

CLIBASIC Language Manual

Commands:

CALL FILENAME\$	Opens and runs FILENAME\$ in the current session. Called programs will retain the same arguments as the parent.
{CHDIR CD} DIR\$	Changes the current directory to DIR\$.
CLS [COLOR]	Clears the screen with optional color.
COLOR {FGC, [BGC] [FGC], BGC}	Sets the foreground color to FGC and the background color to BGC.
DEL {VAR\$ VAR}	Deletes the variable VAR\$ or VAR.
DIM {VAR\$ VAR}, MAX, {INIT\$ INIT}	Makes an array with the max index being MAX and the initial value for each element being INIT/INIT\$.
FILES	List the files and directories in the current directory.
EXEC PROGRAM\$ [{ARG\$/ARG}...]	Runs PROGRAM\$ and passes the remaining arguments to PROGRAM\$.
EXIT [CODE]	Exits with CODE (or 0 if CODE is not supplied).
LOCATE {X, [Y] [X], Y}	Moves the cursor to X, Y.
PRINT [{STRING\$/NUMBER} {,/;} ...]	Prints text on the screen, ';' means print without newline and ',' means print tab.
PUT [{STRING\$/NUMBER} ...]	Puts STRING\$ or NUMBER on the terminal.
QUIT [CODE]	Refer to 'EXIT'.
RESETTIMER	Resets the timer.
RUN FILENAME\$ [{ARG ARG\$}...]	Runs FILENAME\$ in a new session and passes ARG/ARG\$ to it.
{{SET LET} {VAR\$ VAR}, {STRING\$ NUMBER} {VAR\$ VAR} = {STRING\$ NUMBER}}	Sets the variable VAR\$ or VAR to STRING\$ or NUMBER.
SH COMMAND\$	Runs COMMAND\$ in sh on Linux and Command Prompt on Windows
{SRAND SRND} SEED	Seeds the random number generator with SEED.
WAIT SEC	Waits for SEC seconds.

WAITMS MSEC	Waits for MSEC milliseconds.
WAITUS USEC	Waits for USEC microseconds.
_AUTOCMDHIST	Enables automatic history saving (saves to '.clibasic_history' to the user's home directory, remove this file to disable this feature).
_LOADCMDHIST FILENAME\$	Loads the command history from FILENAME\$.
_PROMPT STRING\$	Sets the prompt string to solve to STRING\$.
_PROMPTTAB WIDTH	Sets the prompt tab width to WIDTH.
_RESETTITLE	Resets the terminal title.
_SAVECMDHIST FILENAME\$	Saves the command history to FILENAME\$.
_SHATTRIB {ATTRIB\$ ATTRIB}, {VALUE\$ VALUE}	Sets the 'SH' attribute ATTRIB\$ or ATTRIB to VALUE\$ or VALUE.
_TITLE STRING\$	Sets the terminal title to STRING\$.
_TXTATTRIB {ATTRIB\$ ATTRIB}, {VALUE\$, VALUE}	<p>Sets the text attribute ATTRIB\$ or ATTRIB to VALUE\$ or VALUE.</p> <p>Available attributes:</p> <ul style="list-style-type: none"> 0/RESET 1/BOLD 2/ITALIC 3/UNDERLINE 4/DBL_UNDERLINE/DOUBLE_UNDERLINE 5/SQG_UNDERLINE/SQUIGGLY_UNDERLINE 6/STRIKETROUGH 7/OVERLINE 8/DIM 9/BLINK 10/HIDDEN 11/REVERSE 12/UNDERLINE_COLOR 13/FGC 14/BGC
_TXTLOCK	Stops the keyboard from echoing on the terminal.
_TXTUNLOCK	Undoes the effect of '_TXTLOCK'.

Functions:

<code>ASC (STRING\$, [POSITION])</code>	Returns the ASCII code of character <code>POSITION</code> (starting at and defaulting if not specified to zero) of <code>STRING\$</code> .
<code>BASENAME\$ (FILENAME\$)</code>	Returns the file name out of the file path provided by <code>FILENAME\$</code> .
<code>BGC ()</code>	Returns the current background color.
<code>{CHDIR CD} (DIR\$)</code>	Attempts to change the current directory to <code>DIR\$</code> and returns 0 on success and an error code on failure (the error code is taken directly from the C variable 'errno' set by the C 'chdir()' function).
<code>CHR\$ (CODE)</code>	Returns ASCII character <code>CODE</code> .
<code>CHRAT\$ (STRING\$, POSITION)</code>	Returns the character at <code>POSITION</code> of <code>STRING\$</code> .
<code>CWD\$ ()</code>	Returns the current working directory.
<code>CINT (NUMBER)</code>	Returns <code>NUMBER</code> rounded.
<code>COS (NUMBER)</code>	Returns the cosine of <code>NUMBER</code> .
<code>COSH (NUMBER)</code>	Returns the hyperbolic cosine of <code>NUMBER</code> .
<code>CURX ()</code>	Returns the X position of the cursor.
<code>CURY ()</code>	Returns the Y position of the cursor.
<code>DIRNAME\$ (FILENAME\$)</code>	Returns the directory name out of the file path provided by <code>FILENAME\$</code> .
<code>EXEC (PROGRAM\$ [{ARG\$/ARG}...])</code>	Runs <code>PROGRAM\$</code> , passes the remaining arguments to <code>PROGRAM\$</code> , then returns the exit code of the program or 127 if running <code>PROGRAM\$</code> failed.
<code>EXEC\$ (PROGRAM\$ [{ARG\$/ARG}...])</code>	Runs <code>PROGRAM\$</code> , passes the remaining arguments to <code>PROGRAM\$</code> , then returns the output of <code>PROGRAM\$</code> .
<code>EXP (NUMBER)</code>	Returns the exponent of <code>NUMBER</code> .
<code>FILES\$ ()</code>	Returns a list of the files in the directory.
<code>FGC ()</code>	Returns the current foreground color.

HEIGHT()	Returns the height of the terminal.
HEX\$(NUMBER)	Returns the hexadecimal version of NUMBER.
INKEY\$()	Returns a character from the terminal.
INPUT\$(PROMPT\$)	Returns a string after prompting for PROMPT\$ (asks “?: ” if PROMPT\$ is not supplied).
INT(NUMBER)	Returns NUMBER rounded down.
LCASE\$(STRING\$)	Returns the lower-case version of STRING\$.
LEN(STRING\$)	Returns the length of STRING\$.
LINE\$(LINE, STRING\$)	Returns line LINE (starting at zero) of STRING\$.
LINES(STRING\$)	Returns the line count of STRING\$.
LOG(NUMBER)	Returns the natural logarithm of NUMBER.
LOG10(NUMBER)	Returns the common logarithm of NUMBER.
MOD(NUMBER)	Returns the modulus of NUMBER.
OCT\$(NUMBER)	Returns the octal version of NUMBER.
PI()	Returns Pi.
{RAND RND}({MAX MIN, MAX})	Returns a random number from MIN (0 if MIN is not supplied) to MAX.
SH(COMMAND\$)	Runs COMMAND\$ in the command line or shell and returns the exit status.
SH\$(COMMAND\$)	Runs COMMAND\$ in the command line or shell and returns the text the command outputs.
SIN(NUMBER)	Returns the sine of NUMBER.
SINH(NUMBER)	Returns the hyperbolic sine of NUMBER.
SNIP\$(STRING\$, {TO {[FROM], TO FROM, [TO]}})	Returns the part of STRING\$ defined by FROM through TO.
STR\$(NUMBER)	Returns NUMBER as a string.
TAN(NUMBER)	Returns the tangent of NUMBER.
TANH(NUMBER)	Returns the hyperbolic tangent of NUMBER.
TIME()	Returns the current time in seconds.

<code>TIMEMS()</code>	Returns the current time in milliseconds.
<code>TIMEUS()</code>	Returns the current time in microseconds.
<code>TIMER()</code>	Returns the timer value in seconds.
<code>TIMERMS()</code>	Returns the timer value in milliseconds.
<code>TIMERUS()</code>	Returns the timer value in microseconds.
<code>UCASE\$(STRING\$)</code>	Returns the upper-case version of <code>STRING\$</code> .
<code>VAL(STRING\$[, TYPE])</code>	Returns the numeric value of <code>STRING\$</code> , <code>TYPE</code> is what type the number is (0 = DEC, 1 = HEX, 2 = OCT, etc/not supplied = Auto (scanf autodetect)).
<code>WIDTH()</code>	Returns the width of the terminal.
<code>_ARGC()</code>	Returns how many arguments were passed to the program
<code>_ARG\$([N])</code>	Returns argument <code>N</code> or all arguments except #0 if <code>N</code> is not provided. Argument 0 is the full/real path to the program file.
<code>_BIT\$()</code>	Returns the executable bit format.
<code>_ENV\$(STRING\$)</code>	Returns the content of the environment variable defined by <code>STRING\$</code> .
<code>_ENVSET(STRING\$)</code>	Returns 1 if the environment variable defined by <code>STRING\$</code> is set and 0 otherwise.
<code>_HOME\$()</code>	Returns the path to the user's home directory.
<code>_OS\$()</code>	Returns the current operating system name.
<code>_PROMPT\$()</code>	Returns the prompt string.
<code>_STARTCMD\$()</code>	Returns the full/real path to the command used to start CLIBASIC.
<code>_TXTLOCK()</code>	Returns 1 if the text lock is in effect and 0 otherwise.
<code>_VER\$()</code>	Returns the CLIBASIC version.

Logic Commands:

DO	Begins a DO block.
DOWHILE CONDITION	Begins a DO block while CONDITION is true.
ELSE	Inverts an IF command.
ENDIF	Ends an IF block.
FOR VAR, INIT, CONDITION, I	Begins a FOR block, sets VAR to INIT and loops while adding I to VAR while CONDITION is true.
IF CONDITION	Begins an IF block and runs commands if CONDITION is true.
LOOP	Jumps to the beginning of a DO/LOOP block.
LOOPWHILE CONDITION	Jumps to the beginning of a DO/LOOP block if CONDITION is true.
NEXT	Jumps to the beginning of a FOR block.
REM	Comments out one command.

Symbols:

?	Shortcut to PRINT.
'	Comment until the end of the line.
#	Comment until the end of the line.