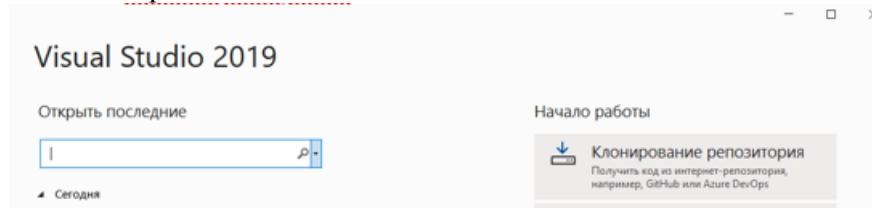
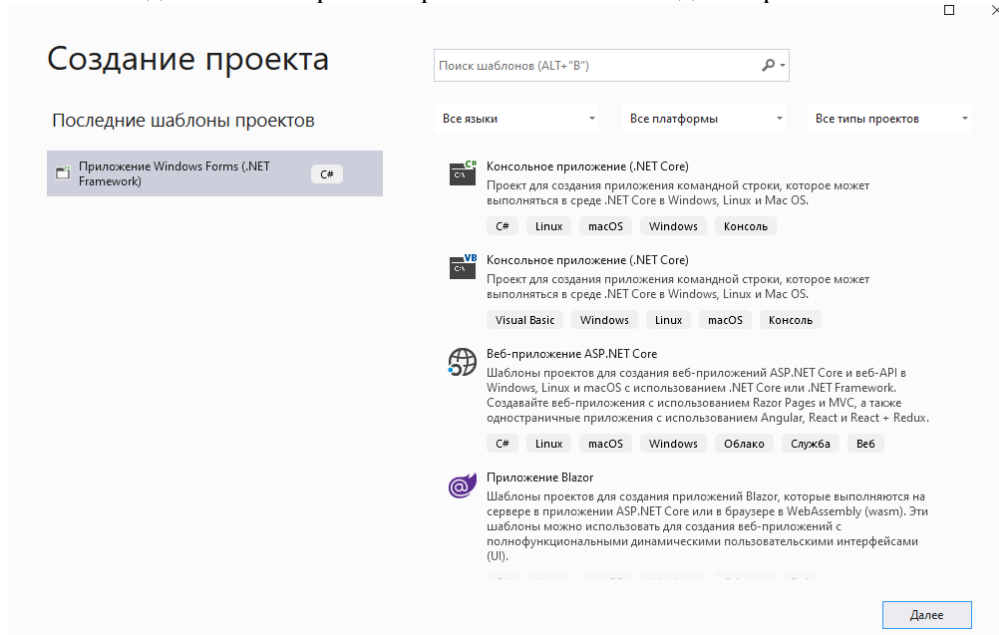


Лабораторная работа №2-часть 1. Работа с DNS в .NET

Шаг 1. Открываем Microsoft Visual Studio



Шаг 2. Создаем новый проект через меню «Файл/Создать/Проект»



Выбираем шаблон Visual C#, приложение Windows Forms. В поле «Имя» вводим имя проекта, например, «PrimerIP». В поле «Расположение» указываем папку для хранения проекта. После этого нажимаем «ОК».

После создания проекта, он содержит в себе одно окно Form1 и следующие файлы с исходным кодом:

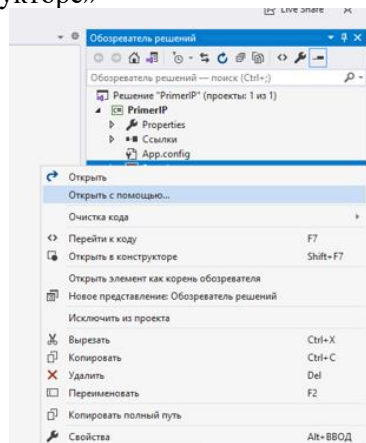
Program.cs – главный файл программы, создается средой автоматически;

Form1.cs – файл исходного кода окна, которые должен дополнять программист;

Form1.Designer.cs – файл исходного кода окна, который среда Visual Studio создает и обновляет автоматически; содержит код для создания управляющих элементов окна (кнопок, текстовых полей и т.п.)

Просмотреть список файлов можно в «Обозревателе решений». Вызвать его можно через меню «Вид/Обозреватель решений».

В середине окна располагается либо конструктор формы, как на рисунке, либо исходный код файла. Для переключения между конструктором и исходным кодом надо в обозревателе решений щелкнуть правой кнопкой на файле «Form1.cs» и в контекстном меню выбрать пункт «Перейти к коду» или «Открыть в конструкторе»



Шаг 3. Создание визуального интерфейса приложения

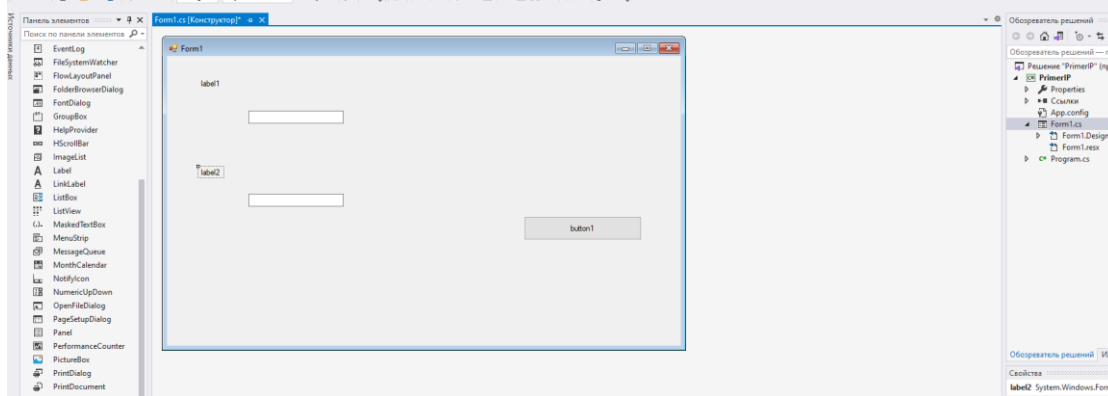
Визуальные компоненты добавляются на форму из «Панели элементов». Для вызова этой панели выберите пункт меню «Вид/Панель элементов». Для удобства можно раскрыть панель на вкладке «Все формы Windows Forms». Для дальнейшего выполнения лабораторной понадобятся следующие компоненты.

Button – кнопка

TextBox – текстовое поле

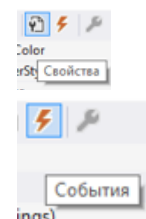
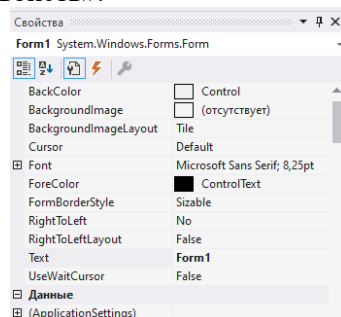
Label – надпись.

Размер формы, размер элементов и их положение можно менять с помощью мыши.



Шаг 4. Задание свойств визуальных элементов.

Все визуальные элементы имеют некоторые свойства, которые можно и нужно менять для достижения необходимого вида. Просмотреть свойства можно в окне «Свойства» (справа внизу), которое открывается по команде «Вид/Окно свойств».



Данное окно имеет пиктограммы: Свойства (вызывается кнопочкой со списком правее кнопки «AZ») и События (кнопка с молнией). Свойства определяют то, как элемент будет называться и выглядеть. События позволяют создать обработчики событий – код, вызываемый при определенных обстоятельствах, например, нажатии кнопки, изменении размеров окна и т.д.

Одним из самых важных свойств элемента является его имя – Name. Именно по имени можно в исходном коде окна обратиться к этому элементу.

Далее расположите элементы так, как показано это ниже и введите следующие свойства элементов:

Для элемента label1 задайте значение поля Text равным «Адрес», а для label2 – «Результат».

Для элемента textBox1 задайте Name равным «textBoxHost», свойство Text – пустым.

Для элемента button задайте Name равным «buttonFind», а свойство Anchor – «Top,Right».

Для элемента textBox2 задайте Name равным «textBoxLog», свойство Text – пустым, свойство Multiline равным «True», а свойство Anchor – «Top, Bottom, Left, Right».

Свойство Anchor (якорь) определяет, как будет перемещаться на форме визуальный элемент при изменении размеров окна. Если какая-то сторона элемента «заякорена», то она будет сохранять свое положение относительно соответствующей стороны окна.

Шаг 5. Добавление обработчика для кнопки buttonFind.

Выделите кнопку buttonFind в конструкторе в окне свойств перейдите на вкладку «События». Найдите там событие Click и дважды щелкните мышкой в правой колонке таблицы в строке этого события (или просто щелкните по данному объекту на форме в режиме конструктора).

После этого среда Visual Studio создаст «код-болванку» для события и откроет исходный код окна.

```
Form1.cs* X Form1.cs [Конструктор]*
PrimerIP
1 using System;
2 using System.Collections.Generic;
3 using System.ComponentModel;
4 using System.Data;
5 using System.Drawing;
6 using System.Linq;
7 using System.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10
11 namespace PrimerIP
12 {
13     Ссылка: 2
14     public partial class Form1 : Form
15     {
16         Ссылка: 1
17         public Form1()
18         {
19             InitializeComponent();
20         }
21         Ссылка: 1
22         private void buttonFind_Click(object sender, EventArgs e)
23         {
24         }
25     }
26 }
```

Каждая форма (окно) описывается одним классом (в данном случае Form1), который расширяет стандартный класс Form (окно без всяких визуальных элементов). Обработчики событий – это методы (функции) этого класса. После создания обработчика события Click кнопки buttonFind среда Visual Studio создает в классе Form1 пустой метод buttonFind_Click. Далее в этот метод можно добавить полезный код, выполняющий нужные по замыслу программиста действия.

Шаг 6. Реализация полезного функционала.

Ниже приведен исходный код, который должен быть набран, чтобы написать приложение для получения и вывода всех IP-адресов, связанных с данным именем хоста.

```
Form1.cs X Form1.cs [Конструктор]
PrimerIP
1 using System;
2 using System.Collections.Generic;
3 using System.ComponentModel;
4 using System.Data;
5 using System.Drawing;
6 using System.Linq;
7 using System.Text;
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10 using System.Net; // подключаем библиотеку для работы с сетью
11
12 namespace PrimerIP
13 {
14     Ссылка: 2
15     public partial class Form1 : Form
16     {
17         Ссылка: 1
18         public Form1()
19         {
20             InitializeComponent();
21         }
22         Ссылка: 1
23         private void buttonFind_Click(object sender, EventArgs e)
24         {
25             // Добавить строку в textBoxLog перед выводом результатов
26             textBoxLog.AppendText("Результат для " + textBoxHost.Text + "\n");
27             //Получаем список IP-адресов, связанных с данным адресом хоста
28             //и записываем результат в массив addresses
29             IPAddress[] addresses = Dns.GetHostAddresses(textBoxHost.Text);
30             //для каждого адреса выполняем действия
31             //в переменной addr хранится адрес из списка
32             foreach (IPAddress addr in addresses)
33             {
34                 // выводим в textBoxLog адрес addr в виде текста
35                 textBoxLog.AppendText(addr + "\n");
36             }
37         }
38     }
39 }
```

Встроенные библиотеки .NET предоставляют широкие, удобные и простые средства для работы с сетью. Чтобы получить доступ к этой библиотеке, в начале файла исходного кода необходимо добавить строку «using System.Net;».

Для работы с DNS в этой библиотеке создан специальный класс Dns. Этот класс содержит набор удобных и полезных методов. В частности, метод (функция) GetHostAddresses позволяет получить список всех IP-адресов, связанных с именем хоста, передаваемым в качестве аргумента для этого метода.

Пользователь вводит адрес хоста в текстовое поле объекта textBoxHost. Чтобы прочитать или записать текст в это поле необходимо обратиться к его свойству textBoxHost.Text типа string (строка). Именно эта строка передается как аргумент в метод Dns.GetHostAddresses. После вызова этого метода, он сам обращается к DNS-серверам и возвращает в качестве результата массив из IP-адресов – объектов типа IPAddress. Этот результат сохраняется в переменной addresses.

Текстовое поле может хранить не одну строку, а многострочный (многоабзачный) текст, если у него поле Multiline равно true, как в случае с полем textBoxLog. Для добавления текста в конец этого поля предназначен метод AppendText. Вначале обработчика buttonFind_Click в поле textBoxLog добавляется строка, подсказывающая пользователю, для какого имени хоста (textBoxHost.Text) ищется список IP-адресов. Последовательность символов \n означает, что в данном месте текста должен произойти переход на новую строку (на новый абзац).

Цикл foreach (IPAddress addr in addresses) позволяет по очереди выполнить однотипные действия для всех адресов из массива (списка) addresses. В нашем случае это действие – печать адреса в текстовое поле textBoxHost.

Шаг 7. Тестирование приложения.

После того как исходный код введен правильно, через команду меню «Отладка/Начать отладку» можно запустить приложение и проверить его работоспособность. Обратите внимание, что некоторые имена хостов, например www.yandex.ru, связаны не с одним единственным IP-адресом, а сразу с несколькими.

Попробуйте определить адреса разных сайтов, например, страницы Филиала.

