

# C#

## СОВЕТЫ ПРОГРАММИСТАМ



Использование стандартных алгоритмов

Использование особенностей интегрированной среды разработки IDE

Работа с формами, элементами управления

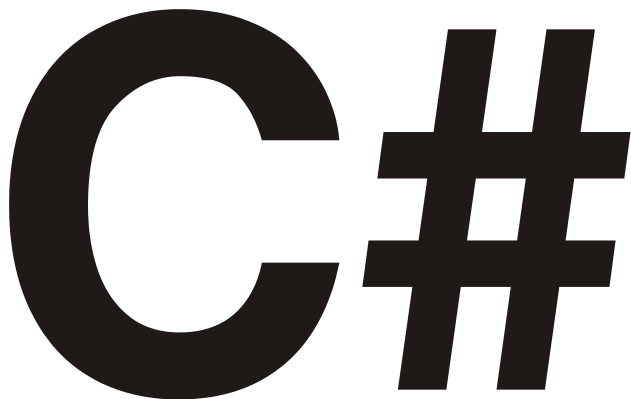
Использование технологий WMI, WSH

Вызов системных функций Windows API



АЛЕКСАНДР КЛИМОВ

**Александр Климов**



**СОВЕТЫ  
ПРОГРАММИСТАМ**

Санкт-Петербург

«БХВ-Петербург»

2008

УДК 681.3.068+800.92С#  
ББК 32.973.26-018.1  
К49

**Климов А. П.**

К49 С#. Советы программистам. — СПб.: БХВ-Петербург, 2008. — 544 с.: ил. + CD-ROM

ISBN 978-5-9775-0174-3

Книга представляет собой сборник советов, алгоритмов и готовых примеров программ на языке С# в среде MS Visual Studio 2005/2008 из различных областей: работа с формами и элементами управления, папками и файлами, мышью и клавиатурой, мультимедиа и графикой, использование технологий WMI и WSH, взаимодействие с MS Office и другими приложениями, работа в локальной сети и Интернете, особенности использования функций Windows API и др.

На компакт-диске размещены примеры из книги, а также демонстрационная версия справочника по функциям Windows API для .NET Framework и общения Windows для Visual Basic .NET и С#.

*Для программистов*

УДК 681.3.068+800.92С#  
ББК 32.973.26-018.1

**Группа подготовки издания:**

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Анна Кузьмина</i>
Компьютерная верстка	<i>Наталья Смирновой</i>
Корректор	<i>Наталья Першакова</i>
Дизайн обложки	<i>Елены Беляевой</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 26.02.08.

Формат 70×100<sup>1/16</sup>. Печать офсетная. Усл. печ. л. 43,86.

Тираж 2500 экз. Заказ №

"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию № 77.99.60.953.Д.002108.02.07 от 28.02.2007 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов  
в ГУП "Типография "Наука"  
199034, Санкт-Петербург, 9 линия, 12

ISBN 978-5-9775-0174-3

© Климов А. П., 2008  
© Оформление, издательство "БХВ-Петербург", 2008

# Оглавление

<b>ВСТУПЛЕНИЕ .....</b>	<b>1</b>
Для кого предназначена книга.....	2
Благодарности .....	2
Требования .....	2
Чего вы не найдете в этой книге .....	3
Исходные коды .....	4
Обратная связь .....	4
<b>ГЛАВА 1. ОБЩЕЕ .....</b>	<b>5</b>
Часто задаваемые вопросы .....	5
Можно ли запустить программу, написанную на C#, без .NET Framework? ...	5
В каком редакторе писать программы? .....	5
Использование в качестве переменных русских символов .....	7
Псевдонимы.....	7
Копирующий строковой литерал .....	8
Символ @ перед идентификатором .....	8
Как узнать, присвоено ли переменной значение .....	9
Как это назвать?.....	10
Какая разница между <i>string</i> и <i>System.String</i> ?.....	10
Выберите свои правила наименования.....	11
Правила для названий классов и методов .....	11
Советы по созданию эффективных и масштабируемых приложений.....	11
Сопряжение .....	11
Наследование .....	12
Минимизация кода.....	12
Экономия ресурсов .....	12
Создание автоматически обновляемых приложений .....	13
Заключение .....	13
<b>ГЛАВА 2. СТРОКИ, ДАТЫ, ЧИСЛА .....</b>	<b>15</b>
Строки.....	15
Простейшие операции со строками .....	15

Входит ли строка в другую строку? .....	16
Преобразование строки в число .....	17
Вставка специального символа .....	17
Создание строки из повторяющихся символов .....	19
Метод <i>String.Format</i> .....	19
Преобразование строки в объект <i>Color</i> .....	19
Проверка строки на пустоту .....	20
Переворачиваем строку .....	21
Сжатие длинных имен файлов .....	22
Печатающийся текст .....	23
Бегущая строка .....	24
Как соединять строки .....	25
Что лучше: <i>Parse</i> или <i>TryParse</i> ? .....	26
Сравнение и сортировка строк .....	27
Даты .....	28
Как получить текущую дату .....	28
Дата и время в разных форматах .....	29
Как использовать дату и время в приложении .....	31
Сложить и вычесть временной интервал из дат .....	32
Вычисление разницы между датами .....	33
Как определить, является ли год високосным? .....	34
Вычисление даты католической Пасхи .....	35
Числа .....	36
Преобразование числа в шестнадцатеричную систему счисления .....	36
Как перевести число в двоичную систему счисления? .....	37
Как перевести число в восьмеричное или шестнадцатеричное представление? .....	37
Является ли выражение числом? .....	37
Создание собственной функции <i>IsNumeric</i> на C# .....	38
Создание уникального идентификатора .....	39
Перечисления .....	40
Как получить все элементы перечисления .....	40
Заключение .....	42
<b>ГЛАВА 3. АЛГОРИТМЫ .....</b>	<b>43</b>
Найти наименьшее и наибольшее значение из трех чисел .....	43
Массив строк .....	44
Преобразование градусов в радианы и радианов в градусы .....	45
Четное или нечетное число .....	46
Получить старшее и младшее слова из числа .....	47

Преобразование градусов по Фаренгейту в градусы по Цельсию .....	48
Генерирование случайного цвета .....	49
Подсчет суммы всех целых чисел диапазона .....	49
Нахождение простых чисел .....	50
Вывод программой своего исходного кода .....	53
Заключение .....	54

## **ГЛАВА 4. ИНТЕГРИРОВАННАЯ СРЕДА РАЗРАБОТКИ (IDE) .....**

Удобные клавиатурные команды .....	55
Получение списка всех назначенных клавиш .....	56
Настройка назначенных клавиш .....	59
Показ назначенных клавиш во всплывающих подсказках .....	62
Селектор оконных конфигураций .....	62
Фрагменты кода (code snippets) .....	65
Создание XML-файла .....	68
Встроенные фрагменты кода .....	72
Распространение собственных фрагментов кода .....	72
Настройка стартовой страницы Visual Studio .....	73
Настройки для групповой работы .....	73
Создание файла параметров .....	74
Размещение файла параметров в пути UNC .....	74
Параметр <i>/resetuserdata</i> .....	75
Ряд мелких советов .....	75
Как показывать нумерацию строк в редакторе кода? .....	76
Как изменить цвет для регионов кода? .....	76
Вертикальное выделение текста .....	76
Альтернативный метод поиска строк .....	76
Множественное копирование в буфер обмена .....	78
Как управлять фрагментами кода в Visual Studio 2008? .....	78
Быстрое комментирование и раскомментирование фрагментов кода .....	78
Отображение IntelliSense .....	78
Прозрачная подсказка IntelliSense .....	79
Перемещение от открывающей скобки к закрывающей скобке .....	79
Сворачивание/разворачивание блока (региона, функции, цикла и т. п.) .....	79
Анимация при автоматическом скрывании панелей .....	79
Вариант загрузки справочной системы .....	80
Путь к файлу .....	80
Быстрый переход к папке, содержащей исходные коды проекта .....	80
Изменение шаблона заготовки метода в C#, генерируемого автоматически .....	81

Вспомнить название пространства имен .....	82
Удобный способ вызвать Smart Tag .....	82
Создание собственных шаблонов приложений .....	83
Работа в полноэкранном режиме .....	84
Быстрый поиск в списках .....	84
Поле <i>Find</i> .....	84
Окно <i>Command</i> .....	85
Диалоговое окно <i>Find and Replace</i> .....	86
Еще о настройках .....	88
Соккрытие статусной строки .....	88
Число показываемых последних файлов .....	88
Многодокументный интерфейс .....	88
Управление панелями Auto Hide и Close .....	88
Меню <i>Window</i> .....	89
Переключение между окнами .....	89
Помоги себе и команде Visual Studio, или пишем логи .....	89
Графические файлы для проектов .....	89
Надстройки .....	90
Надстройки сторонних разработчиков .....	90
GhostDoc .....	90
SmartPaster .....	92
PInvoke.NET .....	93
Paste as Visual Basic .....	95
Заключение .....	95

## **ГЛАВА 5. ЭКРАН И ФОРМЫ.....97**

Экран .....	99
Как определить разрешение экрана .....	99
Как определить рабочую область экрана без панели задач? .....	99
Как изменить разрешение экрана программным путем .....	100
Формы .....	103
Как вывести форму в центре экрана? .....	103
Как задать позицию формы на экране? .....	103
Как программно свернуть или развернуть форму? .....	104
Поддержка тем рабочего стола Windows .....	104
Как узнать, используются ли темы Windows XP? .....	105
Как отобразить форму без передачи ей фокуса? .....	105
Как не отображать форму при запуске программы? .....	106
Как сделать так, чтобы форма отбрасывала тень? .....	107
Как вывести запрос при закрытии формы? .....	108

Выбираем варианты закрытия формы .....	109
Скрытие значка формы на панели задач и при нажатии комбинации клавиш <Alt>+<Tab> .....	110
Как отобразить форму на весь экран?.....	111
Как установить ограничение на минимальный и максимальный размер окна? .....	111
Как отловить момент сворачивания или разворачивания формы?.....	112
Как запретить пользователю перемещать форму по экрану?.....	113
Как перемещать форму, не имеющую заголовка?.....	114
Еще два способа буксировки формы, не имеющей заголовка .....	116
Как добиться эффекта полупрозрачности у формы.....	118
Перемещение формы за заголовок.....	119
Неактивная форма.....	120
Как создать формы без границ и заголовка?.....	121
Как убрать кнопку X из заголовка формы? .....	121
Убрать кнопку X при помощи управляемого кода.....	123
Создать окно произвольной формы .....	123
Создание дырявой формы .....	125
Как создать форму в виде текста?.....	127
Смена темы Windows XP .....	128
Как форме получать уведомления о нажатии кнопок, когда фокус ввода находится в каком-либо элементе управления формы? .....	129
Как получить список всех открытых форм, принадлежащих приложению?.....	130
Сохранение настроек формы .....	131
Создание и использование параметров командной строки .....	131
Установить фоновый цвет в родительской MDI-форме .....	134
Запрет на запуск второй копии приложения.....	135
Как передавать значения между формами Windows Forms.....	136
Способ первый .....	136
Второй способ .....	137
Заключение.....	138

## **ГЛАВА 6. ЭЛЕМЕНТЫ УПРАВЛЕНИЯ.....139**

Общие советы.....	139
Как добавить элемент управления на форму во время выполнения программы? .....	139
Как пройти по всем элементам управления на форме?.....	140
Как изменить цвет границы ( <i>Border</i> ) у элемента управления? .....	141
Окантовка вокруг элемента управления.....	142



Как программно перевести фокус на следующий/предыдущий (в порядке TAB) элемент управления? .....	143
Как изменить Z-порядок элемента управления? .....	144
Как узнать размеры строки в пикселах, отображаемой в каком-нибудь элементе управления? .....	145
Как сделать элемент управления произвольной формы? .....	146
Кнопки ( <i>Button</i> ) .....	148
Как установить кнопку по умолчанию для формы? .....	148
Как установить кнопку отмены ( <i>Cancel</i> ) для формы? .....	148
Как программно вызвать событие <i>Click</i> у кнопки? .....	148
Как создать западающую кнопку? .....	149
Список ( <i>ListBox</i> ) .....	149
Автоматическая прокрутка списка .....	149
Подгоняем ширину списка под самый длинный текст .....	150
Как заполнить список именами файлов, перетаскиваемых из Проводника? .....	150
Разделить список цветными линиями и заполнить цветным текстом .....	151
Поле со списком ( <i>ComboBox</i> ) .....	153
Подгоняем ширину поля со списком под самый длинный текст .....	153
Поддержка автозавершения .....	153
Как раскрыть поле со списком программным способом? .....	154
Как запретить раскрытие списка? .....	155
Как изменить высоту элементов списка у элемента управления <i>ComboBox</i> ? .....	156
Как установить желаемую высоту выпадающего списка у <i>ComboBox</i> ? .....	157
Как использовать <i>ComboBox</i> для редактирования данных в <i>Listview</i> ? .....	157
Текстовые поля ( <i>TextBox</i> ) .....	158
Подсчет числа строк в многострочном текстовом поле .....	158
Фильтрация заданных символов при вводе с клавиатуры .....	159
Как заблокировать контекстное меню в текстовом поле? .....	160
Запрет вставки текста из буфера обмена Windows .....	160
Как ввести многострочный текст в текстовое поле программно? .....	161
Как сделать так, чтобы символы вводились в нужном регистре? .....	162
Как избавиться от звукового сигнала при нажатии на клавишу ввода? .....	162
Как выделить текст программным способом? .....	163
Элемент <i>RichTextBox</i> .....	164
Просмотр форматированного текста RTF .....	164
Как управлять цветом и шрифтами в <i>RichTextBox</i> ? .....	165
Как управлять текстом-гиперссылкой в <i>RichTextBox</i> ? .....	166
Поддержка Drag'n'Drop .....	166

Как определить наличие полос прокрутки в элементе <i>RichTextBox</i> ? .....	167
Как запретить вставку .....	168
Элемент управления <i>MaskedTextBox</i> .....	169
Элемент <i>DateTimePicker</i> .....	169
Как показать пустой текст, если в <i>DateTimePicker</i> не выбрана дата? .....	169
Как программно раскрыть <i>DateTimePicker</i> ? .....	170
Элементы <i>Label</i> и <i>Panel</i> .....	170
Полупрозрачная надпись.....	170
Использование <i>Label</i> в виде разделительной линии как элемент дизайна .....	171
Элемент <i>LinkLabel</i> .....	172
Отображение лишь части текста в виде ссылки .....	173
Несколько ссылок в одном <i>LinkLabel</i> .....	174
<i>NotifyIcon</i> — значок в области уведомлений .....	175
Как создать мигающий значок в области уведомлений? .....	176
Как создать анимированный значок в области уведомлений? .....	177
Свертывание формы вместо закрытия приложения .....	178
Элемент <i>ListView</i> .....	179
Как убрать выделение элемента в <i>ListView</i> программно?.....	179
Как программно выбрать элемент в <i>ListView</i> ? .....	179
Как сортировать элемент управления <i>ListView</i> по колонкам .....	180
Изменение цвета подэлементов <i>ListView</i> программным путем.....	185
Элемент управления <i>ToolTip</i> .....	186
Почему пользователь не видит подсказки в стиле <i>Balloon</i> ?.....	186
Многострочная подсказка .....	187
Меню .....	187
Фон для меню .....	188
Как добавить контекстное меню элементу управления? .....	188
Как определить, какой элемент вызвал контекстное меню? .....	188
Автоматическое закрытие контекстного меню через заданный промежуток времени .....	189
Дерево ( <i>TreeView</i> ) .....	190
Как показать подсказку над узлом <i>TreeView</i> ? .....	190
Вкладки ( <i>TabControl</i> ).....	192
Программное переключение на другую вкладку.....	192
Установка фокуса на элементе управления на вкладке во время загрузки формы .....	192
Как вывести ярлычки внизу вкладки <i>TabControl</i> ? .....	193
Добавление новой вкладки .....	193
Удаление вкладки .....	193
Как вставить вкладку в определенную позицию? .....	194

Элемент <i>PerformanceCounter</i> .....	195
Как создать счетчик производительности процессора? .....	195
<i>StatusBar</i> и <i>StatusStrip</i> .....	197
Как изменить шрифт и фон для <i>StatusBar</i> .....	197
Элементы <i>FlowLayoutPanel</i> и <i>TableLayoutPanel</i> .....	198
Элемент <i>DataGrid</i> .....	198
Элемент <i>DataGridView</i> .....	199
Создание собственных элементов управления .....	200
Как скрыть свойство или метод от IntelliSense в редакторе кода? .....	200
Как скрыть свойства и события из редактора свойств <i>PropertyGrid</i> при создании собственного элемента управления? .....	201
Как запретить изменять размер элемента управления во время разработки? .....	202
Как во время разработки позволить выбирать значение свойства из нескольких predefined в поле со списком? .....	203
Как добиться того, чтобы свойство моего элемента управления было видно в разделе <i>DataBindings</i> окна свойств? .....	204
Как сделать свой элемент управления, выступающий в роли контейнера для других элементов управления во время разработки? .....	204
Как присвоить свой значок для собственного элемента управления в панели инструментов? .....	205
Создание собственного элемента управления <i>SmoothProgressBar</i> .....	206
Создание элемента <i>SmoothProgressBar</i> .....	208
Создание клиентской программы для тестирования .....	213
Заключение .....	215

## **ГЛАВА 7. ГРАФИКА .....**

Преобразование цвета в HTML-формат .....	217
Как преобразовать цвет в целое число? .....	218
Как получить доступ к определенному пикселу изображения? .....	218
Как нарисовать точку? .....	218
Как получить цвет любой выбранной точки экрана? .....	219
Как нарисовать прямоугольник с закругленными краями? .....	222
Установка фонового изображения .....	223
Как сделать снимок экрана? .....	225
Сохранить изображение элемента управления или формы .....	225
Как получить прокручиваемый рисунок? .....	226
Получение негатива изображения .....	226
Сделать изображение серым .....	227
Как создать затемненную картинку .....	228

Эффект недоступной кнопки .....	229
Как нарисовать вдавленный и выпуклый текст? .....	230
Как получить контурный текст.....	231
Как отразить текст в зеркальном отражении? .....	232
Как повернуть текст под некоторым углом?.....	233
Вот новый поворот (из песни группы "Машина времени").....	234
Бегущая градиентная строка.....	235
Скроллинг текста .....	236
Анимированные картинки .....	238
Как сохранить изображение из буфера обмена в файл .....	239
Шрифты и печать.....	240
Получение списка установленных шрифтов.....	240
Использование собственных шрифтов .....	242
Получение списка установленных принтеров.....	243
Как распечатать документ? .....	244
Как показать окно предварительного просмотра перед печатью .....	246
Заключение .....	247

## **ГЛАВА 8. РАБОТА С МЫШЬЮ И КЛАВИАТУРОЙ .....**

Мышь .....	249
Как скрыть и показать указатель мыши?.....	249
Как установить позицию указателя мыши? .....	250
Анимированные курсоры.....	251
Мышеловка.....	251
Право выбора .....	252
Меняем кнопки мыши местами .....	254
Как узнать координаты мыши? .....	255
Как преобразовать экранные координаты в клиентские (для данного элемента) и наоборот? .....	255
Как двигать указателем мыши программно? .....	256
Как выполнить эмуляцию щелчков мыши? .....	258
Рисование.....	259
Работа с клавиатурой.....	262
Как переключать раскладки клавиатуры? .....	262
Как получить текущий язык ввода? .....	263
Как послать нажатия клавиш программно? .....	263
Как включать и выключать индикаторы клавиш <Caps Lock>, <Num Lock> и <Scroll Lock>?.....	264
Как определить состояние клавиш-индикаторов?.....	266
Последнее нажатие на клавишу или на кнопку мыши.....	267
Заключение .....	269

<b>ГЛАВА 9. ПРИЛОЖЕНИЯ .....</b>	<b>271</b>
Работа с процессами .....	271
Как получить полное имя файла запущенного приложения? .....	271
Как получить путь к папке, из которой запущено приложение? .....	272
Как запустить другой исполняемый файл из своего приложения? .....	272
Как закрыть все копии Блокнота? .....	274
Запуск программы по имени файла .....	275
Как узнать число процессоров в системе? .....	276
Как приостановить выполнение программы на несколько секунд? .....	277
Как получить список всех процессов, запущенных в системе? .....	277
Как получить список только оконных процессов на моей машине? .....	278
Как получить список определенных процессов? .....	279
Получение списка процессов на удаленной машине .....	279
Как открыть почтовый клиент, установленный по умолчанию, и установить необходимые параметры для отправки письма? .....	280
Определение операционной системы пользователя .....	281
Определение версии .NET Framework и ее сервис-пака .....	283
.NET Framework 1.0 .....	284
.NET Framework версий 1.1, 2.0 3.0 и v3.5 (Orcas) .....	284
.NET Framework 3.0 .....	285
Определение папки установки .NET Framework .....	294
Номер сборки .....	295
Обновление номера версии сборки в автоматическом режиме .....	295
Вызов файла справки CHM .....	296
Получение номера версии файла и другую информацию .....	296
Определение имени пользователя системы .....	298
Как определить, имеет ли ваша система мышшь, узнать число кнопок у мыши, размер вашего монитора и другую информацию? .....	298
Как зарегистрировать файлы DLL и OCX? .....	300
Извлечение строки или значка из ресурсов .....	301
Сохранение настроек приложения .....	302
Работа с реестром .....	304
Определение архитектуры операционной системы .....	306
Добавление программы в автозагрузку .....	306
Получение информации об изменениях в системе .....	308
Как узнать, что пользователь изменил разрешение экрана? .....	308
Изменение времени .....	309
Консольные приложения .....	310
Журналы событий .....	311
Как найти доступные журналы событий на компьютере? .....	311

Чтение и запись логов в журнал событий .....	311
Запись в журнал .....	312
Очистка записей в журнале событий .....	313
Создание собственного журнала событий.....	313
Удаление собственного журнала событий .....	314
Измерение времени выполнения кода в приложении.....	314
Измерение с помощью функций Windows API.....	315
Измерение с помощью метода <i>Ticks</i> .....	316
Измерение с помощью <i>TickCount</i> .....	316
Класс <i>StopWatch</i> .....	317
Заключение .....	318

## **ГЛАВА 10. ДИСКИ, ПАПКИ И ФАЙЛЫ .....**

Диски.....	319
Как получить список логических дисков? .....	319
Как узнать тип диска и его свойства? .....	320
Папки .....	321
Как получить список папок?.....	321
Как проверить существование папки?.....	322
Как переименовать папку?.....	322
Как удалять папки?.....	323
Как выбрать папку? .....	324
Как получить путь для папки Мои документы и других специальных папок Windows? .....	324
Свойства папки .....	326
Размеры папки.....	327
Как написать свой Проводник? .....	328
Файлы.....	330
Как получить список файлов в папке?.....	331
Как получить список папок и файлов? .....	331
Как получить список файлов по маске? .....	332
Как узнать, существует ли файл? .....	332
Как получить имя файла из полного пути файла?.....	332
Как получить расширение файла из полного пути? .....	333
Как создать, удалить, переместить файл? .....	333
Как установить атрибуты у файла? .....	334
Свойства файла .....	334
Как извлечь информацию о файле? .....	335
Как создать временный файл? .....	336
Как создать уникальное имя для файла? .....	336

Как ограничить доступ к файлу? .....	337
Как работать с бинарными файлами? .....	338
Как работать с текстовыми файлами? .....	339
Как добавить текст в существующий файл? .....	340
Построчное чтение текстового файла .....	340
Загрузить текстовый файл в список? .....	341
Как получить короткое имя файла из длинного файла и наоборот? .....	343
Как удалить файл в Корзину .....	344
Как записать и прочитать текст в различных кодировках? .....	346
Как прочитать XML-файлы? .....	347
Сравнение двух файлов .....	351
Отслеживание изменений в файловой системе .....	353
Как установить уровень доступа к файлу? .....	355
Заключение .....	356

## **ГЛАВА 11. БИБЛИОТЕКА WSH .....**

**357**

Создание ярлыка .....	357
Получение списка установленных в системе принтеров .....	358
Установка принтера по умолчанию .....	359
Получение списка сетевых дисков .....	359
Заключение .....	360

## **ГЛАВА 12. WMI .....**

**361**

Использование WMI на удаленной машине .....	361
Информация об операционной системе .....	362
Информация о компьютере .....	364
Информация о производителе .....	365
Получение информации о процессорах .....	366
Информация о свойствах видеоконтроллера .....	369
Получение свойств приводов компакт-дисков .....	370
Информация о параметрах загрузки Windows .....	371
Информация о сетевом адаптере .....	372
Информация о мониторе .....	373
Материнская плата .....	373
Вывод списка общих ресурсов .....	374
Информация о логических дисках .....	375
Перезагрузка компьютера .....	376
Дополнительный пример .....	377
Заключение .....	378

<b>ГЛАВА 13. МУЛЬТИМЕДИА .....</b>	<b>379</b>
Звуковые сигналы .....	379
Функция Windows API <i>Beep</i> .....	379
Функция Windows API <i>MessageBeep</i> .....	379
Функция <i>Beep</i> для Visual Basic .....	380
Звуковые файлы .....	380
И снова о <i>Beep</i> .....	381
Как проигрывать звуки разных форматов? .....	382
Воспроизведение MIDI и MP3 через неуправляемый код.....	382
Извлечение информации из файлов MP3.....	383
Взаимодействие с Winamp.....	392
Заключение.....	393
<b>ГЛАВА 14. РАЗРАБОТКА ЛОКАЛИЗОВАННЫХ ПРИЛОЖЕНИЙ.....</b>	<b>395</b>
Общая информация о локализации.....	397
Локализирующие идентификаторы .....	398
Культура.....	399
Приложение Culture Explorer.....	400
Разработка многоязычного приложения .....	403
Разделяй и властвуй.....	405
Сопутствующие сборки.....	405
Заключение.....	406
<b>ГЛАВА 15. MICROSOFT OFFICE.....</b>	<b>407</b>
Excel .....	407
Раннее связывание .....	407
Автоматизация Excel или как работать массивами .....	409
Позднее связывание.....	414
Outlook .....	417
Как получить сообщения из папки Входящие? .....	417
Получение уведомлений о новых письмах.....	419
VSTO.....	420
Заключение.....	421
<b>ГЛАВА 16. ЛОКАЛЬНАЯ СЕТЬ И ИНТЕРНЕТ.....</b>	<b>423</b>
Информация о сети.....	423
Как получить хост, порт, протокол из веб-адреса? .....	423
Как получить IP-адрес компьютера, используя DNS? .....	424



Как получить NETBIOS-имя машины? .....	425
Как получить IP-адрес локальной машины? .....	425
Ping .....	426
Проверка доступности веб-адреса.....	428
Подключен ли компьютер к Интернету?.....	429
Пересылка данных по протоколу HTTP .....	430
Как послать запрос GET и отобразить полученные данные? .....	430
Как скачать файл из Интернета? .....	431
Передача файлов по протоколу FTP .....	432
Закачка файла на FTP-сервер.....	433
Получение оглавления папки.....	435
Загрузка файлов .....	437
Отправка писем через SMTP .....	438
Использование браузера Mozilla Firefox.....	439
Работа с локальной сетью .....	443
Как получить имя текущего пользователя?.....	443
Как выяснить, подключена ли локальная система к сети, и узнать используемый тип соединения? .....	444
Получение списка всех компьютеров локальной сети.....	445
Список SQL-серверов при помощи управляемого кода .....	453
Как получить дату и время удаленного компьютера?.....	454
Заключение .....	457

## **ГЛАВА 17. ФУНКЦИИ WINDOWS API.....459**

Вызов функций Windows API, имеющих выходной строковый параметр <i>char*</i> .....	460
Изменение типа, применяемого для маршалинга по умолчанию .....	461
Вызов функций, требующих <i>struct</i> .....	462
Работа с функциями обратного вызова в C#.....	463
Создание собственной управляемой библиотеки .....	463
Примеры использования функций API.....	464
Блокировка компьютера.....	464
Является ли текущий пользователь администратором? .....	464
Мигание заголовка формы .....	465
Форматирование дисков.....	466
Открытие и закрытие лотка привода компакт-дисков .....	467
Создание собственного пункта в системном меню .....	468
Работа с конфигурационными файлами INI.....	471
Извлечение значков из файлов .....	473
Вызов диалогового окна <i>Смена значка</i> .....	474

Продолжаем работать со значками .....	475
Панель задач, кнопка <i>Пуск</i> и часы в области уведомлений.....	477
Смена обоев Рабочего стола .....	481
Использование функций обратного вызова .....	484
Получение списка кодовых страниц, установленных в системе.....	484
Заключение .....	485
<b>ГЛАВА 18. НОВИНКИ VISUAL STUDIO 2008 .....</b>	<b>487</b>
Новшества в C# 3.0 .....	487
Неявно типизированные переменные .....	487
Инициализация объектов .....	488
LINQ .....	489
Вывод чисел из заданного массива с условием .....	489
Ключевое слово <i>Where</i> .....	490
Увеличение на единицу ряда чисел.....	490
Вывод имени числа.....	491
Вывод строк из массива в разных регистрах .....	491
Оператор <i>Take</i> .....	492
Оператор <i>TakeWhile</i> .....	493
Оператор <i>Skip</i> .....	493
Оператор <i>SkipWhile</i> .....	494
Заключение .....	494
<b>ГЛАВА 19. ССЫЛКИ НА ИНТЕРЕСНЫЕ МЕСТА.....</b>	<b>495</b>
Сайты .....	495
Блоги .....	496
Заключение .....	497
<b>ОПИСАНИЕ КОМПАКТ-ДИСКА .....</b>	<b>499</b>
<b>ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ.....</b>	<b>501</b>



# Вступление

Когда читатель берет в руки новую книгу, он пытается понять, действительно ли она ему нужна. Надеюсь, что эта книга оправдает ваши ожидания. В ней собраны самые разные советы, которые я отбирал для себя, чтобы впоследствии использовать их в своей практике. Нет никакого смысла изобретать заново велосипед, когда уже существует готовое решение, которое сэкономит время и силы. Я попытался выстроить систему советов от простого к сложному, хотя, конечно, это деление весьма условно. Надо сказать, что подборки советов по наиболее стандартным вопросам существуют для всех языков программирования. Достаточно набрать в поисковой системе ключевые слова "faq", "tips", "tricks" или их русские аналоги "чаво", "советы", "трюки", и вы увидите, как много сайтов имеют специальные страницы, на которых собраны самые популярные вопросы и ответы программистов. В конце книги вы найдете главу, в которой перечислены наиболее популярные интернет-сайты, посвященные советам по языку C# и .NET Framework. Рекомендуется регулярно посещать эти страницы, так как советы постоянно пополняются и обновляются вслед за выходом новых версий самого языка C# и платформы .NET Framework.

Автор не ставил перед собой цель охватить весь спектр вопросов, которые возникают у программистов, иначе книга писалась бы до бесконечности. Единственное отличие книжных советов от советов, выложенных на сайтах, — более подробное объяснение или комментарии. Бывает так, что ответ на какой-нибудь вопрос слишком лаконичен, и читателю (особенно новичку) не вполне ясно, когда возникает подобная проблема и почему нужно обратить внимание на ту или иную деталь. Все советы имеют практическую ценность, так как программисты не задают вопросы из праздного любопытства. Кто-то проводит бессонные ночи, пытаясь решить поставленную задачу, и понимает, что его знаний не хватает для правильного ответа. Тогда он принимает решение обратиться на форумы за помощью. И находятся добрые люди, которые уже сталкивались с этой проблемой, и они отвечают или дают рекомендации, которые направят программиста в нужное русло в поисках решения. Практически на любом форуме есть такие бескорыстные помощники, которые много знают в своей области и всегда готовы поделиться своими знаниями. Я очень уважаю таких людей, которые находят время, чтобы исследовать чужую проблему, поковыряться в коде и посоветовать свое решение. Огромное им спасибо!

Книгу не обязательно начинать читать с первой страницы. Используйте книгу как справочный материал. Просмотрите оглавление книги и, если увидите тему, которая вас особенно интересует, начните прямо с нее. Может, вы откроете для себя что-то новое, и прочитанная глава даст вам толчок в развитии давно заброшенного проекта. А потом обязательно прочтите остальные главы, потому что я уверен, что и другие полезные советы тоже пригодятся вам в работе.

## Для кого предназначена книга

Эта книга будет интересна программистам, пишущим свои программы на языке C#. Книга не является самоучителем. В магазинах имеется достаточно большой выбор книг для начинающих, в которых излагаются основы программирования. Предполагается, что читатель уже знаком с основами языка C# и .NET Framework и хотел бы расширить свои познания. У каждого программиста свой уровень подготовки, и трудно угодить всем. Поэтому в книге были собраны советы разной степени сложности.

## Благодарности

В книге приведены сотни советов от самых разных людей. Авторство многих советов уже невозможно установить, но, тем не менее, огромная благодарность их авторам от людей, которые делают первые шаги в программировании. Я тоже говорю огромное спасибо этим авторам, благодаря которым и я изучал тонкости языка, будь то C#, Visual Basic, C++, Delphi, JavaScript и т. д. Однако в книге присутствуют и советы, чье авторство известно. И я хочу отдельно выразить свою признательность этим людям.

## Требования

Большинство примеров в книге, как это видно из названия, написаны на языке программирования C# с использованием среды разработки Visual Studio. Когда я только приступал к работе над книгой, то писал примеры на Visual Studio 2005. Уже во время работы над книгой появилась бета-версия Visual Studio 2008, и я начал проверять написанные примеры заново. А когда я заканчивал написание книги, то вышла и финальная версия Visual Studio 2008.

Чтобы не отставать от времени, я переделал все проекты под новую версию среды разработки, воспользовавшись мастером обновлений. Таким образом, представленные на прилагаемом к книге компакт-диске примеры можно использовать как в Visual Studio 2008, так и в Visual Studio 2005, а некоторые примеры подойдут и для Visual Studio .NET 2003.

На самом деле в приводимых примерах важна сама идея. Поэтому опытные программисты могут воспроизвести код из примеров в среде разработки, которой они привыкли пользоваться. А те, кто имеет опыт написания программ на разных языках, могут переложить код из книги на другой язык. Я сам иногда дублировал код, написанный на C#, на Visual Basic .NET или, наоборот, из Visual Basic .NET переводил код на язык C#. Предполагается, что все примеры будут использоваться на платформе .NET Framework 2.0, хотя на самом деле большинство советов прекрасно подходит для .NET Framework 1.1. В некоторых случаях я буду намеренно обращать внимание на новинки, которые появились в новой версии .NET Framework 2.0.

Несомненно, новая версия среды разработки Visual Studio 2008 принесет с собой новые советы и трюки. Уже сейчас многие аналитики говорят о новой революции в программировании. Как вы уже догадались, речь идет об особом языке запросов LINQ, который встроен в Visual Studio 2008. В конце книги я приведу несколько примеров использования LINQ, хотя эта тема заслуживает отдельной книги. Также надо отметить тот факт, что в 2007 году вышла в свет новая операционная система Windows Vista. Надо признать, что отношение к этой системе в России, да и во всем мире, сложилось неоднозначное. Но, тем не менее, нам все равно придется переходить на эту систему. Поэтому в конце книги я приведу несколько примеров, которые используют новые возможности Windows Vista.

## **Чего вы не найдете в этой книге**

Книга — не резиновая игрушка, которую можно растягивать до невероятных размеров. Поэтому считаю своим долгом предупредить читателя, чего вы не найдете в этой книге. Есть темы, которые требуют отдельного разговора и которые интересны более специализированной группе разработчиков. Например, в этой книге вы не найдете советов, связанных с ASP.NET 2.0 или Microsoft SQL Server 2005. Безусловно, разработка веб-приложений или баз данных является важной задачей для многих программистов, но поищите советы в других книгах. Кроме того, в книге не будет советов, относящихся к платформе .NET Framework 3.0. Возможно, вы в курсе, что название .NET

Framework 3.0 является всего лишь маркетинговым ходом. На самом деле, платформа .NET Framework 3.0 является комбинацией .NET Framework 2.0 и новых библиотек Workflow. Эти библиотеки — тоже тема для отдельной книги.

## Исходные коды

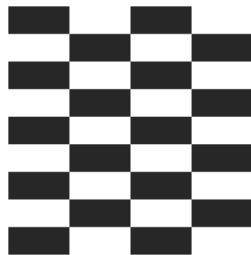
Я всячески рекомендую писать код своими руками, а не копировать из готовых источников. Но если код слишком велик, а идея совета понятна, то возникает соблазн скопировать нужный кусок кода и вставить в свой проект. Поэтому все представленные советы можно найти на компакт-диске, который в обязательном порядке должен прилагаться к книге. Все примеры распределены по папкам, которые имеют названия, соответствующие номерам глав. Внутри каждой из этих папок находятся дополнительные папки, в которых уже содержатся файлы проектов. Если вы пользуетесь средой разработки Visual Studio 2008, то вам достаточно отыскать в папке файл с расширением SLN и запустить его.

Кроме того, на диске вы найдете электронный справочник по функциям Windows API и сообщениям Windows для Visual Basic .NET и C#. На самом деле справочник является платным, и я стараюсь ежемесячно обновлять его. Но вы получите один выпуск справочника совершенно бесплатно! Это небольшой подарок от меня. Если справочник вам понравится, то вы можете приобрести его за небольшую плату и получать обновления.

## Обратная связь

Если вы считаете, что я незаслуженно обошел в книге какие-то темы, сообщите мне об этом. Я постараюсь учесть все пожелания читателей и исправиться в следующей книге. Кроме того, на сайте <http://netsources.narod.ru> будут выкладываться интересные исходные коды, которые могут заинтересовать вас. Если вы захотите поделиться своим интересным проектом, то можете прислать его на мой электронный адрес [rusproject@mail.ru](mailto:rusproject@mail.ru). Жду ваших писем!

# ГЛАВА 1



## Общее

Прежде чем мы перейдем к трюкам, нужно сказать несколько общих слов о простых вещах, которые помогут нам настроиться на нужный лад. Опытные программисты могут пропустить эту главу.

## Часто задаваемые вопросы

Многие новички, которые только начинают осваивать новый для себя язык `C#`, ищут ответы на довольно простые вопросы. Мы рассмотрим некоторые из них.

### **Можно ли запустить программу, написанную на `C#`, без `.NET Framework`?**

Нет, нельзя. Чтобы ваша программа, написанная на `C#`, заработала на компьютере пользователя, необходимо установить специальные библиотеки. Скачать необходимые компоненты можно бесплатно с сайта Microsoft. В операционную систему Windows Vista платформа `.NET Framework 2.0` уже встроена. Также она поставляется вместе с пакетом обновлений Service Pack 2 для Windows XP.

### **В каком редакторе писать программы?**

На самом деле программы можно писать даже в стандартном Блокноте, который имеется в любой версии Windows. Очень многие профессиональные программисты так и поступают в отдельных случаях. При таком подходе создается полностью чистый код без добавления различных дополнительных



строка кода, который вставляют визуальные среды разработки. Я думаю, каждый должен хотя бы раз написать программу таким способом, тем более что здесь нет ничего сложного. Для того чтобы писать программы таким способом, вам необходимо скачать .NET Framework SDK, в состав которого входит компилятор `csc.exe`. При помощи данного компилятора и создается полноценное приложение, написанное на чистом C#. Запустите программу Блокнот и напишите строки, приведенные в листинге 1.1.

#### Листинг 1.1. Простая программа, написанная в Блокноте

```
// Консольная программа на C#,  
// написанная в Блокноте  
class NotepadProgram  
{  
    public static void Main()  
    {  
        System.Console.WriteLine("C#. Трюки для программиста");  
        System.Console.ReadLine();  
    }  
}
```

Сохраните текст программы, приведенный в листинге 1.1, в файле под именем `notepad.cs` и запустите в командной строке команду

```
csc.exe notepad.csc
```

В результате у вас получится файл `notepad.exe`. Более подробно процесс создания программ при помощи блокнота описывать мы не будем, желающие сами найдут дополнительную информацию по этому вопросу. Кроме консольных приложений, можно создавать и обычные Windows-приложения. В этом случае в командной строке нужно указывать дополнительные параметры.

Но будем реалистами, подавляющее число программистов не используют отдельный текстовый редактор и компилятор для создания программ. Поэтому не будем надолго задерживаться на этой теме. Можно только добавить, что существуют более удобные оболочки для написания кода, чем Блокнот, имеющие подсветку синтаксиса, выбор различных параметров компилятора и т. д. Но все же большинство программистов используют пакет Visual Studio 2008 или ее усеченный вариант Visual C# 2008. Хочется отметить прекрасный подарок от Microsoft — выпуск бесплатных Express-версий Visual Studio 2008, с помощью которых можно писать полноценные приложения практически любой сложности.

Далее я приведу несколько простых советов и правил, которые помогут новичкам быстрее освоить новый язык программирования.

## Использование в качестве переменных русских символов

В отличие, скажем, от Visual Basic 6.0, в C# используются символы Unicode при написании кода. Поэтому вы можете выбирать в качестве имен переменных, например, буквы греческого алфавита, иврит или русские буквы. Таким образом, можно изменить программу так, что она будет похожа на программу, написанную на встроенном языке 1С, как в листинге 1.2.

### Листинг 1.2. Использование русских символов в коде

```
// Русские псевдонимы
using целое = System.Int16;
using дробное = System.Single;
```

```
целое переменная1 = 7;
дробное переменная2 = 3.14F;
```

## Псевдонимы

Обратите внимание на оператор `using`. Как правило, этот оператор используют для указания пространства имен, что позволяет сократить написание кода. Но оператор `using` можно применять не только для указания пространств имен, но и для сокращенной записи классов, то есть использовать его для описания псевдонимов. Запомните, что псевдонимы записываются вне описания класса. В приведенном выше примере мы используем псевдоним `целое` для целочисленного типа `System.Int16`. Чтобы закрепить материал с псевдонимами, приведу еще один пример (листинг 1.3).

### Листинг 1.3. Использование псевдонимов

```
using System;
using строка = System.String; // псевдоним строка для строкового типа
// System.String
```

```
class MainClass
{
    static void Main()
    {
        строка ИмяКота = "Рыжик";
        Console.WriteLine("Моего кота зовут " + ИмяКота);
    }
}
```

## Копирующий строковой литерал

В C# можно использовать специальный копирующий строковой литерал. Копирующий строковой литерал начинается с символа @, за которым следует строка в кавычках. Во многих случаях это очень удобно. Например, подобный литерал часто используется при указании полного имени файла. Без использования этого литерала пришлось бы дублировать обратные слэши, используемые в пути файла:

```
string filename = "c:\\windows\\temp\\samples\\test.txt";
```

Дублировать слэши приходится из-за того, что компилятор принимает их за начало escape-последовательности. Но если вы поставите знак @, а содержимое строки заключите в кавычки, то сама строка в этом случае принимается без модификации:

```
string filename = @"c:\windows\temp\samples\test.txt";
// так проще и удобнее
```

## Символ @ перед идентификатором

Символ @ также можно использовать перед идентификатором. Такая позиция символа @ позволяет использовать имя, даже если оно является ключевым словом. На самом деле эта возможность вряд ли пригодится в вашей практике. Но, возможно, вам придется подключать какую-нибудь библиотеку, написанную командой разработчиков на каком-нибудь другом языке программирования. Предположим, что существует такой язык, как Visual CatPro, в котором слова *this*, *else*, *foreach* не являются зарезервированными. И автор библиотеки, не подозревая о языке C#, спокойно использовал эти слова в качестве названий классов, функций и т. п. Если вы используете эту библиотеку, то вам придется проставить символ @ перед ключевым словом, как в лис-

тинге 1.4 он стоит перед `this`, чтобы иметь возможность работать с библиотекой без проблем.

#### Листинг 1.4. Использование символа @ в зарезервированных словах

```
using System;
using строка = System.String;

class MainClass
{
    static void Main()
    {
        строка ИмяКота = "Рыжик";
        int @this = 5;
        Console.WriteLine("Моему коту " + @this + " лет");
    }
}
```

## Как узнать, присвоено ли переменной значение

Чтобы узнать, присвоено ли переменной какое-нибудь значение, можно воспользоваться новым типом `Nullable`, который появился в `.NET Framework 2.0`. В листинге 1.5 приведен пример объявления переменной как `Nullable` и проверки ее значения.

#### Листинг 1.5. Проверка на присвоение значения переменной

```
Nullable<bool> bFlag = null;

// Снимите комментарий со следующей строки,
// чтобы присвоить переменной значение
//bFlag = true;

if (bFlag.HasValue)
    this.Text = "Переменная bFlag имеет значение: " + bFlag.Value;
else
    MessageBox.Show("Переменной bFlag не присвоено никакого значения");
```

Существует и короткая формы записи для `Nullable`:

```
bool? bFlag = null; // альтернативный вариант
```

Объявление переменной как `Nullable` делает доступными такие свойства, как `HasValue` и `Value`, которые помогают узнать, было ли присвоено переменной значение. Для использования `Nullable` подходят все стандартные типы переменных, см. листинг 1.6.

#### Листинг 1.6. Проверка на присвоение значения переменным разных типов

```
int? i = 10;           // integer
double? dl = 3.14;    // double
bool? flag = null;    // bool
char? letter = 'a';   // символ
int?[] arr = new int?[10]; // массив
```

## Как это назвать?

Правильный выбор наименований для различных объектов — залог хорошей читаемости вашего кода. Не думайте, что читать его придется другим людям, и что нет смысла облегчать им жизнь. Если через годик после написания кода вы заглянете в него, то обнаружите, что вы уже и сами чувствуете себя, будто код написал кто-то другой. Поэтому придерживайтесь определенных правил, хотя бы для собственного удобства.

## Какая разница между *string* и *System.String*?

Некоторые начинающие программисты смущаются, когда видят в коде либо `string`, либо `String`, памятуя, что язык `C#` различает регистр. Но на самом деле разницы между этими словами никакой. Это дело привычки. Одни программисты (например, Ч. Петцольд) привыкли использовать подход `C#` и применяют слово `string`, а другие (например, Д. Рихтер) призывают придерживаться синтаксиса `.NET Framework`. Вот небольшая таблица соответствий:

<code>string</code>	<code>System.String</code>
<code>short</code>	<code>System.Int16</code>
<code>ushort</code>	<code>System.UInt16</code>

Только ни в коем случае не используйте сочетание `System.string`, потому что класса `string` в пространстве имен `System` не существует.

## Выберите свои правила наименования

Microsoft призывает программистов использовать определенные правила наименования при создании проектов. В документации MSDN можно найти рекомендации по этому вопросу. Также внутри компании, в которой вы работаете, могут придерживаться собственных правил. Но в любом случае не воспринимайте рекомендации как догму. Старайтесь выработать свою схему именования в проектах, с которой вам будет удобно работать. Например, у меня осталась привычка с Visual Basic 6.0 давать префиксы в именах элементов управления. Например, кнопки начинаются с префиксов `but`, а текстовые поля с префикса `txt`. Когда я набираю в редакторе кода префикс и затем нажимаю комбинацию клавиш `<Ctrl>+<Пробел>`, то IntelliSense выводит список всех имеющихся кнопок или текстовых полей в проекте. Но желательно придерживаться и общепринятых правил.

## Правила для названий классов и методов

Обычно классы представляют объекты, а методы — действия. Поэтому при наименовании классов или методов придерживайтесь следующего правила. Для наименования классов используйте имена существительные — `Cat`, `Machine`, `Girl`. При создании методов используйте глаголы в следующей нотации — `MoveLeft`, `ShowMessage` и т. д. В этом случае вам и другим программистам будет проще ориентироваться в коде.

## Советы по созданию эффективных и масштабируемых приложений

В одном из номеров журнала MSDN Magazine автор Кен Спенсер в ответ на вопрос от одного читателя приводит несколько полезных советов и рекомендаций по созданию эффективного и масштабируемого приложения. Я приведу несколько таких полезных советов, которые мне показались очень интересными, в моей интерпретации.

## Сопряжение

Сопряжение определяет, насколько тесна связь между двумя кусками кода в вашей программе? Как показывает практика, чем слабее такая связь, тем