

В книге рассматриваются следующие вопросы:

- Как писать асинхронный код вручную и как механизм `async` скрывает неприглядные детали.
- Новые способы повышения производительности серверного кода в приложениях ASP.NET.
- Совместная работа `async` и WinRT в приложениях для Windows 8.
- Смысл ключевого слова `await` в `async`-методах.
- В каком потоке .NET выполняется асинхронный код в каждой точке программы.
- Написание асинхронных API, согласованных с паттерном Task-based Asynchronous Pattern (TAP).
- Распараллеливание программ для задействования возможностей современных компьютеров.
- Измерение производительности `async`-кода и сравнение с альтернативными подходами.

Книга рассчитана на опытных программистов на C#, но будет понятна и начинающим. Она изобилует примерами кода, который можно использовать в своих программах.

УДК 004.438С#
ББК 32.973.26-018.2

Электронное издание на основе печатного издания: Асинхронное программирование в C# 5.0 / А. Дэвис. — Москва : ДМК Пресс, 2013. — 120 с. — ISBN 978-5-94074-886-1. — Текст : непосредственный.

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

В соответствии со ст. 1299 и 1301 ГК РФ при устранении ограничений, установленных техническими средствами защиты авторских прав, правообладатель вправе требовать от нарушителя возмещения убытков или выплаты компенсации.

ISBN 978-5-89818-335-6

© 2012 Alex Davies. This translation is published and sold by permission of O'Reilly Media, Inc., which owns or controls all rights to publish and sell the same.

© Оформление, перевод на русский язык
ДМК Пресс, 2013



ОГЛАВЛЕНИЕ

Предисловие	9
Предполагаемая аудитория.....	9
Как читать эту книгу	10
Принятые соглашения	10
О примерах кода.....	11
Как с нами связаться	11
Благодарности	12
Об авторе	12
Глава 1. Введение	13
Асинхронное программирование	13
Чем так хорош асинхронный код?	14
Что такое async?	15
Что делает async?	15
Async не решает все проблемы.....	17
Глава 2. Зачем делать программу асинхронной ...	18
Приложения с графическим интерфейсом пользователя для настольных компьютеров	18
Аналогия с кафе	19
Серверный код веб-приложения	20
Еще одна аналогия: кухня в ресторане.....	21
Silverlight, Windows Phone и Windows 8	22
Параллельный код	23
Пример.....	24
Глава 3. Написание асинхронного кода вручную ...	26
О некоторых асинхронных паттернах в .NET.....	26
Простейший асинхронный паттерн	28
Введение в класс Task	29
Чем плоха реализация асинхронности вручную?	30
Переработка примера с использованием написанного вручную асинхронного кода.....	31

Глава 4. Написание асинхронных методов 33

Преобразование программы скачивания значков к виду, использующему <code>async</code>	33
<code>Task</code> и <code>await</code>	34
Тип значения, возвращаемого асинхронным методом	36
<code>Async</code> , сигнатуры методов и интерфейсы	37
Предложение <code>return</code> в асинхронных методах	38
Асинхронные методы «заразны»	39
Асинхронные анонимные делегаты и лямбда-выражения	40

Глава 5. Что в действительности делает `await` ... 41

Приостановка и возобновление метода	41
Состояние метода	42
Контекст	43
Когда нельзя использовать <code>await</code>	44
Блоки <code>catch</code> и <code>finally</code>	44
Блоки <code>lock</code>	45
Выражения LINQ-запросов	46
Небезопасный код	47
Запоминание исключений	47
Асинхронные методы до поры исполняются синхронно	48

Глава 6. Паттерн TAP 50

Что специфицировано в TAP?	50
Использование <code>Task</code> для операций, требующих большого объема вычислений	52
Создание задачи-марионетки	53
Взаимодействие с прежними асинхронными паттернами	55
Холодные и горячие задачи	56
Предварительная работа	56

Глава 7. Вспомогательные средства для асинхронного кода 58

Задержка на указанное время	58
Ожидание завершения нескольких задач	59
Ожидание завершения любой задачи из нескольких	61
Создание собственных комбинаторов	62
Отмена асинхронных операций	64
Информирование о ходе выполнения асинхронной операции	65

Глава 8. В каком потоке выполняется мой код? ... 67

До первого <code>await</code>	67
-------------------------------------	----

Во время асинхронной операции.....	68
Подробнее о классе SynchronizationContext.....	69
await и SynchronizationContext.....	69
Жизненный цикл асинхронной операции	70
Когда не следует использовать SynchronizationContext	73
Взаимодействие с синхронным кодом.....	74
Глава 9. Исключения в асинхронном коде	76
Исключения в async-методах, возвращающих Task.....	76
Незамеченные исключения	78
Исключения в методах типа async void.....	79
Выстрелил и забыл	79
AggregateException и WhenAll.....	80
Синхронное возбуждение исключений	81
Блок finally в async-методах	82
Глава 10. Организация параллелизма с помощью механизма async	83
await и блокировки.....	83
Акторы.....	85
Использование акторов в C#	86
Библиотека Task Parallel Library Dataflow	87
Глава 11. Автономное тестирование асинхронного кода.....	90
Проблема автономного тестирования в асинхронном окружении	90
Написание работающих асинхронных тестов вручную	91
Поддержка со стороны каркаса автономного тестирования	91
Глава 12. Механизм async в приложениях ASP.NET	93
Преимущества асинхронного веб-серверного кода.....	93
Использование async в ASP.NET MVC 4	94
Использование async в предыдущих версиях ASP.NET MVC	94
Использование async в ASP.NET Web Forms	95
Глава 13. Механизм async в приложениях WinRT ...	97
Что такое WinRT?	97
Интерфейсы IAsyncAction и IAsyncOperation<T>	98
Отмена	99

Информирование о ходе выполнения	99
Реализация асинхронных методов в компоненте WinRT	100

Глава 14. Подробно о преобразовании асинхронного кода, осуществляемом компилятором 102

Метод-заглушка	102
Структура конечного автомата.....	103
Метод MoveNext	105
Наш код.....	106
Преобразование предложений return в код завершения.....	106
Переход в нужное место метода.....	106
Приостановка метода в месте встречи await.....	107
Возобновление после await.....	107
Синхронное завершение	107
Перехват исключений.....	108
Более сложный код	108
Разработка типов, допускающих ожидание	109
Взаимодействие с отладчиком	110

Глава 15. Производительность асинхронного кода 112

Измерение накладных расходов механизма async.....	112
Async и блокирующая длительная операция	113
Оптимизация асинхронного кода для длительной операции	116
Async-методы и написанный вручную асинхронный код.....	116
Async и блокирование без длительной операции	117
Оптимизация асинхронного кода без длительной операции	118
Резюме	119