

Miscellaneous

`paths.uname()`

Returns up to three strings describing the operating system.

The first string is a system name, e.g., “Windows”, “Linux”, “Darwin”, “FreeBSD”, etc.

The second string is the network name of this computer.

The third string indicates the processor type.

`paths.is_win()`

Returns true if the operating system is Microsoft Windows.

`paths.is_mac()`

Returns true if the operating system is Mac OS X.

`paths.getregistryvalue(key,subkey,value)`

Query a value in the Windows registry value.

Causes an error on other systems.

`paths.findprogram(progname,...)`

Finds an executable program named “progname” and returns its full path.

If none is found, continue searching programs named after the following arguments and return the full path of the first match.

All the directories specified by the PATH variable are searched.

Under windows, this also searches the “App Path” registry entries.

paths.thisfile([arg])

Calling `paths.thisfile()` without argument inside a lua file returns the full pathname of the file from which it is called. This function always returns `nil` when called interactively.

Calling `paths.thisfile(arg)` with a string argument `arg` returns the full pathname of the file `arg` relative to the directory containing the file from which function `paths.thisfile` is invoked. This is useful, for instance, to locate files located in the same directory as a lua script.

paths.dofile(filename)

This function is similar to the standard Lua function `dofile` but interprets `filename` relative to the directory containing the file that contains the call to `paths.dofile`, or to the current directory when `paths.dofile` is called interactively.