



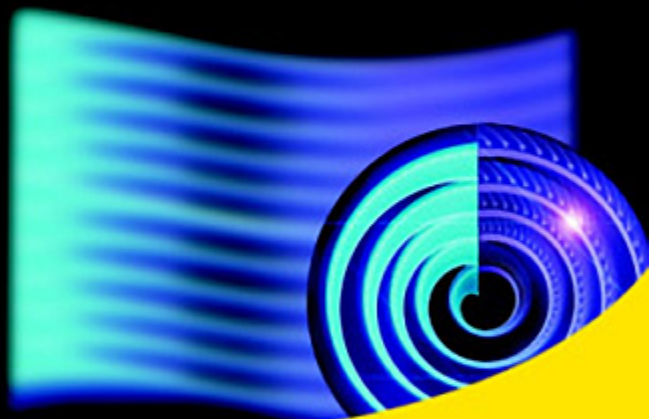
Самоучитель

Никита Культин

Основы программирования

Microsoft®

В Visual C++ 2010



Среда и технология разработки

Использование базовых компонентов

Программирование графики

Работа с Microsoft Access и Microsoft SQL
Server Compact Edition

Создание справочной системы и установщика

УДК 681.3.068
ББК 32.973.26-018.1
К90

Культин Н. Б.
К90 Основы программирования в Microsoft® Visual C++ 2010. — СПб.: БХВ-Петербург, 2010. — 384 с.: ил. + CD-ROM — (Самоучитель)
ISBN 978-5-9775-0520-8

Книга является пособием для начинающих по программированию в Microsoft Visual C++ 2010. В ней в доступной форме изложены принципы визуального проектирования и событийного программирования, на примерах показана технология создания программ различного назначения. Приведено описание среды разработки и базовых компонентов. Рассмотрены вопросы программирования графики, разработки программ работы с базами данных Microsoft Access и Microsoft SQL Server Compact Edition. Уделено внимание отладке программ, созданию справочной системы, установке созданной программы на компьютер пользователя. В справочнике приведено описание базовых компонентов и наиболее часто используемых функций. Прилагаемый компакт-диск содержит проекты, рассмотренные в книге.

Для начинающих программистов

УДК 681.3.068
ББК 32.973.26-018.1

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Игорь Шишигин</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Анна Кузьмина</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Елены Беляевой</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 31.05.10.

Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 30,96.

Тираж 2000 экз. Заказ №

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

Санитарно-эпидемиологическое заключение на продукцию
№ 77.99.60.953.Д.005770.05.09 от 26.05.2009 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 978-5-9775-0520-8

© Культин Н. Б., 2010
© Оформление, издательство "БХВ-Петербург", 2010

Оглавление

Предисловие.....	1
ЧАСТЬ I. MICROSOFT VISUAL C++ 2010.....	3
Глава 1. Среда разработки Microsoft Visual C++ 2010.....	5
Установка.....	5
Первый взгляд.....	7
Справочная система.....	13
Глава 2. Первый проект.....	15
Начало работы над проектом.....	15
Форма.....	16
Компоненты.....	20
Событие.....	30
Функция обработки события.....	31
Структура проекта.....	36
Главный модуль.....	37
Модуль формы.....	38
Компиляция.....	44
Ошибки.....	46
Предупреждения.....	47
Запуск программы.....	47
Исключения.....	48
Обработка исключения.....	49
Внесение изменений.....	54
Завершение работы над проектом.....	58
Установка приложения на другой компьютер.....	59
Глава 3. Базовые компоненты.....	61
<i>Label</i>	61
<i>TextBox</i>	64

<i>Button</i>	69
<i>CheckBox</i>	73
<i>RadioButton</i>	77
<i>GroupBox</i>	82
<i>ComboBox</i>	86
<i>PictureBox</i>	91
<i>ListBox</i>	98
<i>ListView</i>	103
<i>ImageList</i>	107
<i>ToolTip</i>	109
<i>Panel</i>	110
<i>CheckedListBox</i>	112
<i>Timer</i>	114
<i>NumericUpDown</i>	117
<i>StatusStrip</i>	122
<i>NotifyIcon</i>	125
<i>ToolStrip</i>	130
<i>MenuStrip</i>	133
<i>OpenFileDialog</i>	136
<i>SaveFileDialog</i>	139

ЧАСТЬ II. ПРАКТИКУМ ПРОГРАММИРОВАНИЯ 149

Глава 4. Графика 151

Графическая поверхность	153
Карандаши и кисти	154
Карандаш	154
Кисть	157
Графические примитивы	162
Линия	164
Ломаная линия	169
Прямоугольник	169
Точка	170
Многоугольник	171
Эллипс и окружность	172
Дуга	173
Сектор	173
Текст	179
Битовые образы	184
Анимация	187

Глава 5. Базы данных 197

База данных и СУБД	197
Локальные и удаленные базы данных	197
Структура базы данных	198

Компоненты доступа к данным	199
Создание базы данных	199
База данных Microsoft Access	200
Доступ к данным	201
Отображение данных	213
Выбор информации из базы данных	218
Работа с базой данных в режиме формы	221
Сервер баз данных Microsoft SQL Server Compact Edition	227
Среда Microsoft SQL Server Management Studio	228
Создание базы данных	228
База данных "Контакты"	232
Развертывание приложения работы с базой данных Microsoft SQL Server Compact Edition	243
Глава 6. Консольное приложение	245
Создание консольного приложения	249
Запуск консольного приложения	254
Глава 7. Отладка программы.....	255
Классификация ошибок	255
Предотвращение и обработка ошибок.....	257
Отладчик	260
Трассировка программы	260
Точки останова программы	261
Наблюдение значений переменных	262
Глава 8. Справочная информация	265
Справочная система HTML Help	265
Подготовка справочной информации.....	266
Основы HTML	267
Microsoft HTML Help Workshop	269
Файл проекта	269
Оглавление	272
Идентификаторы разделов.....	275
Компиляция.....	276
Отображение справочной информации.....	277
Глава 9. Создание установщика	281
Новый проект.....	281
Общая информация	283
Файлы	284
Команда запуска программы	286
Интерфейс	288
Диалоги	288

Построение	291
Установка	291
Глава 10. Примеры программ.....	293
Экзаменатор.....	293
Требования к программе.....	294
Файл теста	294
Форма	297
Доступ к файлу теста.....	298
Текст программы.....	301
Запуск программы	310
Сапер	312
Правила и представление данных	312
Форма	314
Игровое поле.....	316
Начало игры	316
Игра	319
Справочная информация.....	323
Информация о программе.....	324
Текст программы.....	327
Глава 11. Краткий справочник.....	337
Форма	337
Компоненты.....	338
<i>Button</i>	338
<i>ComboBox</i>	340
<i>ContextMenuStrip</i>	340
<i>CheckBox</i>	341
<i>CheckedListBox</i>	342
<i>GroupBox</i>	343
<i>ImageList</i>	344
<i>Label</i>	344
<i>ListBox</i>	345
<i>MenuStrip</i>	346
<i>NotifyIcon</i>	347
<i>NumericUpDown</i>	347
<i>OpenFileDialog</i>	348
<i>Panel</i>	349
<i>PictureBox</i>	349
<i>RadioButton</i>	350
<i>ProgressBar</i>	351
<i>SaveFileDialog</i>	352
<i>TextBox</i>	353
<i>ToolTip</i>	354
<i>Timer</i>	355

Графика	355
Графические примитивы.....	355
Карандаш.....	357
Кисть.....	358
Типы данных.....	360
Целый тип	360
Вещественный тип	360
Символьный и строковый типы	360
Функции	361
Функции преобразования.....	361
Функции манипулирования строками	362
Функции манипулирования датами и временем	364
Функции манипулирования каталогами и файлами	365
Математические функции.....	367
События.....	368
Исключения	369
Приложение. Описание компакт-диска	371
Предметный указатель	373

Visual

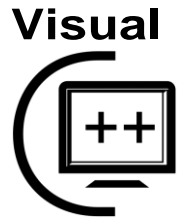


ЧАСТЬ I

Microsoft Visual C++ 2010

В этой части книги приведено краткое описание среды разработки Microsoft Visual C++ 2010; на примере программы "Конвертер" показан процесс разработки приложения Windows Forms; дано описание и приведены примеры использования базовых компонентов.

Глава 1.	Среда разработки Microsoft Visual C++ 2010
Глава 2.	Первый проект
Глава 3.	Базовые компоненты



ГЛАВА 1

Среда разработки Microsoft Visual C++ 2010

Установка

Среда разработки Microsoft Visual Studio 2010, в которую входит Microsoft Visual C++, доступна в трех вариантах: Professional, Premium и Ultimate. Каждый комплект представляет собой набор инструментов, обеспечивающих разработку высокоэффективных приложений для Win32- и .NET-платформ, Web-приложений. Различие вариантов состоит в составе компонентов — чем выше уровень пакета (от Professional к Ultimate), тем больше возможностей и инструментов поддержки процесса разработки он предоставляет программисту. Так, например, в Ultimate включены инструменты UML-моделирования и анализа, в версии Premium есть только возможность просмотра UML-диаграмм, а в версии Professional работа с UML-диаграммами не поддерживается.

Microsoft Visual Studio 2010 может работать в среде операционных систем Microsoft:

- Windows XP (x86) Service Pack 3 (кроме Starter Edition);
- Windows XP (x64) Service Pack 2 (кроме Starter Edition);
- Windows Vista (x86, x64) Service Pack 1 (кроме Starter Edition);
- Windows 7 (x86, x64);
- Windows Server 2003 (x86, x64) Service Pack 2;
- Windows Server 2003 R2 (x86, x64);
- Windows Server 2008 (x86, x64) Service Pack 2;
- Windows Server 2008 R2 (x64).

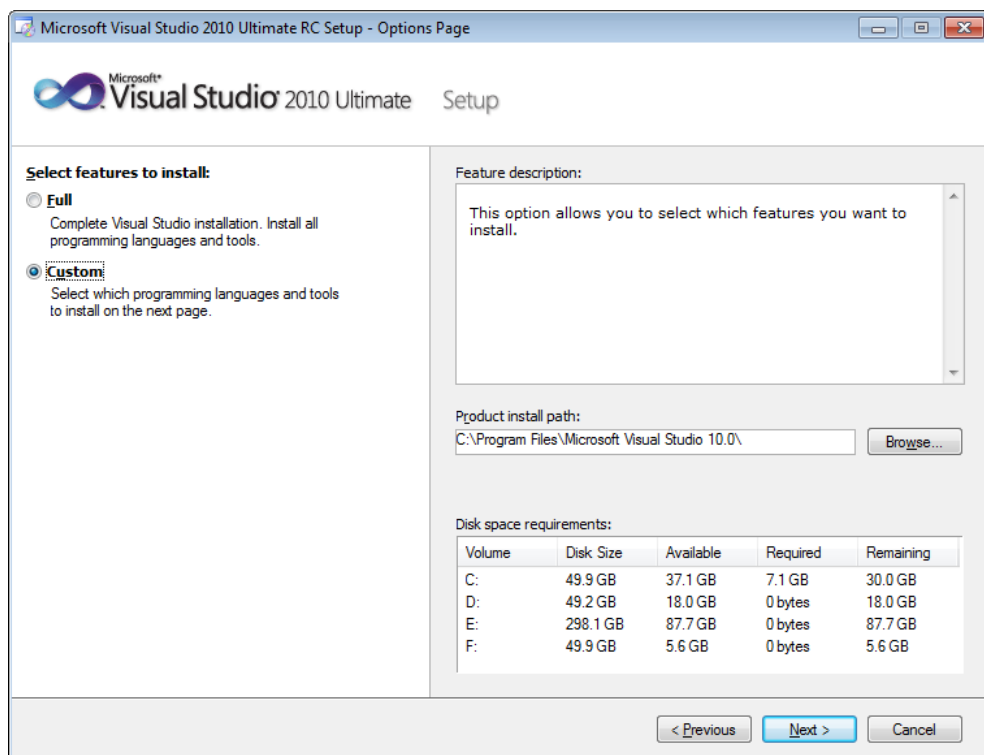
Особых требований, по современным меркам, к ресурсам компьютера Microsoft Visual Studio не предъявляет: процессор должен быть с частотой не

ниже 1,6 ГГц, 1 Гбайт оперативной памяти (2 Гбайт для x64-процессора), порядка 7 Гбайт свободного места на жестком диске (для установки Visual C++ достаточно чуть больше 3 Гбайт).

Установка выполняется с DVD-диска, на котором помимо Visual Studio находятся компоненты, необходимые для установки и работы среды.

Процесс установки активизируется автоматически, после того как установочный диск будет помещен в дисковод. Если автоматический запуск программ запрещен, то надо вручную запустить установщик (файл setup.exe), который находится в корне установочного диска.

Установщик проверяет версию Windows, затем — Microsoft .NET Framework. Если версия операционной системы не соответствует необходимым требованиям, процесс установки прерывается. В случае если версия .NET Framework ниже 4, то на компьютер устанавливаются компоненты Microsoft .NET Framework 4. После этого непосредственно начинается установка Microsoft Visual Studio.



Сначала на экране появляется окно лицензионного соглашения, затем установщик предлагает выбрать вариант установки: **Full** (полный) или **Custom** (частичный). Выбрав вариант установки **Custom** (рис. 1.1), в следующем окне (рис. 1.2) программист может выбрать компоненты Visual Studio, которые надо установить (точнее, отказаться от установки ненужных).

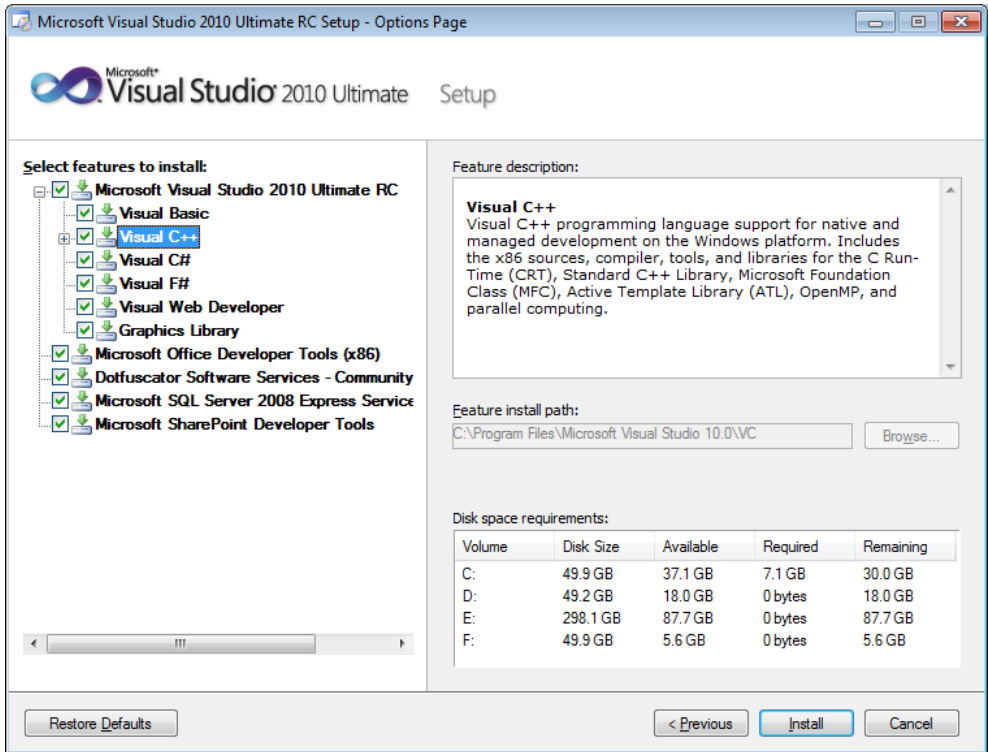


Рис. 1.2. Установка Visual C++: выбор устанавливаемых компонентов

Следует обратить внимание на то, что во время инсталляции Visual Studio справочная система на компьютер не устанавливается. Установка справочной системы выполняется из среды разработки (о том, как это сделать, см. в разд. "Справочная система" далее в этой главе).

Первый взгляд

Чтобы начать работу в Microsoft Visual C++, надо запустить Microsoft Studio — сделать щелчок на кнопке **Пуск** и в меню **Все программы** ▶ **Microsoft Visual Studio 2010** выбрать **Microsoft Visual Studio 2010**.

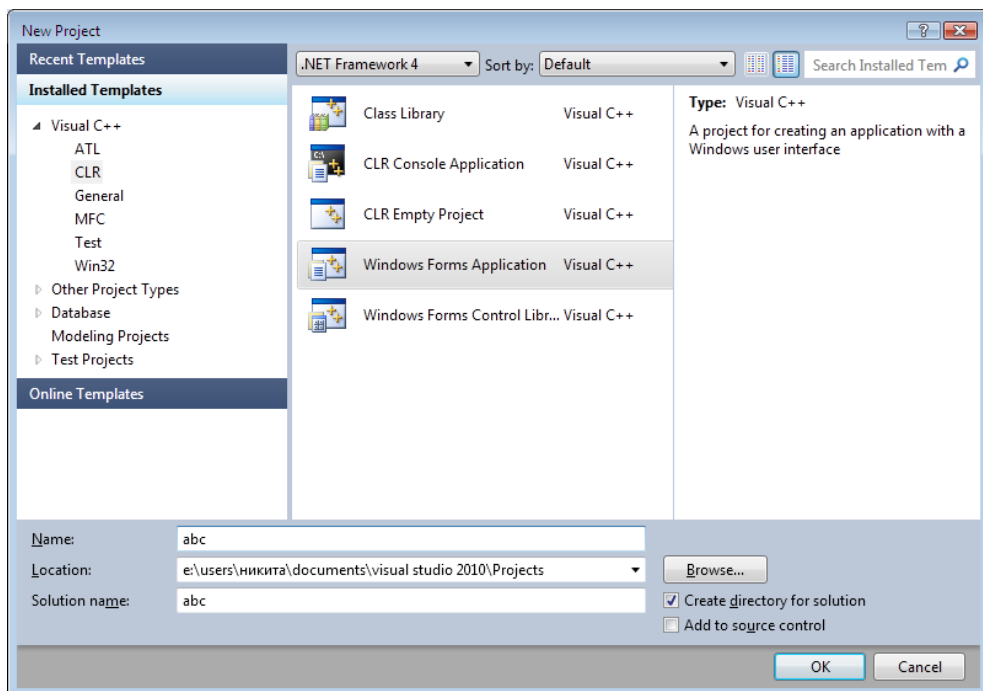


Рис. 1.3. В окне **New Project** надо выбрать язык разработки, тип приложения и задать имя проекта

Для того чтобы приступить к созданию программы в Visual C++ или, как принято говорить, начать работу над *проектом*, надо:

1. В меню **File** окна Microsoft Visual Studio выбрать команду **New ► Project**.
2. В открывшемся окне **New Project** (рис. 1.3) раскрыть список **Visual C++** и выбрать тип приложения — **CLR**.
3. Выбрать вид приложения — **Windows Forms Application Visual C++**.
4. В поле **Name** ввести имя проекта и нажать кнопку **OK**.

В результате описанных действий будет создан *проект* Visual Studio — совокупность файлов, необходимых для создания программы. Проект создастся в папке проектов Visual Studio 2010 (по умолчанию это C:\Users\User\Documents\Visual Studio 2010\Projects, где *User* — имя пользователя в системе), в новой папке (обратите внимание на установленный флажок **Create directory for solution** — создать каталог для решения). Имя папки проекта определяет содержимое поля **Name** (имя проекта).

Окно среды Microsoft Visual Studio 2010 в начале работы над новым проектом приведено на рис. 1.4. В заголовке окна отображается имя проекта, над которым в данный момент идет работа.

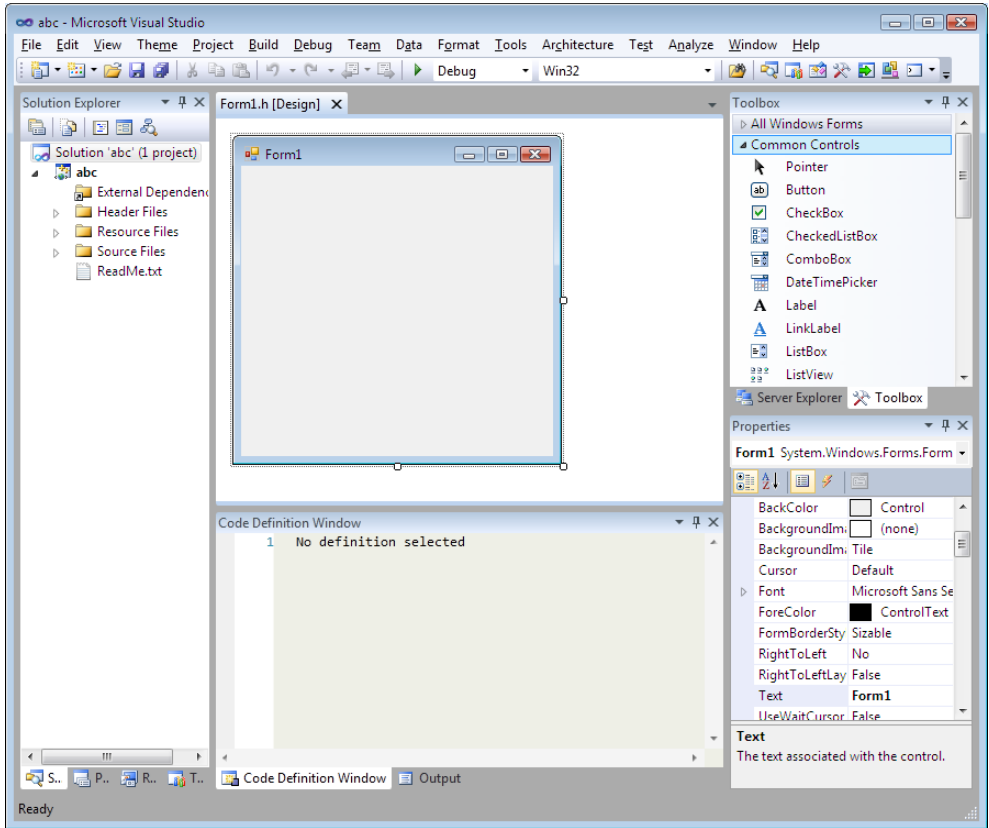


Рис. 1.4. Окно среды разработки Visual C++ в начале работы над новым проектом Windows Forms

В верхней части окна находится строка меню и область отображения панелей инструментов. По умолчанию в области отображения панелей инструментов выводится панель **Standard** (рис. 1.5). Чтобы сделать доступными другие панели инструментов, надо выбрать команду **View ► Toolbars** и в раскрывшемся списке сделать щелчок на имени нужной панели.

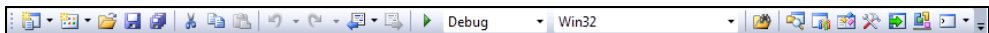


Рис. 1.5. Панель **Standard**

Центральную часть окна Visual C++ занимает окно конструктора (Designer) формы (рис. 1.6). В нем находится *форма* — заготовка окна приложения (окно программы во время его разработки принято называть формой). В окне дизайнера формы в графической форме отображаются инструкции програм-

мы (обратите внимание, в заголовке окна указано имя h-файла), обеспечивающие создание окна. Чтобы их увидеть (рис. 1.7), выберите в меню **View** команду **Code**.

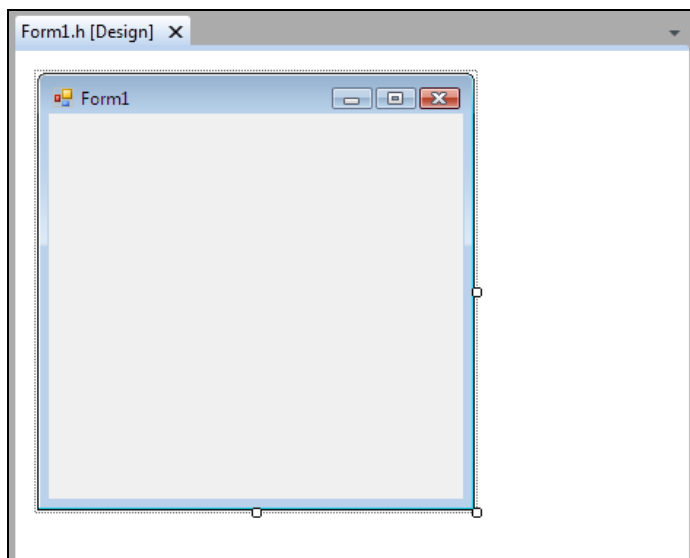


Рис. 1.6. Окно конструктора формы

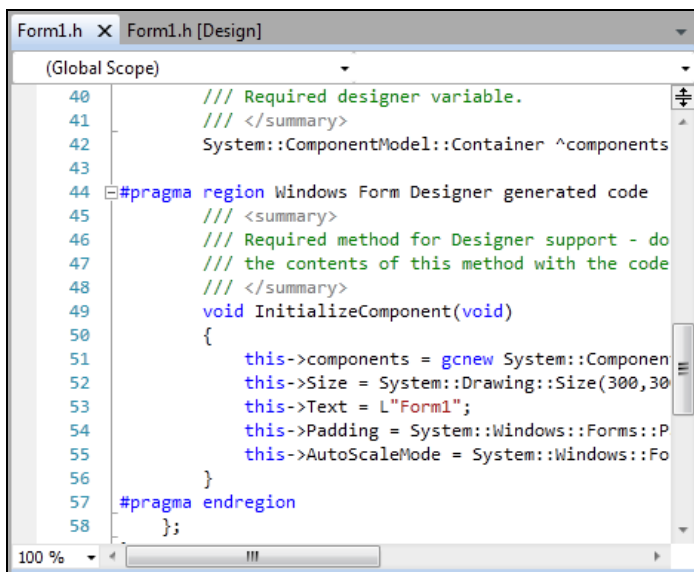
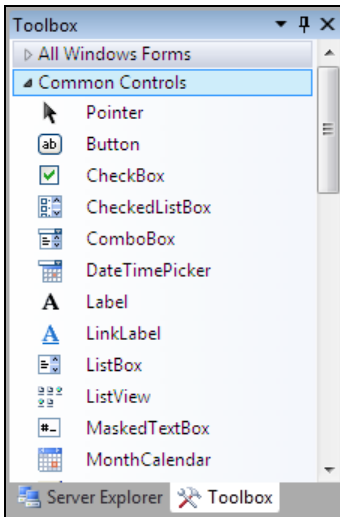
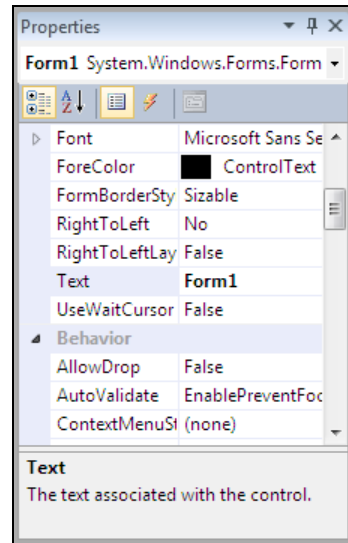


Рис. 1.7. Инструкции, обеспечивающие создание формы, находятся в секции Windows Form Designer generated code

В правой части главного окна Visual Studio обычно отображаются окна **Toolbox** и **Properties**.

В окне **Toolbox** (рис. 1.8) находятся *компоненты*, которые можно поместить (перетащить) на форму. Компонент — это объект, реализующий некоторую функциональность. Например, в группе **Common Controls** находятся компоненты, реализующие пользовательский интерфейс (Label — область отображения текста; TextBox — поле ввода/редактирования текста; Button — командная кнопка), а в группе **Data** — компоненты доступа к базам данных.

В окне **Properties** (рис. 1.9) отображаются *свойства* выбранного (выделенного) в данный момент объекта — компонента или, если ни один из элементов не выделен, самой формы. Обратите внимание, имя и тип объекта отображаются в верхней части окна **Properties**.

Рис. 1.8. Окно **Toolbox**Рис. 1.9. Окно **Properties**

Окно **Properties** используется для редактирования значений свойств *объектов* и, как следствие, изменения их вида и поведения. Например, чтобы изменить текст в заголовке формы, надо изменить значение свойства `Text`.

При отображении свойства могут быть объединены в группы по функциональному признаку (**Categorized**) или упорядочены по алфавиту (**Alphabetical**). Чтобы изменить способ отображения, надо сделать щелчок на соответствующей кнопке, находящейся в верхней части окна **Properties** (рис. 1.10).

Сделав в окне **Properties** щелчок на кнопке **Events** (рис. 1.11), можно увидеть *события* (рис. 1.12), которые способен воспринимать выбранный объект

(компонент или форма). Событие — это то, что происходит во время работы программы. Например, командная кнопка может реагировать на щелчок кнопкой мыши — событие `Click`.

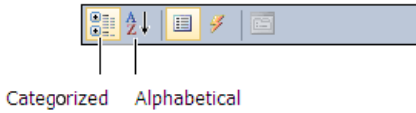


Рис. 1.10. Кнопки управления способом группировки свойств

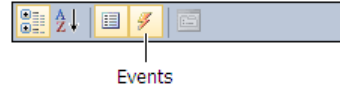


Рис. 1.11. Чтобы увидеть события, которые способен воспринимать объект, нажмите кнопку **Events**

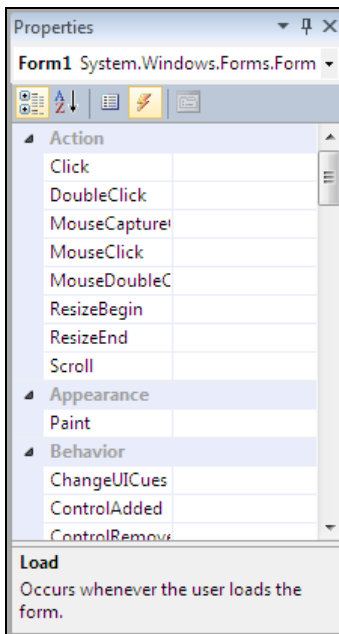


Рис. 1.12. События, которые может воспринимать объект (в данном случае — форма)

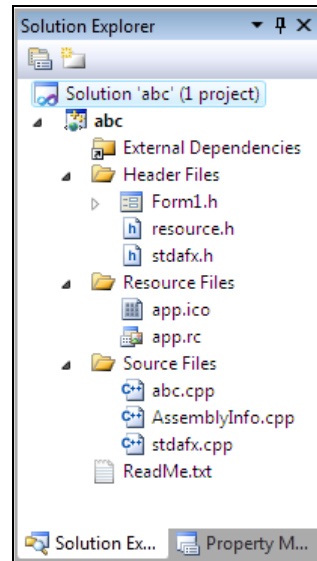


Рис. 1.13. В окне **Solution Explorer** отображается структура проекта

В окне **Solution Explorer** (рис. 1.13) отображается структура проекта. В нем перечислены файлы, образующие проект. Функция `main` находится в главном файле проекта (в рассматриваемом примере это `abc.cpp`), описание класса формы — в `h`-файле (в рассматриваемом примере — в файле `Form1.h`). Здесь же отображаются имена файлов ресурсов (`app.rc`) и значка (`app.ico`) проекта.

Окно **Solution Explorer** удобно использовать для быстрого доступа к нужному элементу проекта.

Справочная система

Сразу после установки Visual Studio справочная информация недоступна. Чтобы получить возможность доступа к ней, надо выполнить настройку справочной системы.

Visual Studio позволяет программисту выбрать источник справочной информации — локальная справочная информация (находящаяся на компьютере разработчика) или online (предоставляемая Microsoft через Интернет).

Чтобы справочная информация стала доступной, надо:

1. В меню **Help** выбрать команду **Manage Help Settings**.
2. В окне **Manage Help Settings** сделать щелчок на ссылке **Chose online or local help**.
3. В окне **Settings** выбрать вариант **I want to use local help** (использовать локальную справочную информацию) и нажать кнопку **OK**.
4. В окне **Manage Help Settings** сделать щелчок на ссылке **Find content on disk**.
5. В появившемся окне **Find content on disk** сделать щелчок на кнопке **Browse**, открыть находящийся на установочном диске, в папке ProductDocumentation, файл контента справочной информации (Help-ContentSetup.msha) и выбрать разделы справочной информации, которые надо скопировать с установочного диска на компьютер.

ГЛАВА 2

Первый проект

Процесс разработки программы в Microsoft Visual C++ рассмотрим на примере — создадим *приложение* (программы, предназначенные для решения прикладных задач, принято называть приложениями), позволяющее посчитать доход по вкладу (рис. 2.1).

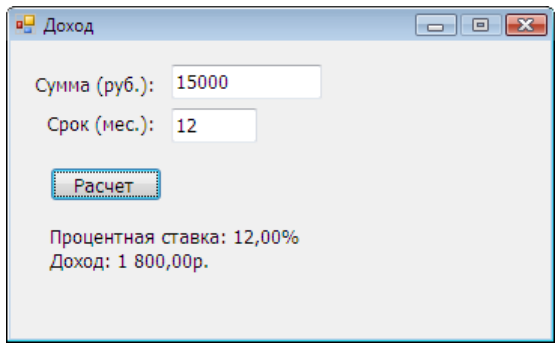


Рис. 2.1. Окно программы "Доход"

Начало работы над проектом

Чтобы начать работу над новым приложением (проектом), надо:

1. В меню **File** выбрать команду **New ▶ Project**.
2. В открывшемся окне **New Project** в списке **Visual C++** выбрать сначала тип приложения — **CLR**, а затем вид приложения — **Windows Forms Application**.
3. В поле **Name** ввести имя проекта — **profit** и нажать кнопку **OK**.

Форма

Работа над приложением начинается с создания стартовой *формы* — главного окна программы.

Сначала нужно установить требуемые значения свойств формы, затем — поместить на форму необходимые *компоненты* (поля ввода информации, командные кнопки, поля отображения текста и др.) и выполнить.

Настройка формы (а также компонентов) осуществляется путем изменения значений *свойств*. Свойства *объекта* (формы, компонента) определяют его вид и поведение. Например, свойство `Text` определяет текст заголовка окна, а свойство `StartPosition` — положение окна в момент появления его на экране.

Основные свойства формы (объекта `WinForm`) приведены в табл. 2.1.

Таблица 2.1. Свойства формы (объекта `WinForm`)

Свойство	Описание
<code>Name</code>	Имя формы
<code>Text</code>	Текст в заголовке
<code>Size</code>	Размер формы. Уточняющее свойство <code>Width</code> определяет ширину, свойство <code>Height</code> — высоту
<code>StartPosition</code>	Положение формы в момент первого появления на экране. Форма может находиться в центре экрана (<code>CenterScreen</code>), в центре родительского окна (<code>CenterParent</code>). Если значение свойства равно <code>Manual</code> , то положение формы определяется значением свойства <code>Location</code>
<code>Location</code>	Положение формы на экране. Расстояние от верхней границы формы до верхней границы экрана задает уточняющее свойство <code>Y</code> , расстояние от левой границы формы до левой границы экрана — уточняющее свойство <code>X</code>
<code>FormBorderStyle</code>	Тип формы (границы). Форма может представлять собой обычное окно (<code>Sizable</code>), окно фиксированного размера (<code>FixedSingle</code> , <code>Fixed3D</code>), диалог (<code>FixedDialog</code>) или окно без кнопок Свернуть и Развернуть (<code>SizeableToolWindow</code> , <code>FixedToolWindow</code>). Если свойству присвоить значение <code>None</code> , у окна не будет заголовка и границы
<code>ControlBox</code>	Управляет отображением системного меню и кнопок управления окном. Если значение свойства равно <code>False</code> , то в заголовке окна кнопка системного меню, а также кнопки Свернуть , Развернуть , Заккрыть не отображаются
<code>MaximizeBox</code>	Кнопка Развернуть . Если значение свойства равно <code>False</code> , то находящаяся в заголовке окна кнопка Развернуть недоступна
<code>MinimizeBox</code>	Кнопка Свернуть . Если значение свойства равно <code>False</code> , то находящаяся в заголовке окна кнопка Свернуть недоступна

Таблица 2.1 (окончание)

Свойство	Описание
Icon	Значок в заголовке окна
Font	Шрифт, используемый по умолчанию компонентами, находящимися на поверхности формы. Изменение значения свойства приводит к автоматическому изменению значения свойства Font всех компонентов формы (при условии, что значение свойства компонента не было задано явно)
ForeColor	Цвет, наследуемый компонентами формы и используемый ими для отображения текста. Изменение значения свойства приводит к автоматическому изменению соответствующего свойства всех компонентов формы (при условии, что значение свойства Font компонента не было задано явно)
BackColor	Цвет фона. Можно задать явно (выбрать на вкладке Custom или Web) или указать элемент цветовой схемы (выбрать на вкладке System)
Opacity	Степень прозрачности формы. Форма может быть непрозрачной (100%) или прозрачной. Если значение свойства меньше 100%, то сквозь форму видна поверхность, на которой она отображается

Для изменения значений свойств объектов используется окно **Properties**. В левой колонке окна перечислены свойства объекта, *выбранного* в данный момент, в правой — указаны значения свойств. Имя выбранного объекта отображается в верхней части окна **Properties**.

Чтобы в заголовке окна отображалось название программы, надо изменить значение свойства `Text`. Для этого следует щелкнуть левой кнопкой мыши в поле значение свойства `Text` (в поле появится курсор), ввести в поле редактирования текст **Доход** и нажать клавишу <Enter> (рис. 2.2).

При выборе некоторых свойств, например `FormBorderStyle`, справа от текущего значения свойства появляется значок раскрывающегося списка. Очевидно, что значение таких свойств можно задать путем выбора из списка (рис. 2.3).

Некоторые свойства являются сложными. Они представляют собой совокупность других (уточняющих) свойств. Например, свойство `Size`, определяющее размер формы, представляет собой совокупность свойств `Width` и `Height`. Перед именами сложных свойств стоит значок **▶**, в результате щелчка на котором раскрывается список уточняющих свойств (рис. 2.4). Значение уточняющего свойства можно задать (изменить) обычным образом — ввести нужное значение в поле редактирования.

Размер формы можно изменить и с помощью мыши, точно так же, как и любого окна, т. е. путем перемещения границы. По окончании перемещения