ГАЛИНА ДОВБУШ
АНАТОЛИЙ ХОМОНЕНКО

Visual C++

НА ПРИМЕРАХ

СОЗДАНИЕ ПРИЛОЖЕНИЙ В СРЕДЕ VISUAL C++

ОСНОВЫ ПРОГРАММИРОВАНИЯ НА C++

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

ОРГАНИЗАЦИЯ ВВОДА-ВЫВОДА И ОБРАБОТКА ИСКЛЮЧЕНИЙ

СОЗДАНИЕ ПРИЛОЖЕНИЙ API WINDOWS И MFC

+CD

bhv®

УДК 681.3.068+800.92VisualC++
ББК 32.973.26-018.1
Д58

Довбуш, Г. Ф.

Д58 Visual C++ на примерах / Г. Ф. Довбуш, А. Д. Хомоненко / Под ред. проф. А. Д. Хомоненко. — СПб.: БХВ-Петербург, 2007. — 528 с.: ил.

ISBN 978-5-94157-918-1

Рассмотрены интерфейс системы программирования Visual C++, техника создания и отладки проектов приложений в среде Visual Studio 2005. Описаны основы языка C++: типы данных и операции, приемы программирования разветвлений и циклов, техника работы со статическими и динамическими массивами, использование функций. Рассмотрены классы и объекты, механизм множественного и одиночного наследования, перегрузка операторов и шаблоны классов, понятия ввода-вывода данных и классификация, принципы работы с потоками и файлами, стандартные классы потоков, форматированный ввод-вывод базовых типов, дополнительные возможности ввода-вывода. Освещена обработка исключений. Показаны особенности создания приложений API Windows и MFC. Представлены внутренняя их организация, создание диалоговых окон и меню, механизм обработки сообщений, работа с картой сообщений. Приводятся многочисленные примеры отлаженных программ. На компакт-диске содержатся тексты листингов примеров программ, приведенных в книге.

Для начинающих программистов

УДК 681.3.068+800.92VisualC++
ББК 32.973.26-018.1

Группа подготовки издания:

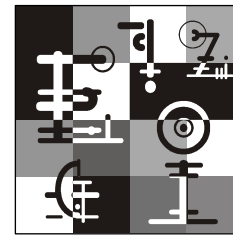
Главный редактор	*Екатерина Кондукова*
Зам. главного редактора	*Игорь Шишигин*
Зав. редакцией	*Григорий Добин*
Редактор	*Елена Кашлакова*
Компьютерная верстка	*Натальи Караваевой*
Корректор	*Виктория Пиотровская*
Дизайн серии	*Игоря Цырульникова*
Оформление обложки	*Елены Беляевой*
Зав. производством	*Николай Тверских*

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 28.09.07.
Формат $70\times100^{1}/_{16}$. Печать офсетная. Усл. печ. л. 42,57.
Тираж 3000 экз. Заказ №
"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

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

ISBN 978-5-94157-918-1

Оглавление



Глава 3

Создание консольного приложения

Консольное приложение — это программа, которая выполняется из командной строки окна DOS или Windows и не имеет графического интерфейса. Проект консольного приложения создается пустым и предполагает добавление в него исходных файлов вручную.

Среда программирования MVC++ использует концепцию рабочей области (solution), что на один уровень абстракции выше, чем проект (project). Проект — это множество всех файлов, необходимых для построения исполняемого файла. В одной рабочей области может содержаться несколько проектов. Несмотря на наличие в рабочей области нескольких проектов, работать можно только над одним, называемым активным. Для рабочей области создается отдельная папка, включающая каталог \Debug и несколько файлов, основным из которых является конфигурационный файл области с расширением `.sln`. Внутри рабочей области каждый проект имеет собственный каталог, в котором находятся конфигурационный файл проекта с расширением `.vcproj` и исходные файлы проекта с расширениями `.h` (заголовочные файлы) и `.cpp` (файлы реализации). Во время создания проекта в его каталоге создается пустая папка \Debug, где будут храниться объектные файлы `.obj`. Исполняемые файлы `.exe` всех проектов рабочей области записываются в каталог рабочей области.

Запуск MVC++

Для запуска среды создания приложений MVC++ необходимо последовательно выполнить следующие действия:

1. Нажать кнопку **Пуск** (Start).
2. Выбрать пункт меню **Программы** (Programs).

3. Выбрать пункт меню **Visual Studio 2005**. В результате откроется стартовое окно, которое показано на рис. 3.1.

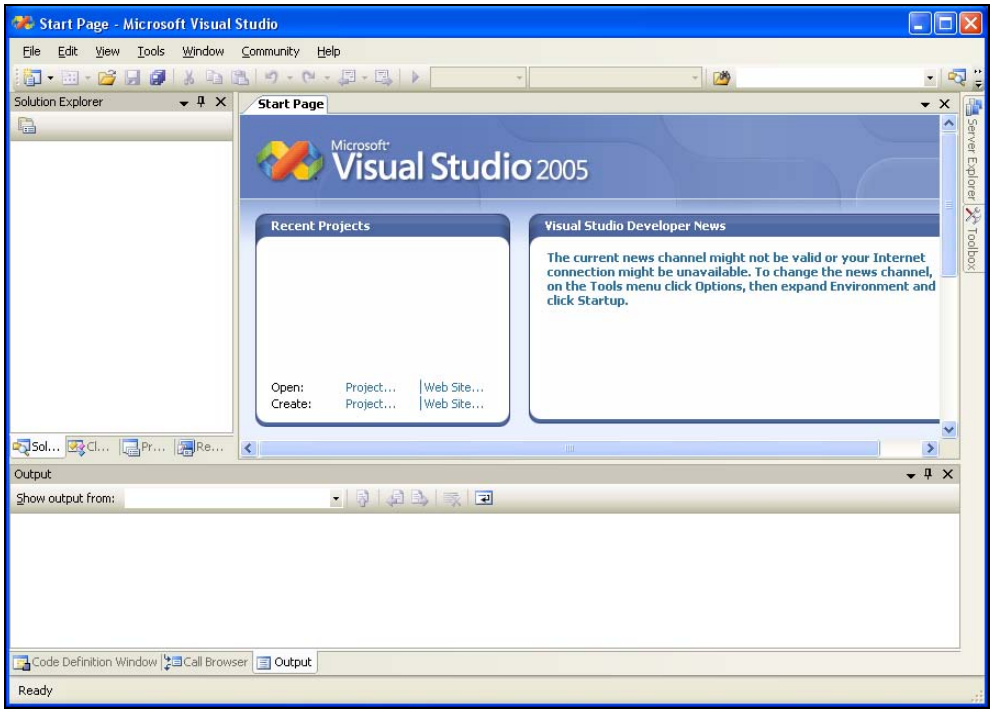


Рис. 3.1. Стартовое окно MVC++

В верхней части окна расположено меню, с помощью которого создаются, изменяются и выполняются программы C++. После выбора пункта меню появляется выпадающее меню, в котором следует выбрать соответствующий пункт. Состояние меню и окон меняется в зависимости от выполняемых программистом действий.

Создание проекта в новой рабочей области

Как упоминалось, рабочая область (solution) создается с целью объединения схожих по тематике проектов. Для создания проекта консольного приложения в новой рабочей области необходимо:

1. Выбрать пункт меню **File → New → Project**.

2. В открывшемся окне **New Project**:
   - в строке **Location** ввести каталог расположения рабочей области или выбрать этот каталог в окне **Project Location**, нажав кнопку **Browse**;
   - в нижней части окна в строке **Solution Name** вместо **<Enter_Name>** набрать название рабочей области;
   - убедиться, что установлен флажок **Create directory for solution**;
   - в строке **Name** вместо **<Enter_Name>** ввести имя проекта;
   - в левой части окна **Project types** выбрать тип проекта **Visual C++ General**;
   - в правой части окна **Templates** выбрать шаблон проекта **EmptyProject**;
   - нажать кнопку **OK**.

На рис. 3.2 демонстрируется вид окна **New Project** при создании проекта p1 в новой рабочей области Exercise, которая будет расположена на диске E.

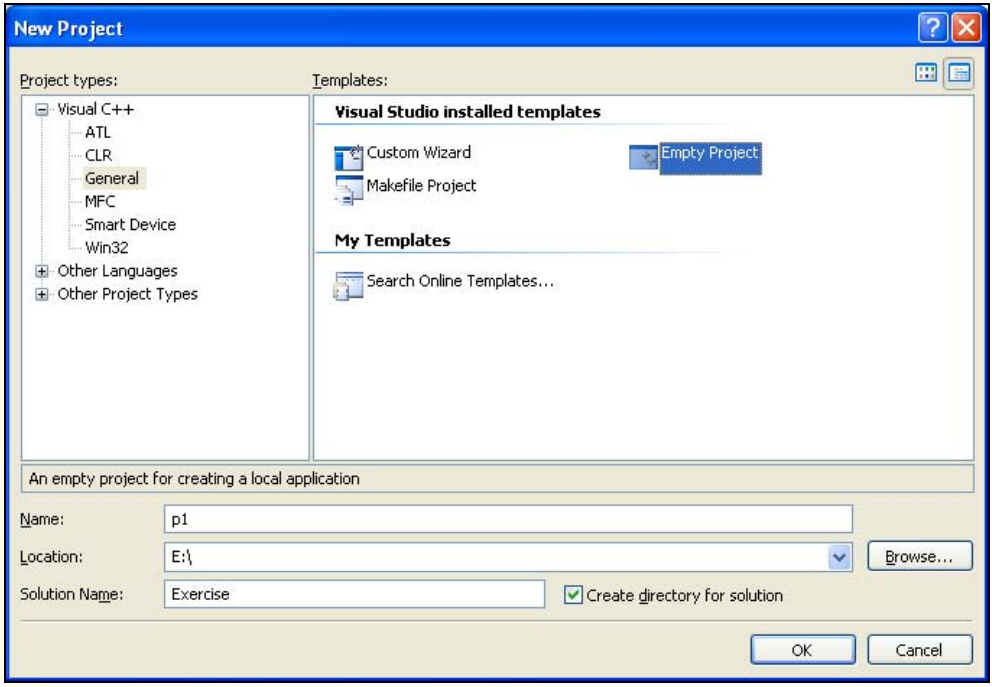


Рис. 3.2. Создание проекта в новой рабочей области

В результате работы мастера MVC++ на диске E в корневом каталоге будет создана папка \Exercise, внутри которой будет сформирован файл рабочей

области Exercise.sln, а также папка \Debug для хранения исполняемых файлов рабочей области. В браузере активной вкладки **Solution Explorer** правой части экрана будет выведено состояние рабочей области, как показано на рис. 3.3.

Рис. 3.3. Состояние рабочей области после создания проекта

Открытие существующей рабочей области

Открыть существующую рабочую область можно различными способами, например, используя MVC++ или проводник.

Для открытия области с помощью MVC++ необходимо выполнить следующие действия:

1. Открыть MVC++.
2. Выбрать нужную область из списка **Recent Project** в стартовом окне.

Если в списке стартового окна нет названия необходимой рабочей области:

1. Выбрать меню **File → Open Project/Solution**.

2. В окне **Open Project** выделить файл рабочей области и открыть его, нажав кнопку **Open**.

Примечание

Если во вкладке **Solution Explorer** браузера MVC++ уже имеется открытая рабочая область, то проекты вновь открываемой рабочей области могут быть добавлены в состав существующей. Для этого следует в окне **Open Project** установить переключатель **Add to Solution** и нажать кнопку **Open**. В результате проекты будут добавлены в уже открытую рабочую область. Если в добавлении проектов нет необходимости, должен быть установлен переключатель **Close Solution**, тогда открытая рабочая область закроется, и будет открыта выбранная в окне **Open Project**.

Для открытия рабочей области при помощи проводника Windows следует выполнить такие шаги:

1. Открыть Проводник.
2. Выделить файл рабочей области (например, **Exercise.sln**).
3. Дважды щелкнуть левой кнопкой мыши по названию файла.

Создание нового проекта в рабочей области

Для создания нового проекта консольного приложения внутри рабочей области проще всего выполнить следующую последовательность действий:

1. Нажать правую кнопку мыши на имени рабочей области в браузере активной вкладки **Solution Explorer**.
2. В появившемся меню выбрать пункт **Add**.
3. В появившемся меню выбрать пункт **New Project**.
4. В окне Add New Project:
   - в левой части окна **Project types** выбрать тип проекта **General**;
   - в правой части окна **Templates** выбрать шаблон проекта **Empty Project**;
   - в нижней части окна в строке **Name** вместо **<Enter_name>** набрать без пробелов название проекта;
   - нажать кнопку **OK**.

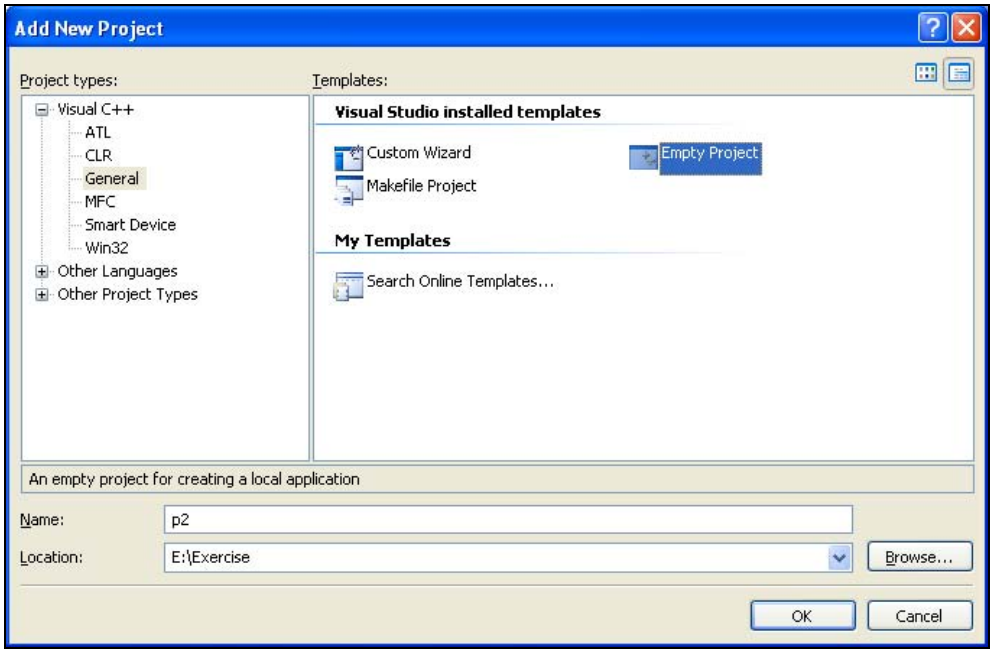


Рис. 3.4. Создание нового проекта в открытой рабочей области

Рис. 3.5. Состояние браузера рабочей области с проектами

На рис. 3.4 показано окно при создании нового проекта p2 в области Exercise.

В результате на диске E в каталоге \Exercise будет создана папка p2, внутри которой будет сформирован файл проекта консольного приложения p2.vcproj. Состояние браузера в активной вкладке **Solution Explorer** показано на рис. 3.5.

Активизация существующего проекта

Рабочая область, как правило, содержит множество проектов. Чтобы сделать проект активным, следует во вкладке **Solution Explorer** выделить имя нужного проекта, нажать правую кнопку мыши и выбрать в меню пункт **Set as StartUp Project**. Название активного проекта в браузере выделяется при этом полужирным шрифтом.

Добавление исходных файлов в проект

В проект консольного приложения необходимо вручную включить исходные файлы: заголовочные файлы (Header Files) и файлы реализации (Source Files).

Включение обоих файлов производится одинаково. Разница состоит только в выборе шаблона: для заголовочного файла шаблон называется Header File (.h), а для файла реализации — C++ File (.cpp).

Далее указаны действия, необходимые для подключения файла реализации:

1. Нажать правую кнопку мыши на имени проекта в браузере Solution Explorer.
2. В появившемся меню выбрать пункт **Add**.
3. В появившемся меню выбрать пункт **Add New Item**.
4. В окне Add New Item:
   - выбрать в правой части окна **Categories** категорию Code;
   - выбрать в левой части окна **Templates** шаблон **C++ File (.cpp)**;
   - ввести в окне **Name** вместо **<Enter_name>** имя файла;
   - нажать кнопку **Add**.

На рис. 3.6 приводится окно для включения файла с именем main в проекте p1 рабочей области Exercise.

В результате на диске в каталоге проекта будет создан файл main.cpp, и в редакторе MVC++ появится окно и вкладка для работы с этим файлом. Состояние главного окна MVC++ после включения файла реализации показано на рис. 3.7.

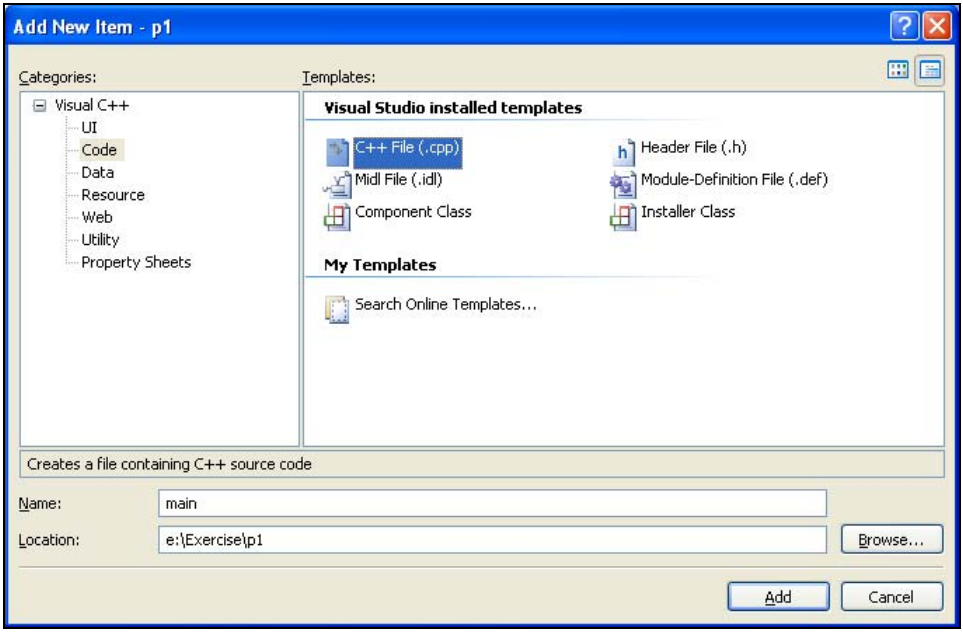


Рис. 3.6. Добавление файла реализации в проект

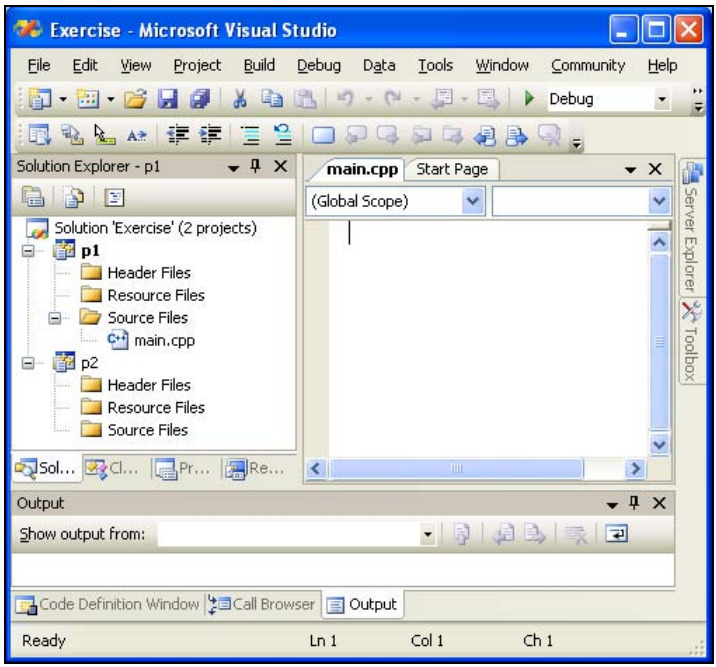


Рис. 3.7. Состояние главного окна после подключения файла реализации

Примечание

Имена исходных файлов задаются автором программы. В начале книги рассматриваются простейшие проекты, состоящие только из одного файла, который содержит исходный код главной функции `main ( )`. Для проектов, состоящих из нескольких файлов, рекомендуется имя заголовочного файла и файла реализации главной функции выбирать таким же, как и имя проекта, а для файлов с описанием и реализацией используемых в программе классов использовать имена самих классов, но без первого символа `C`, с которого обычно они начинаются.

Ввод и редактирование исходного текста осуществляется в активном окне редактора. Редактор поддерживает команды **Cut**, **Copy** и **Paste**, **Undo** и **Redo**, расположенные в меню **Edit**, где также показаны горячие клавиши и пиктограммы для этих действий.

В листинге 3.1 приведен текст первой программы, которая просто выводит текст на экран. Из этой программы будет создаваться исполняемый файл.

Листинг 3.1. Первая программа

```
#include <iostream>
using namespace std ;
int main ( )
{
       cout << "I will be super programmer!" << endl ;
return 0 ;
}
```

Активизация исходного файла для редактирования

Если в окне редактора существует вкладка с именем файла, требующего редактирования, то достаточно щелкнуть левой кнопкой мыши по вкладке. Редактор сделает окно редактирования этого файла активным.

Если такой вкладки не существует, то необходимо дважды щелкнуть левой кнопкой мыши по имени файла в окне **Solution Explorer**. Файл откроется и загрузится в активное окно редактора, в котором появится вкладка с именем открытого файла.

Если файла не существует в проекте, то следует его добавить в проект так, как описано в предыдущем разделе, и приступить к его редактированию.

Сохранение и закрытие файла

Для сохранения файла из активного окна редактора под тем же именем можно либо щелкнуть левой кнопкой мыши по пиктограмме сохранения на панели инструментов, либо выбрать в меню **File → Save**.

Для сохранения файла из активного окна под другим именем необходимо выбрать в меню **File → Save As**. В этом случае в открывшемся окне **Save File As** указать новое имя, а при необходимости выбрать новый каталог, после чего нажать кнопку **Save**. Следует помнить, что сохраненный под другим именем файл не будет являться файлом проекта.

Для закрытия активного файла нажимается кнопка закрытия с крестиком в правом верхнем углу окна редактора.

Трансляция файлов реализации

Следует помнить, что компилируются только файлы реализации, имеющие расширение cpp. Перед трансляцией файл реализации может быть как активным (то есть быть открытым в редакторе), так и нет. Удобнее, если файл будет активным.

Если файл реализации активен, то для его трансляции можно выполнить одно из двух действий:

1. Нажать комбинацию клавиш <Ctrl> + <F7>.
2. Выбрать в меню **Build → Compile**.

Если файл не активен, то выбрать его в списке файлов активной вкладки **Solution Explorer**, нажать правую кнопку мыши и выбрать в меню строку **Compile**.

В случае успешной трансляции исходного файла в папку проекта \Debug компилятор запишет объектный файл с расширением obj и с именем исходного файла. Получив сообщение об успехе (рис. 3.8) после компиляции всех включенных в проект файлов реализации, можно переходить к компоновке.

Компилятор может обнаружить ошибки, тогда внизу экрана будут выведены сообщения о найденных синтаксических ошибках. Для быстрого перемещения к содержащей ошибку строке необходимо дважды щелкнуть левой кнопкой мыши по сообщению об ошибке в нижней части экрана. На рис. 3.9 продемонстрировано состояние окон MVC++ после компиляции программы с ошибками. При вводе текста программы не были набраны кавычки, обрамляющие текстовую константу.

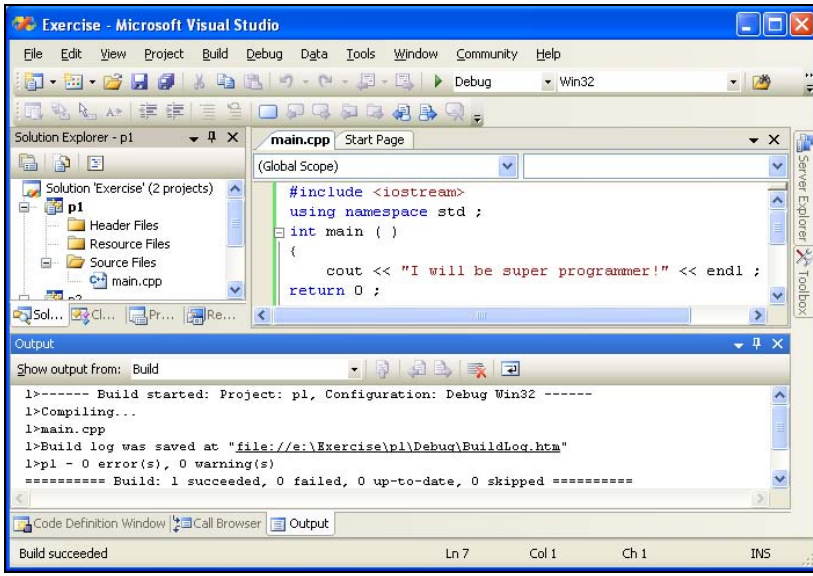


Рис. 3.8. Сообщение об успешной компиляции файла

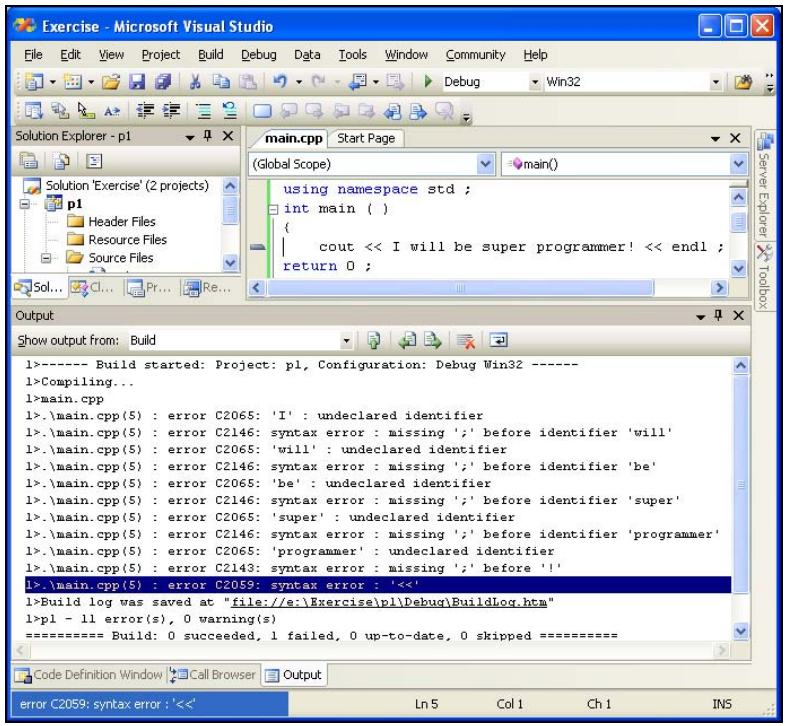


Рис. 3.9. Сообщения об ошибках

Примечание

Компилятор последовательно выполняет проверку синтаксиса программы, поэтому сначала сообщается о неопределенных идентификаторах переменных, а также о пропущенных символах завершения оператора, и только в конце обнаруживает неверную запись операции вывода. Начинающие программисты ошибаются в самых простых операторах. Часто после исправления первой из 127 обнаруженных компилятором ошибок не остается ни одной.

Если компилятор выведет сообщения об ошибках, следует исправить эти ошибки и повторить трансляцию.

Компоновка

Для компоновки приложения из объектных файлов достаточно выбрать в меню **Build → Build**.

Если компоновщик не обнаружит ошибок, то в нижней части экрана появится сообщение, которое показано на рис. 3.10. В результате успешной компоновки в папку проекта \Debug рабочей области будет записан исполняемый файл с расширением exe и с именем проекта. Файл можно вызывать на исполнение.

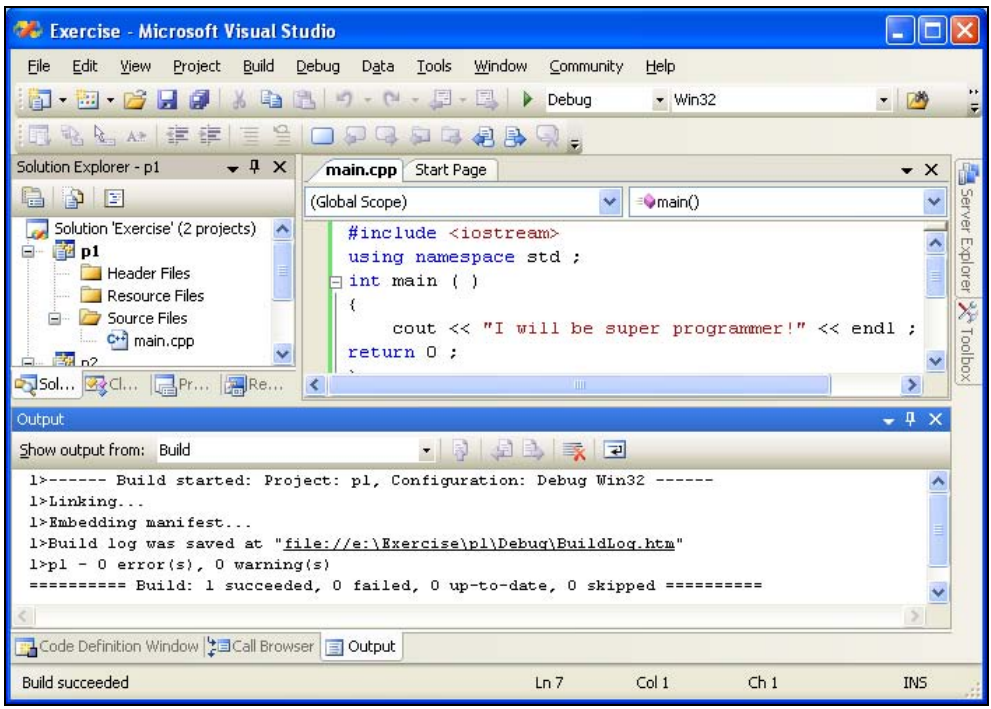


Рис. 3.10. Сообщение об успешной компоновке приложения