

# **CLIBASIC Language Manual**

## Commands:

CALL FILENAME\$	Opens and runs FILENAME\$ in the current session.
{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.
FILES	List the files and directories in the current directory.
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\$	Runs FILENAME\$ in a new session.
{{SET LET} {VAR\$ VAR}, {STRING\$  NUMBER} {VAR\$ VAR} = {STRING\$  NUMBER}}	Sets the variable VAR\$ or VAR to STRING\$ or NUMBER.
{SH EXEC} 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).

<code>_DEBUGOFF</code>	Disables the printing of debug text.
<code>_DEBUGON</code>	Enables the printing of debug text.
<code>_LOADCMDHIST FILENAME\$</code>	Loads the command history from <code>FILENAME\$</code> .
<code>_PROMPT STRING\$</code>	Sets the prompt string to solve to <code>STRING\$</code> .
<code>_PROMPTTAB WIDTH</code>	Sets the prompt tab width to <code>WIDTH</code> .
<code>_RESETTITLE</code>	Resets the terminal title.
<code>_SAVECMDHIST FILENAME\$</code>	Saves the command history to <code>FILENAME\$</code> .
<code>_SHATTRIB {ATTRIB\$ ATTRIB}, {VALUE\$ VALUE}</code>	Sets the 'SH' attribute <code>ATTRIB\$</code> or <code>ATTRIB</code> to <code>VALUE\$</code> or <code>VALUE</code> .
<code>_TITLE STRING\$</code>	Sets the terminal title to <code>STRING\$</code> .
<code>_TXTATTRIB {ATTRIB\$ ATTRIB}, {VALUE\$, VALUE}</code>	<p>Sets the text attribute <code>ATTRIB\$</code> or <code>ATTRIB</code> to <code>VALUE\$</code> or <code>VALUE</code>.</p> <p>Available attributes:</p> <ul style="list-style-type: none"> <li>0/RESET</li> <li>1/BOLD</li> <li>2/ITALIC</li> <li>3/UNDERLINE</li> <li>4/DBL_UNDERLINE/DOUBLE_UNDERLINE</li> <li>5/SQG_UNDERLINE/SQUIGGLY_UNDERLINE</li> <li>6/STRIKETROUGH</li> <li>7/OVERLINE</li> <li>8/DIM</li> <li>9/BLINK</li> <li>10/HIDDEN</li> <li>11/REVERSE</li> <li>12/UNDERLINE_COLOR</li> <li>13/FGC</li> <li>14/BGC</li> </ul>
<code>_TXTLOCK</code>	Stops the keyboard from echoing on the terminal.
<code>_TXTUNLOCK</code>	Undoes the effect of ' <code>_TXTLOCK</code> '.

## 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> .
---	--

BGC ( )	Returns the current background color.
{CHDIR CD} (DIR\$)	Attempts to change the current directory to DIR\$ 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).
CHR\$ (CODE)	Returns ASCII character CODE.
CWD\$ ( )	Returns the current working directory.
CINT (NUMBER)	Returns NUMBER rounded.
COS (NUMBER)	Returns the cosine of NUMBER.
COSH (NUMBER)	Returns the hyperbolic cosine of NUMBER.
CURX ( )	Returns the X position of the cursor.
CURY ( )	Returns the Y position of the cursor.
EXP (NUMBER)	Returns the exponent of NUMBER.
FILES\$ ( )	Returns a list of the files in the directory.
FGC ( )	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 if NUMBER.

<code>MOD (NUMBER)</code>	Returns the modulus of NUMBER.
<code>OCT\$ (NUMBER)</code>	Returns the octal version of NUMBER.
<code>PI ()</code>	Returns Pi.
<code>{RAND RND} ({MAX MIN, MAX})</code>	Returns a random number from MIN (0 if MIN is not supplied) to MAX.
<code>{SH EXEC} (COMMAND\$)</code>	Runs COMMAND\$ in sh on Linux and Command Prompt on Windows and returns the exit status.
<code>SIN (NUMBER)</code>	Returns the sine of NUMBER.
<code>SINH (NUMBER)</code>	Returns the hyperbolic sine of NUMBER.
<code>STR\$ (NUMBER)</code>	Returns NUMBER as a string.
<code>TAN (NUMBER)</code>	Returns the tangent of NUMBER.
<code>TANH (NUMBER)</code>	Returns the hyperbolic tangent of NUMBER.
<code>TIME ()</code>	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 STRING\$.
<code>VAL (STRING\$[, TYPE])</code>	Returns the numeric value of STRING\$, TYPE is what type the number is (0 = DEC, 1 = HEX, 2 = OCT, etc/not supplied = Auto (sscanf autodetect)).
<code>WIDTH ()</code>	Returns the width of the terminal.
<code>__BITS\$ ()</code>	Returns the executable bit format.
<code>__DEBUG ()</code>	Returns 1 if in debug mode and 0 otherwise.
<code>__ENV\$ (STRING\$)</code>	Returns the content of the environment variable defined by STRING\$.
<code>__ENVSET (STRING\$)</code>	Returns 1 if the environment variable defined by STRING\$ is set and 0 otherwise.

<code>_HOME\$()</code>	Returns the path to the user's home directoy.
<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:

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

## Symbols:

<code>?</code>	Shortcut to <code>PRINT</code> .
<code>?;</code>	Shortcut to <code>PRINT;</code> .
<code>'</code>	Comment until the end of the line.

