

Введение

C# 2008 для “чайников” — это тщательно переработанное издание с большим количеством новой информации (в том числе новых примеров программ) и усовершенствованной организацией материала. Так что даже если у вас уже есть предыдущее издание, все равно имеет смысл внимательно прочесть настоящее.

Язык программирования C# является мощным и уже не совсем новым потомком более ранних языков C, C++ и Java. Программирование на нем доставляет много удовольствия, в чем можно убедиться, изучая эту книгу.

Язык C# был создан Microsoft как главная часть ее инициативы .NET. Возможно, из политических соображений компания Microsoft направила спецификации языка C# в Комитет международных стандартов ассоциации ECMA (произносится как “эк-ма”) летом 2000 года, задолго до внедрения платформы .NET. Теоретически любая компания может разработать собственную версию языка C#, предназначенную для работы в любой операционной системе и на любой машине, которая больше калькулятора.

Когда вышло первое издание настоящей книги, компилятор языка C# Microsoft был единственным, и ее инструментальный пакет Visual Studio .NET предлагал только одну возможность программирования на языке C#. С тех пор, однако, Visual Studio претерпел три существенных изменения. Последняя вышедшая версия — Visual Studio 2008. И по крайней мере еще два игрока вступили в игру C#.

В настоящее время можно писать и компилировать программы на языке C# на множестве машин, работающих под управлением Unix, при помощи реализаций Mono или Portable .NET платформы .NET и языка C#. Mono (www.go-mono.com) является программным проектом с открытым исходным кодом, финансируемым компанией Novell Corporation. Версия 1.2 вышла в ноябре 2006 года. Хотя проект Mono и отстает от платформы .NET компании Microsoft, он быстро развивается.

Реализация Portable .NET фирм Southern Storm Software и DotGNU (www.dotgnu.org/pnet.html) также является проектом с открытым исходным кодом. Во время написания этой книги текущей версией проекта Portable .NET была 0.7.0.

И Mono, и его менее разработанный двойник Portable .NET (www.dotgnu.org/pnet.htm) предназначены для выполнения программ C# в Windows и различных операционных системах семейства Unix, включая Linux и Macintosh компании Apple. Во время создания этой книги проект Portable .NET работал на большем количестве платформ, в то время как проект Mono гордится более полной реализацией платформы .NET. Так что выбор между ними может быть затруднен, в зависимости от вашего проекта, платформы и целей. (Книги по программированию для этих платформ уже есть в продаже. Посетите сайты книжных магазинов.)



Программное обеспечение с открытым исходным кодом создается начинающими группами программистов-добровольцев и обычно является бесплатным.

Переносимость языка C# и других языков платформы .NET выходит далеко за рамки настоящей книги. Но можно ожидать, что в ближайшие годы программы C# для Windows, которые можно научиться создавать по этой книге, будут работать на различном аппаратном обеспечении и для всех типов операционных систем, что соответствует требованиям для языка Java компании Sun Microsystems о работе на любой машине. Это, конечно же, хорошая вещь, даже для Microsoft. Переносимость — это вопрос, над которым в настоящее время идет интенсивная работа, так что, несомненно, все трудности и препятствия на пути к истинной универсальной переносимости языка C# будут преодолены. Но этот путь является уже не только путем Microsoft.

Сегодня пакет Visual Studio компании Microsoft содержит наиболее развитые версии языка C# и платформы .NET, а также набор инструментов к ним с богатыми возможностями для программирования.

Что нового в C# 3.0

В версию 3.0 языка C# был внесен ряд изменений. В этой книге рассматриваются следующие значительные нововведения.

- ✓ **Запросы, интегрированные в язык (Language Integrated Query — LINQ).** LINQ позволяет работать с данными, используя синтаксис C# (подобно тому, как целые поколения программистов работали с базами данных, используя SQL). LINQ не только упрощает код работы с базами данных, но и позволяет писать запросы к XML-файлам. Глава 17, “Выражения запросов”, представляет собой введение в LINQ; в ней рассматривается применение нового синтаксиса запросов для работы с произвольными коллекциями данных, включая массивы C# и коллекции.
- ✓ **Расширенные методы, анонимные типы и пр.** Большое количество языковых возможностей делают C# особенно удобным для работы с базами данных с применением LINQ.
- ✓ **HashSet.** В C# появился новый класс коллекции HashSet, о котором подробно рассказывается в главе 5, “Коллекционирование”.

Оставляя в стороне сложные нововведения, следует упомянуть о нескольких простых элементах в соответствующих разделах книги. (Не нужно беспокоиться, если сейчас вам что-то непонятно. Все станет ясно в процессе чтения.)

Кроме новейших возможностей C# 3.0, в данной книге описано множество более старых, которые, тем не менее, не были рассмотрены в предыдущем издании. Вы найдете здесь полномасштабное описание классов коллекций C#, дополнительную информацию о строках, интерфейсах и исключениях, а также массу советов и примечаний.

Об этой книге

Назначение этой книги — научить читателя языку программирования C#. Однако для реального написания программ необходима специальная среда кодирования. Мы уверены, что большинство читателей будут использовать Microsoft Visual Studio, хотя могут быть и другие варианты. Основывая книгу на Visual Studio, мы попытались сделать ее долю в книге минимально необходимой. Можно было бы просто сказать: “Запускайте

свою программу каким угодно образом”, но вместо этого мы говорим: “Запускайте свою программу C# в Visual Studio, нажав клавишу <F5>”. Мы хотим, чтобы вы сосредоточились на самом языке C#, а не на том, как работают такие простые вещи.

Вполне понятно, что многие читатели, если не большинство, захотят использовать C# для создания графических приложений Windows и приложений Web, поскольку язык C# является мощным средством разработки подобных программ, но это всего лишь одна из областей применения C#. В данной же книге, в первую очередь, обращается внимание на C# как на *язык программирования*. Если вас интересуют графические Windows-программы, зайдите на мой сайт `csharp102.info`. Но вы должны хорошо усвоить основы C#, прежде чем переходить к программированию для Windows.

Также вполне понятно, что некоторые опытные пользователи будут применять C# для создания сетевых и распределенных приложений, Web-приложений и приложений для работы с базами данных. Однако из-за издательских ограничений включить эти темы в данную книгу не представляется возможным. В книге *C# 2008 для “чайников”* распределенное программирование, программирование баз данных и другие подобные вопросы не рассматриваются, как не рассматриваются и новые технологии типа Windows Presentation Foundation, Windows Communication Foundation или Windows Workflow Foundation. В книге очень кратко рассмотрена платформа .NET — по той простой причине, что могущество языка C# во многом исходит из библиотек классов .NET Framework, которые используются этим языком.

Что необходимо для чтения этой книги

Для того чтобы просто запустить программы, сгенерированные C#, нужна как минимум *общезыковая исполняющая среда* (Common Language Runtime — CLR). Visual Studio 2008 копирует систему CLR на вашу машину во время процедуры установки. В качестве альтернативы можно загрузить весь пакет .NET, включая компилятор языка C# и множество других полезных инструментов, зайдя на Web-сайт компании Microsoft по адресу <http://msdn.microsoft.com>. Ищите там *набор инструментальных средств разработки программного обеспечения .NET* (Software Development Toolkit — SDK).



Если все, что вам надо, — это C#, можете взять бесплатную версию Visual Studio под названием Visual C# 2008 Express по адресу msdn.microsoft.com/vstudio/express. Версия Express включает новые возможности C# 3.0. В качестве альтернативного решения можно рассмотреть SharpDevelop (www.icsharpcode.net) — бесплатный аналог Visual Studio, который я разместил на сайте настоящей книги. (На момент написания этой книги в SharpDevelop еще не были реализованы все новые возможности C# 3.0/.NET 3.5, но работа в этом направлении интенсивно велась.)



Большинство приведенных здесь программ можно при необходимости создавать и в более ранних средах Visual Studio, например в Visual Studio 2005. Исключениями являются программы, которые содержат новые возможности, доступные только в языке C# 3.0 и перечисленные выше, в разделе “Что нового в C# 3.0”.

Как использовать эту книгу

Создавая настоящую книгу, авторы преследовали цель сделать ее максимально простой в использовании, поскольку изучение нового языка — дело и так достаточно сложное. Зачем же его еще больше усложнять? Книга разделена на части. В части I представлено введение в программирование на C# с использованием Visual Studio. В ней пошагово излагается создание программ на C#. Авторы настоятельно рекомендуют прочесть эту часть, прежде чем двигаться дальше. Даже если у вас уже есть опыт программирования, примите к сведению, что базовая структура программы, созданная в части I, применяется во всей книге.

Главы, приведенные в частях II–V, являются самостоятельными. Они написаны так, чтобы можно было открыть книгу на любой из них и начать чтение. Если вы новичок в программировании или в программировании на C-образных языках, прочтите полностью часть II. Но если вы просто возвращаетесь назад, чтобы освежить в памяти конкретную тему, у вас не возникнет никаких проблем при переходе от одного раздела к другому и вам не нужно будет повторно перечитывать предыдущие 20 страниц.

И, конечно же, книгу завершают традиционные “Великолепные десятки” и дополнительные главы. Много интересного можно найти и на моем сайте по адресу `csharp102.info`.

Структура книги

Ниже приведено краткое содержание каждой части книги.

Часть I. Начнем, пожалуй...

В этой части шаг за шагом рассматривается написание минимальной консольной программы C#, которая используется в других частях книги. Информация о графическом программировании приведена на моем сайте.

Часть II. Основы программирования в C#

На базовом уровне пьесы Шекспира — это всего лишь набор слов, связанных вместе. С этой же точки зрения 90% любой программы C#, которую вы когда-либо напишете, состоят из создания переменных, выполнения арифметических действий и управления ходом выполнения программы. В части II внимание уделяется этим основным операциям. В нее также включен материал о классах коллекций C#, в том числе о новом классе `HashSet`.

Часть III. Использование объектов

Одно дело — объявлять переменные где-либо в программе, добавлять и убирать их и совсем другое — создавать реальные программы для реальных людей. В части III внимание уделяется такой организации данных, чтобы их было легко использовать при создании программы.

Часть IV. Объектно-ориентированное программирование

Вы можете соединять части самолета так, как пожелаете, но пока вы не сложите их правильно, у вас не будет ничего, кроме кучи деталей. И только когда вы соберете самолет так, что сможете запустить двигатели и использовать подъемную силу крыла, вы сможете лететь куда угодно.

В части IV объясняется, как превратить набор данных в реальный объект — объект, который имеет внутренние члены и может моделировать свойства реальных вещей. Здесь представлена сущность объектно-ориентированного программирования, и даже совершенный новичок в объектно-ориентированном программировании сможет понять, что же это такое.

Часть V. C# 3.0

После того как самолет оторвется от земли, он должен будет куда-нибудь лететь. В этой части вы бесстрашно летите на встречу с новыми возможностями C# 3.0. В частности, вы встретитесь с запросами, интегрированными в язык (Language Integrated Query — LINQ). Объекты и коллекции объектов, с которыми вы познакомились в предыдущих частях, пользуясь метафорой полета, обретают новые крылья с запросами LINQ. Ну а чтобы увеличить скорость полета, в книгу включены новые главы о делегатах и событиях.

Часть VI. Великолепные десятки

Язык C# силен в поиске ошибок в ваших программах — иногда кажется, что он даже слишком хорошо указывает на недостатки. Однако верите вы в это или нет, C# все же пытается принести вам пользу. Каждая обнаруженная им проблема могла бы привести к другим проблемам, которые вам пришлось бы находить и локализовывать самостоятельно.

К сожалению, сообщения об ошибках могут сбивать с толку. В этой части представлено десять наиболее общих сообщений об ошибках C# времени компиляции, их значение и способы исправления этих ошибок.

О Web-сайте

На Web-сайте данной книги содержится масса интересной информации. Прежде всего, на нем можно найти все исходные тексты этой книги. Кроме того, там же имеется набор полезных утилит. Утилита SharpDevelop не рекомендуется для полномасштабной разработки коммерческих программ, но весьма полезна для написания небольших приложений или быстрого внесения изменений, чтобы не ожидать загрузки Visual Studio. Она полностью подходит для компиляции всех исходных текстов данной книги (за исключением, может быть, новых возможностей LINQ). Инструмент Reflector позволяет посмотреть, во что превращает компилятор ваш исходный текст C#. Инструмент тестирования NUnit, очень популярный среди программистов на C#, выполняет проверку вашего кода проще, чем из Visual Studio или SharpDevelop.

И, конечно, не забудьте о файле `ReadMe`, содержащем всю наиболее свежую информацию.

Пиктограммы, используемые в этой книге

В данной книге используются следующие пиктограммы для выделения важной информации.



Этой пиктограммой помечены технические сведения, которые можно пропустить при первом чтении.



Данной пиктограммой выделены советы, которые могут сохранить много вашего времени и усилий.



Это необходимо запомнить, так как это важно.



Это также следует запомнить, иначе проблема настигнет вас тогда, когда вы меньше всего ее ожидаете, и создаст одну из действительно трудно локализуемых ошибок.



Данной пиктограммой помечен код, который можно найти на сайте книги. Эта возможность предназначена для того, чтобы избавить вас от лишнего набора исходного текста, если ваши пальцы начали дрожать. Но не злоупотребляйте ею — вы лучше поймете C#, если будете набирать код программ самостоятельно.

Соглашения, используемые в этой книге

В книге принято несколько соглашений. Термины, которые не являются общепотребляемыми словами, например имена переменных, напечатаны таким шрифтом. Листинги программ выделены следующим образом:

```
use System;
namespace MyNameSpace
{
    public class MyClass
    {
    }
}
```

Каждый листинг сопровождается ясным и понятным комментарием. В отличие от небольших фрагментов, полные исходные тексты программ содержатся на сайте книги.

Наконец, здесь вы увидите стрелки, как, например, во фразе “Выберите команду меню File⇒Open With⇒Notepad”. Это означает, что сначала необходимо выбрать меню File, затем из появившегося раскрывающегося меню выбрать Open With, и наконец из следующего подменю выбрать Notepad.

Web-сайт

В отличие от предыдущих изданий книги на этот раз все исходные тексты примеров, как и дополнительные главы, содержатся не на прилагаемом компакт-диске, а в Web. Код и дополнительные главы представлены на двух следующих сайтах.

- ✓ www.dummies.com
Здесь, кроме кодов и глав данной книги, можно найти материал из других книг серии ...для “чайников”, на которые имеются ссылки в этой книге.
- ✓ csharp102.info (мой личный сайт)
Здесь представлен тот же материал и множество разных статей на темы, которые я хотел, но не смог включить в книгу. Вы найдете также массу ссылок на инструменты для программистов и прочие ресурсы C#.

И, конечно, не забудьте о файле ReadMe (он есть на обоих сайтах), содержащем всю наиболее свежую информацию.

Что дальше

Очевидно, что первым шагом должно быть изучение языка C# (в идеальном случае — с помощью книги *C# 2008 для “чайников”*, конечно). Вы можете потратить несколько месяцев на написание простых программ C#, прежде чем сделаете следующий шаг — освоите создание приложений Windows. Вам придется потратить еще много месяцев на приложения Windows, прежде чем вы начнете создавать программы, предназначенные для распространения через Интернет.

Поддерживать знания языка C# можно несколькими способами. Прежде всего, обратитесь к официальному источнику — <http://msdn.microsoft.com/msdn>. Кроме того, на различных Web-сайтах для программистов имеется обширный материал по языку C#, включая обсуждения разных вопросов — от способов сохранения исходных файлов до сравнения свойств детерминистической и недетерминистической сборки мусора. Ниже приведено несколько больших сайтов, посвященных C#.

- ✓ msdn.microsoft.com/vcsharp (начальная страница C#, содержащая ресурсы C# и .NET различного рода)
- ✓ blogs.msdn.com/csharpfaq (блог часто задаваемых вопросов по C#)
- ✓ msdn.microsoft.com/vcsharp/team/blogs (включает персональные блоги членов команды разработчиков C#)
- ✓ www.c-sharpcorner.com и www.codeproject.com (основные сайты, посвященные C#, со статьями, блогами, кодом, информацией о вакансиях и прочими ресурсами, связанными с C#)

Я также поддерживаю собственный Web-сайт csharp102.info, содержащий ряд ответов на часто задаваемые вопросы (FAQ). Если вы столкнетесь с чем-то, чего не сможете понять, посетите его — может быть, в FAQ уже есть ответ на ваш вопрос. Кроме того, на сайте содержится список ошибок, которые могли пробраться в книгу. И наконец — имеется в виду действительно крайний случай, — вы можете найти ссылку на мой адрес электронной почты и написать мне.