

KiXtart Quick Reference

Author: Jialong He

Jialong_he@bigfoot.com

http://www.bigfoot.com/~jialong_he

Introduction

What is it and where to get it?

KiXtart is a logon script processor and enhanced batch scripting language for Windows. It comes with MS resource kit and can also be downloaded from: <http://www.kixtart.org>

A simple script

KiXtart is a free-format scripting language. All commands and variables are case insensitive. By default, script engine (kix32.exe) automatically looks for a script with the current user name (*username.kix*). If not found, it looks for a script named "kixtart.kix". Script name can be specified from command line.

```

;=====
; myscript.kix
;=====
? "Hi, @UserID"
    
```

To run this script, in a command window, type:

kix32 [myscript.kix](#)

Language Basics

Variables

A variable is prefixed with a dollar sign (\$) and can be used directly. This defines a global variable. You define a local variable with DIM command. Anything after semi-colon (;) is comments.

```

$MyVal = 10          ; global variable
DIM $YourVal        ; local variable
$YourVal = 30       ;
DIM $MyArray[10]    ; one-dim array
    
```

NOTE: there are spaces on both sides of equal sign "=".

Operators

+,-,*,/	Add, Sub, Mul, Div
&,	bitwise AND, OR
>,<>=,<	Greater Than, Less Than, Equal, Not Equal
<=, >=	Less than or equal, Greater than or equal
AND, OR, NOT	Logical AND, OR, NOT

Flow Control

IF statement

IF expression
statements
ELSE
statements
ENDIF

```

IF INGROUP("Admins")
    ; similar to IF INGROUP("Admins") > 0
    ; do stuff
ENDIF
    
```

FOR statement

FOR \$counter = start **TO** end [**STEP** step]
statements
NEXT

```

For $Count = 0 To 10 Step 2
    ? $Count
Next
    
```

WHILE statement

WHILE "expression"
statements
LOOP

DO statement

DO
statements
UNTIL "expression"

FOR EACH statement

FOR EACH \$element **IN** group
statements
NEXT

```

Dim $MyArray[10]
For Each $Element In $MyArray
    ? $Element
Next
    
```

SELECT ... CASE statement

SELECT
CASE expression
statements
CASE expression
statements
ENDSELECT

User define a function

Function name [(argument1, argument2, [optional]argumentx)]
[statements]
[name = expression]
EndFunction

```

;=====
; Define a function
;=====
Function MyFunc($Str)
    ? $Str
EndFunction
    
```

MyFunc("Hi, there!") ; call user func

GOTO, Call Other script

```

GOSUB <label>
GOTO <label>
CALL "scriptname"

? "In main program"
GOSUB "MySum"
? "Back to main"
EXIT 1

:MySum
? "In subroutine now..."
RETURN
    
```

Build-in Commands and Functions

Console I/O

[?] "text"	print text on console
REDIRECTOUTPUT ("filename", overwrite)	redirect output from console to a file
CLS	clear console screen
AT (row, column)	position current cursor
GET \$var	read a single character into a variable
KBHIT ()	Checks the console for keyboard input
GETS \$var	read a line from keyboard into a variable
SETCONSOLE ("mode")	change the display state of console window
SETTITLE ("title")	set console window title
BIG, SMALL	set display character size
BOX (tl_row, tl_col, br_row, br_col, "line style")	Draw a box on screen
color F[+]/B[+]	set foreground/background color

```

$rc=BOX(10, 10, 12, 15,
"+++|+++| ")
    
```

	Example: COLOR w+/b
Text File	
DISPLAY filename	Display content of the file.
OPEN (FHandle, "filename", mode)	Open text file
READLINE (FHandle)	Read a line from opened text file
WRITELINE (FHandle, "LineToWrite")	Write a line to the opened text file
CLOSE (FHandle)	Close text file
ReadProfileString ("filename", "section", "key")	Read/Write INI file
WriteProfileString ("filename", "section", "key", "string")	
<pre>IF Open(3, "MySettings.ini") = 0 \$x = ReadLine(3) WHILE @ERROR = 0 ? "Line read: [" + \$x + "]" \$x = ReadLine(3) LOOP Close (3) ELSE ? "Open failed: " + @ERROR ENDIF</pre>	
DOS commands	
CD, MD, RD	change working / make / remove directory
Copy, Move, Del	copy / move / delete files
Go [drive]	change current drive
Conversion functions	
abs(), int(), fix()	absolute value, round to integer.
asc (char)	return ASCII value
val ("number")	return value of a numerical string ? val ("125") + 5
chr (char_code)	return character from its ASCII code, \$Message= "Hello " + @USERID + chr(13) + chr(10) + "Welcome to our network."
DECtoHEX (Decal value)	Returns the hexadecimal representation of a decimal value.
RND (Range) SRnd (Seed)	random number generator
CDbl (expression) CInt (expression) CStr (expression)	Convert to Double, Integer or String

Printer Management	
ADDPRINTERCONNECTION ("printer name") DELPRINTERCONNECTION ("printer name") SETDEFAULTPRINTER ("printer name")	
<pre>If AddPrinterConnection ("\\vleerbeer\hp laserjet 4") = 0 ? "Added printer connection...." If SetDefaultPrinter ("\\vleerbeer\hp laserjet 4") = 0 ? "Set default printer to HP LaserJet 4...." Endif Endif</pre>	
User Group Management	
INGROUP ("group name") ENUMGROUP (Index) ENUMLOCALGROUP (Index)	
<pre>If InGroup("Domain Users") DISPLAY "z:\users.txt" Endif</pre>	
Registry Functions	
AddKey ("subkey") DelKey ("subkey") ReadValue ("subkey", "entry") WriteValue ("subkey", "entry", "expression", "data type") DelValue ("subkey", "entry") DelTree ("subkey") ReadType ("subkey", "entry") ENUMKEY ("subkey", index) ENUMVALUE ("subkey", index) KEYEXIST ("subkey") LOADHIVE ("key", "filename") UNLOADHIVE ("key") LOADKEY ("subkey", "filename") SAVEKEY ("subkey", "filename")	
<pre>\$RC = AddKey("HKEY_CURRENT_USER\EZReg") If \$RC = 0 ? "Key added...." Endif</pre>	
Event Log	
BACKUPEVENTLOG ("eventlog", "backupfile")	backup event log. <pre>\$rc=BackupEventLog("Application" , "C:\eventbackups\application.evt")</pre>

CLEARVENTLOG ("eventlog")	clear event log \$rc=ClearEventLog("Application")
LOGEVENT (type, ID, message, target, source)	log an event to the log file \$RC = LogEvent(0 , 1 , "Logon script completed successfully" , "" , "MyEvent")
File Info	
GETDISKSPACE ("drive")	Returns disk space in KB.
GETFILESIZE ("filename")	Returns the size of a file in bytes
GETFILEATTR ("filename") SETFILEATTR ("filename", attributes)	Get/Set file attributes
EXIST ("filename")	Test if file exist IF EXIST ("c:\config.sys") DISPLAY "c:\config.sys" ENDIF
COMPAREFILETIMES ("file1", "file2")	Compares the date and time of two files.
GETFILETIME ("filename")	Returns the date and time information of a file.
GETFILEVERSION ("filename", "versionfield")	Returns a version information string of a file.
String Operation	
SUBSTR ("string", start, length) LEFT ("string", length) RIGHT ("string", length) INSTR ("string1", "string2") INSTRREV ("string1", "string2") LCASE ("string") UCASE ("string") LEN ("string") LTRIM ("string") RTRIM ("string") TRIM ("string") JOIN (array, "delimiter") SPLIT ("string", "delimiter", count)	
Misc Commands	
BEEP	Beep sound on PC speaker
PLAY	Play a wave file through the sound card
SLEEP <second>	pause for a few seconds
EXIT	Exits from the current script
QUIT	Exits from KiXtart completely.

RETURN	return to SUB script (see GOSUB).
RUN "command"	run external program. Current script does not wait. RUN "%COMSPEC% /e:1024 /c DIR C: "
SHELL "command"	run external command, current script wait till external command finish. SHELL "%COMSPEC% /C COPY Z:\FILE.TXT C:\ "
SET "var=string" SETL "var=string" SETM "var=string"	manipulate environment variables
SETTIME "source"	synchronize time with a time source
USE	connect to a share, similar to "net use" command
EXPANDENVIRONMENTVARS ("env string")	Expands an environment variable ? ExpandEnvironmentVars("%SystemRoot%")
IsDeclared (variable)	test if variable is defined
VARTYPE (\$variable)	return variable type ID
VarTypeName (\$variable)	return variable type name
MemorySize (type)	return available memory
MESSAGEBOX ("message", "title", style, time-out)	Displays a standard dialog box in Windows. \$Selection = MessageBox("Hi, there", "Greeting", 36) IF \$Selection = 6 ? "Yes selected ENDIF
SENDKEYS ("keys") SETFOCUS ("Title")	feed key strokes to an application Run("notepad.exe") SetFocus("Untitled - Notepad") Sleep(1) SendKeys("From SendKey")
SENDMESSAGE ("recipient", "message")	send message to a user (similar to "net send" command)
SETOPTION ("option", "value")	change options of KiXtart script engine.
SetSystemState (mode, force)	change power state of computer
SetWallpaper	set wall paper

("my.bmp", mode)	
SIDTONAME ("SID")	translate SID to user name
UBound (array)	largest index in an array
CREATEOBJECT ("serverclassname.type pename") GETOBJECT ("objectname")	COM object operation
LOGOFF (force)	log off current user
SHUTDOWN ("computer", "message", wait, force, reboot)	Shutdown/Reboot computer
ENUMIPINFO (index, type)	Enumeration of TCP/IP information.

Build-in Macros

Macros can be used anywhere an expression is expected. For example, ? "Hi "+ @UserID	
@Address	Address of the network adapter
@Build	Build number of the operating system
@Comment	User comment (Description field in User Manager)
@CPU	Name of the CPU (e.g.: "Intel Pentium III")
@CRLF	Returns a carriage-return line-feed combination
@CSD	CSD information (eg: "Service Pack 2")
@CurDir	Current directory
@Date	Date (in the format YYYY/MM/DD)
@Day	Day of the week (Monday, Tuesday, etc.)
@Domain	Domain or workgroup the computer belongs to
@DOS	Version of Windows NT
@Error	Return code of the most recent command or function.
@FullName	Full name of current user
@HomeDir	Short name of the directory part of home directory
@HomeDrive*	Drive letter of drive containing home directory
@HomeShr	Server and share name part of home directory
@HostName	Fully qualified TCP/IP host name (including TCP/IP domain name)
@InWin	Operating system: 1 = Windows NT; 2 = Windows 9x
@IPaddressX	TCP/IP address (possible values for X are 0 - 3).
@KiX	Version of KiXtart
@LanRoot	Directory where network software resides (usually Systemroot\System32)
@LDomain*	Logon domain

@LDrive	Drive that is redirected to \\logonserver\NETLOGON
@LM	Version of network software
@LogonMode	If 1, indicates that KiXtart assumes to be running during the logon sequence
@LongHomeDir	Long name of the directory part of home directory
@LServer	Logon server
@MaxPWAge	Maximum password age
@MDayNo	Day of the month (1-31)
@MHz	Approximation of the CPU speed (not available on Windows 9x)
@MonthNo	Month number, beginning with January (1-12)
@Month	Name of the month
@MSecs	Milliseconds part of the current time
@PrimaryGroup*	Current user's primary group
@Priv	User's privilege level (GUEST, USER, ADMIN)
@ProductSuite	OS Suite.
@ProductType	OS type.
@PWAge	Password age
@RAS	Number of active Remote Access Service (RAS) connections
@RServer*	KXRPC server used for the current session
@ScriptDir	Directory of current script
@ScriptName	Name of the current script
@SError	Error text corresponding with @ERROR
@SID*	Current user's Windows NT Security Identifier (SID)
@Site**	Name of the site in which the system resides
@StartDir	Directory from which KiXtart was started
@SysLang	Full English name of the language of the operating system.
@Ticks	Number of milliseconds elapsed since computer was last booted.
@Time	Current time (in the format HH:MM:SS)
@UserID	Current user's Windows NT user ID
@UserLang	Full English name of the language selected by the current user.
@WDayNo	Days since Sunday (1 - 7)
@Wksta	Computer name
@WUserID	Current user's Windows user ID
@YDayNo	Days since January 1 (1 - 365)
@Year	Current year