

# Лабораторная работа №7

**ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ  
АВТОМОБИЛЬНО-ДОРОЖНЫЙ УНИВЕРСИТЕТ**  
**ФАКУЛЬТЕТ МЕХАТРОНИКИ ТРАНСПОРТНЫХ СРЕДСТВ**

**Кафедра информатики**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**  
**по проведению практических работ по дисциплине «Программирование»**  
**для студентов специальности 6.050201 «Системная инженерия»**

Разработчик - доцент кафедры информатики  
кандидат технических наук,  
старший научный сотрудник  
Тимонин Владимир Алексеевич

Харків 2012

## Лабораторная работа №7

### Исследование возможностей интегрированной среды разработки Visual C# для создания приложений по управлению буфером обмена.

**Цель работы** – исследовать возможности интегрированной среды разработки Visual Studio 2010 и получить практические навыки по созданию приложений по управлению буфером обмена

#### 1. Теоретические сведения

Класс Clipboard предоставляет статические методы, которые облегчают передавать данные через буфер обмена.

Основными методами данного класса являются:

- **Clear()** - удаляет все данные из буфера обмена;
- **ContainsAudio()** - запрос на наличие данных в буфер обмена WaveAudio формат данных.
- **ContainsData(string format)** - запрос на наличие данных в указанном формате данных, где format - формат данных, в котором необходимо выполнить поиск

Основными форматами данных являются (полный перечень форматов представлен в классе DataFormats):

- **Bitmap** - определяет формат данных растрового изображения;
  - **Dif** - задает формат данных формата обмена данными windows (обмена данными);
  - **Rtf** - определяет Формат RTF (Rich Text Format) формат данных;
  - **Text** - определяет ANSI формат текстовых данных;
  - **UnicodeText** - определяет Юникод формат текстовых данных;
  - **WaveAudio** - задает волновой формат аудиоданных
  - **ContainsImage()** - запрос на наличие данных в буфер обмена Bitmap формат данных;
  - **ContainsText()** - запрос на наличие данных в буфер обмена UnicodeText формат.
- Возвращаемое значение при любом запросе – тип Boolean, если буфер обмена содержит данные заданного формата – значение true, в противном случае - false.

- **GetAudioStream()** - возвращает поток данных в буфер обмена WaveAudio формат данных;
- **GetData(string format)** - извлекает данные из буфера обмена в указанном формате, где format - строка, задающая формат данных для получения.
- **GetDataObject()** - возвращает объект данных, представляющий все содержимое буфера обмена;

- **GetImage()** - возвращает объект, содержащий данные из буфера обмена в Bitmap формат;
- **GetText()** - возвращает строку, содержащую UnicodeText данные в буфере обмена;
- **GetText(TextDataFormat)** - возвращает строку, содержащую текстовые данные из буфера обмена;

- **SetAudio(Byte[])** - сохранение звуковых данных (WaveAudio формат данных) в буфере обмена.

Аудиоданные указываются в виде массива байтов

- **SetAudio(Stream)** - сохранение звуковых данных (WaveAudio формат данных) в буфере обмена.

Аудиоданные указываются в виде потока

- **SetData(string format, Object data)** - сохраняет указанные данные в буфере обмена в указанном формате, где format - строка, задающая формат для хранения данных, data –объект, представляющий данные для сохранения в буфере обмена.

- **SetImage(BitmapSource image)** - хранит Bitmap данные в буфере обмена, где image - объект типа BitmapSource, содержащий данные изображения для сохранения в буфере обмена. Растровое изображение представляет изображения как массив битов в памяти;

- **SetText(String)** - хранит UnicodeText данные в буфере обмена;

- **SetText(String, TextDataFormat)** - сохраняет текстовые данные из буфера обмена в указанном

формате данных text. UnicodeText сохраняемые данные определяются в виде строки.



**Пример 1.** Нижеприведенный пример программного кода демонстрирует возможности записи в буфер обмена и извлечения из буфера обмена текста. Нажатие кнопки **button1** записывает текст в буфер обмена, нажатие кнопки **button2** – считывает содержимое из буфера обмена, нажатие кнопки **button3** – очищает буфер обмена.

```
private void button1_Click(object sender, EventArgs e)
{
    string Text = "Демонстрационный текст";
    Clipboard.SetDataObject(Text);
}
private void button2_Click(object sender, EventArgs e)
{
    // объявляем объект-получатель
    IDataObject MyText = Clipboard.GetDataObject();
    textBox1.Text = MyText.GetData(DataFormats.Text).ToString();
}
private void button3_Click(object sender, EventArgs e)
{
    Clipboard.Clear();
}
```



**Пример 2.** Нижеприведенный пример программного кода демонстрирует возможности отображения рисунка из буфера обмена (при реализации программного кода необходимо поместить в буфер обмена рисунок, например с помощью графического редактора Paint).

```
private void button1_Click(object sender, EventArgs e)
{
    // объявляем объект-получатель
    IDataObject MyPicture = Clipboard.GetDataObject();
    Bitmap MyBitmap;
    // Проверка - данные в буфере обмена представлены в формате Bitmap?
    if (MyPicture.GetDataPresent(DataFormats.Bitmap) == true)
    {
        MyBitmap = (Bitmap)MyPicture.GetData(DataFormats.Bitmap);
        pictureBox1.Image = MyBitmap;
    }
}
```



**Пример 3.** Нижеприведенный пример программного кода демонстрирует возможности отображения рисунка из буфера обмена (при реализации программного кода необходимо поместить в буфер обмена рисунок, например с помощью графического редактора Paint).

```
private void button4_Click(object sender, EventArgs e)
{
    // объявляем объект-получатель
    IDataObject MyPicture = Clipboard.GetDataObject();
    Bitmap MyBitmap;
    // Проверка - данные в буфере обмена представлены в формате Bitmap?
    if (MyPicture.GetDataPresent(DataFormats.Bitmap) == true)
    {
```

```

MyBitmap = (Bitmap)MyPicture.GetData(DataFormats.Bitmap);
MyBitmap.Save(@"d:\Ris1.bmp");
}
}
}

```

## 2. Рабочее задание



**Задание 1.** Разработать приложение «Работа с буфером обмена», которое позволяет продемонстрировать взаимодействие с текстовым редактором «Блокнот».

Приложение должно выполнять следующие функции:

- вставлять в компонент **TextBox** текст, скопированный в текстовом редакторе «Блокнот», и записывать в файл;
- считывать текст из файла и помещать в буфер обмена текст, скопированный в полном объеме или частично из компонента **TextBox**. С помощью текстового редактора «Блокнот» проверить работоспособность приложения.

При разработке интерфейса приложения использовать компонент **ToolStrip** (один из вариантов интерфейса представлен на рис. 1), в котором используются элементы

- **toolStripComboBox1** - предназначен для выбора файла, в который записывается текст;
- **toolStripButton1** - предназначен для выполнения команды на запись текста в файл;
- **toolStripComboBox2** - предназначен для выбора файла, из которого считывается текст в компонент **TextBox**;
- **toolStripButton2** - предназначен для выполнения команды считывания текста из файла;
- **toolStripButton3** - предназначен для выполнения команды печати текста;
- **toolStripButton4** - предназначен для выполнения команды очистки буфера обмена;
- **toolStripButton5** - предназначен для выполнения команды копирования в буфера обмена;
- **toolStripButton6** - предназначен для выполнения команды вставки из буфера обмена;
- **toolStripButton7** - предназначен для выполнения команды выдачи справки

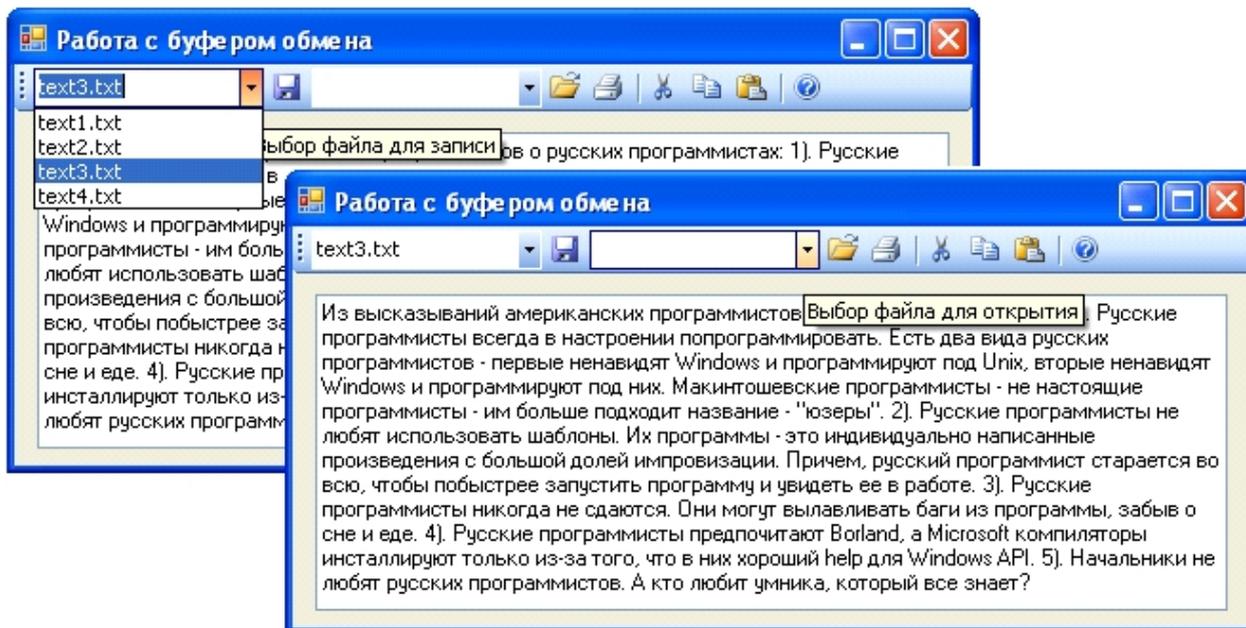


Рис. 1. Внешний вид приложения «Работа с буфером обмена»

## 3. Контрольные вопросы

### Литература

1. Голощапов А.Л. Microsoft Visual Studio 2010. – СПб.: БХВ-Петербург, 2011. – 544 с.: ил.
2. Культин НБ. Microsoft Visual C# в задачах и примерах. – СПб.: БХВ-Петербург, 2009. – 320 с.:

ил.

3. Лабор В.В. Си Шарп: Создание приложений для Windows. – Мн.: Харвест, 2003. – 384 с.
4. Петцольд Ч. Программирование для Microsoft Windows на C#. В 2-х томах. Том 1. Пер. с англ. - М.: «Русская Редакция», 2002.- 576 с.: ил.
5. Петцольд Ч. Программирование для Microsoft Windows на C#. В 2-х томах. Том 2. Пер. с англ. - М.: «Русская Редакция», 2002.- 624 с.: ил.
6. Троелсен Э. Язык программирования C# 2010 и платформа .NET 4.0. Пер. с англ. - М.: Издательский дом "Вильямс", 2011. — 1392 с.: ил.
7. Фаронов В.В. Программирование на языке C#. – СПб.: Питер, 2007. – 240 с.: ил.
8. Фленов М.Е. Библия C#. - СПб.: БХВ-Петербург, 2011.– 560с.: ил.