

# **Sun City Summerlin Computer Club**

## **Seminar**

### **Beginning Programming With Visual Basic Script Edition**

**Tom Burt  
August 31, 2006**

This seminar is an experiment. It will present a quick overview of the fundamentals of computer programming, including two “real world” examples.

The audience is experienced computer users who can think logically and translate a procedure or other activity into a sequence of steps.

For this introduction, we will mainly use the VBScript language, which is a dialect of Microsoft BASIC. It is very similar to Visual Basic for Applications (VBA), which is built into most MS Office applications. It is also similar to the Visual Basic 6 programming language.

The VBScript language runtime executive and programming documentation are available free from Microsoft’s Windows Scripting web site. (See the web links at the end of the presentation).

If time permits, we’ll also look briefly at Microsoft’s FREE VB.Net 2005 Express Edition.

## Where to Find the Materials

---

### Sun City Summer Computer Club Website Seminars:

- <http://www.scsc.com/smnr>
  
- Acrobat file of these slides and Notes:
  - [http://www.scsc.com/smnr/VBScript\\_Beginning\\_Programming.pdf](http://www.scsc.com/smnr/VBScript_Beginning_Programming.pdf)
  
- ZIP file of the Examples:
  - [http://www.scsc.com/smnr/Programming\\_Examples.zip](http://www.scsc.com/smnr/Programming_Examples.zip)

TNB August 31, 2006

Intro - Programming VBScript

## Seminar Agenda

---

- Visual Basic Versions and Reference
- Basic Programming Concepts
- VBSCRIPT Language
- -- *Bio Break (~10 min)*
- Program 1 – Ticket Draw
- Program 2 – Clean out Temp folders
- Web Site Links
- Open Q and A

TNB August 31, 2006

Intro - Programming VBScript

We'll spend the first half of the seminar covering the key programming concepts of the VBScript language. VBScript can be used in stand-alone programs, but it also very commonly used in web-site development and in web pages with dynamic content.

Most of these concepts are common to all modern programming languages, though syntax and library functions will differ.

Other popular scripting languages include:

Javascript / Jscript, PERL, PYTHON, PHP

Popular program (EXE) development languages include:

C, C++, Java, VB 6, VB.Net, C#.Net

Complete Eclipse Java Developments can be downloaded free from:

Sun Microsystems and from IBM.

## VB Versions and Reference

---

- VBA is the Visual Basic engine in Excel, Word, Access.
- VB6 is a full featured Windows application tool.
- VB.Net 2005 Express Edition is latest – Free from MS.
- VBSCRIPT language syntax is similar to VBA, VB6
  - Variables are not typed.
  - Fewer statements and functions.
  - Supports ActiveX (OLE) automation.
  - FileSystem object used for most file/folder actions.
  - Weak Developer interface environment (MSE7).
  - Can download a script debugger to help find errors.
  - Good language for standalone scripts.
  - Often used for “active content” in web pages.

<http://www.microsoft.com/technet/scriptcenter/default.mspx>

TNB August 31, 2006

Intro - Programming VBScript

Because it is a scripting language, VBScript is not quite as rich as the full-featured versions of Visual Basic. It's ability to display an interactive graphical user interface is limited.

One way to get visually appealing “graphical“ output is for the script to write a web page into a temporary HTML file and then display that page with IE. Another approach, if you have a web development tool like ASP Web Matrix is to write your programs as a series of web pages and Active Server Pages with embedded VBScript or Javascript. To run Active Server Pages, you need to have the MS “Web Server” (IIS 5 or 6) installed and activated.

VBScript interfaces to the file system and operating system through various standard “objects” that are, effectively, libraries of functions.

## Basic Programming Concepts

---

- **Coding and Running Programs**
- **Variables and Arrays**
- **Constants**
- **Assignment statements**
- **Operators**
- **Flow Control**
  - **If ... Then ... Else ... Endif**
  - **Loops**
  - **Select Case**
  - **Goto**
- **Console Input / Output**
- **VBScript Built-In Library Functions**
- **Data Persistence**

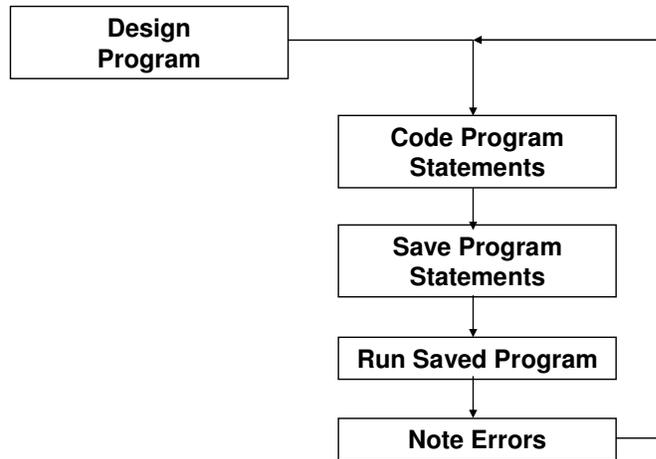
TNB August 31, 2006

Intro - Programming VBScript

The slide above lists the topical areas we will cover before turning to the two sample programs.

## Programming Cycle

---



TNB August 31, 2006

Intro - Programming VBScript

Computer programming is a hybrid of engineering, craft and art.

I find the activity very akin to clay sculpture or woodworking.

You need to come up with a clear idea of where you want to go before you start.

Programming builds up a working whole incrementally. The process includes successive iteration and refinement until the final result is “good enough” to deploy.

One of the challenges, especially as a program becomes complex, is figuring out why it is not working as intended.

VBScript provides a script debugger, which can help locate and diagnose errors in a program.

Our script editor program will be the Microsoft Script Editor (MSE7.EXE), which is provided as part of MS Office. It’s pretty lame, but way better than Notepad.

## Running VBScript Programs

---

- **Save program file as a “.vbs” file.**
  - Text file on your hard drive.
- **Double click the “.vbs” file to run it.**
  - Windows will invoke the scripting runtime to “run” the file.
  - VBScript programs are “interpreted” by the runtime
- **Not easy to stepwise debug the program.**
  - Script debugger is available.
  - Gives some help in finding errors.
- **Can add MsgBox statements to see what’s happening.**

TNB August 31, 2006

Intro - Programming VBScript

When the scripting runtime is installed, it will register as the handler for “.VBS” files.

To run a script just double click on the saved “.VBS” file.

Often, to help follow the script program’s flow and to see what values variables currently have, it is helpful to embed temporary MsgBox statements. MsgBox can display a string and will pause for a user OK before continuing.

## VBScript Variables and Arrays

---

- **Placeholders for values that change during execution**
- **Variable names – declared with DIM statement**
  - Start with a letter.
  - Can have letters, numbers and underscores.
  - E.g.: i, tLast\_Name, yTop, xLeft, Todays\_Date
- **Use simple names for small integers, loop counters, indexes**
- **Use descriptive names for less familiar variables**
  - Makes code more self-documenting.
  - Avoid really long names.
  - VBScript variable names are not case-sensitive.
- **Variables' "scope" depends on where they're declared**
  - Internal to a function or subroutine (private to that module).
  - External to a function or subroutine (accessible by all).

TNB August 31, 2006

Intro - Programming VBScript

```
' In-line calculation
Dim fWidth, fHeight, fDiagonal
fHeight = 3.0
fWidth = 4.0
fDiagonal = Sqr( (fHeight * fHeight ) + (fWidth * fWidth ) )
Msgbox "fDiagonal = " & fDiagonal
=====
' Calculate diagonal using Pythagorean theorem
Function CalcDiagonal (fWidth, fHeight)
    CalcDiagonal = Sqr((fHeight * fHeight ) + (fWidth * fWidth ) )
End Function
Msgbox "CalcDiagonal = " & CalcDiagonal
=====
' Arrays
Dim Values(20)
Values(0)=1.5
Values(1) = 3.0
```

## VBScript Constants

---

- **Numbers**
  - Integers: 1, 25, 32767
  - Floating point: 3.1412597, 10.634 E7
- **Dates and Times**
  - #01/01/06#
  - #07-04-2006 1:35 PM#
- **Strings**
  - "Happy Days"
  - 'Sad days'
- **Booleans**
  - True
  - False

TNB August 31, 2006

Intro - Programming VBScript

VBScript constants are pretty straightforward.

See the above examples.

## VBScript Operators

---

- **Math:**
  - +, -, \*, /, mod
- **String concatenation:**
  - &
- **String Manipulation functions:**
  - Left, Right, Mid, Len
- **Boolean:**
  - And, Or, Xor, Not
- **Operand Grouping**
  - ()

TNB August 31, 2006

Intro - Programming VBScript

Math operators have natural precedence:

\* (multiply) and / (divide) are equal and higher than + or -

+ (add) and - (subtract) are equal

Good idea to use parentheses to group sub-expressions to help readability and ensure desired results.

String concatenation combines two string variables or constants into one. Handy for building up string to display on screen or to write to files.

```
strTemp = "The patient's address is: " & strAddress
```

Boolean operators act on True / False variables, constants or expressions. Most often used in If expressions, but can be used in Boolean calculations:

```
If (Age >= 10) And (Sex = "F") Then
    MsgBox "Applicant qualifies for girl's soccer program."
End If
If (Color = "Red") Or (Color = "Black") Or (Color = "Blue") Then
    MsgBox "Requested item color is in stock."
Else
    MsgBox "Requested item must be ordered from depository."
End If
```

## VBScript Assignment Statements

---

- Variable = Constant
- Variable = Expression
- Variable = function result
- Examples:
  - `i = 20`
  - `Last_Name = "Burt"`
  - `Pi = 3.1415927`
  - `Due_Date = cur_Date+30`
  - `circumference = 2 * pi * radius`
  - `Name = First_Name & " " & Last_Name`
  - `iDays = abs(cur_date - file_date)`
  - `i = instr(File_Name, ".")`
  - `l = len>Last_Name)`
  - `avg = (a + b + c) / 3`

TNB August 31, 2006

Intro - Programming VBScript

Assignment statements set values of variables or return values of functions.  
When a variable is assigned a constant, it implicitly takes on that constant's type.

See the examples on the slide.

## VBScript Flow Control

---

- **If ... Then ... [Else If]... Else ... End If**
- **For ... Next (counted loop)**
- **Do While <condition> ... Loop**
  - (condition based loop, may run zero times)
- **Do Until ... <condition>**
  - (condition based loop, runs at least once)
- **Select Case <case variable or expression>**
  - One or more cases (match)
  - End Case to terminate the Select
- **Goto <label>**
  - Jumps directly to a statement label.
  - Useful occasionally to escape complex If nests.

TNB August 31, 2006

Intro - Programming VBScript

Program statement execution proceeds sequentially until a statement is encountered that requires a decision to be made.

Like other modern languages, VBScript is a “block oriented” language, rather than line-oriented. Blocks of code are demarked by one of the above flow control statement sets.

Each of the above flow control constructs is explored in the next few slides.

## VBScript IF Then Else End If

---

- **If <condition(s)> Then**
  - **Block of statements**
- **[Else If <condition(s)> Then]**
  - **Block of statements**
- **[Else]**
  - **Block of statements**
- **[End If]**
- **A single statement block on the same line as the IF need not have a closing Endif.**

TNB August 31, 2006

Intro - Programming VBScript

```
' Range safety checks (single statement for the code block)
' No End If needed.
If A < 0 Then A = 0
If A > 50 then A = 50
=====
' Bubble Sort
If Cust_Name(i) > CustName (i+1) Then
    xName = Cust_Name(i+1)
    Cust_Name(i+1) = Cust_Name(i)
    Cust_Name(i) = xName
End If
=====
If Color = "Red" Then
    IsInStock = True
Else If Color = "Blue" Then
    IsInStock = True
Else
    IsInStock = False
Endif
```

## VBScript Loops

---

- **For i = 1 to n step 1**
  - **Statements (can reference loop counter)**
- **Next i (counted loop)**
  
- **Do While <condition>**
  - **Statements (may run zero times)**
- **Loop**
  
- **Do Until <condition>**
  - **Statements (runs at least once)**
- **Loop**

TNB August 31, 2006

Intro - Programming VBScript

Loops are used to repeat execution of a block of statements until some condition is met.

Usually loops operate on a sequence of data elements, such as an array or collection.

In the For loop, the loop variable (e.g. i) is incremented by a specified step until the loop limit is reached.

```
For i = 1 to 15
    A(i) = 3 * i + 15
    If i > 10 Then Exit For
Next
```

Do While and Do Until have more flexible Conditions ...

```
KeepRunning = True
Do While KeepingRunning
    <statements>
    If <some condition> Then Exit Do
    If <some other condition> Then KeepRunning = False
Loop
```

## VBScript Select Case

---

- **Used to handle complex sets of If conditions.**
- **Case variable must evaluate to a number or string.**
  
- **Select Case <case variable or expression>**
- **Case <Constant list 1>**
  - **Block of statements**
- **Case <Constant list 2>**
  - **Block of statements**
- ...
- **Case Else**
  - **Block of statements**
- **End Case (to terminate the Select)**

TNB August 31, 2006

Intro - Programming VBScript

Select Case is a shortcut for complex nests of If Else IF statements.

A Case variable may be either a number or a string.

For each case, multiple values may be specified, delimited by constants or a range of values may be specified, separated by a dash.

```
Select Case Color
Case "Red", "Blue", "Green"
    InStock = True
Case Else
    InStock = False
End Select
```

```
Select Case Weight
Case 25 - 50
    WeightRange = 1
Case 51 - 150
    WeightRange = 2
Case Else
    WeightRange = 3
End Select
```

---

# **Bio-Break**

## **10 Minutes**

TNB August 31, 2006

**Intro - Programming VBScript**

## VBScript Console I/O

---

- **InputBox(Prompt, Title)**  
Takes input from a user via a dialog box.
  - The **InputBox()** function displays a prompt in a dialog box, waits for the user to input text or click a button and then returns the contents of the text box.
- **MsgBox(Message, DialogType, Title)**  
Displays a message dialog box
  - The **MsgBox()** function displays a message in a dialog box, waits for the user to click a button and returns a value indicating which button was clicked.

TNB August 31, 2006

Intro - Programming VBScript

InputBox returns a string, representing the value a user typed in.

```
tAnswer = InputBox ("Please enter your name.", "Enter Name")
```

Msgbox displays a message and waits for a response (button click) from the user.

It can be executed as a statement or called as a function to receive a specific response.

```
Msgbox "Debug: A=" & A, , "Debug Value"
```

```
tMsg = "Please confirm you wish to delete file: " & vFileName  
If MsgBox (tMsg, VbYesNo, "Confirm Delete") = VbYes Then  
    <Statements to delete the file>  
End If
```

Note the use of built-in VBScript constants: VbYesNo and VbYes. VBNo is also a built-in constant.

## **VBScript Common Functions**

---

**See VBScript Online Reference for List**

<http://www.microsoft.com/technet/scriptcenter/default.mspx>

TNB August 31, 2006

**Intro - Programming VBScript**

VBScript has a large built in library of functions that you can use in writing programs.

It is worth exploring this library, since the built in functions run much faster than functions written in interpreted VBScript.

The string functions are particularly important, since most scripts will do extensive string manipulation.

## VBScript Simple File I/O

---

- **Use the FileSystem Object**
- **Use a File Object**
  - **Open a file for reading, writing or appending.**
  - **Read strings of characters from a file.**
  - **Write strings of characters to a file.**
  - **Close the file.**
- **See code snippets below.**

TNB August 31, 2006

Intro - Programming VBScript

```
Sub CreateFile()  
    Dim fso, tf  
    Set fso = CreateObject("Scripting.FileSystemObject")  
    Set tf = fso.CreateTextFile("c:\MyFiles\testfile.txt", True)  
    ' Write a line with a newline character.  
    tf.WriteLine("Testing 1, 2, 3.")  
    ' Write three newline characters to the file.  
    tf.WriteLineBlankLines(3)  
    ' Write a line.  
    tf.WriteLine("This is a test.")  
    tf.Close  
End Sub  
  
Sub ReadFiles  
    Dim fso, fl, ts, s1, s2  
    Const ForReading = 1  
    Set fso = CreateObject("Scripting.FileSystemObject")  
    ' Read the contents of the file.  
    Set ts = fso.OpenTextFile("c:\MyFiles\testfile.txt", ForReading)  
    s1 = ts.ReadLine  
    s2 = ts.ReadLine  
    ts.Close  
End Sub
```

## VBScript Example 1

---

### General Meeting Number Drawing Tool

- **Input the starting and ending ticket numbers**
- **Use random number generator:**
  - **Generate 20 + 4 random ticket numbers in two sets.**
  - **Prevent generating same number twice in either set.**
  - **Sort each set to make it easy to find numbers.**
  - **Display them in numbered order on screen.**
  - **Write them to a text file for later reference.**
- **The extra 4 numbers are for handling missing or departed tickets.**

TNB August 31, 2006

Intro - Programming VBScript

We will use the “TicketDraw.vbs” script file for this example.

It is in the Programming\_Examples.zip file.

This script assumes all tickets will be given out consecutively from the same roll of tickets.

## VBScript Example 2

---

### Clean Out the “Temp” Folder

Delete any files not created “today”

- **General Steps**

- Get today's date
- Loop, scanning the “Temp” folder's files / folders.
- Check file's create, last-accessed, last-modified dates
- If older than today, delete the file.
- Otherwise leave it.
- Also look at folders within the “Temp” folder.
- Keep a running count of files deleted and space released.
- At end, report total files deleted and total space released.

TNB August 31, 2006

Intro - Programming VBScript

This example is in “CleanTemp.vbs”.

It is in the Programming\_Examples.zip file.

This script illustrates the use of the FileSystem object, which has a rich and vast set of objects, collections. Properties and methods.

We deal specifically with the Files collection, Subfolders collection, File and Folder objects. These are used to navigate through a designated folder to inspect individual files and decide whether to delete them.

## Web Site Tools Links

---

<http://www.devguru.com/technologies/vbscript/home.asp>

<http://www.devguru.com/home.asp>

[http://www.softpile.com/Development/Java/Review\\_03142\\_index.html](http://www.softpile.com/Development/Java/Review_03142_index.html)

<http://www.microsoft.com/technet/scriptcenter/default.mspx>

<http://www.microsoft.com/downloads/details.aspx?FamilyID=c717d943-7e4b-4622-86eb-95a22b832caa&DisplayLang=en>

<http://www.csidata.com/custserv/onlinehelp/VBSdocs/VBSTOC.htm>

<http://msdn.microsoft.com/vstudio/express/vb/>

TNB August 31, 2006

**Intro - Programming VBScript**

The above hyperlinks go to the major home networking equipment vendors' web sites. All these vendors provide a wealth of technical information.

There is also a link to the Microsoft home networking site, which again has much useful information.

Finally, the Windows XP Help system has extensive information on networking which is worth reading for more information.

## **Open Workshop / Q and A**

---

**Final**

**Questions And Answers**

TNB August 31, 2006

**Intro - Programming VBScript**