

1 эффективность с C#

Визуальные приложения за 10 минут

Не волнуйся, мама. На C# ты
будешь программировать так
быстро, что тушеное мясо больше
никогда не подгорит.



Хотите программировать действительно быстро? C# — это **мощный язык программирования**. Благодаря **Visual Studio** вам не потребуется писать непонятный код, чтобы заставить кнопку работать. Вместо того чтобы запоминать параметры метода для *имени* и для *ярлыка* кнопки, вы сможете **сфокусироваться на достижении результата**. Звучит заманчиво? Тогда переверните страницу и приступим к делу.

Зачем вам изучать C#

C# и ИСР Visual Studio облегчают и ускоряют процесс написания кода.

ИСР, или Интегрированная Среда Разработки, — это программа для редактирования кода, управления файлами и публикации проектов.

Задачи, которые за вас решает ИСР

Чтобы поместить на форму кнопку, вам потребуются большие куски повторяющегося кода.

```
using System;
using System.Collections.Generic;
using System.Windows.Forms;
namespace A_New_Program
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

```
private void InitializeComponent()
{
    this.button1 = new System.Windows.Forms.Button();
    this.SuspendLayout();
    // button1
    //
    this.button1.Location = new System.Drawing.Point(105, 56);
    this.button1.Name = "button1";
    this.button1.Size = new System.Drawing.Size(75, 23);
    this.button1.TabIndex = 0;
    this.button1.Text = "button1";
    this.button1.UseVisualStyleBackColor = true;
    this.button1.Click += new System.EventHandler(this.button1_Click);
    // Form1
    //
    this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.None;
    this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
    this.ClientSize = new System.Drawing.Size(292, 267);
    this.Controls.Add(this.button1);
    this.Name = "Form1";
    this.Text = "Form1";
    this.ResumeLayout(false);
}
```

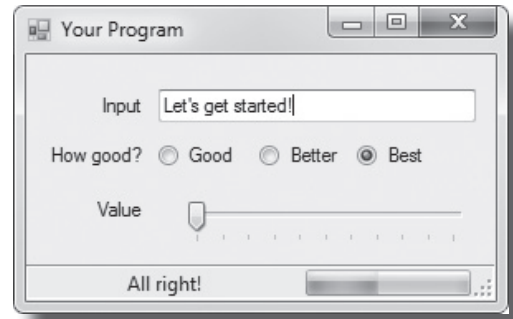
Этот код всего лишь добавляет на форму кнопку. Добавление других элементов может увеличить его в десятки раз.

Преимущества Visual Studio и C#

Язык C#, оптимизированный для программирования в Windows, вместе с Visual Studio позволяет сфокусироваться на непосредственных задачах.



Такое приложение не только лучше выглядит, но и быстрее создается.



C#, ИСР Visual Studio многое упрощают

Язык C# и Visual Studio позволят без дополнительных усилий выполнять следующие задачи:

- 1 **Быстро создавать приложения.** Программировать на C# очень просто. Это мощный, легко осваиваемый язык, а Visual Studio позволяет автоматизировать большинство процессов.
- 2 **Разрабатывать красивый пользовательский интерфейс.** Инструмент Form Designer в Visual Studio превращает создание великолепного пользовательского интерфейса в одну из самых увлекательных задач при разработке приложений на C#. Вам больше не потребуется тратить часы на написание графических элементов с нуля.
- 3 **Создавать базы данных и взаимодействовать с ними.** ИСР снабжена простым интерфейсом для создания баз данных, которые затем легко интегрируются в SQL Server Compact Edition и другие популярные приложения.
- 4 **Фокусироваться на решении РЕАЛЬНЫХ проблем.** За конечный результат работы, разумеется, отвечаете вы и только вы. Но ИСР позволяет концентрироваться на глобальных вещах, взяв на себя:
 - ★ слежение за всеми проектами;
 - ★ упрощенное редактирование кода;
 - ★ отслеживание графики, аудиофайлов, значков и прочих ресурсов;
 - ★ управление базами данных и взаимодействие с ними.

Теперь вместо рутинного написания кода вы можете потратить время на **создание потрясающих программ.**

Скоро вы поймете, что мы имеем в виду.

Избавьте директора от бумаг

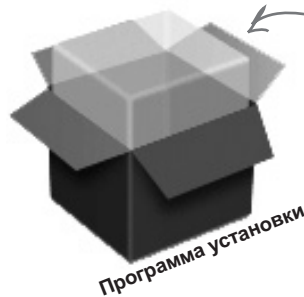
В объективильской фирме по производству бумаги появился новый исполнительный директор. Он любит пешие прогулки, кофе и природу... и с целью сохранения лесов решил перейти на безбумажный документооборот. На выходные он уехал кататься на лыжах в Аспен, а вам приказал к понедельнику написать программу для хранения контактной информации. Если этого не сделать... хм... вы составите компанию предыдущему директору вашей фирмы, который ищет новую работу.



Перед началом работы выясните, что именно нужно пользователю

Перед началом работы над *любым* приложением нужно понять, кто будет его использовать и каким ожиданиям оно должно соответствовать.

- 1 Директор хочет, чтобы программа работала не только на его офисном компьютере, но и на его ноутбуке. Значит, вам потребуется инсталлятор, который позволит поместить нужные файлы на любой компьютер.



Так как приложение должно работать не только в офисе, но и на ноутбуке директора, без установщика не обойтись.

- 2 Другим сотрудникам фирмы также понадобится доступ к контактной информации. Как же в противном случае они сформируют списки рассылки, чтобы сделать фирму лидером по продаже бумаги.



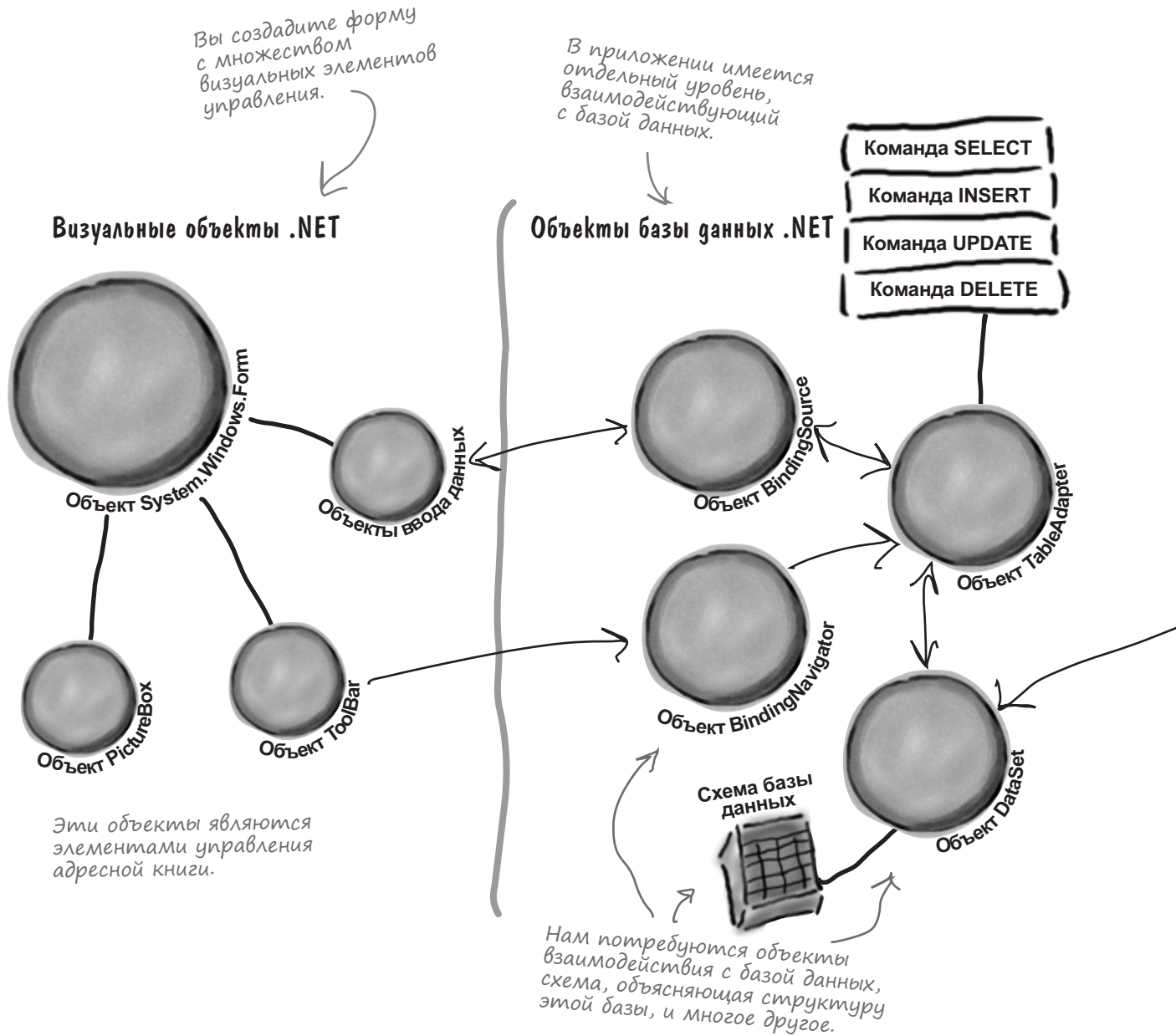
Мы уже знаем, что Visual C# упрощает работу с базами данных. Поместив контактную информацию в базу, вы обеспечите доступ к ней любому сотруднику.

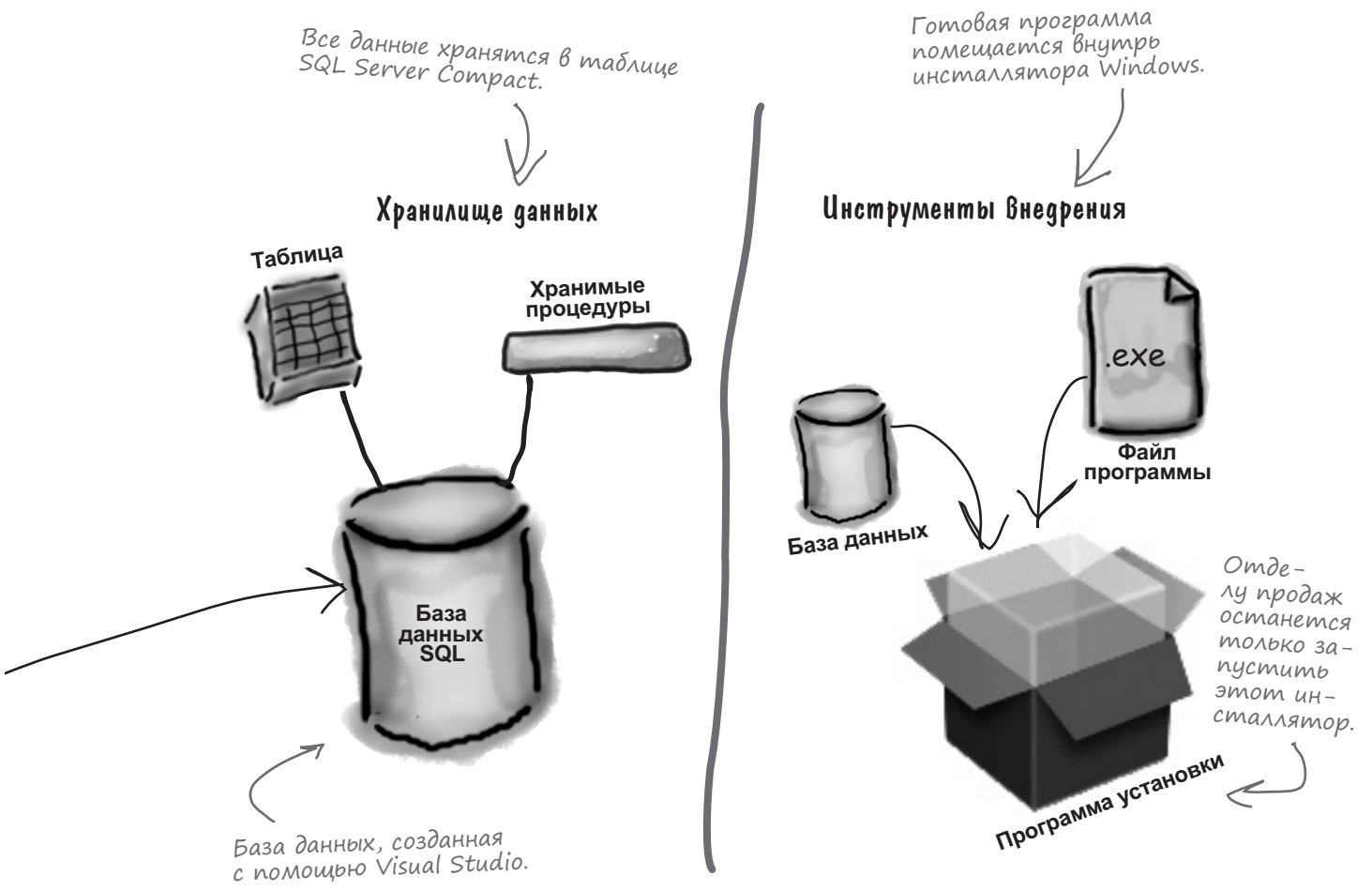
Перед началом работы над программой всегда думайте о нуждах конечных пользователей, только так можно создать по-настоящему качественный продукт!

Что мы собираемся сделать

Нам потребуется приложение с графическим интерфейсом пользователя, объекты, взаимодействующие с базой данных, собственно база данных и программа установки. Вся эта огромная с виду работа будет проделана к концу этой главы.

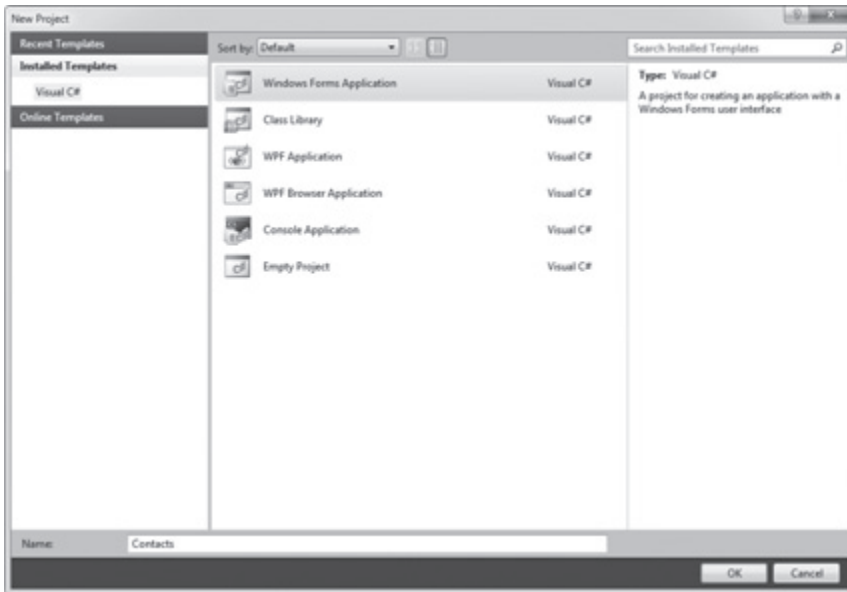
Вот структура нашей будущей программы:





Это вы делаете в Visual Studio

Запустите Visual Studio, если вы еще это не сделали. Пропустите начальную страницу и выберите в меню **File** команду **New Project**. В открывшемся окне диалога **New Project** выберите тип проекта **Windows Forms Application**, а в текстовое поле **Name** в нижней части окна введите название проекта **Contacts**.



Будьте осторожны!

В вашей ИСР все может выглядеть по-другому.

На рисунке показан вид окна **New Project** в *Visual Studio 2010 Express Edition*. В *Professional* или *Team Foundation edition* оно может выглядеть по-другому. Но суть его от этого не меняется.

А это Visual Studio делает за вас

В момент сохранения проекта ИСР создает файлы **Form1.cs**, **Form1.Designer.cs** и **Program.cs**. Они добавляются в окно **Solution Explorer** и по умолчанию сохраняются в папке **My Documents\Visual Studio 2010\Projects\Contacts**.

Команда **Save All** из меню **File** сохраняет все открытые файлы, в то время как команда **Save** — только файл, активный в данный момент.

Этот файл содержит код, определяющий поведение формы.

Этот файл содержит код, запускающий программу и показывающий форму.

Код, определяющий форму и ее объекты.



Возьми в руку карандаш

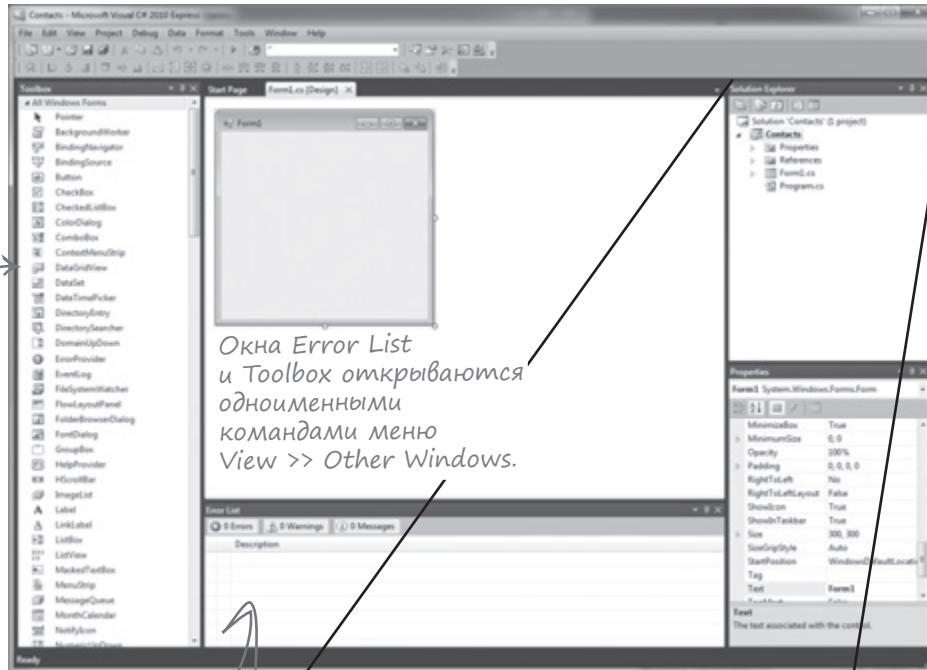


Ниже показан возможный вид экрана. Вы должны понимать назначение большинства окон и файлов. Убедитесь в наличии панелей Toolbox и Error List, **вызываемых одноименными командами меню View >> Other Windows**. В пустые строки впишите назначение каждой части ИСР, как показано в примере.

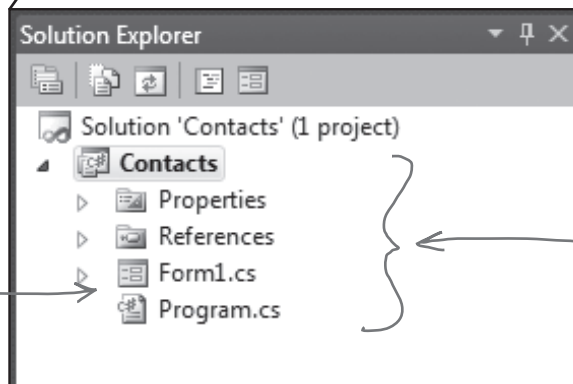
Здесь выбираются инструменты для работы.

Если ваша ИСР выглядит по-другому, воспользуйтесь командой Window>>Reset Window Layout.

Это окно было увеличено, чтобы дать вам больше пространства.



Окна Error List и Toolbox открываются одноименными командами меню View >> Other Windows.



Возьми в руку карандаш

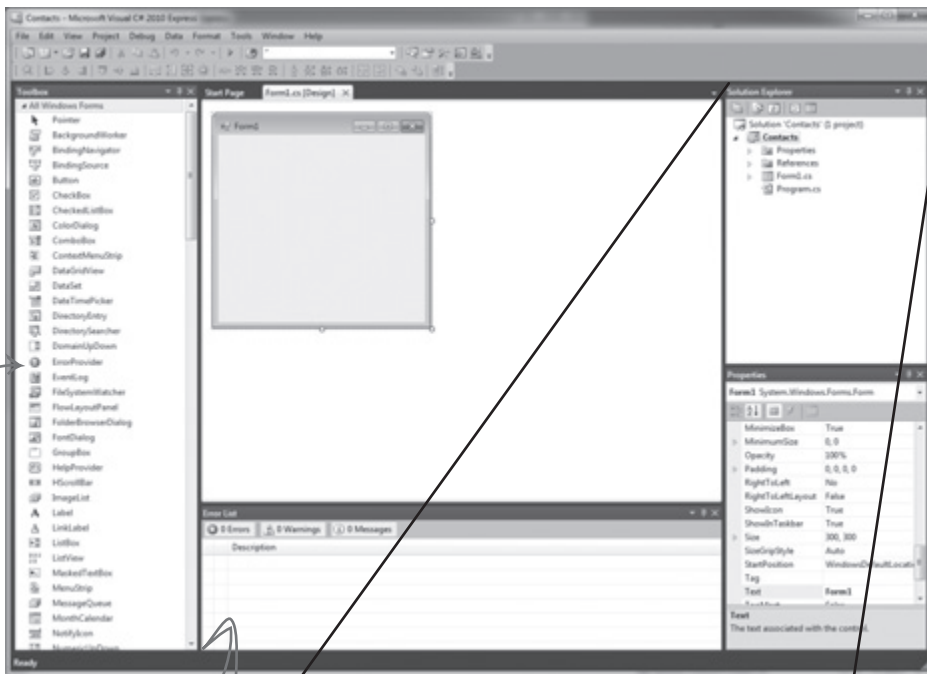


Решение

Итак, вы описали назначение различных частей ИСР Visual Studio C#. Здесь вы можете сравнить написанное с правильными вариантами ответа.

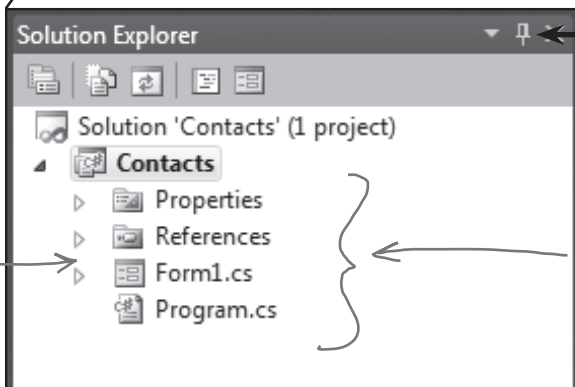
Здесь выбираются инструменты для работы.

Здесь собраны визуальные элементы управления, которые можно перетаскивать на форму.



В окне Error List отображаются вкравшиеся в код ошибки.

Имена автоматически созданных файлов Form1.cs и Program.cs появляются в окне Solution Explorer.



Здесь показаны свойства выбранного элемента управления формы.

Щелчок на этой кнопке включает и отключает функцию автоматического сворачивания окна. У панели Toolbox она включена по умолчанию.

В окне Solution Explorer осуществляется выбор файла.