
Содержание

Об авторе	22
О научном редакторе	22
Благодарности	23
Предисловие	25
Структура книги	27
Книга для всех программирующих	27
Необходимое программное обеспечение	27
Код, доступный в Интернете	27
Что еще почитать	28
От издательства	28
ЧАСТЬ I. ЯЗЫК C#	29
Глава 1. Создание C#	31
Генеалогическое дерево C#	32
Язык C — начало современной эпохи программирования	32
Появление ООП и C++	33
Появление Интернета и Java	33
Создание C#	34
Развитие C#	36
Связь C# со средой .NET Framework	37
О среде NET Framework	37
Принцип действия CLR	38
Управляемый и неуправляемый код	38
Общезыковая спецификация	39
Глава 2. Краткий обзор элементов C#	41
Объектно-ориентированное программирование	41
Инкапсуляция	42
Полиморфизм	43
Наследование	44
Первая простая программа	44
Применение компилятора командной строки csc.exe	45
Применение интегрированной среды разработки Visual Studio	46
Построчный анализ первого примера программы	50
Обработка синтаксических ошибок	53
Незначительное изменение программы	54
Вторая простая программа	54

8 Содержание

Другие типы данных	57
Два управляющих оператора	58
Условный оператор	58
Оператор цикла	60
Использование кодовых блоков	61
Точка с запятой и оформление исходного текста программы	63
Ключевые слова C#	64
Идентификаторы	65
Библиотека классов среды .NET Framework	66
Глава 3. Типы данных, литералы и переменные	67
О значении типов данных	67
Типы значений в C#	68
Целочисленные типы	69
Типы для представления чисел с плавающей точкой	71
Десятичный тип данных	73
Символы	74
Логический тип данных	75
Некоторые возможности вывода	76
Литералы	79
Шестнадцатеричные литералы	80
Управляющие последовательности символов	80
Строковые литералы	81
Более подробное рассмотрение переменных	83
Инициализация переменной	83
Динамическая инициализация	84
Неявно типизированные переменные	85
Область действия и время существования переменных	86
Преобразование и приведение типов	89
Автоматическое преобразование типов	90
Приведение несовместимых типов	91
Преобразование типов в выражениях	93
Приведение типов в выражениях	95
Глава 4. Операторы	97
Арифметические операторы	97
Операторы инкремента и декремента	98
Операторы отношения и логические операторы	101
Укороченные логические операторы	104
Оператор присваивания	106
Составные операторы присваивания	107
Поразрядные операторы	107
Поразрядные операторы И, ИЛИ, исключающее ИЛИ и НЕ	108
Операторы сдвига	114

Поразрядные составные операторы присваивания	117
Оператор ?	117
Использование пробелов и круглых скобок	119
Предшествование операторов	119
Глава 5. Управляющие операторы	121
Оператор if	121
Вложенные операторы if	122
Конструкция if-else-if	124
Оператор switch	125
Вложенные операторы switch	129
Оператор цикла for	129
Некоторые разновидности оператора цикла for	131
Оператор цикла while	137
Оператор цикла do-while	138
Оператор цикла foreach	139
Применение оператора break для выхода из цикла	139
Применение оператора continue	142
Оператор return	143
Оператор goto	143
Глава 6. Введение в классы, объекты и методы	147
Основные положения о классах	147
Общая форма определения класса	148
Определение класса	149
Создание объектов	153
Переменные ссылочного типа и присваивание	154
Методы	155
Добавление метода в класс Building	156
Возврат из метода	158
Возврат значения	159
Использование параметров	162
Добавление параметризованного метода в класс Building	164
Исключение недоступного кода	166
Конструкторы	166
Параметризованные конструкторы	168
Добавление конструктора в класс Building	169
Еще раз об операторе new	170
Применение оператора new вместе с типами значений	170
“Сборка мусора” и применение деструкторов	171
Деструкторы	172
Ключевое слово this	174
Глава 7. Массивы и строки	177
Массивы	177

10 Содержание

Одномерные массивы	178
Многомерные массивы	182
Двумерные массивы	182
Массивы трех и более измерений	183
Инициализация многомерных массивов	184
Ступенчатые массивы	185
Присваивание ссылок на массивы	187
Применение свойства Length	189
Применение свойства Length при обращении со ступенчатыми массивами	191
Неявно типизированные массивы	192
Оператор цикла foreach	194
Строки	198
Построение строк	198
Обращение со строками	199
Массивы строк	203
Постоянство строк	205
Применение строк в операторах switch	206
Глава 8. Подробнее о методах и классах	209
Управление доступом к членам класса	209
Модификаторы доступа	210
Организация закрытого и открытого доступа	212
Практический пример организации управления доступом	212
Передача объектов методам по ссылке	218
Способы передачи аргументов методу	220
Использование модификаторов параметров ref и out	222
Использование модификатора параметра ref	223
Использование модификатора параметра out	224
Использование модификаторов ref и out для ссылок на объекты	227
Использование переменного числа аргументов	229
Возврат объектов из методов	231
Возврат массива из метода	234
Перегрузка методов	235
Перегрузка конструкторов	241
Вызов перегружаемого конструктора с помощью ключевого слова this	245
Инициализаторы объектов	246
Необязательные аргументы	247
Необязательные аргументы и перегрузка методов	249
Необязательные аргументы и неоднозначность	250
Практический пример использования необязательных аргументов	251
Именованные аргументы	252
Метод Main()	254
Возврат значений из метода Main()	254

Передача аргументов методу Main()	255
Рекурсия	257
Применение ключевого слова static	260
Статические конструкторы	265
Статические классы	266
Глава 9. Перегрузка операторов	269
Основы перегрузки операторов	270
Перегрузка бинарных операторов	270
Перегрузка унарных операторов	273
Выполнение операций со встроенными в С# типами данных	277
Перегрузка операторов отношения	281
Перегрузка операторов true и false	283
Перегрузка логических операторов	286
Простой способ перегрузки логических операторов	286
Как сделать укороченные логические операторы доступными для применения	288
Операторы преобразования	292
Рекомендации и ограничения по перегрузке операторов	297
Еще один пример перегрузки операторов	298
Глава 10. Индексаторы и свойства	303
Индексаторы	303
Создание одномерных индексаторов	304
Перегрузка индексаторов	307
Индексаторы без базового массива	310
Многомерные индексаторы	311
Свойства	313
Автоматически реализуемые свойства	318
Применение инициализаторов объектов в свойствах	319
Ограничения, присущие свойствам	320
Применение модификаторов доступа в аксессорах	320
Применение индексаторов и свойств	324
Глава 11. Наследование	329
Основы наследования	329
Доступ к членам класса и наследование	333
Организация защищенного доступа	336
Конструкторы и наследование	337
Вызов конструкторов базового класса	339
Наследование и сокрытие имен	343
Применение ключевого слова base для доступа к скрытому имени	344
Создание многоуровневой иерархии классов	346
Порядок вызова конструкторов	349
Ссылки на базовый класс и объекты производных классов	351

12 Содержание

Виртуальные методы и их переопределение	355
Что дает переопределение методов	359
Применение виртуальных методов	360
Применение абстрактных классов	363
Предотвращение наследования с помощью ключевого слова sealed	367
Класс object	368
Упаковка и распаковка	370
Класс object как универсальный тип данных	372
Глава 12. Интерфейсы, структуры и перечисления	375
Интерфейсы	375
Реализация интерфейсов	377
Применение интерфейсных ссылок	381
Интерфейсные свойства	383
Интерфейсные индексы	385
Наследование интерфейсов	387
Скрытие имен при наследовании интерфейсов	388
Явные реализации	388
Выбор между интерфейсом и абстрактным классом	391
Стандартные интерфейсы для среды .NET Framework	391
Структуры	391
О назначении структур	395
Перечисления	397
Инициализация перечисления	399
Указание базового типа перечисления	399
Применение перечислений	399
Глава 13. Обработка исключительных ситуаций	403
Класс System.Exception	404
Основы обработки исключительных ситуаций	404
Применение пары ключевых слов try и catch	404
Простой пример обработки исключительной ситуации	405
Второй пример обработки исключительной ситуации	407
Последствия перехвата исключений	408
Обработка исключительных ситуаций — “изящный” способ устранения программных ошибок	410
Применение нескольких операторов catch	411
Перехват всех исключений	412
Вложение блоков try	413
Генерирование исключений вручную	414
Повторное генерирование исключений	415
Использование блока finally	416
Подробное рассмотрение класса Exception	418
Наиболее часто используемые исключения	420

Получение производных классов исключений	422
Перехват исключений производных классов	426
Применение ключевых слов checked и unchecked	428
Глава 14. Применение средств ввода-вывода	431
Организация системы ввода-вывода в C# на потоках	431
Байтовые и символьные потоки	432
Встроенные потоки	432
Классы потоков	432
Класс Stream	432
Классы байтовых потоков	434
Классы-оболочки символьных потоков	434
Двоичные потоки	436
Консольный ввод-вывод	436
Чтение данных из потока ввода с консоли	436
Применение метода ReadKey()	438
Запись данных в поток вывода на консоль	440
Класс FileStream и байтовый ввод-вывод в файл	441
Открытие и закрытие файла	441
Чтение байтов из потока файлового ввода-вывода	444
Запись в файл	446
Использование класса FileStream для копирования файла	448
Символьный ввод-вывод в файл	449
Применение класса StreamWriter	449
Применение класса StreamReader	451
Переадресация стандартных потоков	452
Чтение и запись двоичных данных	454
Класс BinaryWriter	454
Класс BinaryReader	455
Демонстрирование двоичного ввода-вывода	457
Файлы с произвольным доступом	461
Применение класса MemoryStream	463
Применение классов StringReader и StringWriter	465
Класс File	467
Копирование файлов с помощью метода Copy()	467
Применение методов Exists() и GetLastAccessTime()	468
Преобразование числовых строк в их внутреннее представление	469
Глава 15. Делегаты, события и лямбда-выражения	473
Делегаты	473
Групповое преобразование делегируемых методов	476
Применение методов экземпляра в качестве делегатов	477
Групповая адресация	478
Ковариантность и контравариантность	481

14 Содержание

Класс System.Delegate	483
Назначение делегатов	483
Анонимные функции	483
Анонимные методы	484
Передача аргументов анонимному методу	484
Возврат значения из анонимного метода	485
Применение внешних переменных в анонимных методах	486
Лямбда-выражения	488
Лямбда-оператор	488
Одиночные лямбда-выражения	489
Блочные лямбда-выражения	492
События	494
Пример групповой адресации события	496
Методы экземпляра в сравнении со статическими методами в качестве обработчиков событий	497
Применение аксессоров событий	500
Разнообразные возможности событий	504
Применение анонимных методов и лямбда-выражений вместе с событиями	504
Рекомендации по обработке событий в среде .NET Framework	506
Применение делегатов EventHandler<EventArgs> и EventHandler	508
Практический пример обработки событий	509
Глава 16. Пространства имен, препроцессор и сборки	513
Пространства имен	513
Объявление пространства имен	514
Предотвращение конфликтов имен с помощью пространств имен	516
Директива using	518
Вторая форма директивы using	520
Аддитивный характер пространств имен	521
Вложенные пространства имен	523
Глобальное пространство имен	524
Применение описателя псевдонима пространства имен ::	524
Препроцессор	528
Директива #define	529
Директивы #if и #endif	529
Директивы #else и #elif	531
Директива #undef	533
Директива #error	533
Директива #warning	534
Директива #line	534
Директивы #region и #endregion	534
Директива #pragma	534
Сборки и модификатор доступа internal	535

Модификатор доступа <code>internal</code>	536
Глава 17. Динамическая идентификация типов, рефлексия и атрибуты	537
Динамическая идентификация типов	537
Проверка типа с помощью оператора <code>is</code>	538
Применение оператора <code>as</code>	539
Применение оператора <code>typeof</code>	540
Рефлексия	541
Класс <code>System.Type</code> — ядро подсистемы рефлексии	541
Применение рефлексии	543
Получение сведений о методах	544
Вызов методов с помощью рефлексии	548
Получение конструкторов конкретного типа	550
Получение типов данных из сборок	555
Полностью автоматизированное обнаружение типов	560
Атрибуты	562
Основы применения атрибутов	563
Сравнение позиционных и именованных параметров	566
Встроенные атрибуты	570
Атрибут <code>AttributeUsage</code>	570
Атрибут <code>Conditional</code>	571
Атрибут <code>Obsolete</code>	572
Глава 18. Обобщения	575
Что такое обобщения	576
Простой пример обобщений	576
Различение обобщенных типов по аргументам типа	580
Повышение типовой безопасности с помощью обобщений	580
Обобщенный класс с двумя параметрами типа	583
Общая форма обобщенного класса	585
Ограниченные типы	585
Применение ограничения на базовый класс	586
Применение ограничения на интерфейс	594
Применение ограничения <code>new()</code> на конструктор	598
Ограничения ссылочного типа и типа значения	599
Установление связи между двумя параметрами типа с помощью ограничения	602
Применение нескольких ограничений	603
Получение значения, присваиваемого параметру типа по умолчанию	604
Обобщенные структуры	606
Создание обобщенного метода	607
Вызов обобщенного метода с явно указанными аргументами типа	609
Применение ограничений в обобщенных методах	610
Обобщенные делегаты	610

16 Содержание

Обобщенные интерфейсы	612
Сравнение экземпляров параметра типа	615
Иерархии обобщенных классов	620
Применение обобщенного базового класса	620
Обобщенный производный класс	622
Переопределение виртуальных методов в обобщенном классе	623
Перегрузка методов с несколькими параметрами типа	625
Ковариантность и контравариантность в параметрах обобщенного типа	626
Применение ковариантности в обобщенном интерфейсе	626
Применение контравариантности в обобщенном интерфейсе	630
Вариантные делегаты	633
Создание экземпляров объектов обобщенных типов	635
Некоторые ограничения, присущие обобщениям	636
Заключительные соображения относительно обобщений	636
Глава 19. LINQ	637
Основы LINQ	638
Простой запрос	639
Неоднократное выполнение запросов	641
Связь между типами данных в запросе	642
Общая форма запроса	643
Отбор запрашиваемых значений с помощью оператора where	644
Сортировка результатов запроса с помощью оператора orderby	646
Подробное рассмотрение оператора select	649
Применение вложенных операторов from	653
Группирование результатов с помощью оператора group	655
Продолжение запроса с помощью оператора into	657
Применение оператора let для создания временной переменной в запросе	659
Объединение двух последовательностей с помощью оператора join	660
Анонимные типы	663
Создание группового объединения	666
Методы запроса	669
Основные методы запроса	669
Формирование запросов с помощью методов запроса	670
Синтаксис запросов и методы запроса	673
Дополнительные методы расширения, связанные с запросами	673
Режимы выполнения запросов: отложенный и немедленный	675
Деревья выражений	676
Методы расширения	678
PLINQ	680

Глава 20. Небезопасный код, указатели, обнуляемые типы и разные ключевые слова	681
Небезопасный код	681
Основы применения указателей	682
Применение ключевого слова unsafe	684
Применение модификатора fixed	685
Доступ к членам структуры с помощью указателя	686
Арифметические операции над указателями	686
Сравнение указателей	688
Указатели и массивы	688
Указатели и строки	690
Многоуровневая непрямая адресация	691
Массивы указателей	692
Создание буферов фиксированного размера	693
Обнуляемые типы	695
Основы применения обнуляемых типов	695
Применение обнуляемых объектов в выражениях	697
Оператор ??	698
Обнуляемые объекты, операторы отношения и логические операторы	699
Частичные типы	700
Частичные методы	701
Создание объектов динамического типа	703
Возможность взаимодействия с моделью COM	707
Дружественные сборки	708
Разные ключевые слова	708
Ключевое слово lock	708
Ключевое слово readonly	709
Ключевые слова const и volatile	710
Оператор using	711
Ключевое слово extern	712
ЧАСТЬ II. БИБЛИОТЕКА C#	717
Глава 21. Пространство имен System	719
Члены пространства имен System	720
Класс Math	721
Структуры .NET, соответствующие встроенным типам значений	727
Структуры целочисленных типов данных	728
Структуры типов данных с плавающей точкой	730
Структура Decimal	735
Структура Char	741
Структура Boolean	748
Класс Array	750
Сортировка и поиск в массивах	763

18 Содержание

Обращение содержимого массива	766
Копирование массива	767
Применение предиката	768
Применение делегата Action	769
Класс BitConverter	771
Генерирование случайных чисел средствами класса Random	773
Управление памятью и класс GC	774
Класс object	776
Класс Tuple	777
Интерфейсы IComparable и IComparable<T>	778
Интерфейс IEquatable<T>	778
Интерфейс IConvertible	779
Интерфейс ICloneable	779
Интерфейсы IFormatProvider и IFormattable	781
Интерфейсы IObservable<T> и IObservable<T>	781
Глава 22. Строки и форматирование	783
Строки в C#	783
Класс String	784
Конструкторы класса String	784
Поле, индексатор и свойство класса String	785
Операторы класса String	786
Заполнение и обрезка строк	808
Вставка, удаление и замена строк	810
Смена регистра	811
Применение метода Substring()	811
Методы расширения класса String	812
Форматирование	812
Общее представление о форматировании	812
Спецификаторы формата числовых данных	814
Представление о номерах аргументов	815
Применение методов String.Format() и ToString() для форматирования данных	816
Применение метода String.Format() для форматирования значений	816
Применение метода ToString() для форматирования данных	819
Определение пользовательского формата числовых данных	820
Символы-заполнители специального формата числовых данных	820
Форматирование даты и времени	824
Определение пользовательского формата даты и времени	827
Форматирование промежутков времени	829
Форматирование перечислений	830
Глава 23. Многопоточное программирование. Часть первая: основы	833
Основы многопоточной обработки	834

Класс Thread	835
Создание и запуск потока	836
Простые способы усовершенствования многопоточной программы	838
Создание нескольких потоков	839
Определение момента окончания потока	841
Передача аргумента потоку	844
Свойство IsBackground	846
Приоритеты потоков	847
Синхронизация	849
Другой подход к синхронизации потоков	853
Класс Monitor и блокировка	855
Сообщение между потоками с помощью методов Wait(), Pulse() и PulseAll()	855
Пример использования методов Wait() и Pulse()	856
Взаимоблокировка и состояние гонки	860
Применение атрибутаMethodImplAttribute	860
Применение мьютекса и семафора	862
Мьютекс	863
Семафор	867
Применение событий	870
Класс Interlocked	873
Классы синхронизации, внедренные в версии .NET Framework 4.0	874
Прерывание потока	875
Другая форма метода Abort()	876
Отмена действия метода Abort()	878
Приостановка и возобновление потока	880
Определение состояния потока	880
Применение основного потока	880
Дополнительные средства многопоточной обработки, внедренные в версии .NET Framework 4.0	882
Рекомендации по многопоточному программированию	882
Запуск отдельной задачи	882
Глава 24. Многопоточное программирование. Часть вторая: библиотека TPL	885
Два подхода к параллельному программированию	886
Класс Task	887
Создание задачи	887
Применение идентификатора задачи	890
Применение методов ожидания	891
Вызов метода Dispose()	895
Применение класса TaskFactory для запуска задачи	895
Применение лямбда-выражения в качестве задачи	896
Создание продолжения задачи	897

Возврат значения из задачи	899
Отмена задачи и обработка исключения AggregateException	901
Другие средства организации задач	905
Класс Parallel	906
Распараллеливание задач методом Invoke()	906
Применение метода For()	909
Применение метода ForEach()	915
Исследование возможностей PLINQ	917
Класс ParallelEnumerable	917
Распараллеливание запроса методом AsParallel()	918
Применение метода AsOrdered()	919
Отмена параллельного запроса	920
Другие средства PLINQ	922
Вопросы эффективности PLINQ	922
Глава 25. Коллекции, перечислители и итераторы	923
Краткий обзор коллекций	924
Необобщенные коллекции	925
Интерфейсы необобщенных коллекций	926
Структура DictionaryEntry	931
Классы необобщенных коллекций	931
Хранение отдельных битов в классе коллекции BitArray	950
Специальные коллекции	953
Обобщенные коллекции	954
Интерфейсы обобщенных коллекций	954
Структура KeyValuePair<TKey, TValue>	960
Классы обобщенных коллекций	960
Параллельные коллекции	983
Сохранение объектов, определяемых пользователем классов, в коллекции	988
Реализация интерфейса IComparable	990
Реализация интерфейса IComparable для необобщенных коллекций	990
Реализация интерфейса IComparable для обобщенных коллекций	992
Применение интерфейса IComparer	994
Применение необобщенного интерфейса IComparer	994
Применение обобщенного интерфейса IComparer<T>	996
Применение класса StringComparison	997
Доступ к коллекции с помощью перечислителя	998
Применение обычного перечислителя	999
Применение перечислителя типа IDictionaryEnumerator	1000
Реализация интерфейсов IEnumerable и IEnumerator	1001
Применение итераторов	1003
Прерывание итератора	1005
Применение нескольких операторов yield	1006

Создание именованного итератора	1006
Создание обобщенного итератора	1008
Инициализаторы коллекций	1009
Глава 26. Сетевые средства подключения к Интернету	1011
Члены пространства имен System.Net	1012
Универсальные идентификаторы ресурсов	1013
Основы организации доступа к Интернету	1014
Класс WebRequest	1015
Класс WebResponse	1017
Классы HttpRequest и HttpResponse	1018
Первый простой пример	1018
Обработка сетевых ошибок	1021
Исключения, генерируемые методом Create()	1021
Исключения, генерируемые методом GetResponse()	1022
Исключения, генерируемые методом GetResponseStream()	1022
Обработка исключений	1022
Класс Uri	1024
Доступ к дополнительной информации, получаемой в ответ по протоколу HTTP	1025
Доступ к заголовку	1026
Доступ к cookie-наборам	1027
Применение свойства LastModified	1029
Практический пример создания программы MiniCrawler	1030
Применение класса WebClient	1034
Приложение. Краткий справочник по составлению документирующих комментариев	1039
Дескрипторы XML-комментариев	1039
Компилирование документирующих комментариев	1041
Пример составления документации в формате XML	1041
Предметный указатель	1044