

Visual Basic .NET

- Разработка интерфейса приложения
- Работа с файлами и организация печати
- Управление данными с помощью ADO.NET
- Применение Web-сервисов при создании интернет-приложений
- Использование графических возможностей GDI+
- Visual Basic .NET 2003 и новые возможности, включенные в версию Visual Basic .NET 2005



САМОУЧИТЕЛЬ

Visual

Basic .NET

Под редакцией
Аркадия Тихонова

Санкт-Петербург

«БХВ-Петербург»

2005

УДК 681.3.068+800.92VB.NET2003

ББК 32.973.26-018.1

K26

Карпов Р. Г. и др.

K26 Самоучитель Visual Basic .NET. / Авторы: Карпов Р. Г., Соколова Н. Е., Степанов А. М., Шевякова Д. А. / Под. ред. А. Ф. Тихонова. — СПб.: БХВ-Петербург, 2005. — 592 с.: ил.

ISBN 5-94157-417-7

Доступно и подробно представлена популярная система программирования Visual Basic .NET 2003, рассмотрены также новые возможности Visual Basic .NET 2005 для создания приложений практически в любой области компьютерных технологий. Описаны стандартные элементы для разработки пользовательского интерфейса, использование в проектах графики, таймеров, ползунков, гиперссылок и др. средства отладки приложений и обработки ошибок. Раскрыты основные понятия объектно-ориентированного программирования и классы. Изложены вопросы работы с графикой с использованием интерфейса GDI+, разработки справочной системы в формате HTML и создания собственного установочного компакт-диска. Большое внимание уделяется приложениям типа клиент/сервер, предназначенным для управления базами данных: создание форм для ввода и редактирования данных с использованием компонентов ADO.NET, создание отчетов с помощью генератора отчетов Crystal Reports и др. Приведена методика разработки приложений, работающих в Интернете.

Для начинающих программистов

УДК 681.3.068+800.92VB.NET2003

ББК 32.973.26-018.1

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. гл. редактора	<i>Игорь Шишигин</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Анна Кузьмина</i>
Компьютерная верстка	<i>Натальи Караваевой</i>
Корректор	<i>Виктория Пиотровская</i>
Дизайн обложки	<i>Игоря Цырульникова</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 26.11.04.

Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 47,73.

Тираж 3000 экз. Заказ №

"БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Гигиеническое заключение на продукцию, товар № 77.99.02.953 Д.001537.03.02 от 13.03.2002 г. выдано Департаментом ГСЭН Минздрава России.

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

ISBN 5-94157-417-7

© Карпов Р. Г., Соколова Н. Е., Степанов А. М., Шевякова Д. А., 2005
© Оформление, издательство "БХВ-Петербург", 2005

Содержание

Введение	1
Как построена книга	2
Специальные элементы книги	3
Глава 1. Первое знакомство с Visual Basic .NET	5
Запуск Visual Studio .NET	6
Главное окно Visual Studio .NET.....	7
Создание нового проекта.....	8
Главное меню	10
Меню <i>File</i>	11
Меню <i>Edit</i>	11
Меню <i>View</i>	12
Меню <i>Project</i>	12
Меню <i>Build</i>	12
Меню <i>Debug</i>	12
Меню <i>Format</i>	13
Меню <i>Tools</i>	13
Меню <i>Window</i>	13
Меню <i>Help</i>	13
Стандартная панель инструментов.....	13
Окно <i>Start Page</i>	15
Окно конструктора форм.....	17
Окно редактора кода	18
Окно <i>Solution Explorer</i>	20
Окно <i>Class View</i>	21
Окно <i>Toolbox</i>	22
Окно <i>Properties</i>	23
Окно <i>Object Browser</i>	25
Окно <i>Locals</i>	26

Окна <i>Watch</i>	27
Окно <i>Command Window</i>	27
Справочная система	28
Окно справочной системы.....	28
Вкладка <i>Contents</i>	31
Вкладка <i>Index</i>	31
Вкладка <i>Search</i>	32
Вкладка <i>Favorites</i>	34
Использование динамической справки.....	35
Глава 2. Основы программирования в Visual Basic .NET.....	37
Переменные.....	37
Имена переменных	37
Типы данных	38
Объявление переменной.....	41
Область видимости переменных	42
Присвоение значения переменной	43
Константы.....	44
Встроенные константы Visual Basic	44
Объявление констант.....	44
Массивы.....	45
Объявление массива.....	46
Объявление массива фиксированного размера.....	46
Объявление динамического массива.....	46
Оформление программного кода	47
Комментарии	47
Размещение оператора на нескольких строках	48
Размещение нескольких операторов на одной строке	48
Программные модули.....	48
Редактирование исходного кода	49
Процедуры	52
Процедуры <i>Sub</i>	52
Процедуры <i>Function</i>	54
Вызов процедур	55
Передача параметров	56
Возвращаемое значение.....	57
Управляющие конструкции и циклы	58
Управляющие конструкции Visual Basic.....	58
Условные выражения.....	59
Конструкция <i>If ... Then</i>	60
Конструкция <i>If ... Then ... Else</i>	61
Конструкция <i>Select Case</i>	62
Циклы.....	63

Цикл <i>For...Next</i>	63
Цикл <i>For Each...Next</i>	65
Цикл <i>Do...Loop</i>	65
Оператор <i>Exit</i>	67

Глава 3. Построение интерфейса пользователя 69

Создание нового проекта	69
Сохранение проекта	71
Выполнение приложения	72
Создание формы	72
Свойства объектов формы	73
Общие для всех объектов свойства	77
Обработка событий	78
Действия, выполняемые с объектами формы	79
Выделение объектов формы	79
Отмена выделения с объектов	80
Перемещение объектов в форме	80
Удаление объектов из формы	80
Изменение размеров объектов	81
Выравнивание объектов формы	81
Линии сетки	82
Порядок обхода объектов формы	83
Настройка параметров формы	83
Расположение формы и ее размеры	84
Заголовок формы	85
Стиль обрамления формы	85
Фон формы	86
Полоса прокрутки	86
События формы	87
Интерфейс	88
Общие рекомендации по разработке интерфейса	88
Типы интерфейсов	88
SDI-интерфейс	89
MDI-интерфейс	90
Интерфейс типа проводника	94
Элементы интерфейса	95
Меню	95
Редактор меню <i>Menu Editor</i>	96
Имя и текст	97
Горячие клавиши	97
Использование флажков	97
Свойства меню для MDI-интерфейса	98
Свойства, определяющие состояние пункта меню	99

Контекстное меню.....	99
Пример создания меню	99
Строка состояния.....	100
Пример создания строки состояния	102
Панель инструментов	103
Свойства элементов панели инструментов	104
Свойства панели инструментов	105
Расположение рисунка на кнопке.....	106
Пример создания панели инструментов.....	106
Диалоговые окна	107
Окно сообщения.....	108
Диалоговое окно открытия файла	111
Диалоговое окно сохранения файла.....	113
Диалоговое окно настройки шрифтов текста	114
Диалоговое окно настройки цветовой палитры.....	116

Глава 4. Использование основных элементов управления

Visual Basic .NET..... 119

Общие свойства элементов управления	119
Метка.....	120
Задание размера.....	121
Задание клавиш быстрого доступа.....	122
Размещение рисунка на надписи	122
Текстовое поле	122
Свойства, определяющие внешний вид	123
Многострочные текстовые поля.....	123
Управление текстом.....	124
Нередактируемые текстовые поля.....	126
Проверка правильности ввода данных	126
Использование текстового поля для ввода пароля	127
Кнопка управления.....	127
Клавиши быстрого доступа.....	128
Кнопка по умолчанию и кнопка отмены.....	128
Стиль оформления кнопки	128
Способы выбора кнопки управления	129
Флажок.....	130
Переключатель	132
Объединение элементов формы.....	133
Элемент управления <i>Panel</i>	134
Элемент управления <i>GroupBox</i>	134
Списки	135
Элемент управления <i>ListBox</i>	136
Добавление элементов в список	136

Удаление элементов из списка	137
Вставка элементов в список	138
Выбор нескольких элементов из списка	138
Доступ к элементам списка	139
Выделенные элементы списка	139
Поиск элемента списка	141
Элемент управления <i>ComboBox</i>	142
Стиль оформления списка	142
Параметры раскрывающегося списка	143
Добавление и удаление элементов списка	143
Доступ к элементам списка	144
Элемент управления <i>CheckedListBox</i>	144
Элементы списка	145
Элемент управления <i>NumericUpDown</i>	146
Значения списка	147
Внешний вид элемента управления	147
Элемент управления <i>DomainUpDown</i>	148
Значения списка	148
Внешний вид элемента управления	149
Пример	149

Глава 5. Дополнительные элементы управления Visual Basic .NET..... 151

Использование в форме графики	151
Элемент управления <i>PictureBox</i>	151
Размер графического объекта	152
Способы загрузки изображения	153
Элемент управления <i>ImageList</i>	154
Полосы прокрутки	157
Размещение полосы прокрутки и настройка свойств	158
Пример использования полос прокрутки	159
Таймер	160
Использование таймера	161
Задание даты	162
Элемент управления <i>MonthCalendar</i>	162
Внешний вид элемента управления	163
Выделение дат	164
Работа с календарем	165
Элемент управления <i>DateTimePicker</i>	166
Внешний вид элемента управления	167
Получаемые значения	168
Вкладки	168
Индикатор прогресса	171
Ползунок	172

Гиперссылка	173
Отдельная гиперссылка	174
Сложные гиперссылки	174
Выбор гиперссылки	175
Вид ссылок.....	176
Элемент управления <i>NotifyIcon</i>	176

Глава 6. Объектно-ориентированное программирование в Visual Basic .NET..... 179

Инкапсуляция	179
Наследование.....	180
Полиморфизм.....	180
Структура класса.....	180
Члены классов.....	184
Поля.....	184
Методы	184
Свойства	185
События.....	188
Создание и удаление классов и экземпляров классов	190
Переопределение методов базовых классов	192
Интерфейсы.....	195
Создание визуальных классов.....	197
Создание класса элемента управления.....	197
Наследование класса элемента управления	206
Создание класса-формы.....	207
Использование Visio для создания структуры классов	209
Создание диаграмм из проектов Visual Basic .NET.....	209
Создание UML-диаграмм и генерация кода на Visual Basic .NET.....	211

Глава 7. Работа с файлами и организация печати..... 217

Основные операции с файлами	217
Работа с информацией о файле.....	218
Удаление файла	221
Перемещение файла	222
Копирование файла	223
Создание файла	224
Открытие файла	225
Переход на заданную позицию в файле.....	227
Считывание данных из текстового файла.....	227
Считывание всех данных целиком из текстового файла	228
Построчное считывание данных из текстового файла.....	228
Считывание данных из текстового файла блоками.....	229

Запись данных в текстовый файл	229
Побайтовые операции с файлами	230
Работа с каталогами и устройствами	232
Получение списка файлов и подкаталогов указанного каталога	233
Получение информации о каталоге	234
Удаление каталога	235
Перемещение каталога	236
Создание каталога	236
Работа с путями к файлам	237
Просмотр окружения	239
Просмотр изменений файловой системы	240
Организация печати	244
Примеры организации печати	246

Глава 8. Управление графикой 251

Первые шаги	252
Структуры пространства имен <i>System.Drawing</i>	253
Задание координат точки	254
Размер объекта	255
Задание параметров прямоугольника	256
Задание цвета	258
Построение линий и фигур	260
Типы линий	260
Прямая линия	262
Ломаная линия	263
Дуга	264
Сплайны	266
Сплайны Безье	266
Основные сплайны	267
Замкнутые сплайны	269
Сектор	270
Прямоугольник и набор прямоугольников	272
Эллипс	273
Многоугольник	274
Путь	275
Заливка фигур	279
Виды заливки фигур	279
Однородная заливка	280
Текстурная заливка	280
Штриховая заливка	281
Градиентная заливка	282
Прямоугольники	285
Эллипс	286

Сектор.....	287
Замкнутый сплайн	289
Многоугольник.....	290
Путь.....	291
Подробнее о градиентной заливке.....	292
Текст.....	296
Шрифт	297
Создание текста.....	298
Формат текста.....	299
Нахождение существующих шрифтов	302
Определение размера строки	303
Изображения	304
Растровое изображение	304
Создание изображения	304
Расположение изображения на форме.....	305
Сохранение изображения	307
Значок.....	309
Дополнительные параметры.....	310
Заливка формы	310
Аффинное преобразование	310
Управление качеством	313
Использование областей	315
Задание области видимости графики	316
Анимационная графика	319
Перемещение изображения	319
Размещение на форме многокадровых изображений.....	321
Глава 9. Мультимедиа в Visual Basic .NET	325
Общие понятия	325
Типы файлов мультимедиа	325
Типы управляемых устройств	326
Интерфейс MCI	327
Элемент управления MCI	327
Разработка простого проигрывателя при помощи элемента управления MCI	334
Проигрыватель аудиофайлов WAV.....	334
Проигрыватель видеофайлов	336
Использование Windows Media Player.....	338
Разработка простого проигрывателя с помощью Windows Media Player.....	339

Глава 10. Создание справочной системы приложения	345
Создание справочной системы в формате HTML	345
Окно HTML Help Workshop.....	346
Определение параметров проекта справочной системы	347
Определение псевдонимов тем.....	348
Определение связи между псевдонимами и индексами тем.....	349
Создание содержания справочной системы	350
Создание ключей для поиска тем	351
Компиляция и тестирование справочной системы.....	353
Использование справочной системы в приложениях	354
Создание кнопки и меню для вызова справочной системы.....	354
Определение темы справочной системы для формы и элементов управления	356
Отображение всплывающей подсказки.....	357
Глава 11. Управление данными.....	359
Особенности ADO.NET	359
Организация хранения данных	360
Организация доступа к данным	361
Объектная модель ADO.NET.....	362
Объект <i>DataSet</i>	363
Объект <i>Connection</i>	364
Объект <i>Command</i>	364
Объект <i>DataAdapter</i>	364
Объект <i>DataReader</i>	364
Подключение компонентов ADO к проекту	365
Пространства имен	367
Создание подключения к базе данных.....	367
Управление данными	369
Передача данных между источником данных и <i>DataSet</i>	372
Объект <i>DataSet</i>	376
Использование <i>DataSet</i> без связывания с таблицами баз данных.....	379
Объект <i>DataTable</i>	383
Использование мастера настройки объекта <i>DataAdapter</i>	384
Отображение данных.....	392
Глава 12. Построение отчетов	395
Создание отчета	395
Соединение с источником данных.....	399
Добавление полей в отчет.....	401
Использование параметров.....	406

Глава 13. Создание интернет-приложений	409
ASP.NET-приложение	409
Основные технологии, используемые при создании Web-приложения	411
HTML 4.0	412
Каскадные таблицы стилей.....	412
Управление поведением тегов	413
HTML DOM 1.0.....	413
ActiveX-объекты.....	413
XML 1.0	413
XML DOM 1.0	413
SOAP	414
Структура проекта Web-приложения	414
Настройка Web-приложения	415
Global.asax	415
Файл Web.config.....	418
Секция <code><appSettings></code>	418
Секция <code><sessionState></code>	419
Секция <code><globalization></code>	419
Секция <code><compilation></code>	419
Секция <code><trace></code>	420
Создание Web-формы	421
Создание презентационной части Web-формы.....	422
Заголовок файла	422
Источники данных	423
Использование источников данных в презентационной части Web-формы.....	423
Заполнение XML-документа	425
Использование каскадной таблицы стилей	427
Управление поведением объектов.....	428
Использование Web-сервисов	430
Вызов Web-сервисов из HTML-страницы	431
Формирование заголовка Web-формы.....	431
Добавление XML-тегов.....	433
Вызов Web-сервиса.....	434
Отображение данных, полученных от Web-сервиса.....	434
Определение поведения для XML-тега.....	435
Создание Web-сервиса.....	436
Средства отладки	439
Создание дистрибутива приложения.....	440
Настройка общих свойств дистрибутива.....	440
Настройка параметров работы приложения и параметров доступа к каталогам приложения	441
Определение файлов приложения	442

Глава 14. Расширенные средства Visual Basic .NET.....	443
Сервисы.....	443
Менеджер сервисов.....	443
Взаимодействие сервисов с рабочим столом.....	444
Обработка исключений в сервисах.....	445
Разработка простого сервиса.....	446
Создание класса для установки сервиса.....	447
Класс <i>ServiceProcessInstaller</i>	448
Класс <i>ServiceInstaller</i>	449
Установка и удаление сервиса.....	449
Многопоточное программирование.....	450
Создание потока для выполнения определенной задачи.....	450
Использование асинхронных делегатов.....	453
Функции, создаваемые компилятором.....	453
Функция <i>BeginInvoke</i>	454
Функция <i>EndInvoke</i>	454
Пример выполнения асинхронных вызовов.....	454
Синхронизация потоков.....	457
Класс <i>Monitor</i>	458
Классы <i>AutoResetEvent</i> и <i>ManualResetEvent</i>	461
Класс <i>Mutex</i>	462
Пример создания многопоточного сервиса.....	463
Исходный код сервиса.....	463
Описание работы сервиса.....	468
Глава 15. Взаимодействие с внешними программами	471
Подключение библиотек к проекту.....	471
Управление Microsoft Word.....	474
Управление Microsoft Excel.....	480
Управление Microsoft PowerPoint.....	483
Глава 16. Отладка программ, обработка ошибок и оптимизация приложений	485
Отладка программ.....	485
Обработка ошибок.....	492
Обработка ошибок с помощью конструкции <i>Try...Catch...Finally</i>	494
Оптимизация приложений.....	495
Оптимизация скорости работы приложения.....	497
Оптимизация размера приложения.....	498
Глава 17. Групповая разработка проекта.....	501
Администрирование SourceSafe.....	502
Запуск SourceSafe.....	502
Настройка.....	502

Работа с пользователями	507
Работа с данными.....	512
Работа пользователя в SourceSafe	513
Иерархия в SourceSafe	516
Работа с проектами.....	516
Работа с файлами проекта.....	517
SourceSafe в среде Visual Basic .NET.....	521
Глава 18. Установка приложения.....	527
Создание инсталлятора	527
Использование мастера установки проекта.....	528
Дополнительная настройка параметров пакета установки	532
Настройка параметров размещения и запуска приложения.....	534
Определение папки, в которой будет установлено приложение.....	534
Добавления ярлыка в меню <i>Пуск</i>	535
Ярлык на рабочем столе клиента.....	536
Завершение создания файла установки приложения.....	537
Глава 19. Обзор новых возможностей Visual Basic 2005	539
Объект <i>My</i>	539
Добавление фрагмента кода в программу.....	541
Редактирование кода во время отладки	542
Использование подсказок в режиме отладки	542
Подсказки при компиляции кода.....	542
Диалоговое окно <i>Immediate</i>	544
Обработка исключений	545
Усовершенствование возможностей Intellisense	545
Упрощение работы с формами	546
Расширения синтаксиса языка.....	547
Перегрузка операторов	547
Новые целочисленные типы данных	549
Частичные типы	549
Новые ключевые слова.....	550
<i>Using</i>	550
<i>Continue</i>	550
Различный уровень доступа для свойств <i>Get</i> и <i>Set</i>	551
Обобщенные типы	551
Создание обобщенных классов.....	552
Управление данными	555
Настройка параметров отладки кода.....	557
XML-комментарии	558
Предметный указатель	559

Введение

О Visual Basic .NET написано достаточно много книг, раскрывающих с разной степенью подробности все возможности этой версии популярного языка программирования. В данной книге мы постарались в первую очередь доступно и подробно показать все новое, что предоставляет пользователю версия Visual Basic .NET, не забывая, конечно, и о возможностях, унаследованных от предыдущих версий.

Обычно Basic ассоциируется с чем-то очень простым в освоении и использовании для программирования. Это действительно так. На заре компьютерных технологий язык BASIC был создан для простых программ и использовался в качестве учебного языка для первых шагов при изучении основ программирования с последующим переходом на более сложные и универсальные языки. Это было заложено в название языка *BASIC — Beginners All-purpose Symbolic Instructional Code*, т. е. многоцелевой код символьных инструкций для начинающих. С прогрессом компьютерных технологий развивался и BASIC. В настоящее время версия Visual Basic .NET дает возможность решать любые современные задачи разработки приложений. При этом Visual Basic .NET остался достаточно простым в освоении, став в то же время мощным современным языком программирования.

С помощью Visual Basic .NET можно создавать приложения практически для любой области современных компьютерных технологий: бизнес-приложения, игры, мультимедиа, базы данных, интернет-приложения. При этом приложения могут быть как простыми, так и очень сложными, в зависимости от поставленной задачи.

Простота и мощь языка Visual Basic .NET позволили сделать его встроенным языком для приложений Microsoft Office. В настоящее время Visual Basic уже не считается учебным языком — знание Visual Basic и его диалектов (VBA, VBScript) становится необходимостью для современного программиста любого уровня.

В Visual Basic .NET используются все самые современные методы программирования: объектно-ориентированная модель, включая наследование визуальных

классов, модель составных объектов COM (Component Object Model), технология программных компонентов ActiveX и другие. Суть этих подходов и их реализацию на примерах можно изучить, прочитав посвященные им главы из этой книги.

Книга рассчитана на широкий круг пользователей. Начинающему программисту материалы данной книги помогут быстро изучить язык и все основные возможности Visual Basic .NET. Книга будет полезна и читателю, имеющему опыт программирования на предыдущих версиях Visual Basic.

Как построена книга

Книга предполагает последовательное изучение материала от простого к сложному. *Глава 1* является вводной. Вы узнаете, как запустить Visual Basic, как получить в нем справочную информацию, познакомитесь с интегрированной средой разработки, с основными рабочими окнами, а также с настройкой среды разработки программы.

В *главе 2* рассматриваются основы программирования в Visual Basic, кратко рассмотрен редактор исходных кодов.

Глава 3 посвящена разработке интерфейса пользователя. Подробно рассмотрен вопрос разработки основ интерфейса: форм и меню. Вы познакомитесь со средствами, предоставляемыми Visual Basic для создания панелей инструментов, диалоговых окон, строки состояния.

Среда Visual Basic .NET включает множество элементов управления, позволяющих создать богатый пользовательский интерфейс. В *главе 4* рассматриваются стандартные элементы управления, используемые наиболее часто, такие как надписи, поля различных типов, флажки, списки и т. д. *Глава 5* познакомит вас с размещением в форме графики (в том числе списки с графическими изображениями), полос прокрутки, таймера, объектов, позволяющих работать с датами, добавлением в форму вкладок, облегчающих размещение и группировку большого объема данных, индикатора процесса выполнения, ползунка и гиперссылок.

Язык Visual Basic .NET является объектно-ориентированным языком. В *главе 6* вы познакомитесь с основными понятиями объектно-ориентированного программирования, такими как инкапсуляция, наследование, полиморфизм. Что такое члены класса, как создавать классы и экземпляры класса, переопределять методы базовых классов, использовать интерфейсы, реализуемые классом, и создавать структуры классов с помощью UML-диаграмм, вы также узнаете из этой главы.

При проектировании приложения достаточно часто возникает необходимость в работе непосредственно с файлами. В *главе 7* подробно рассматриваются вопросы добавления, удаления файлов или каталогов, записи данных в файлы, чтения из них данных как программно, так и в интерактивном режиме.

Глава 8 посвящена работе с графикой с использованием нового интерфейса GDI+. В *главе 9* речь идет о применении в приложениях мультимедиа.

О том, как разработать для своего приложения эффективную справочную систему в формате HTML, вы узнаете из *главы 10*.

Глава 11 посвящена элементам управления данными. Из нее вы узнаете о создании форм для ввода и редактирования данных с использованием компонентов ADO.NET.

Рассмотрению одной из главных задач информационной системы, представлению данных в виде отчетов с помощью генератора отчетов Crystal Reports посвящена *глава 12*. Изучив материал *главы 13*, вы познакомитесь с методикой создания приложений, работающих в Интернете.

Глава 14 посвящена расширенным средствам Visual Basic .NET: созданию сервисов (системных служб Windows) и многопоточному программированию.

Использованию в разработке приложений возможностей других программ, таких, например, как программы, входящие в пакет Microsoft Office, посвящена *глава 15*.

О том, какие средства предоставляет в распоряжение разработчика Visual Basic по отладке приложений и обработке ошибок, вы узнаете из *главы 16*.

Вопросам бесконфликтной работы группы программистов над одним приложением посвящена *глава 17*.

В *главе 18* рассматривается создание инсталлятора, который позволяет устанавливать разработанное вами приложение на компьютере пользователя.

Новые возможности и усовершенствования, включенные в версию Visual Basic 2005, кратко описаны в *главе 19*.

Специальные элементы книги

В книге есть много особых, специальным образом выделенных вставок. В них содержится дополнительная информация, облегчающая чтение и поиск информации.

Замечание

В замечаниях речь идет о последствиях, к которым приводят те или иные действия.

Совет

В советах рассказывается о некоторых хитростях, которые следует знать, чтобы наиболее эффективно использовать возможности Visual Basic.

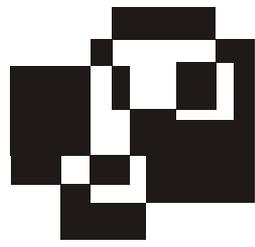
Предупреждение

Предостережения должны помочь вам избежать проблем. В них сказано, чего следует опасаться, а также что нужно делать, чтобы избежать ошибок.

В книге используются различные виды шрифта:

- новые термины выделены *курсивом* и *разреженным интервалом* между символами;
- тот текст, который вы должны вводить с клавиатуры, выделен ***полужирным курсивом***;
- команды меню, наименования кнопок, вкладок, опций, флажков, диалоговых окон, областей и т. п. выделены **полужирным шрифтом**;
- две клавиши, соединенные знаком плюс, — это комбинация клавиш; нажмите первую клавишу и, не отпуская ее, нажмите вторую, а затем отпустите обе;
- названия функций, свойств, методов, баз данных, таблиц, полей таблиц выделены **моноширинным шрифтом**.

При работе с книгой читателю необходимо иметь в виду, что рядом с наименованиями команд меню, диалоговых окон и располагаемых в них элементах даны переводные эквиваленты, а не названия локальной версии.



Глава 1

Первое знакомство с Visual Basic .NET

Вы приступаете к работе с Visual Basic .NET. Многие из того, с чем вам придется работать (меню, панели инструментов, диалоговые окна), покажутся знакомыми, т. к. оно характерно для среды Windows. Но по сравнению с предыдущими версиями теперь не существует отдельной среды разработки для Visual Basic. В Visual Studio .NET все языки Microsoft используют одну и ту же интегрированную среду разработки (Integrated Development Environment, IDE).

Чтобы понять особенности новой версии Visual Studio, необходимо, прежде всего, познакомиться с платформой .NET Framework.

.NET Framework — это новая вычислительная платформа, которая упрощает разработку приложений в сильно распределенном окружении Интернета. Платформа .NET Framework предлагает следующие возможности:

- ❑ предоставление среды выполнения кода, которая уменьшает конфликты между разными версиями языков, гарантирует безопасное выполнение кода, устраняет проблемы, связанные с переносом сред;
- ❑ работа с различными типами приложений, размещенных как в Интернете, так и на локальном компьютере;
- ❑ установление стандартов для поддержки платформы .NET Framework другими языками;
- ❑ создание непротиворечивой объектно-ориентированной среды программирования, в которой код объекта может храниться и выполняться локально, выполняться локально, но быть распределенным в Интернете или выполняться удаленно.

Платформа .NET Framework состоит из двух частей: *единой среды исполнения* (Common Language Runtime, CLR) и *библиотеки классов*. Единая среда исполнения управляет кодом во время его выполнения, обеспечивая управление памятью, потоком и удаленное управление и гарантируя безопасность. Код, генерируемый CLR, называется *управляемым кодом*, а код,

не использующий возможности CLR, называется неуправляемым. Библиотека классов является всесторонней, объектно-ориентированной коллекцией типов, которую можно использовать для разработки приложений, начиная с традиционных приложений с командной строкой и с использованием графического пользовательского интерфейса и заканчивая приложениями, использующими Web-формы и XML Web-сервисы.

Запуск Visual Studio .NET

Для запуска программы из главного меню Windows выполните следующие действия:

1. Нажмите кнопку **Пуск**, расположенную в нижней части экрана.
2. В открывшемся главном меню Windows выберите команду **Программы**. Появится меню данной команды.
3. Выберите в меню команду **Microsoft Visual Studio .NET 2003**.
4. В появившемся подменю выберите опцию **Microsoft Visual Studio .NET 2003**. На экране появится главное окно Visual Studio .NET (рис. 1.1).

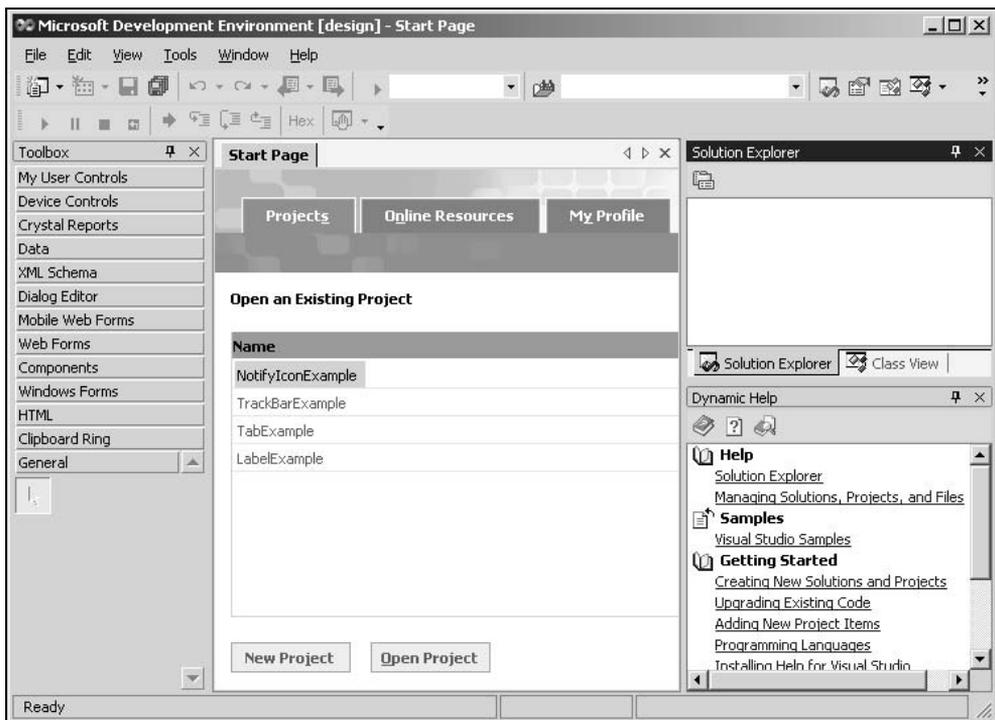


Рис. 1.1. Главное окно Visual Studio .NET

Замечание

При запуске программы может открываться пустая рабочая среда, диалоговые окна **Open Project** (Открыть проект) и **New Project** (Новый проект), страница **Start Page** (как показано на рис. 1.1) или запускаться последнее решение. Для настройки этого параметра используется вкладка **My Profile** (Мой профиль) окна **Start Page** (Начальная страница).

Совет

Для более быстрого запуска программы Visual Studio .NET можно создать на рабочем столе ярлык и назначить ему клавиши быстрого вызова. Тогда будет достаточно нажать заданную комбинацию клавиш для запуска программы. Кроме того, для удобства запуска приложения можно использовать панель **Быстрый запуск** системы Windows.

Главное окно Visual Studio .NET

На рис. 1.1 показано главное окно Visual Studio .NET, каким оно выглядит после запуска программы.

В нем можно выделить несколько основных объектов: стандартная панель инструментов, окна **Start Page** (Начальная страница), **Solution Explorer** (Обозреватель решений), **Class View** (Классы), **Toolbox** (Инструментарий) и **Dynamic Help** (Динамическая справка).

Visual Studio .NET предоставляет в распоряжение пользователя набор самых разнообразных *панелей инструментов* для отладки и запуска программ, создания баз данных и многого другого. Эти панели инструментов содержат кнопки, назначение которых зависит от функций конкретной панели инструментов. После запуска Visual Studio .NET на экране отображается *стандартная панель инструментов*.

Диалоговое окно **Start Page** (Начальная страница) позволяет открывать недавно использовавшиеся проекты, осуществляет поиск примеров как из справочной системы, так и Интернета, содержит различные ссылки на сайты, которые могут помочь при работе с Visual Studio .NET, а также позволяет настроить интегрированную среду для большего удобства при работе.

В отличие от предыдущей версии, где проекты и их файлы отображались в окне **Project Explorer** (Обозреватель проекта), в Visual Studio .NET с этой целью используется окно **Solution Explorer** (Обозреватель решений). Проекты и файлы входят в состав *решения*, а не *группы*, как было ранее. В окне **Solution Explorer** (Обозреватель решений) размещаются проекты решения со своими файлами, а также файлы, не относящиеся ни к одному из проектов, но принадлежащие только решению.

Окно **Class View** (Классы) содержит перечень классов текущего проекта, а также различные методы, свойства, события объектов текущего проекта.

Для получения более подробной информации об объектах используется диалоговое окно **Object Browser** (Просмотр объектов). Оно позволяет искать и исследовать элементы, их свойства, методы, события, находящиеся в проектах и ссылках на них, как бы представляя собой внутреннюю библиотеку.

Новым в среде Visual Studio является появление окна **Dynamic Help** (Динамическая справка). При работе окно динамической справки постоянно обновляется и предлагается справочная информация, относящаяся к текущим действиям. Также справка теперь может отображаться как в виде отдельного окна, так и внутри рабочей области.

К средству, призванному облегчить разработку, относится также окно **Toolbox** (Инструментарий), отображающее элементы, используемые в проектах Visual Studio .NET. К таким элементам могут относиться компоненты .NET и COM, HTML-объекты, фрагменты кода и текст.

В главном диалоговом окне Visual Studio .NET могут отображаться и другие окна, но они будут рассмотрены позднее.

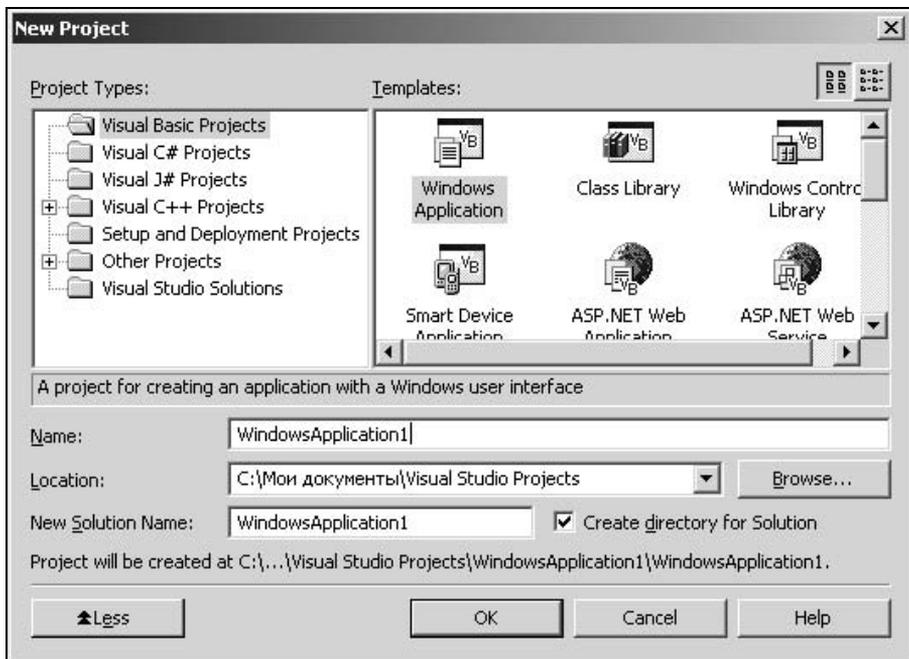
Создание нового проекта

Для создания нового проекта используется диалоговое окно **New Project** (Новый проект) (рис. 1.2). Для его открытия выполните одно из следующих действий:

- на вкладке **Projects** (Проекты) страницы **Start Page** (Начальная страница) нажмите кнопку **New Project** (Новый проект);
- из меню **File** (Файл) выберите пункт **New** (Новый) и в раскрывшемся меню — команду **Project** (Проект);
- нажмите кнопку **New Project** (Новый проект)  стандартной панели инструментов.

Visual Studio .NET предлагает создание приложений на различных языках. Остановимся на проектах, использующих язык Visual Basic.

- Windows Application** (Windows-приложение). Предназначен для создания традиционного Windows-приложения и приложения клиент/сервер, пользовательский интерфейс для которых проектируется с помощью форм Windows. Для формы можно задавать определенные характеристики и располагать на ней различные элементы управления. Этот тип проекта соответствует стандартному приложению **Standard EXE** в Visual Basic 6.0.
- Class Library** (Библиотека классов). Создает класс или компонент, который может использоваться в других приложениях. Является аналогом ActiveX DLL и ActiveX EXE проектов в Visual Basic 6.0. Этот тип проекта не предполагает создания форм.

Рис. 1.2. Диалоговое окно **New Project**

- ❑ **Windows Control Library** (Библиотека элементов управления Windows). Является аналогом ActiveX Controls в Visual Basic 6.0. Позволяет создавать индивидуальные средства управления для форм Windows.
- ❑ **Smart Device Application** (Приложение для интеллектуальных устройств). Visual Studio .NET предлагает программную модель разработки приложений для интеллектуальных устройств. Используя дополнительные инструменты и компоненты платформы .NET Framework, можно создать, запустить, отладить приложения, работающие на платформе .NET Compact Framework (платформа .NET Framework, предназначенная для устройств) в карманных компьютерах, мобильных телефонах и других устройствах, ограниченных по ресурсам.
- ❑ **ASP.NET Web Application** (Web-приложение для ASP.NET). Позволяет создать динамическое Web-приложение для ASP.NET, которое может включать HTML-страницы, компоненты Web, использующие язык XML, COM-компоненты и код на языке VB.NET.
- ❑ **ASP.NET Web Service** (Web-сервис для ASP.NET). Данный тип создает Web-сервис, который может быть доступен другим Web-сервисам и приложениям через Интернет. Благодаря стандартным протоколам Web-сервис может использоваться при различных реализациях, на любых платформах и с разными устройствами.

- ❑ **ASP.NET Mobile Web Application** (Изменяемое Web-приложение для ASP.NET). Предназначен для разработки Web-страниц, которые смогут отображаться на экранах мобильных телефонов, а также для создания приложений, которые будут работать на самих устройствах.
- ❑ **Web Control Library** (Библиотека элементов управления Web). Позволяет создавать собственные элементы управления для использования в Web-приложениях.
- ❑ **Console Application** (Консольное приложение). Создает приложения, которые, как правило, не поддерживают пользовательские графические интерфейсы, и представляет собой отдельный исполняемый файл.
- ❑ **Windows Service** (Сервис Windows). Создает сервисы Windows, более известные как NT-сервисы, способные формировать исполняемые приложения. Эти сервисы могут автоматически запускаться при загрузке компьютера, быть приостановлены и перезапущены. Они не имеют пользовательских интерфейсов, а информация выводится в файл протокола. Данные сервисы идеальны для серверов, которые требуют длительной работы и не взаимодействуют с другими пользователями, работающими на том же компьютере.
- ❑ **Empty Project** (Пустой проект). Создает пустой проект, содержащий только необходимую для хранения приложения структуру файлов. Какие-либо ссылки, компоненты и файлы нужно добавлять вручную.
- ❑ **Empty Web Project** (Пустой Web-проект). Предназначен для опытных пользователей, которые хотят начать программировать с пустой среды и добавлять собственные выполняемые функции.
- ❑ **New Project in Existing Folder** (Новый проект в существующей папке). Размещает новый проект в папке существующего приложения, чтобы использовать файлы этого проекта при программировании. Данный шаблон может применяться для локальных и Web-проектов.

Замечание

В отличие от предыдущей версии, любые созданные файлы сохраняются на диске, даже если просто создать проект и выйти без сохранения. Файлы сохраняются в момент создания проекта.

Главное меню

При работе с Visual Studio .NET можно использовать как кнопки панели инструментов, так и строку меню, расположенную в верхней части экрана, все команды которого являются иерархическими. При выборе определенной команды открывается ее подменю. Главное меню может содержать следующие пункты: **File** (Файл), **Edit** (Правка), **View** (Вид), **Project** (Проект), **Build**

(Сборка), **Debug** (Отладка), **Data** (Данные), **Format** (Формат), **Tools** (Сервис), **Window** (Окно) и **Help** (Справка).

Меню *File*

Меню **File** (Файл) содержит команды, связанные с доступом к файлам. Эти команды позволяют создавать новые файлы, проекты и решения, открывать существующие, закрывать, сохранять и печатать их. Также можно добавлять новые и существующие проекты и элементы (формы, классы, файлы и т. д.) в решение.

В меню включено также подменю **Source Control** (Управление версиями), предназначенное для управления версиями проектов и организации групповой разработки.

Команда **Advanced Save Options** (Дополнительные параметры сохранения) позволяет определить параметры кодирования языка и выбрать операционную систему для обеспечения совместимого форматирования данных на различных компьютерах.

В нижней части меню располагаются имена последних открываемых файлов и проектов, которые предоставляют возможность быстрого открытия любого из них. Последней командой меню **File** (Файл) является команда **Exit** (Выход), которая предназначена для выхода из Visual Studio .NET.

Меню *Edit*

Меню **Edit** (Правка) имеется во многих приложениях Windows. В Visual Studio .NET команды этого меню используются при создании форм, редактировании программ.

Данное меню содержит стандартные команды редактирования: отменить предыдущую команду, вернуть предыдущую команду, вставить, вырезать фрагмент текста, выделить весь текст, удалить выделенное, а также команды поиска и замены фрагмента текста и перемещение к определенной строке.

Команда **Bookmarks** (Закладки) данного меню дает возможность вставлять закладки в текст программы.

С помощью команд пункта **IntelliSense** меню **Edit** (Правка) можно просмотреть список членов определенного класса, структуры, объединения или пространства имен и вставить в код подходящий, получить информацию о числе, типах и именах параметров методов и свойств, дописать слова, являющиеся именами методов, команд.

Команды пункта **Outlining** (Скрытие) позволяют скрыть часть кода программы с помощью символа , отменить использование этого символа, раскрыть весь код программы. Нажав на плюс рядом с указанным символом, можно просмотреть скрытый код.

В данном меню также содержится команда **Advanced** (Дополнительные), которая позволяет выделять текст как комментарий, показать непечатаемые символы, установить перенос слов, сделать все выделенные символы строчными или прописными и т. д.

Меню *View*

Данный пункт меню содержит команды вызова окон среды Visual Studio .NET. С помощью этих команд могут открываться окна редактора программного кода, конструктора формы, свойств объектов, обозревателя решений и другие окна.

Команда **Toolbars** (Панели инструментов) позволяет управлять отображением всевозможных панелей инструментов.

Внизу меню **View** (Вид) располагается команда **Property Pages** (Страницы свойств), открывающая окно для определения общих настроек текущего проекта или решения.

Меню *Project*

Меню **Project** (Проект) содержит команды, позволяющие добавлять в проект и удалять из него элементы проекта, такие как форма, программный модуль, класс, а также дающие возможность добавлять ссылки на подключаемые библиотеки.

С помощью команды **Set StartUp Projects** (Задать запускаемые проекты) этого меню можно задать запускаемый проект данного решения и определить, окажется ли такой проект единственным или их будет несколько.

Последней командой меню **Project** (Проект) является команда **Properties** (Свойства), позволяющая открыть окно свойств проекта или решения.

Меню *Build*

Меню **Build** (Сборка) содержит команды, помогающие скомпоновать решение или проект.

Меню *Debug*

В меню **Debug** (Отладка) расположены команды, предназначенные для отладки и запуска приложения. С помощью команд этого меню можно запустить приложение на выполнение, установить точки останова программы, осуществить пошаговое выполнение приложения, открыть специальные окна для отладки.

Меню *Format*

Этот пункт меню доступен при работе в конструкторе форм. Меню **Format** (Формат) содержит команды, управляющие выравниванием текста и объектов, заданием размеров объектов и определением интервалов между ними. Однако при работе с различными конструкторами становятся доступными и дополнительные команды.

Внизу этого меню располагается команда **Lock Controls** (Блокировка элементов управления), которая позволяет сделать недоступным перемещение элементов управления и сохранить их размер. С ее помощью фиксируются все элементы управления, включая саму форму.

Меню *Tools*

Меню **Tools** (Сервис) содержит средства для настройки среды разработки, создания макросов, а также команды запуска дополнительных утилит.

Меню *Window*

В меню **Window** (Окно) располагаются команды, которые управляют открытыми на экране окнами. С помощью этих команд можно упорядочивать, скрывать окна и переходить из одного окна в другое. Кроме того, команды данного меню позволяют активизировать любое открытое окно.

Меню *Help*

Help (Справка) — последняя команда меню главного окна. Используя команды данного меню, можно вызвать справочную систему с различными вариантами представления информации, открыть окна **Dynamic Help** (Динамическая справка) и **Start Page** (Начальная страница), а также получить сведения о системе.

Стандартная панель инструментов

В Visual Studio .NET содержится большое количество панелей инструментов для отладки и запуска программ, задания расположения элементов на форме и многого другого. Познакомимся со стандартной панелью инструментов (рис. 1.3), которая используется во всех режимах работы.



Рис. 1.3. Стандартная панель инструментов

Назначение кнопок стандартной панели инструментов описано в табл. 1.1.

Таблица 1.1. Назначение кнопок стандартной панели инструментов

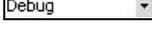
Кнопка	Название	Назначение
	New Project (Новый проект)	Создает новый проект или пустое решение
	Add New Item (Добавить новый элемент)	Позволяет добавить в проект новый или существующий элемент, форму, модуль, класс, компонент или элемент управления
	Open File (Открыть файл)	Открывает существующий файл
	Save (Сохранить)	Сохраняет открытый файл
	Save All (Сохранить все)	Сохраняет все открытые файлы
	Cut (Вырезать)	Удаляет выделенный текст или выделенные объекты и помещает их в буфер
	Copy (Копировать)	Копирует в буфер выделенный текст или выделенные объекты, не удаляя их
	Paste (Вставить)	Вставляет содержимое буфера
	Undo (Отменить)	Отменяет выполненные действия
	Redo (Восстановить)	Восстанавливает отмененные действия
	Navigate Backward (Переместиться назад)	Перемещает курсор в позицию, в которой он находился ранее
	Navigate Forward (Переместиться вперед)	Перемещает курсор в последующую позицию
	Start (Запустить)	Запускает программу на выполнение
	Solution Configurations (Конфигурация решения)	Определяет конфигурацию программы: отладочная или окончательная версия. Можно создать собственную
	Find in Files (Найти в файлах)	Осуществляет поиск информации по контексту
	Solution Explorer (Обозреватель решения)	Открывает окно обозревателя решений
	Properties Window (Окно свойств)	Открывает окно Properties (Свойства), используемое для настройки свойств решения, проекта, файлов и их элементов

Таблица 1.1 (окончание)

Кнопка	Название	Назначение
	Object Browser (Обзор объектов)	Открывает окно Object Browser (Просмотр объектов), позволяющее просматривать классы, свойства, методы, события и константы выбранных библиотек
	Toolbox (Инструментарий)	Открывает окно Toolbox (Инструментарий), содержащее элементы, используемые разработчиком при создании приложения
	Class View (Классы)	Открывает окно Class View (Классы), содержащее перечень классов, методов свойств и событий текущего проекта

Замечание

Если по внешнему виду кнопки вы не можете определить ее назначение, установите курсор мыши на кнопку. При этом под курсором появится всплывающая подсказка с ее наименованием.

По умолчанию в главном окне программы Visual Studio .NET всегда присутствует стандартная панель инструментов, если только вы не удалили ее с экрана. Если для работы необходима стандартная панель инструментов, выберите в подменю **Toolbars** (Панели инструментов) меню **View** (Вид) команду **Standard** (Стандартная).

После установки Visual Studio .NET стандартная панель инструментов размещается в верхней части главного окна, но она, как и все остальные панели инструментов, может перемещаться в любое место экрана. Для этого установите курсор мыши в любое свободное от кнопок место на панели инструментов, нажмите кнопку мыши и, не отпуская ее, переместите панель на новое место.

Окно *Start Page*

Окно **Start Page** (Начальная страница) (рис. 1.4) позволяет просмотреть последние использовавшиеся проекты, осуществить поиск примеров программ как из справки, так и Интернета, настроить интегрированную среду разработки, а также просмотреть ссылки на сайты, содержащие новости о продукте Visual Studio, документацию, учебные пособия.

Начальная страница автоматически открывается при запуске Visual Studio .NET. Если же окно **Start Page** (Начальная страница) не появилось, его можно вызвать с помощью команды **Show Start Page** (Отобразить начальную страницу) меню **Help** (Справка).