

Microsoft® Small Basic

Средства отладки

Предполагаемое время работы с этим уроком: 1 час



Средства отладки

В этом уроке вы изучите следующее.

Использование операции `TextWindow.WriteLine` для отладки программ.



Отладка

Отладка используется для обнаружения и устранения неполадок компьютерных программ. Все программисты развивают свои навыки обнаружения ошибок и отладки программ.

Даже незначительная ошибка в программе может свести на нет все усилия программиста!

При работе с большими программами программисты создают вспомогательный код отладки, не включаемый в итоговую программу. Этот код помогает определять место, в котором происходит сбой выполнения программы.



TextWindow.WriteLine как вспомогательное средство отладки

Как же выполняется отладка программ в Small Basic?

Для отладки программы используется операция **TextWindow.WriteLine**. Эта операция выступает в качестве вспомогательной программы отладки и предоставляет сведения, помогающие выполнить отладку программы.



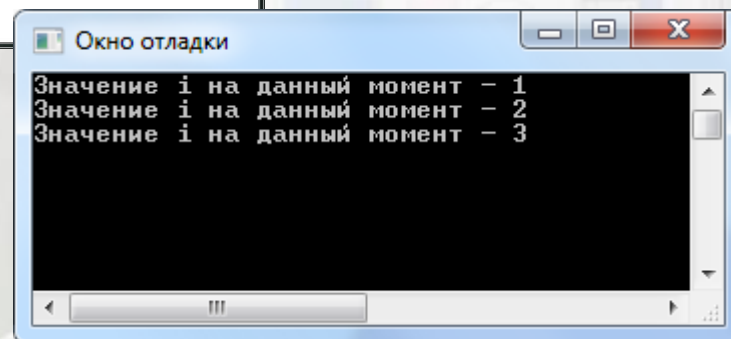
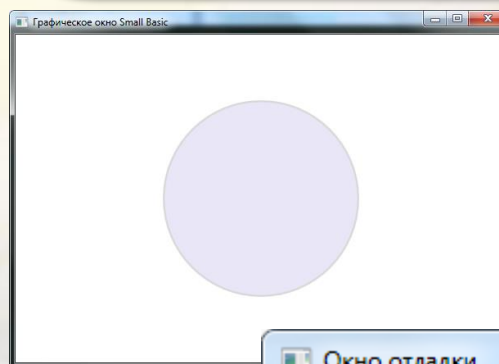
TextWindow.WriteLine как вспомогательное средство отладки

Рассмотрим пример использования операции **TextWindow.WriteLine** для отладки.

В этой программе сначала в графическом окне отображается эллипс. Если программа выполняется не так, как ожидается, необходима ее отладка! В этот момент можно использовать операцию **TextWindow.WriteLine** в качестве отладчика. Эта операция помогает проследить значение «i» для каждой итерации цикла **For...EndFor**. Значение «i» отображается в отдельном текстовом окне, позволяя просто обнаружить ошибку и исправить ее.

```
GraphicsWindow.Height = 400
GraphicsWindow.Width = 600
Ellipse = Shapes.AddEllipse(200,200)
Shapes.Move(Ellipse,200,100)

For i = 1 To 5
  Program.Delay(1000)
  Shapes.SetOpacity(Ellipse,5*i)
  Shapes.Zoom(Ellipse,i * 0.4,i * 0.4)
  TextWindow.WriteLine("Значение i на данный момент - " + i)
EndFor
```

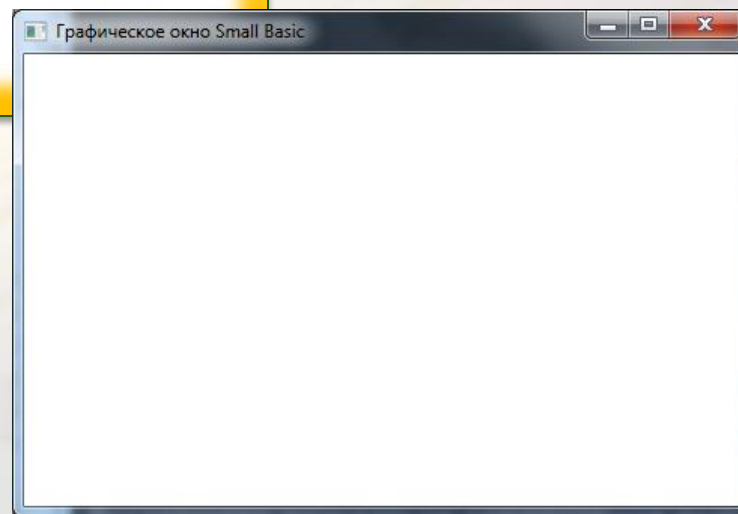


TextWindow.WriteLine как вспомогательное средство отладки

Теперь используем операцию **TextWindow.WriteLine** в другой программе и посмотрим, как она может помочь в отладке программы.

```
GraphicsWindow.Show()  
GraphicsWindow.DrawImage(Flickr.GetPictureOfMoment(), 0, 0)  
Program.Delay(2000)  
GraphicsWindow.Clear()
```

В этом примере объект **Flickr** используется для загрузки фотографии с веб-сайта www.flickr.com и ее отображения в графическом окне.

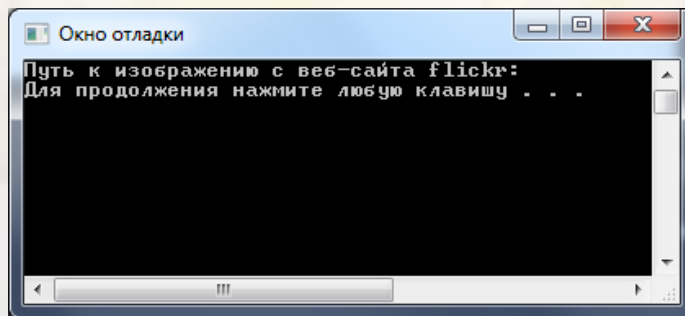
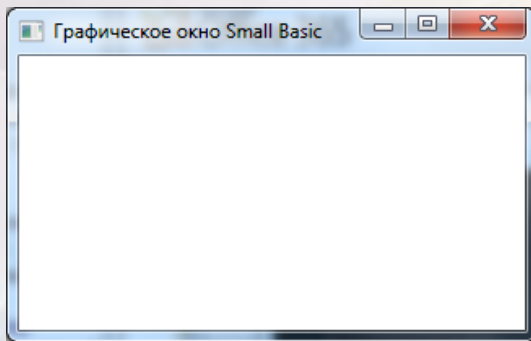


Запрошенное изображение не появляется в графическом окне. Пора приступить к отладке кода!

TextWindow.WriteLine как вспомогательное средство отладки

Возможно, отсутствует подключение к Интернету или у вас нет доступа к веб-сайту.

```
GraphicsWindow.Show()  
GraphicsWindow.DrawImage(Flickr.GetPictureOfMoment(), 0, 0)  
TextWindow.WriteLine("Путь к изображению с веб-сайта flickr: " + Flickr.GetPictureOfMoment())  
Program.Delay(2000)  
GraphicsWindow.Clear()
```



Как вы может видеть, в текстовом окне не отображается путь изображения, это означает, что отсутствует подключение к Интернету.

Для проверки этого можно использовать операцию **TextWindow.WriteLine**, чтобы отобразить путь изображения с веб-сайта. Если подключение к Интернету работает верно, в текстовом окне появится путь к изображению. В противном случае в текстовом окне путь не будет отображен.

TextWindow.WriteLine как вспомогательное средство отладки

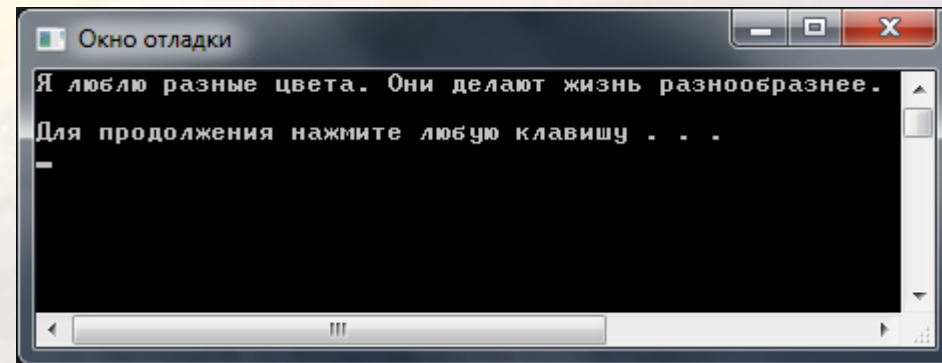
Теперь напишем программу для копирования файла и отображения его содержимого в текстовом окне.

В этом примере при успешном выполнении операции **CopyFile** в текстовом окне отображается содержимое файла.

```
Sourcepath = Program.Directory + "\\Source\data.txt"  
Destpath = Program.Directory + "\\dest"  
File.CopyFile(Sourcepath, Destpath)  
TextWindow.WriteLine(File.ReadContents(destPath + "\\data.txt"))
```

Содержимое файла может не отображаться в текстовом окне по следующим причинам.

- Указанный источник или назначение могут быть неверными.
- Файл может отсутствовать в указанном источнике.
- Имя указанного файла может быть неверным.

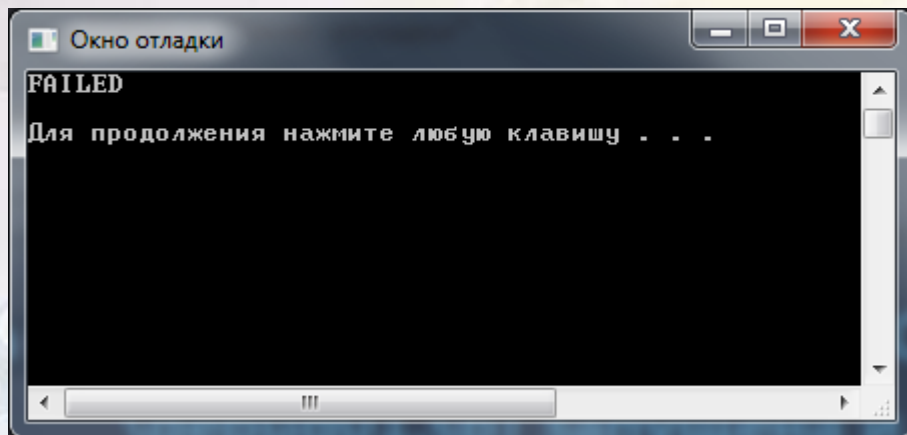


Как определить ошибку? Пора снова приступить к отладке кода...

TextWindow.WriteLine как вспомогательное средство отладки

Используем операцию **TextWindow.WriteLine** для обнаружения ошибки.

```
Sourcepath = Program.Directory + @"\Source\mydata.txt"  
Destpath = Program.Directory + "\dest"  
TextWindow.WriteLine(File.CopyFile(Sourcepath, Destpath))  
TextWindow.WriteLine(File.ReadContents(destPath + "\data.txt"))
```



Можно изменить программу и использовать **TextWindow.WriteLine** для отображения возвращенного значения операции **CopyFile** в текстовом окне. Возвращенное значение может быть SUCCESS (Успешно) или FAILED (Сбой). Поэтому, если программа не выполняется из-за ошибки операции **CopyFile**, вы немедленно узнаете об этом!

Подведем итоги...



Поздравляем! Вы изучили следующее.

- ✚ Использование операции **TextWindow.WriteLine** для отладки программ.

Продемонстрируйте свои знания

Программа создает слайд-шоу изображений в графическом окне. Все эти изображения хранятся в локальной папке. При каждом нажатии клавиши мыши изображение в графическом окне меняется.

Использование операции **TextWindow.WriteLine** в качестве вспомогательной программы отладки для отображения пути каждого изображения в текстовом окне.

