

VBA Foundations, Part 6

A Tutorial in VBA for Beginners—The Sixth in a Twelve Part Series

Richard L. Binning / rbinning@attbi.com

Welcome to this months edition of VBA Foundations. As you may have already heard, next month's PaperSpace™ will become a part of a new magazine called Augiworld™. This represents an exciting opportunity for a much wider and diverse audience. The four to five pages of in-depth VBA instruction for beginners are going to be shortened to a maximum of two pages when published in Augiworld format. As a result, the format that you have become accustomed to is going to change. I have gotten an agreement to continue the long (4-5 page) format throughout the rest of the series. This longer version of the articles will be available for download at the AUGI website and will include code examples and more in-depth coverage and explanations of the concepts covered. When Augiworld debuts in September, this column will be included in the PaperSpace portion of Augiworld as a two page condensed version, kind of an executive summary. Look for download instructions in the inaugural issue and stay tuned for more exciting VBA instruction in the coming months.

Often in your work you may wish to pop up a message while your program is working in the background either to let the user know what is happening, or just to advertise some new function or ability of your program. When you see these types of “pop-ups” at the start of the program they are called splash screens although they can and often do pop-up at other times as well. VBA has a built in “Timer” function that will time a process. You can use this function to pop up your splash screens to make your programs more professional in operation. There are many of these highly useful functions built in to VBA; some for converting values or types of data, some for performing math operations, others for reading or writing to files on your hard-drive or network, still others for managing files such as creating or deleting files or folders, and finally some functions to check or verify your data and data type.

As you know, I can't extol the virtues of the Visual Basic language and interface enough. This month will be no exception. In this months issue we will again explore the code window and discover some of these built in functions designed to assist you in getting your projects done in an easy and straightforward manner. These are built in functions that you'll use in virtually all your projects for manipulating words and phrases, converting numbers to other formats, date functions and more. As I mentioned, VBA has many built in functions that can assist you in your development efforts. First, I will list and describe many of the more useful functions, and then I will show you an easy way to find them while writing your code. We are going to cover three main groups of built in functions in this article; Conversion functions, math functions, and file management functions. See the Tables below for some common functions.

Common Math Functions:

These functions will take

	Function Name	Description	How to Use it
Common Math Functions	Abs	Returns the Absolute value of whatever data type is passed into the function. Negative values become positive.	Abs (Somenumber)
	Atn	The Atn function takes the ratio of two sides of a right triangle (number) and returns the corresponding angle in radians as a Double data type.	Atn (Somenumber)
	Cos	The Cos function takes an angle and returns the ratio of two sides of a right triangle.	Cos (Somenumber)
	Rnd	Returns a Single data Type containing a random number between 0 and 1.	Rnd [(Somenumber or Expression)]
	Sgn	Returns a Variant data Type containing an integer specifying the sign. Returns 1 for positive, 0 for Zero, and -1 for negative.	Sgn (SomeNumber)
	Sin	The Sin function takes an angle and returns the ratio of two sides of a right triangle as a Double data type.	Sin (number)
	Sqr	Returns a Double data type which is the square root of the number passed into it.	Sqr (number)
	Tan	Tan takes an angle and returns the ratio as a Double data type of two sides of a right triangle.	Tan (number)

(Continued on page 2)

VBA Foundations, Part 6

A Tutorial in VBA for Beginners—The Sixth in a Twelve Part Series

Richard L. Binning / rbinning@attbi.com

you back to those heady days of your high school math classes. Remember all those theorems you had to painstakingly develop? I know its all rushing back to you. These built in functions should save you a lot of painful math. Now make sure you track down your math teachers and tell them that you are indeed using those concepts in real life after all.

Common Conversion Functions: These functions are going to save your bacon at times when you are delousing your code. Still can't get past that dreaded "Type Mismatch" error? Check your variable and use one of these handy functions to set your data right. Note: You can use the UCase and LCase functions to force text to behave properly on your drawings too, it also makes comparison of data easier.

Common File Management Functions: Tired of popping open Microsoft Window's Explorer to whip your directories and drives into shape? Use these functions to get a grip on what belongs where in your document management strategy.

Okay, now that we've covered these functions you still have a passel of words, concepts and usage notes to remember when you are working right? Yes, but they are

	Function Name	Description	How to Use it
Common Conversion Functions	Asc	Returns an Integer (0 - 255) representing the ascii character code. Used when you wish to check for valid characters, keys pressed, etc.	Asc(SomeString)
	Chr	Returns a String containing the character associated with the ascii code number passed into it. Used opposite of Asc function.	Chr(Somenumber)
	Format	Returns a Variant (String) containing an expression formatted as numbers, dates and times, strings, etc.	Format(expression[, format[, firstdayofweek[, firstweekofyear]])] Note: last two arguments are optional constants.
	CBool	Evaluates the expressions passed into the function and returns a boolean (true, false) value.	CBool(expression) Example: If A and B are defined as containing the same value then Cbool(A = B) would return "TRUE"
	CStr	Converts a numeric value to a String. If you get a type mismatch error passing a number back to your program when it is expecting a string then wrap your value up in this function.	CStr(SomeNumber)
	CInt	Changes a string or other numeric value to an Integer data type.	CInt(Somenumber)
	CDbl	Changes a string or other numeric value to a Double data type.	CDbl(number)
	LCase	Changes passed string to all lowercase letters.	LCase(String)
UCase	Changes passed string to all Uppercase letters.	UCase(String)	

	Function Name	Description	How to Use it
File and Directory Functions	ChDir	Change directory or folder. The ChDir statement changes the default directory but not the default drive.	ChDir "Some Drive and Folder name" Example: ChDir "C:\pagedir" will change the directory to c:\pagedir from any folder on the c drive.
	ChDrive	Change the drive	ChDrive "Some Drive letter"
	FileCopy	Copy a file from one location to another.	FileCopy source, destination Note: To ensure consistent programming practices, both the source and destination arguments should point to a valid drive, path, and filename although a filename is all that is required.
	MkDir	Make a directory or folder.	MkDir path Note: If a drive is not specified then the folder will be created on the current drive.
	Rmdir	Remove or delete a directory or folder.	Rmdir path Note: If no drive is specified, Rmdir removes the directory or folder on the current drive. Errors will occur if path contains files. Use Kill function to delete all files prior to removing directory.
	CurDir	Return the current path.	Dim varReturnValue varReturnValue = CurDir()
	Dir	Used to check existence of a particular file or group of files. Can be used to check on files with particular attributes such as read only, hidden, etc.	Dir[(pathname[, attributes])] Note: pathname and attributes are optional arguments. See online help for assistance with this function.
	Kill	Delete one or more files from the computer.	Kill path Note: Kill supports the use of multiple-character (*) and single-character (?) wildcards to specify multiple files.

(Continued on page 3)

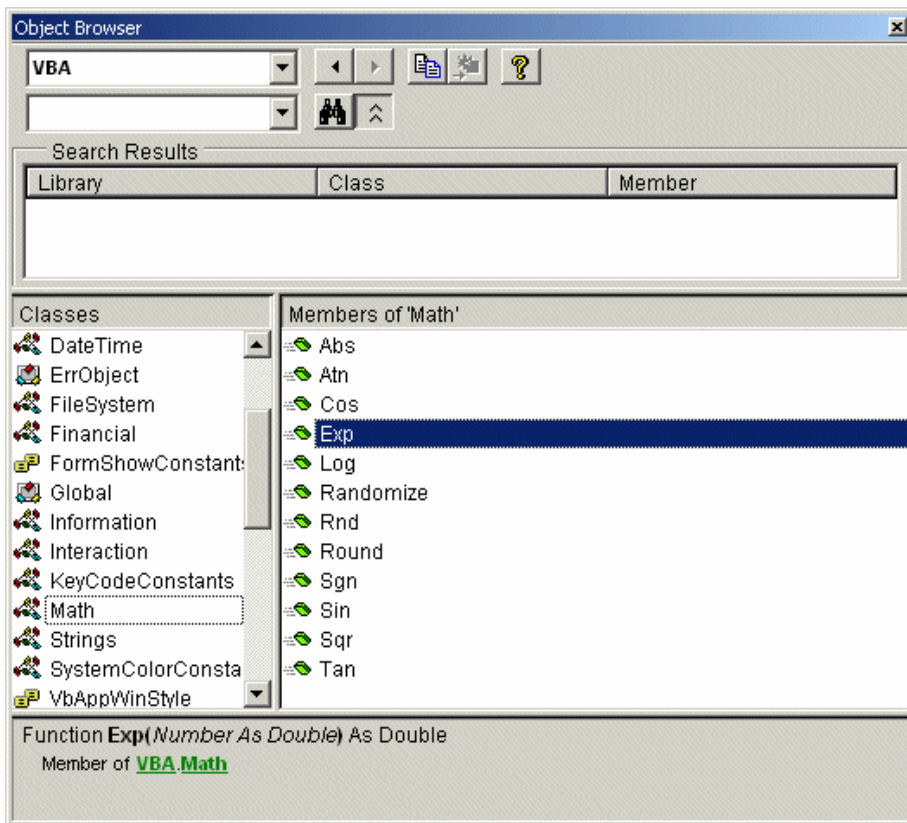
VBA Foundations, Part 6

A Tutorial in VBA for Beginners—The Sixth in a Twelve Part Series

Richard L. Binning / rbinning@attbi.com

in a handy format for you to cut out and paste to the wall of your cubical. In addition, I am going to share a little secret with you. Open your editor and a VBA macro. Now in the code window type the following: VBA. (Don't forget the "dot operator" after the three letters of your favorite programming environment.) Did you see what happened? That's right! VBA knows that you wish to access one of the built in functions, so it uses the Intellisense™ feature and immediately the VBE popped up a window with available code for you as a guideline. In that window you will find all of the above and many more of the built in functions available to you.

Perhaps you typed in vba. And then browsed to the function you were looking for and hit your tab key to select it. Now what? If you find yourself searching for some help with the particular function or method you've chosen, simply highlight the word in your VBA editor and hit the "F1" key to pop open the built in help topic file pointing to that particular function or keyword which is highlighted. It doesn't get any easier than this now does it?



You can also search through the Object browser to find these and many more functions by selecting "VBA" in the project/library pull down of the Object browser as shown in the graphic. Notice I've got the Math functions collection shown in the class window. For extra credit check out the derived math functions in the online help system.

Note: You can highlight a word or function in this window and either hit your "F1" key or click on the yellow question mark as shown at the top of the window in the graphic to display the built-in help topic associated with your selection. See the bottom pane of the graphic window? It shows the syntax or how the function is defined so that you can utilize the function correctly.

Please stay tuned as we kick off the inaugural issue of Augiworld with more explorations of the VBA editor and language. As always feel free to send in your questions and comments to the email address at the top of this article. An Email with the subject line "Built-ins" will get you a zipped copy of the graphics used in this article in glorious color suitable for framing. If you are not already a guild member please join the VBA guild and others. These guilds are in place for you and all are welcome both beginners and experts alike. See you on the guilds or next month in Augiworld magazine.