

Практическая работа 14

Использование объектов библиотеки классов NET Framework

Цель занятия

Получить практические навыки использования объектов библиотеки классов .NET Framework

Перечень оборудования и программного обеспечения

Персональный компьютер
Microsoft Office (Word, Visio)
Microsoft Visual Studio 2010

Краткие теоретические сведения

Описание библиотеки классов

Платформа .NET - многоязыковая среда, открытая для свободного включения новых языков, создаваемых не только Microsoft, но и третьими фирмами. Все языки, включаемые в платформу .NET, должны опираться на единый каркас, роль которого играет .NET Framework. Каркас .NET также содержит библиотеку классов (Class Library). Она служит тем же целям, что и любая библиотека классов, входящая в каркас. Библиотека содержит более сотни интерфейсов и классов, объединенных в группы по тематике. Каждая группа задается пространством имен (namespace), корневое пространство носит имя System. Классы библиотеки связаны отношением наследования, все классы являются наследниками класса System.Object. Для классов библиотеки, равно как и для классов в языке C#, не определено множественное наследование.

Например, если при построении нового проекта в C# выбрать в качестве типа проекта Windows Application, то каркас приложения, построенный по умолчанию, будет включать следующие пространства имен из библиотеки классов:

```
using System;  
using System.Drawing;  
using System.Collections;  
using System.ComponentModel;  
using System.Windows.Forms;  
using System.Data;
```

Корневое пространство имен System содержит фундаментальные и базовые классы - общие для всех языков ссылочные типы и типы-значения, события, обработчики событий и средства обработки исключений. Пространство имен Drawing обеспечивает доступ к GDI. В это пространство

входят классы Graphics, Bitmap, Brush, Point, Rectangle и т. п. Пространство имен Collections содержит интерфейсы и классы, определяющие различные коллекции объектов - списки, очереди, массивы, хэшируемые таблицы, словари. Единый каркас многоязыковой среды программирования во многом способствует компонентной ориентированности не только отдельного языка, но и всей среды .NET.

Библиотека классов - это статическая составляющая каркаса. В .NET Framework есть и динамическая составляющая - агент, задающий общую среду выполнения, CLR (Common Language Runtime). Роль этой среды весьма велика - в ее функции входит управление памятью, потоками, безопасностью, компиляция из промежуточного байт-кода в машинный код и многое другое. Важный элемент CLR - мощный механизм сборки мусора (garbage collector), управляющий работой с памятью типа "куча".

Язык C# в полной мере позволяет использовать все возможности CLR. Код, создаваемый на C#, по большей части безопасен. Среда CLR выполняет массу других проверок, следя за тем, чтобы не появлялись неопределенные указатели, не было выхода за границы массива и т. д.

Библиотека классов .NET Framework предоставляет библиотеку проверенного кода, предназначенного для повторного использования, который разработчики могут вызывать из своих приложений.

Библиотека базовых классов .NET Framework содержит основные функциональные возможности, такие как:

- передача данных по сети;
- система безопасности;
- удаленная обработка;
- диагностика;
- ввод/вывод;
- базы данных;
- язык XML;
- Web-службы, которые позволяют использовать интерфейсы компонентов в любом месте Internet;
- Web-программирование;
- пользовательский интерфейс операционной системы Windows.

Перечень наиболее используемых пространств имен

Пространство имён	Назначение
System	
System.Data	Для обращения к базам данных
System.Data.Common	
System.Data.OleDb	
System.Data.SqlClient	
System.Collections	Классы для работы с контейнерными объектами
System.Diagnostics	Классы для трассировки и отладки кода

System.Drawing	Классы графической поддержки
System.Drawing.Drawing2D	
System.Drawing.Printing	
System.IO	Поддержка ввода-вывода
System.Net	Поддержка передачи данных по сетям
System.Reflection	Работа с пользовательскими типами во время выполнения приложения
System.Reflection.Emit	
System.Runtime.InteropServices	Поддержка взаимодействия с “обычным кодом” – DLL, COM-серверы, удалённый доступ
System.Runtime.Remoting	
System.Security	Криптография, разрешения
System.Threading	Работа с потоками
System.WEB	Работа с web-приложениями
System.Windows.Forms	Работа с элементами интерфейса Windows
System.XML	Поддержка данных в формате XML

Создание проектов с использованием библиотеки базовых классов

Использование таймера

В .NET Framework доступно несколько классов Timer, которые могут применяться для обеспечения вызова методов по истечении определенного промежутка времени, либо для выполнения операций с заданной периодичностью. Этот компонент предназначен для работы в среде WindowsForms.

Методы Timer:

Start() Начинает вызывать событие Elapsed, задавая для свойства Enabled значение true.

Stop() Прекращает вызывать событие Elapsed, задавая для свойства Enabled значение false.

Пример:

Создайте форму, поместите на форму элементы label, textbox, Timer
 В textBox1 будет указано время в минутах до выключения компьютера
 В переменной ShutTime будет храниться время выключения компьютера.

Для работы модуля добавьте **using System.Diagnostics;**

Добавьте следующий код:

```
DateTimeShutTime;
private void button1_Click(object sender, EventArgs e)
{
```

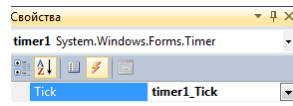
```
ShutTime = DateTime.Now.AddMinutes(Convert.ToInt32(textBox1.Text));
label1.Text = "До выключения осталось " + textBox1.Text + " мин.";
```

```

timer1.Start();
    }
private void timer1_Tick(object sender, EventArgs e)
    {
if (DateTime.Now < ShutTime)
    {
TimeSpants = ShutTime - DateTime.Now;
label1.Text = "До выключения осталось " + ts.Minutes + " мин. " + ts.Seconds +
" сек.";
    }
else
    {
Close();
Process.Start("Shutdown.exe", "-s");
    }
    }
    }
}

```

Для элемента timer1 в свойствах Tick.



добавьте событие

Использование блокировки потока using System.Threading;

```
Thread.Sleep(TimeSpan.FromSeconds(5));
```

Запуск процессов

Часто возникает необходимость узнать, какие процессы запущены на компьютере, и определить параметры запуска, например, путь к приложению, или запустить дочерний процесс.

Пример:

```

private void butExePath_Click(object sender, EventArgs e)
    {
string appPath = Application.ExecutablePath;
    // Выводим полный путь к файлу
MessageBox.Show(appPath);
    }
private void butBaseDirectory_Click(object sender, EventArgs e)
    {
    // Выводим путь к папке, откуда запущено приложение
MessageBox.Show(System.AppDomain.CurrentDomain.BaseDirectory);
    }

```

```

MessageBox.Show(Application.StartupPath);
    }

private void butRunNotepad_Click(object sender, EventArgs e)
    {
        // создаем НОВЫЙ процесс
        Process proc = new Process();
        // Запускаем Блокнот
        proc.StartInfo.FileName = @"Notepad.exe";
        proc.StartInfo.Arguments = "";
        proc.Start();
    }
private void butRunIE_Click(object sender, EventArgs e)
    {
        // Запускаем Блокнот с файлом test.txt
        //Process.Start("notepad.exe", "test.txt");

        // Запускаем браузер InternetExplorer с заданным адресом
        Process.Start("iexplore.exe", "yandex.ru");
    }

```

Работа с электронной почтой

```

using System.Net.Mail;
using System.Net;

    var client = new SmtpClient("smtp.mail.ru");

    varmsg = new MailMessage("from", "to");

    msg.Subject = "My test message";
    msg.Body =
        "Hello, my friend! Just imagine, that I just written a incredible program
to send this meaningless letter to you a billion times only by several clicks!";
    msg.SubjectEncoding = Encoding.UTF8;
    msg.BodyEncoding = Encoding.UTF8;
        // приоритет
    msg.Priority = MailPriority.High;
    // тело письма в html?
    msg.IsBodyHtml = false;

    client.Credentials = new NetworkCredential("login", "password");
    client.Send(msg);

```

Задания

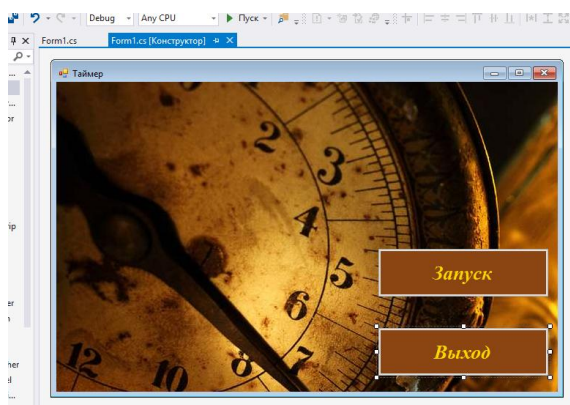
- 1 Изучить теоретические сведения и задание к работе
- 2 В соответствии с вариантом задания разработать и отладить программный модуль

Порядок выполнения работы

Задание 2 (Пример выполнения)

Разработка модуля, содержащего форму, на которую помещена кнопка и элемент управления Timer. При нажатии на кнопку через 3 минуты должен вызваться графический редактор Adobe Photoshop с конкретным файлом calendar.jpg.

1 Интерфейс формы в VisualStudio



2 Текст программы.

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
using System.Diagnostics;  
using System.IO;  
using System.Windows.Forms;  
using System.Reflection;
```

```
namespace WindowsFormsApplication14
```

```
{
```

```
    public partial class Form1 : Form
```

```
    {
```

```
        public Form1()
```

```
        {
```

```
        ...
```

```
        }
```

```
        private void button2_Click(object sender, EventArgs e)
```

```
        {
```

```
            Close();
```

```
        }
```

```
    }
```

```
}
```

3 VisualStudio (фрагмент).

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Diagnostics;
using System.IO;
using System.Windows.Forms;
using System.Reflection;

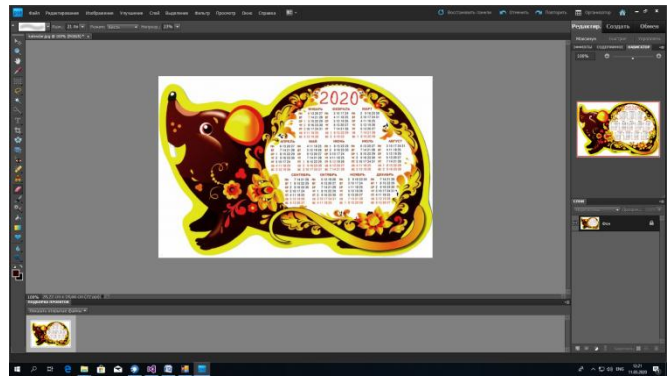
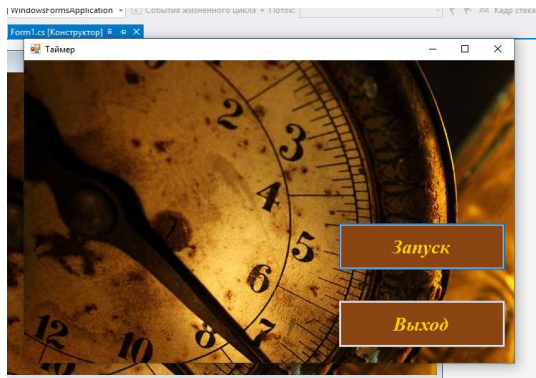
namespace WindowsFormsApplication14
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;
        }

        private void button1_Click(object sender, EventArgs e)
        {
            timer1.Interval = 3000;
            timer1.Enabled = true;
            timer1.Start();
        }

        private void timer1_Tick(object sender, EventArgs e)
        {
            Process proc = new Process();
            proc.StartInfo.FileName = @'C:\Program Files (x86)\Adobe\Elements 9 Organizer\Photoshop Elements 9.0.exe';
            proc.StartInfo.Arguments = @'C:\Users\Гуньяков\\Desktop\pr\kalendar.jpg';
            proc.Start();
            timer1.Stop();
            timer1.Enabled = false;
        }

        private void button2_Click(object sender, EventArgs e)
        {
            Close();
        }
    }
}
```

4 Результаты работы программы:



Содержание отчета

- 1 Название работы
- 2 Цель работы
- 3 Технические средства обучения
- 4 Задания (условия задач)
- 5 Порядок выполнения работы
- 6 Ответы на контрольные вопросы
- 7 Вывод

Варианты к практической работе:

1 Разработка модуля, содержащего форму, на которую помещена кнопка и элемент управления Timer. При нажатии на кнопку через 4 минуты должен вызваться графический редактор Paint с конкретным файлом.

2 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку через интервал времени, указанный в textBox должен вызваться графический редактор Paint.

3 Разработка модуля, содержащего форму, на которую помещена кнопка, элемент управления Timer. При нажатии на кнопку должен вызваться диспетчер задач через 3 минуты.

4 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку должен вызваться диспетчер задач через интервал, указанный в textbox.

5 Разработка модуля, содержащего форму, на которую помещена кнопка и элемент управления Timer. При нажатии на кнопку через 5 минут должен вызваться Блокнот с конкретным текстовым файлом.

6 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку должен вызваться Блокнот через интервал, указанный в textbox.

7 Разработка модуля, содержащего форму, на которую помещена кнопка и элементы управления Timer и textBox. При нажатии на кнопку должен вызваться проводник через интервал в секундах, указанный в textbox.

8 Разработка модуля, содержащего форму, на которую помещена кнопка и элемент управления Timer. При нажатии на кнопку через 20 секунд должен вызваться Калькулятор.

9 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку должен вызваться Калькулятор через интервал, указанный в textBox.

10 Разработка модуля, содержащего форму, на которую помещена кнопка и элемент управления Timer. При нажатии на кнопку через 30 секунд должен вызваться Монитор ресурсов.

11 Разработка модуля, содержащего форму, на которую помещена кнопка и элемент управления Timer. При нажатии на кнопку через интервал, указанный в textBox должен вызваться Монитор ресурсов.

12 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку через интервал времени, указанный в textBox должен вызваться графический редактор Paint 3D.

13 Разработка модуля, содержащего форму, на которую помещена кнопка и элемент управления Timer. При нажатии на кнопку через 40 секунд должен вызваться графический редактор Paint 3D с указанным графическим файлом.

14 Разработка модуля, содержащего форму, на которую помещена кнопка и элемент управления Timer. При нажатии на кнопку через 40 секунд должен вызваться графический редактор Corel Draw с указанным файлом.

15 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку через интервал времени, указанный в textBox должен вызваться графический редактор Corel Draw.

16 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку должен вызваться Блокнот через интервал, указанный в textBox.

17 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку должен вызваться Калькулятор через интервал, указанный в textBox.

18 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку через интервал времени, указанный в textBox должен вызваться графический редактор Paint с указанным файлом.

19 Разработка модуля, содержащего форму, на которую помещена кнопка, элемент управления Timer. При нажатии на кнопку должен вызваться диспетчер задач через 30 секунд.

20 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку должен вызваться графический редактор Paint 3D с заданным файлом и через интервал времени, указанный в textBox, должно произойти закрытие этого файла.

21 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку через интервал времени, указанный в textBox должен вызваться графический редактор Paint 3D.

22 Разработка модуля, содержащего форму, на которую помещена кнопка, элементы управления Timer и textBox. При нажатии на кнопку через интервал времени, указанный в textBox должен вызваться графический редактор Corel Draw.

23 Разработка модуля, содержащего форму, на которую помещена кнопка и элемент управления Timer. При нажатии на кнопку должен вызваться Калькулятор, а через 20 секунд закрыться.

Контрольные вопросы

1. Для чего используется библиотека классов .NET Framework?
2. Основные функции библиотеки классов .NET Framework.
3. Каркас приложения, построенный по умолчанию, если при построении нового проекта в C# выбрать в качестве типа проекта Console Application, будет включать какие пространства имен из библиотеки классов?
4. Какое пространство имен необходимо подключить в программе для работы с базами данных, web-приложениями?
5. Какое пространство имен необходимо подключить в программе для работы таймера?
6. Какие пространства имен вы подключали в своем приложении?

Используемая литература

1. Гниденко, И. Г. Технология разработки программного обеспечения: учеб. пособие для СПО / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — М.: Издательство Юрайт, 2017.
2. Шарп Джон Ш26 Microsoft Visual C#. Подробное руководство. 8-е изд. — СПб.: Питер, 2017.
3. Васильев А.Н. Программирование на C# для начинающих. Основные сведения. – Москва: Эксмо, 2018.
4. Васильев А.Н. Программирование на C# для начинающих. Особенности языка. – Москва: Эксмо, 2019.
5. <http://msdn.microsoft.com/ru-ru/library/67ef8sbd.aspx>.