[Перейти на главную страничку сайта (список статей, файлы для скачивания)](http://www.script-coding.com/index.html)

[ФОРУМ (здесь можно обсудить эту статью, а также любые проблемы программирования на различных макроязыках и в скриптовых средах)](http://forum.script-coding.com/)

**Библиотека svcsvc.dll и её объектная модель**

**Назначение и установка**

Библиотека svcsvc.dll - небольшой ActiveX, который может быть использован в любых средах, которые в состоянии выступить в роли OLE-клиента, например, в MS Office (VBA), 1C:Предприятии, административных скриптах Windows (.vbs и .js) и т.д. Авторы программы (в порядке подключения к проекту): Александр Орефков, Алексей Антипин, Дмитрий Трошин, Бердаков Дмитрий, Алексей Диркс, Михаил Клопков. Библиотека svcsvc.dll распространяется свободно. Вы можете скачать библиотеку [здесь](http://www.script-coding.com/svcsvc.zip) (версия 1.0.0.2 от 18.12.2004г., архив 24 141 байт).

Возможности, предоставляемые библиотекой:

* Преобразование строк и целых текстовых файлов из кодировки ANSI в OEM и обратно.
* Вызов стандартных диалогов Windows для интерактивного выбора файла или папки пользователем.
* Интерактивный выбор пользователем значений из списка (в т.ч. древовидного, а также списка с флажками). Возможность вывода списка под позицию курсора или мыши, возможность интерактивной фильтрации списка, в т.ч. по вхождению нескольких подстрок.
* Вывод контекстного меню, которое может подстраиваться под позицию курсора или мыши.

Порядок установки:

* Скопируйте библиотеку на жёсткий диск вашего компьютера, например в такой каталог:   
    
  **C:\WINDOWS\system32\**
* Зарегистрируйте библиотеку с помощью **regsvr32.exe**. Командная строка регистрации библиотеки может выглядеть, например, так:   
    
  **regsvr32.exe "C:\WINDOWS\system32\svcsvc.dll"**

ProgID единственного объекта Service, предоставляемого библиотекой - "Svcsvc.Service".

**Методы**

|  |  |
| --- | --- |
| **Метод** | **Описание** |
| AnsiToOEM(ByVal Ansi As String) As String | Преобразовывает строку из кодировки ANSI в OEM. |
| OEMToAnsi(ByVal OEM As String) As String | Преобразовывает строку из кодировки OEM в ANSI. |
| FileA2O(ByVal Path As String) | Преобразовывает текстовый файл из кодировки ANSI в OEM. |
| FileO2A(ByVal Path As String) | Преобразовывает текстовый файл из кодировки OEM в ANSI. |
| SelectFile(ByVal ForSave As boolean, ByVal StartFileName As String, ByVal Filter As String, ByVal MultiSelect As boolean = False) As String | Открывает стандартный диалог выбора файла, возвращает путь к выбранному файлу или пустую строку. Параметры:   * ForSave - True для диалога "Сохранить", False для диалога "Открыть". * StartFileName - Начальное имя файла. При указании полного пути диалог откроется на заданной папке. * Filter - Фильтр файлов. Представляет собой строку с описаниями фильтров и их масками, разделенные символом "|". Например: "**Текстовые файлы (\*.txt)|\*.txt|Все файлы|\***". * MultiSelect - True для выбора нескольких файлов, False для выбора одного файла. Необязательный (по умолчанию - False). |
| SelectFolder(ByVal Caption As String, ByVal StartFolder As String = "", ByVal Flags As Long = 0, ByVal Root = "") As String | Открывает стандартный диалог выбора папки, возвращает путь к выбранному файлу или папке или пустую строку. Параметры:   * Caption - текст подсказки. * StartFolder - начальный каталог (необязательный). * Flags - опции (необязательный, по умолчанию - 0). Возможные значения (значения можно суммировать):   + &H1 - включать только объекты файловой системы.   + &H10 - отображать поле ввода.   + &H1000 - искать компьютеры.   + &H2000 - искать принтеры.   + &H4000 - включать файлы. * Root - корневая папка (необязательный). Пользователь не сможет пройти в окне выбора выше этой папки. Если параметр не указан, корневой папкой будет рабочий стол Windows. |
| SelectValue(ByVal Values As String, ByVal Caption As String, ByVal Mark As boolean = False) As String | Выбор значения из списка (возможно с флажками). Возвращает выбранное значение. При пометке нескольких значений возвращает все помеченные значения, разделенные переводами строк. Параметры:   * Values - Строка со значениями для выбора. Каждое значение должно располагаться на отдельной строке (через перевод строки). При отметке нескольких значений в конце каждого значения можно указать "|c" (латинская). В этом случае значение изначально будет отмечено. * Caption - Заголовок окна выбора. * Mark - True для возможности пометки нескольких значений, False для выбора одного значения. Необязательный (по умолчанию - False).   Пользователь может изменять размер диалога для выбора значений. |
| SelectInTree(ByVal Values As String, ByVal Caption As String, ByVal Mark As boolean = False, ByVal OnlyItems As boolean = True) As String | Выбор значение из дерева значений с флажками. Параметры:   * Values - строка со значениями, каждое из которых размещено в отдельной строке (через перевод строки). Для задания вложенных значений используется символ табуляции в начале строки. После каждого значения через символ "|" можно указать модификаторы:   + c или C (от Checked) - значение изначально будет отмечено.   + e или E (от Expanded)- узел изначально будет "развёрнут". * Caption - заголовок окна. * Mark - True для отметки нескольких значений, False для выбора одного значения. Необязательный (по умолчанию - False). * OnlyItems - необязательный (по умолчанию - True). Действует только при выборе одного значения. True - выбирать только значения, не имеющие потомков, False - выбирать любое значение.   Метод возвращает при отмене пустую строку, при выборе - полный "путь" к выбранному значению, разделенный "\". При пометке нескольких значений возвращает полные "пути" к каждому выбранному значению, разделенные переводами строк. |
| FilterValue(ByVal Values As String, ByVal Flags As Short = 0, ByVal Caption As String = "", ByVal X As Long=0, ByVal Y As Long = 0) As String | Открывает окошко выбора из списка с возможностью фильтрации по вхождению нескольких подстрок (вводить через пробел). По мере ввода текста список автоматически фильтруется. Параметры:   * Values - строка со значениями для выбора, каждое из которых расположено на отдельной строке (через переводы строк). * Flags - опции. Возможные значения (значения можно суммировать):   + 1 - фильтровать по вхождению подстрок. Если флаг не указан, значения будут фильтроваться с начала строки.   + 2 - вывести окно в позиции мыши.   + 4 - вывести окно в позиции курсора.   + 8 - вывести окно в указанных координатах.   + 16 - допускать ввод своих значений. В этом случае функция возвратит то, что пользователь набрал в окне ввода. Иначе будет возвращён выбранный пункт списка.   + 32 - использовать заголовок. В этом случае список выбора выведется с указанным заголовком. * Caption - заголовок для списка в случае, если установлен флаг 32. Необязательный (по умолчанию - ""). * X - X-координата для вывода окошка (необязательный). * Y - Y-координата для вывода окошка (необязательный). |
| PopupMenu(ByVal Values As String, ByVal ShowIn As short = 0, X As Long =0, Y As Long=0) As String | Отображает контекстное меню. Параметры:   * Values - строка со значениями, каждое из которых размещено в отдельной строке (через перевод строки). Для задания вложенных значений используется символ табуляции в начале строки. В каждом значении можно через символ "|" указать флаг и идентификатор пункта меню. Флаги:   + c или C (от Checked) - пункт меню будет с галочкой.   + d или D (от Disabled) - пункт будет недоступен.   Например: "**НекийПункт|c|Command1**".   * ShowIn - вариант "привязки меню" (необязательный, по умолчанию - 0):   + 0 - меню будет выведено в текущем расположении указателя мыши.   + 1 - меню будет выведено в текущем расположении курсора.   Любое другое значение - меню будет выведено в указанных координатах.   * X - X-координата для вывода меню (необязательный). * Y - Y-координата для вывода меню (необязательный).   Метод возвращает строку с выбранным пунктом меню. Если для пункта меню задан идентификатор, то будет возвращён идентификатор. |

**Примеры кода**

Примеры кода приводятся на языке VBScript для административных сценариев Windows.

Преобразования строк из одной кодировки в другую:

**Set SvcService = CreateObject("Svcsvc.Service")  
Set FSO = CreateObject("Scripting.FileSystemObject")  
Set WshShell = CreateObject("WScript.Shell")  
  
'Создание текстового файла C:\Test.txt  
Set TxtStream = FSO.OpenTextFile("C:\Test.txt", 2, True)  
'Запись в файл строки в кодировке OEM  
TxtStream.Write(SvcService.AnsiToOEM("Строка в кодировке OEM"))  
TxtStream.Close  
'Отображение файла в редакторе MS-DOS  
WshShell.Run "edit C:\Test.txt", 1, True  
  
'Открытие текстового файла C:\Test.txt на чтение  
Set TxtStream = FSO.OpenTextFile("C:\Test.txt", 1)  
MsgBox SvcService.OEMToANSI(TxtStream.ReadAll)  
TxtStream.Close  
'Удаление файла C:\Test.txt  
FSO.DeleteFile "C:\Test.txt", True**

Преобразования текстового файла целиком из одной кодировки в другую:

**Set SvcService = CreateObject("Svcsvc.Service")  
Set FSO = CreateObject("Scripting.FileSystemObject")  
Set WshShell = CreateObject("WScript.Shell")  
  
'Создание текстового файла C:\Test.txt  
Set TxtStream = FSO.OpenTextFile("C:\Test.txt", 2, True)  
'Запись в файл строки в кодировке OEM  
TxtStream.Write(SvcService.AnsiToOEM("Строка в кодировке OEM"))  
TxtStream.Close  
'Отображение файла в редакторе MS-DOS  
WshShell.Run "edit C:\Test.txt", 1, True  
  
'Преобразование файла в кодировку ANSI  
SvcService.FileO2A "C:\Test.txt"  
'Отображение файла в Notepad'е  
WshShell.Run "notepad C:\Test.txt", 1, True  
  
'Преобразование файла обратно в кодировку OEM  
SvcService.FileA2O "C:\Test.txt"  
'Отображение файла в редакторе MS-DOS  
WshShell.Run "edit C:\Test.txt", 1, True  
'Удаление файла C:\Test.txt  
FSO.DeleteFile "C:\Test.txt", True**

Вызов стандартного диалога Windows для интерактивного выбора файла пользователем:

**Set SvcService = CreateObject("Svcsvc.Service")  
Sel = SvcService.SelectFile(False, "C:\boot.ini", "INI-файлы|\*.ini")  
MsgBox "Выбран файл: """ & Sel & """"**

Вызов стандартного диалога Windows для интерактивного выбора папки пользователем:

**Set SvcService = CreateObject("Svcsvc.Service")  
Sel = SvcService.SelectFolder("Выбираем папку", "C:\Program Files", &H10 + &H4000, "C:\Program Files")  
MsgBox "Выбран путь: """ & Sel & """"**

Выбор значения из списка:

**Set SvcService = CreateObject("Svcsvc.Service")  
Sel = SvcService.SelectValue("раз" & vbCrLf & "два" & vbCrLf & "три", "Заголовок", True)  
MsgBox "Выбрано: """ & Sel & """"**

Выбор значения из древовидного списка:

**Set SvcService = CreateObject("Svcsvc.Service")  
Tree = "Часть №1" & vbCrLf & vbTab & "Значение №1" & vbCrLf & vbTab & "Значение №2" & vbCrLf & "Часть №2"  
Sel = SvcService.SelectInTree(Tree, "Выбор из дерева", True, False)  
MsgBox "Выбрано: " & vbCrLf & Sel**

Выбор значения из списка с возможностью фильтрации:

**Set SvcService = CreateObject("Svcsvc.Service")  
Sel = SvcService.FilterValue("раз" & vbCrLf & "два" & vbCrLf & "три", &H1+&H2+&H20, "Заголовок")  
MsgBox "Выбрано """ & Sel & """"**

Вывод контекстного меню:

**Set SvcService = CreateObject("Svcsvc.Service")  
Tree = "Часть №1" & vbCrLf & vbTab & "Значение №1|c|MyMenu" & vbCrLf & vbTab & "Значение №2|c" & vbCrLf & "Часть №2|d"  
Sel = SvcService.PopupMenu(Tree, 0)  
MsgBox "Выбрано """ & Sel & """"**

**Версия 1.0.0.3**

Вы можете скачать библиотеку [здесь](http://www.script-coding.com/svcsvc1003.zip) (версия 1.0.0.3 от 27.03.2006г., архив 24 837 байт). Все изменения касаются метода *FilterValue*:

* Новый флаг "64" во втором параметре Flags: использовать параметр заголовка как инициализирующее значение фильтра, то есть третий параметр (заголовок) устанавливается как текст в поле ввода и список появляется уже отфильтрованным по этому значению.
* Шестой необязательный параметр метода - ширина списка в пикселах (если 0 или не указан, то список выводится с шириной по умолчанию).
* Седьмой необязательный параметр метода - первоначальная высота списка в пикселах.
* При смене текущей строки списка текст в поле ввода не изменяется. Отсюда: если есть "16" во флагах (допускать ввод своих значений), при нажатии Enter метод возвращает выделенную строку списка, Esc - строку, введённую в поле ввода. При автоматической фильтрации списка (при наборе текста) строка, совпадающая со строкой в поле ввода, автоматически выделяется.
* При нажатии вверх/вниз в начале/конце списка выделяется последняя/первая строка (осуществляется "циклический" переход по списку).
* При открытии списка всегда выделен первый пункт (раньше это было только при отсутствии "16" во флагах).
* Выводится объемная рамка для поля ввода, а для элементов списка выводятся иконки.

**Версия 1.0.0.4**

Вы можете скачать библиотеку [здесь](http://www.script-coding.com/svcsvc1004.zip) (версия 1.0.0.4 от 12.04.2006г., архив 25 489 байт). Все изменения касаются метода *FilterValue*:

* Если шестой необязательный параметр метода равен 1, список выводится с автошириной по самому длинному пункту.
* Если при перемещении по списку нажат Shift, текст текущего пункта списка переносится в поле ввода.
* Нажатия Ctrl+Home/Ctrl+End переносят курсор в начало/конец текста в поле ввода.

**Версия 1.0.0.5**

Вы можете скачать библиотеку [здесь](http://www.script-coding.com/svcsvc1005.zip) (версия 1.0.0.5 от 10.05.2006г., архив 26 845 байт). Добавлены свойства и методы:

|  |  |
| --- | --- |
| GetWindowText(WH, IsList) | Получает текст окна. Параметры:   * WH - необязательный, по умолчанию 0. Хэндл окна, текст которого нужно получить. Если 0 или не указан - используется текущий фокус. * IsList - необязательный, по умолчанию False. Если True, передаваемый хэндл считается списком (ListBox) или выпадающим списком (ComboBox) и возвращается его содержимое, разбитое разделителями строк \r\n. |
| SetWindowText(WH, Text) | Устанавливает текст окна. Параметры:   * WH - необязательный, по умолчанию 0. Хэндл окна, которому нужно установить текст. Если 0 или не указан - используется текущий фокус. * Text - текст, который необходимо установить. |
| Version | Текущая версия библиотеки. |

Добавлен новый флаг "128" во втором параметре *Flags* метода *FilterValue*: список заполняется из активного списка (ListBox) или выпадающего списка (ComboBox), а затем этот список позиционируется на выбранном значении.

Пример вызова метода FilterValue:

**Set Svc = CreateObject("Svcsvc.Service")**

**Svc.FilterValue "", 1 + 128, "", 0, 0, 1**

Пример получения текста активного окна:

**Set Svc = CreateObject("Svcsvc.Service")**

**MsgBox Svc.GetWindowText()**

Пример установки текста активного окна:

**Set Svc = CreateObject("Svcsvc.Service")**

**Svc.SetWindowText , "Текст!"**

**Версия 1.0.0.6**

Вы можете скачать библиотеку [здесь](http://www.script-coding.com/svcsvc1006.zip) (версия 1.0.0.6 от 05.01.2007г., архив 21 529 байт). Исправлены мелкие ошибки при работе с визуальными списками (метод *FilterValue*).

**Версия 1.0.0.8**

Вы можете скачать библиотеку [здесь](http://www.script-coding.com/svcsvc1008.zip) (версия 1.0.0.8 от 23.01.2007г., архив 31 201 байт). Изменения:

* Сортировка списка фильтра значений (FilterValue, флаг 256).
* Автоширина списка фильтра значений (FilterValue, флаг 512).
* Добавлены пиктограммы из библиотеки SelectDlg, входящей в поставку скриптов для [OpenConf](http://www.script-coding.com/OpenConf.html), для отображения в списках. Формат задания значений списка:**НомерПиктограммы#Значение|ДополнительныеДанные**, т.е. символ '#' разделяет номер пиктограммы и значение, а '|' - значение и данные.
* Указание флага FilterValue 16 больше не препятствует возврату дополнительных значений (заданных через разделитель "|" в списке).
* Устранена потеря GDI объектов при каждом вызове FilterValue (количество GDI объектов можно наблюдать в диспетчере задач для каждого процесса).
* Добавлен метод **SetImageList(ByVal Values As String)** для задания произвольного списка пиктограмм. Формат задания пиктограмм: строка с путями к пиктограммам, разделенными символами перевода строк.
* Устранен нежелательный селект текста при вызове FilterValue с флагом 32.

Пример, иллюстрирующий использование стандартных и собственных пиктограмм:

**Function Add(str, str1, str2, str3)**

**Add = str & VbCrLf & str1 & "#" & str2 & "|" & str3**

**End Function**

**Set obj = CreateObject("Svcsvc.Service")**

**str = ""**

**str = Add(str, "1", "Первый", "1-й")**

**str = Add(str, "2", "Второй", "2-й")**

**str = Add(str, "3", "Третий", "3-й")**

**str = Add(str, "4", "Четвертый", "4-й")**

**str = Add(str, "5", "Пятый", "5-й")**

**str = Add(str, "6", "Шестой", "6-й")**

**str = Add(str, "7", "Седьмой", "7-й")**

**str = Add(str, "8", "Восьмой", "8-й")**

**str = Add(str, "9", "Девятый", "9-й")**

**str = Add(str, "10", "Десятый", "10-й")**

**str = Add(str, "11", "Одиннадцатый, здесь дефолтная картинка!", "11-й")**

**str = Add(str, "12", "Двенадцатый, это очень очень очень длинный тест автоширины окна", "12-й")**

**' стандартные картинки (10 штук)**

**rez = obj.FilterValue(str, 1+2+16+32+512, "выбор", 0, 0, 0, 0)**

**MsgBox rez**

**' устанавливаем произвольные картинки**

**' картинка "C:\WINDOWS\Пузыри.bmp" существует, а "C:\НесуществующаяКартинка.bmp" - нет (тест на ошибку)**

**img = "C:\WINDOWS\Пузыри.bmp" & VbCrLf & "C:\НесуществующаяКартинка.bmp"**

**' поскольку "НесуществующаяКартинка.bmp" загрузить не удалось -**

**' возврат будет "НесуществующаяКартинка.bmp" (ошибочные картинки)**

**MsgBox obj.SetImageList(img)**

**' выбор с установленными картинками (одна картинка)**

**rez = obj.FilterValue(str, 1+2+16+32+512, "выбор", 0, 0, 0, 0)**

**MsgBox rez**

**' устанавливаем стандартные картинки обратно**

**MsgBox obj.SetImageList("")**

**' снова стандартные картинки**

**rez = obj.FilterValue(str, 1+2+16+32+512, "выбор", 0, 0, 0, 0)**

**MsgBox rez**

**Версия 1.0.1.0**

Вы можете скачать библиотеку [здесь](http://www.script-coding.com/svcsvc1010.zip) (версия 1.0.1.0 от 01.02.2007г., архив 31 477 байт). Исправлена ошибка метода *SelectFile*, приводившая к невозможности выбора большого количества файлов с флагом MultiSelect - True для выбора нескольких файлов.

Составление - Людоговский Александр

[Перейти на главную страничку сайта (список статей, файлы для скачивания)](http://www.script-coding.com/index.html)

© 2007 [http://www.script-coding.com](http://www.script-coding.com/) При любом использовании материалов сайта обязательна **ссылка** на него как на источник информации, а также сохранение **целостности** и **авторства** материалов.