

Сохранение и чтение массивов при помощи OpenFileDialog и SaveFileDialog в Pascal ABC NET.

Окна открытия и сохранения файла представлены классами `OpenFileDialog` и `SaveFileDialog`. Они имеют во многом схожую функциональность, поэтому рассмотрим их вместе.

`OpenFileDialog` и `SaveFileDialog` имеют ряд общих свойств, среди которых можно выделить следующие:

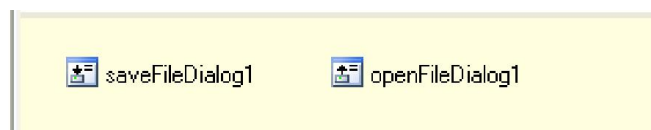
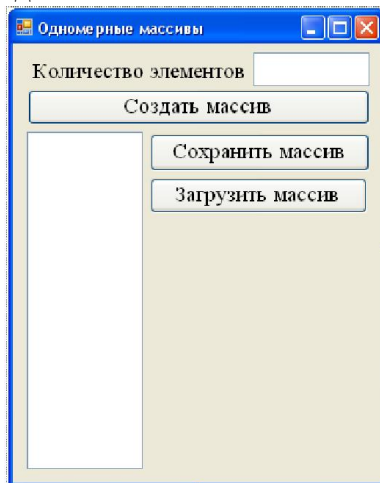
- `DefaultExt`: устанавливает расширение файла, которое добавляется по умолчанию, если пользователь ввел имя файла без расширения
- `AddExtension`: при значении `true` добавляет к имени файла расширение при его отсутствии. Расширение берется из свойства `DefaultExt` или `Filter`
- `CheckFileExists`: если имеет значение `true`, то проверяет существование файла с указанным именем
- `CheckPathExists`: если имеет значение `true`, то проверяет существование пути к файлу с указанным именем
- `FileName`: возвращает полное имя файла, выбранного в диалоговом окне
- `Filter`: задает фильтр файлов, благодаря чему в диалоговом окне можно отфильтровать файлы по расширению. Фильтр задается в следующем формате `Название_файлов|*.расширение`. Например, `Текстовые файлы(*.txt)|*.txt`. Можно задать сразу несколько фильтров, для этого они разделяются вертикальной линией `|`. Например, `Bitmap files (*.bmp)|*.bmp|Image files (*.jpg)|*.jpg`
- `InitialDirectory`: устанавливает каталог, который отображается при первом вызове окна
- `Title`: заголовок диалогового окна

Отдельно у класса `SaveFileDialog` можно еще выделить пару свойств:

- `CreatePrompt`: при значении `true` в случае, если указан не существующий файл, то будет отображаться сообщение о его создании
- `OverwritePrompt`: при значении `true` в случае, если указан существующий файл, то будет отображаться сообщение о том, что файл будет перезаписан

Чтобы отобразить диалоговое окно, надо вызвать метод `ShowDialog()`.

Добавим на форму, создающую массив две дополнительные кнопки и два не визуальных компонента `OpenFileDialog` и `SaveFileDialog`. Они отображаются не на форме, а внизу на отдельной панели.



Процедуры сохранения и чтения массива будут выглядеть следующим образом:

```
procedure Form1.button2_Click(sender: Object; e: EventArgs);  
var f:String;  
begin  
    // настраиваем фильтр при сохранении файла  
    saveFileDialog1.Filter := 'Text files (*.txt)|*.txt|All files (*.*)|*.*';  
    // вызываем метод открытия диалогового окна  
    saveFileDialog1.ShowDialog();  
    // в f будем хранить имя файла для сохранения массива  
    f := saveFileDialog1.FileName;  
    // сохраняем содержимое textBox2.Text в файл  
    System.IO.File.WriteAllText(f, textBox2.Text);  
end;  
  
procedure Form1.button3_Click(sender: Object; e: EventArgs);  
var f:String;  
begin  
    // настраиваем фильтр при открытии файла  
    openFileDialog1.Filter := 'Text files (*.txt)|*.txt|All files (*.*)|*.*';  
    // вызываем метод открытия диалогового окна  
    openFileDialog1.ShowDialog();  
    // в f будем хранить имя файла из которого будем считывать массив  
    f := openFileDialog1.FileName;  
    // переносим содержимое файла в textBox2.Text  
    textBox2.Text := System.IO.File.ReadAllText(f);  
    // что бы пользователь пощелкал мышкой :)  
    MessageBox.Show('                Файл открыт                ');  
end;
```