

版本 2

.DLL 命令 生成声音_ , 整数型, "kernel32.dll", "Beep", 公开, , Beep,用于生成简单的声音 Long, TRUE (非零)表示成功, 否则返回零。会设置 GetLastError

.参数 声音频率, 整数型, , dwFreq,声音频率 (从 37Hz到 32767Hz) Long, 声音频率 (从 37Hz到 32767Hz)。在 windows95中忽略

.参数 声音持续时间, 整数型, , dwDuration,参见相关帮助 Long, 声音的持续时间, 以毫秒为单位。如为-1, 表示一直播放声音, 直到再次调用该函数为止。在 windows95中会被忽略

.DLL 命令 取字节数据地址_ , 整数型, "kernel32.dll", "lstrcpyn", 公开, lstrcpyn

.参数 数据 1, 字节型, 传址, lpString1

.参数 数据 2, 字节型, 传址, lpString2

.参数 数据长度, 整数型, , iMaxLength

.DLL 命令 取字节数组数据地址_ , 整数型, "kernel32.dll", "lstrcpyn", 公开, lstrcpyn

.参数 数据 1, 字节型, 传址 数组, lpString1

.参数 数据 2, 字节型, 传址 数组, lpString2

.参数 数据长度, 整数型, , iMaxLength

.DLL 命令 取字节集数据地址_ , 整数型, "kernel32.dll", "lstrcpyn", 公开, lstrcpyn

.参数 数据 1, 字节集, 传址, lpString1

.参数 数据 2, 字节集, 传址, lpString2

.参数 数据长度, 整数型, , iMaxLength

.DLL 命令 取文本数据地址_ , 整数型, "kernel32.dll", "lstrcpyn", 公开, lstrcpyn

.参数 数据 1, 文本型, 传址, lpString1

.参数 数据 2, 文本型, 传址, lpString2

.参数 数据长度, 整数型, , iMaxLength

.DLL 命令 取文本数组数据地址_ , 整数型, "kernel32.dll", "lstrcpyn", 公开, lstrcpyn

.参数 数据 1, 文本型, 传址 数组, lpString1

.参数 数据 2, 文本型, 传址 数组, lpString2

.参数 数据长度, 整数型, , iMaxLength

.DLL 命令 取整数数据地址_ , 整数型, "kernel32.dll", "lstrcpyn", 公开, lstrcpyn

.参数 数据 1, 整数型, 传址, lpString1

.参数 数据 2, 整数型, 传址, lpString2

.参数 数据长度, 整数型, , iMaxLength

.DLL 命令 取整数数组数据地址_ , 整数型, "kernel32.dll", "lstrcpyn", 公开, lstrcpyn

.参数 数据 1, 整数型, 传址 数组, lpString1

.参数 数据 2, 整数型, 传址 数组, lpString2

.参数 数据长度, 整数型, , iMaxLength

.DLL 命令 lstrcpyn, 整数型, "kernel32.dll", "lstrcpyn", 公开, lstrcpyn

.参数 数据 1, 文本型, 传址, lpString1

.参数 数据 2, 文本型, 传址, lpString2

.参数 数据长度, 整数型, , iMaxLength

.DLL 命令 置线程优先级_ , 整数型, "kernel32", "SetThreadPriority", 公开, \$(b)设定线程的优先级 非零表示成功, 零表示失败。会设置 GetLastError

.参数 线程句柄, 整数型, , hThread

.参数 优先权, 整数型, , nPriority 例如:-2 更过自己去搜索下

.DLL 命令 SetThreadPriority, 整数型, "kernel32", "SetThreadPriority", 公开, \$(b)设定线程的优先级别 非零表示成功, 零表示失败。会设置 GetLastError
.参数 线程句柄, 整数型, , hThread
.参数 优先权, 整数型, , nPriority 例如:-2 更过自己去搜索下

.DLL 命令 置进程优先级_, 整数型, "kernel32", "SetPriorityClass", 公开, \$(b)设置一个进程的优先级别 进程的优先级, 零表示失败。会设置 GetLastError
.参数 进程句柄, 整数型
.参数 级别, 整数型, , 例如:64 更过自己去搜索下

.DLL 命令 SetPriorityClass, 整数型, "kernel32", "SetPriorityClass", 公开, \$(b)设置一个进程的优先级别 进程的优先级, 零表示失败。会设置 GetLastError
.参数 进程句柄, 整数型
.参数 级别, 整数型, 数组, 例如:64 更过自己去搜索下

.DLL 命令 取线程优先级_, 整数型, "kernel32", "GetThreadPriority", 公开, GetThreadPriority
.参数 线程句柄, 整数型, , 线程句柄

.DLL 命令 取特定进程优先级_, 整数型, "kernel32", "GetPriorityClass", 公开, GetPriorityClass\$(b)获取特定进程的优先级别 进程的优先级, 零表示失败。会设置 GetLastError
.参数 进程句柄, 整数型, , 进程句柄;

.DLL 命令 GetThreadPriority, 整数型, "kernel32", "GetThreadPriority", 公开, GetThreadPriority
.参数 线程句柄, 整数型, , 线程句柄

.DLL 命令 GetPriorityClass, 整数型, "kernel32", "GetPriorityClass", 公开, GetPriorityClass\$(b)获取特定进程的优先级别 进程的优先级, 零表示失败。会设置 GetLastError
.参数 进程句柄, 整数型, , 进程句柄;

.DLL 命令 取当前线程伪句柄_, 整数型, "kernel32", "GetCurrentThread", 公开, GetCurrentThread \$(b)获取当前线程的一个伪句柄 当前线程的伪句柄

.DLL 命令 取当前进程伪句柄_, 整数型, "kernel32", "GetCurrentProcess", 公开, GetCurrentProcess\$(b)获取当前进程的一个伪句柄 当前进程的伪句柄 一般是-1

.DLL 命令 GetCurrentThread, 整数型, "kernel32", "GetCurrentThread", 公开, GetCurrentThread\$(b)获取当前线程的一个伪句柄 当前线程的伪句柄一般是-2

.DLL 命令 GetCurrentProcess, 整数型, "kernel32", "GetCurrentProcess", 公开, GetCurrentProcess\$(b)获取当前进程的一个伪句柄 当前进程的伪句柄

.DLL 命令 退出线程_, 整数型, "kernel32", "ExitThread", 公开, ExitThread
.参数 线程句柄, 整数型

.DLL 命令 ExitThread, 整数型, "kernel32", "ExitThread", 公开, ExitThread
.参数 线程句柄, 整数型

.DLL 命令 关机对话框_, 整数型, "shell32.dll", "#60", 公开, #60
.参数 参数, 整数型, , YourGuess

.DLL 命令 复制文件_, 整数型, "kernel32.dll", "CopyFileA", 公开
.参数 源文件名, 文本型, , lpExistingFileName, 被复制的文件名
.参数 目标文件名, 文本型, , lpNewFileName, 目标文件名
.参数 是否覆盖, 整数型, , bFailIfExists, 非 0 不覆盖, 0 覆盖

.DLL 命令 删除文件_, 整数型, "kernel32.dll", "DeleteFileA", 公开, 删除指定文件
.参数 文件名, 文本型, , lpFileName, 欲删除文件的名字

.DLL 命令 最小化窗口_， 整数型, "user32.dll", "CloseWindow", 公开, 非 0 表示成功, 0 表示失败

.参数 窗口句柄, 整数型, , hwnd, 欲最小化窗口的窗口句柄

.DLL 命令 取文件属性_， 整数型, "kernel32", "GetFileAttributesA", 公开, 1、#只读文件; 2、#隐藏文件; 4、#系统文件; 16、#子目录; 32、#存档文件 要判断是否设置了某个属性, 在返回值与想要得知的属性值之间使用“位与”命令进行比较。如果所得的结果不为零, 则表示设置了这个属性值。如果取文件属性失败, 将返回 -1。

.参数 文件名, 文本型

.DLL 命令 取文件扩展名_， 整数型, "shlwapi.dll", "PathFindExtensionA", 公开, 无扩展名返回空 得到的返回整数值要进过 指针到文本代码 才可以取得

.参数 文件路径, 文本型, , pszPath

.DLL 命令 删除扩展名_， "shlwapi.dll", "PathRemoveExtensionA", 公开, 比如: C:\Documents and Settings\Administrator\桌面\l.exe 返回 C:\Documents and Settings\Administrator\桌面\l

.参数 文件路径, 文本型, , pszPath 得到的变量保存在文本路径中

.DLL 命令 NtShutdownSystem, 整数型, "ntdll.dll", , 公开

.参数 se_， 整数型, , 0 为关机, 1 为重起

.DLL 命令 RtlAdjustPrivilege, 整数型, "ntdll.dll", , 公开

.参数 se_， 整数型

.参数 true_， 逻辑型

.参数 hprocess, 整数型

.参数 ret, 整数型, 传址

.DLL 命令 播放声音文件_， 整数型, "winmm.dll", "PlaySoundA", 公开, 播放声音文件

.参数 声音文件名, 文本型, , lpzName

.参数 模块句柄, 整数型, , hModule 一般填写 1

.参数 标志, 整数型, , dwFlags 一般填写 0

.DLL 命令 是否为管理员_， 整数型, "advpack.dll", "IsNTAdmin", 公开, ' 返回值非 0 为管理员

.参数 参数 1, 整数型, , dwReserved, 默认 0

.参数 参数 2, 整数型, , lpdwReserved, 默认 0

.DLL 命令 ShellExecute, 整数型, "ShellExecuteA", 公开, 执行命令

.参数 窗口句柄, 整数型, , 父窗口句柄, 可为 0 hwnd 0 指定一个窗口的句柄, 有时候, windows 程序有必要在创建自己的主窗口前显示一个消息框

.参数 操作方式, 文本型, , lpOperation“open” 指定字符串“open”来打开 lpFile 文档, 或指定“Print”来打印它

.参数 链接地址, 文本型, , lpFile 要打开的网页地址 或 mailo:709760015@qq.com 想用关联程序打印或打开一个程序名或文件名

.参数 命令行参数, 文本型, , lpParameters“”或"0" 如 lpzFile 是可执行文件, 则这个字符串包含传递给执行程序的参数

.参数 完整路径, 文本型, , lpDirectory“”或"0" 想使用的完整路径

.参数 窗口显示方式, 短整数型, , nShowCmd 5 或 1 定义了如何显示启动程序的常数值。参考 ShowWindow 函数的nCmdShow 参数;

.DLL 命令 SetWindowTextW, 整数型, "user32.dll", "SetWindowTextA", 公开

.参数 窗口句柄, 整数型

- .参数 窗口标题, 文本型
- .DLL 命令 取标题文本长度_, 整数型, "user32.dll", "GetWindowTextLengthA", 公开
 - .参数 窗口句柄, 整数型
- .DLL 命令 取窗口标题_, 整数型, "user32", "GetWindowTextA", 公开, 一个窗体的标题 (caption) 文字, 或者一个控件的内容 (在 vb 里使用: 使用 vb 窗体或控件的caption 或 text 属性) 复制到 lpString 的字串长度; 不包括空中止字符。会设置 GetLastError
 - .参数 窗口句柄, 整数型, , 欲获取文字的那个窗口的句柄
 - .参数 缓冲区, 文本型, , 预定义的一个缓冲区, 至少有 cch+1 个字
 - .参数 缓冲尺寸, 整数型, , lp 缓冲区的长度;
- .DLL 命令 设置窗口标题, 整数型, "user32.dll", "SetWindowTextA", 公开
- .DLL 命令 WritePrivateProfileString, 整数型, "kernel32", "WritePrivateProfileStringA", 公开, 相当于写配置项
 - .参数 节名称, 文本型, , lpApplicationName String, 要在其中写入新字串的小节名称。这个字串不区分大小写
 - .参数 配置项名称, 文本型, , lpAppName, 要设置的小节。这个字串不区分大小写 Any, 要设置的项名或条目名。这个字串不区分大小写。用 vbNullString 可删除这个小节的所有设置项
 - .参数 欲写入值, 文本型, , lpString, 项和值字串的一个列表。
 - .参数 配置项文件名, 文本型, , lpFileName, 初始化文件的名字。
- .DLL 命令 取当前目录_, 整数型, "kernel32", "GetCurrentDirectoryA", 公开
 - .参数 缓冲区长度, 整数型
 - .参数 缓冲区, 文本型
- .DLL 命令 GetPrivateProfileStringA, 整数型, "kernel32", "GetPrivateProfileStringA", 公开, 相当于读配置项
 - .参数 节名称, 文本型, , lpApplicationName String, 要在其中写入新字串的小节名称。这个字串不区分大小写
 - .参数 配置项名称, 文本型, , lpAppName, 要设置的小节。这个字串不区分大小写 Any, 要设置的项名或条目名。这个字串不区分大小写。用 vbNullString 可删除这个小节的所有设置项
 - .参数 默认文本, 文本型, , lpString, 项和值字串的一个列表。
 - .参数 返回文本, 文本型, , String, 指定一个字串缓冲区, 长度至少为nSize
 - .参数 返回文本大小, , , Long, 指定装载到lpReturnedString 缓冲区的最大字符数量
 - .参数 配置项文件名, 文本型, , lpFileName, 初始化文件的名字。
- .DLL 命令 初始化配置项_, 整数型, "kernel32.dll", "WritePrivateProfileSectionA", 公开, \$(b)
 - .参数 节名称, 文本型, , lpAppName, 要设置的小节。这个字串不区分大小写
 - .参数 配置项名称, 文本型, , lpString, 项和值字串的一个列表。
 - .参数 配置项文件名, 文本型, , lpFileName, 初始化文件的名字。
- .DLL 命令 获取配置项列表_, 整数型, "kernel32", "GetPrivateProfileSectionA", 公开, 相当于取配置节名
 - .参数 节名称, 文本型, , lpAppName
 - .参数 返回文本, 文本型, , lpReturnedString
 - .参数 返回文本大小, 整数型, , nSize
 - .参数 配置项文件名, 文本型, , lpFileName
- .DLL 命令 MessageBox, 整数型, , "MessageBox A", 公开

- .参数 窗口句柄, 整数型
- .参数 提示信息, 文本型
- .参数 窗口标题, 文本型
- .参数 按钮值, 整数型
- .DLL 命令 MessageBox W, 整数型, "MessageBox W", 公开
 - .参数 窗口句柄, 整数型
 - .参数 提示信息, 文本型
 - .参数 窗口标题, 文本型
 - .参数 按钮值, 整数型
- .DLL 命令 strlenA, 整数型, "kernel32.dll", "strlenA", 公开
 - .参数 文本 lpString, 文本型
- .DLL 命令 strcatA, 整数型, "kernel32", "strcatA", 公开
 - .参数 文本 lpString1 被赋值的文本, 文本型, lpString1
 - .参数 文本 lpString2 用作赋值的文本, 文本型, lpString2
- .DLL 命令 strcpyA, 整数型, "kernel32.dll", "strcpyA", 公开
 - .参数 文本 lpString1 被赋值的文本, 文本型
 - .参数 文本 lpString2 用作赋值的文本, 文本型
- .DLL 命令 GetCurrentDirectoryA, 整数型, "kernel32", "GetCurrentDirectoryA", 公开
 - .参数 缓冲区长度, 整数型
 - .参数 缓冲区, 文本型
- .DLL 命令 目录是否为空_, 整数型, "shlwapi.dll", "PathIsDirectoryEmptyA", 公开, 是返回非 0, 不存在或不是空目录返回 0
 - .参数 路径文本, 文本型, pszPath
- .DLL 命令 退出进程_, 整数型, "kernel32", "ExitProcess", 公开
 - .参数 进程句柄, 整数型
- .DLL 命令 PathIsDirectoryEmptyA, 整数型, "shlwapi.dll", "PathIsDirectoryEmptyA", 公开, 是返回非 0, 不存在或不是空目录返回 0
 - .参数 路径文本, 文本型, pszPath
- .DLL 命令 ExitProcess, 整数型, "kernel32", "ExitProcess", 公开
 - .参数 进程句柄, 整数型
- .DLL 命令 取得 Win 目录_, 整数型, "kernel32.dll", "GetWindowsDirectoryA", 公开, 取得 WINDOWS 返回的值后面没有"\ " 你可以自己增加上去
 - .参数 缓冲区, 文本型
 - .参数 尺寸, 整数型
- .DLL 命令 GetWindowsDirectoryA, 整数型, "kernel32.dll", "GetWindowsDirectoryA", 公开, 取得 WINDOWS 返回的值后面没有"\ " 你可以自己增加上去
 - .参数 缓冲区, 文本型
 - .参数 尺寸, 整数型
- .DLL 命令 取计算机名_, 整数型, "kernel32", "GetComputerNameA", 公开, TRUE (非零) 表示成功, 否则返回零。会设置 GetLastError
 - .参数 字符串缓冲区, 文本型, lpBuffer, 随同计算机名载入的字符串缓冲区
 - .参数 缓冲区长度, 整数型, nSize, 缓冲区的长度。这个变量随同返回计算机名的实际长度载入
- .DLL 命令 GetComputerNameA, 整数型, "kernel32", "GetComputerNameA", 公开, TRUE (非

零) 表示成功, 否则返回零。会设置 GetLastError

.参数 字符串缓冲区, 文本型, lpBuffer, 随同计算机名载入的字符串缓冲区

.参数 缓冲区长度, 整数型, 传址, nSize, 缓冲区的长度。这个变量随同返回计算机名的实际长度载入

.DLL 命令 取系统目录, 逻辑型, "shell32.dll", "SHGetSpecialFolderPathA", 公开, ' 只要改变 API“取系统目录”参数中的“标识”值, 就可以取不同的目录 0 我的桌面 1 临时目录 5 我的文档 6 我的收藏夹 7 我的启动 11 我的开始菜单 20 系统字体 36 Windows 安装目录 37 系统目录 [99 更多]

.参数 窗口句柄, 整数型

.参数 路径, 文本型

.参数 标识, 整数型, ' 只要改变 API“取系统目录”参数中的“标识”值, 就可以取不同的目录

.参数 是否创建, 逻辑型, , 真: 不存在就创建, 假: 不创建

.DLL 命令 SHGetSpecialFolderPathA, 逻辑型, "shell32.dll", "SHGetSpecialFolderPathA", 公开, ' 只要改变 API“取系统目录”参数中的“标识”值, 就可以取不同的目录

.参数 窗口句柄, 整数型

.参数 路径, 文本型

.参数 标识, 整数型, ' 只要改变 API“取系统目录”参数中的“标识”值, 就可以取不同的目录

.参数 是否创建, 逻辑型, , 真: 不存在就创建, 假: 不创建

.DLL 命令 GetSystemDirectory, 整数型, , "GetSystemDirectoryA", 公开

.参数 缓冲区, 文本型

.参数 缓冲区长度, 整数型

.DLL 命令 取 system32 目录, 整数型, , "GetSystemDirectoryA", 公开

.参数 缓冲区, 文本型

.参数 缓冲区长度, 整数型

.DLL 命令 GetTempPathA, 整数型, , "GetTempPathA", 公开

.参数 缓冲区长度, 整数型

.参数 缓冲区, 文本型

.DLL 命令 取临时目录, 整数型, , "GetTempPathA", 公开

.参数 缓冲区长度, 整数型

.参数 缓冲区, 文本型

.DLL 命令 鼠标左右键, 整数型, "user32.dll", "SwapMouseButton", 公开, 决定是否互换鼠标左右键的功能 TRUE (非零) 表示鼠标按钮的功能在调用这个函数之前已经互换; 否则返回零

.参数 鼠标按钮的功能, 整数型, , 倘若为 TRUE (非零), 则互换两个鼠标按钮的功能。FALSE 则恢复正常状态;

.DLL 命令 SwapMouseButton, 整数型, "user32.dll", "SwapMouseButton", 公开, 决定是否互换鼠标左右键的功能 TRUE (非零) 表示鼠标按钮的功能在调用这个函数之前已经互换; 否则返回零

.参数 鼠标按钮的功能, 整数型, , 倘若为 TRUE (非零), 则互换两个鼠标按钮的功能。FALSE 则恢复正常状态;

.DLL 命令 取屏幕句柄, 整数型, "user32", "GetDesktopWindow", 公开, 返回屏幕句柄

.DLL 命令 GetDesktopWindow, 整数型, "user32", "GetDesktopWindow", 公开, 返回屏幕句柄

- .DLL 命令 删除文件, 整数型, "kernel32", "DeleteFileA", 公开, 非零表示成功, 零表示失败
 - .参数 欲删除文件的名字, 文本型
- .DLL 命令 DeleteFile, 整数型, "kernel32", "DeleteFileA", 公开, 非零表示成功, 零表示失败
 - .参数 欲删除文件的名字, 文本型
- .DLL 命令 建立模板目录_, 整数型, "kernel32.dll", "CreateDirectoryExA", 公开, Long, 非零表示成功, 零表示失败。
 - .参数 模板目录, 文本型
 - .参数 新目录名称, 文本型, lpNewDirectory, 新目录名称
 - .参数 安全属性, 整数型, lpSecurityAttributes 一般为 0 即可
- .DLL 命令 CreateDirectoryExA, 整数型, "kernel32.dll", "CreateDirectoryExA", 公开, Long, 非零表示成功, 零表示失败。
 - .参数 模板目录, 文本型
 - .参数 新目录名称, 文本型, lpNewDirectory, 新目录名称
 - .参数 安全属性, 整数型, lpSecurityAttributes 一般为 0 即可
- .DLL 命令 十六进制转十进制_, 整数型, "shlwapi.dll", "StrToIntExA", 公开, StrToIntExA 成功返回 1(非 0) 失败返回 0 可以判断是不是输入的不是 0 到 9 A 到 F
 - .参数 转换文本, 文本型, pszString "0x" + 转换文本 pszString 一般为 如果标志为 0 的话 则直接填写 转换文本 如果标志为 1 的话 转换文本="0x"+转换文本
 - .参数 标志, 整数型, dwFlags 1 dwFlags 标志为 0 则为 10 进制 标志为 1 则为十六进制
 - .参数 返回值, 整数型, piRet 得到的 10 进制
- .DLL 命令 删除指定目录_, 整数型, "kernel32.dll", "RemoveDirectoryA", 公开, RemoveDirectory, 删除指定目录, 在调用这个函数前, 目录必须为空
 - .参数 目录的名字, 文本型, lppathname, String, 要删除的那个目录的名字
- .DLL 命令 创建目录_, 整数型, "shell32.dll", "SHCreateDirectoryExA", 公开, 非零则成功 0 为失败
 - .参数 窗口句柄, 整数型, hwnd
 - .参数 路径, 文本型, pszPath
 - .参数 参数, 整数型, psa, any 一般为 0 即可
- .DLL 命令 RemoveDirectoryA, 整数型, "kernel32.dll", "RemoveDirectoryA", 公开, RemoveDirectory, 删除指定目录, 在调用这个函数前, 目录必须为空
 - .参数 目录的名字, 文本型, lppathname, String, 要删除的那个目录的名字
- .DLL 命令 添加新分区_, "kernel32", "DefineDosDeviceA", 公开,
 - .参数 类型, 整数型, 0 为创建 3 为删除
 - .参数 新区名, 文本型, 比如: H: ()
 - .参数 创建虚拟新区的目录, 文本型, 当你选择 3 时, 一定要为空
- .DLL 命令 调用系统关于窗口, 整数型, "shell32.dll", "Shell AboutA", 公开
 - .参数 窗口句柄, 整数型
 - .参数 窗口标题, 文本型
 - .参数 说明, 文本型
 - .参数 图标句柄, 整数型, 比如 _启动窗口.发送信息 (127, 1, 0)
- .DLL 命令 ShellAboutA, 整数型, "shell32.dll", "Shell AboutA", 公开
 - .参数 窗口句柄, 整数型
 - .参数 窗口标题, 文本型

- .参数 说明, 文本型
- .参数 图标句柄, 整数型, , 比如 _启动窗口.发送信息 (127, 1, 0)
- .DLL 命令 发送消息_, 整数型, "user32", "SendMessageA", , , 调用一个窗口的窗口函数, 将一条消息发给那个窗口。除非消息处理完毕, 否则该函数不会返回。SendMessageBynum, ; SendMessageByString 是该函数的“类型安全”声明形式 由具体的消息决定
 - .参数 窗口句柄, 整数型, , 要接收消息的那个窗口的句柄
 - .参数 信息值, 整数型, , 消息的标识符
 - .参数 参数一, 逻辑型, , 具体取决于消息
 - .参数 参数二, 整数型, , 具体取决于消息;
- .DLL 命令 发送消息__, 整数型, "user32", "SendMessageA", , , 调用一个窗口的窗口函数, 将一条消息发给那个窗口。除非消息处理完毕, 否则该函数不会返回。SendMessageBynum, ; SendMessageByString 是该函数的“类型安全”声明形式 由具体的消息决定
 - .参数 窗口句柄, 整数型, , 要接收消息的那个窗口的句柄
 - .参数 信息值, 整数型, , 消息的标识符
 - .参数 参数一, 整数型, , 具体取决于消息
 - .参数 参数二, 整数型, , 具体取决于消息;
- .DLL 命令 SendMessageA, 整数型, "user32", "SendMessageA", , , 调用一个窗口的窗口函数, 将一条消息发给那个窗口。除非消息处理完毕, 否则该函数不会返回。SendMessageBynum, ; SendMessageByString 是该函数的“类型安全”声明形式 由具体的消息决定
 - .参数 窗口句柄, 整数型, , 要接收消息的那个窗口的句柄
 - .参数 信息值, 整数型, , 消息的标识符
 - .参数 参数一, 逻辑型, , 具体取决于消息
 - .参数 参数二, 整数型, , 具体取决于消息;
- .DLL 命令 DefineDosDeviceA, "kernel32", "DefineDosDeviceA", 公开,
 - .参数 类型, 整数型, , 0 为创建 3 为删除
 - .参数 新区名, 文本型, , 比如: H: ()
 - .参数 创建虚拟新区的目录, 文本型, , 当你选择 3时, 一定要为空
- .DLL 命令 取鼠标位置_, 整数型, "user32", "GetCursorPos", 公开
 - .参数 位置, 鼠标坐标 GetCursorPos
- .DLL 命令 GetCursorPos, 整数型, "user32", "GetCursorPos", 公开
 - .参数 位置, 鼠标坐标 GetCursorPos
- .DLL 命令 取鼠标所在窗口句柄_, 整数型, "user32.dll", "WindowFromPoint", 公开, ,
 - .参数 横坐标, 整数型, , xPoint, x 点值
 - .参数 纵坐标, 整数型, , yPoint, y 点值
- .DLL 命令 WindowFromPoint, 整数型, "user32.dll", "WindowFromPoint", 公开, ,
 - .参数 横坐标, 整数型, , xPoint, x 点值
 - .参数 纵坐标, 整数型, , yPoint, y 点值
- .DLL 命令 取指定点窗口句柄_, 整数型, "user32.dll", "WindowFromPoint", 公开, ,
 - .参数 横坐标, 整数型, , xPoint, x 点值
 - .参数 纵坐标, 整数型, , yPoint, y 点值
- .DLL 命令 取进程版本_, 整数型, "kernel32.dll", "GetProcessVersion", 公开
 - .参数 进程标识符ProcessId, 整数型, , ProcessId
- .DLL 命令 GetProcessVersion, 整数型, "kernel32.dll", "GetProcessVersion", 公开
 - .参数 进程标识符ProcessId, 整数型, , ProcessId

- .DLL 命令 取前台窗口句柄_ , 整数型, "user32", "GetForegroundWindow", 公开
- .DLL 命令 GetForegroundWindow, 整数型, "user32", "GetForegroundWindow", 公开
- .DLL 命令 取得当前活动窗口句柄_ , 整数型, "user32", "GetForegroundWindow", 公开
- .DLL 命令 SHCreateDirectoryExA, 整数型, "shell32.dll", "SHCreateDirectoryExA", 公开, 非零则成功 0 为失败
 - .参数 窗口句柄, 整数型, , hwnd
 - .参数 路径, 文本型, , pszPath
 - .参数 参数, 整数型, , psa, any 一般为 0 即可
- .DLL 命令 窗口置顶_ , 整数型, "user32.dll", "BringWindowToTop", 公开
 - .参数 窗口句柄, 整数型, , hwnd, 欲带至顶部的那个窗口的句柄
- .DLL 命令 运行 WinExec_ , 整数型, , "WinExec", 公开, 成功则返回 3 失败返回 2
 - .参数 欲运行的命令行, 文本型, , 欲运行的文件名
 - .参数 被运行程序窗口显示方式, 整数型, , : 0、#隐藏窗口; 1、#普通激活; 2、#最小化激活; 3、#最大化激活; 4、#普通不激活; 5、#最小化不激活
- .DLL 命令 运行 WinExec_1, 整数型, "WinExec", 公开, 成功则返回 3 失败返回 2
 - .参数 欲运行的命令行, 文本型, , 欲运行的文件名
 - .参数 被运行程序窗口显示方式, 整数型, , : 0、#隐藏窗口; 1、#普通激活; 2、#最小化激活; 3、#最大化激活; 4、#普通不激活; 5、#最小化不激活
- .DLL 命令 提取文件图标_ , 整数型, "shell32.dll", "ExtractIconA", 公开, 判断一个可执行文件或 DLL 中是否有图标存在, 并将其提取出来
 - .参数 程序句柄, 整数型, , hInst, 当前应用程序的实例句柄也可用 GetWindowWord 函数取得拥有一个窗体或控件的实例的句柄
 - .参数 文件名, 文本型, , lpzExeFileName, 在其中提取图标的那个程序的全名
 - .参数 图标索引, 整数型, , nIndex, 欲获取的图标的索引
- .DLL 命令 ExtractIconA, 整数型, "shell32.dll", "ExtractIconA", 公开, 判断一个可执行文件或 DLL 中是否有图标存在, 并将其提取出来
 - .参数 程序句柄, 整数型, , hInst, 当前应用程序的实例句柄
 - .参数 文件名, 文本型, , lpzExeFileName, 在其中提取图标的那个程序的全名
 - .参数 图标索引, 整数型, , nIndex, 欲获取的图标的索引
- .DLL 命令 投递消息, 整数型, "user32", "PostMessageA", 公开, 调用一个窗口的窗口函数, 将一条消息发给那个窗口。除非消息处理完毕, 否则该函数不会返回。SendMessageBynum, ; SendMessageByString 是该函数的“类型安全”声明形式 由具体的消息决定
 - .参数 窗口句柄, 整数型, , 要接收消息的那个窗口的句柄。
 - .参数 信息值, 整数型, , 消息的标识符。
 - .参数 参数 1, 整数型, , 具体取决于消息。
 - .参数 参数 2, 整数型, , 具体取决于消息。
- .DLL 命令 PostMessageA, 整数型, "user32", "PostMessageA", 公开, 调用一个窗口的窗口函数, 将一条消息发给那个窗口。除非消息处理完毕, 否则该函数不会返回。SendMessageBynum, ; SendMessageByString 是该函数的“类型安全”声明形式 由具体的消息决定
 - .参数 窗口句柄, 整数型, , 要接收消息的那个窗口的句柄。
 - .参数 信息值, 整数型, , 消息的标识符。
 - .参数 参数 1, 整数型, , 具体取决于消息。
 - .参数 参数 2, 整数型, , 具体取决于消息。

.DLL 命令 **BringWindowToTop**, 整数型, "user32.dll", "BringWindowToTop", 公开
.参数 窗口句柄, 整数型, , hwnd, 欲带至顶部的那个窗口的句柄

.DLL 命令 **寻找顶级窗口_**, 整数型, "user32.dll", "FindWindowA", 公开, FindWindow, 寻找窗口列表中第一个符合指定条件的顶级窗口 参数填写正确的话返回窗口句柄
.参数 窗口类名, 文本型, , lpClassName, 参见相关帮助
.参数 窗口文本, 文本型, , lpWindowName, 参见相关帮助

.DLL 命令 **获取指定窗口句柄_**, 整数型, "user32.dll", "FindWindowA", 公开, FindWindow, 寻找窗口列表中第一个符合指定条件的顶级窗口 参数填写正确的话返回窗口句柄
.参数 窗口类名, 文本型, , lpClassName, 参见相关帮助
.参数 窗口文本, 文本型, , lpWindowName, 参见相关帮助

.DLL 命令 **FindWindowA**, 整数型, "user32.dll", "FindWindowA", 公开, FindWindow, 寻找窗口列表中第一个符合指定条件的顶级窗口 参数填写正确的话返回窗口句柄
.参数 窗口类名, 文本型, , lpClassName, 参见相关帮助
.参数 窗口文本, 文本型, , lpWindowName, 参见相关帮助

.DLL 命令 **清空回收站_**, 整数型, "shell32.dll", "SHEmptyRecycleBinA", 公开, SHEmptyRecycleBin
.参数 窗口句柄, 整数型, , hwnd
.参数 根路径, 文本型, , pszRootPath 可以为整数 如果类型是文本设置为空即可 如果类型是整数 填写 0 即可
.参数 标志, 整数型, , dwFlags 表示为 0 则提示是否清空回收站 7 则不提示直接清空回收站

.DLL 命令 **取剪辑板窗口句柄_**, 整数型, "user32.dll", "GetOpenClipboardWindow", 公开, \$(b) 返回打开 CLIPBOARD 的窗口句柄

.DLL 命令 **清空剪辑板_**, 整数型, "user32.dll", "EmptyClipboard", 公开, 清除当前 Windows 系统剪辑板中的所有数据

.DLL 命令 **EmptyClipboard**, 整数型, "user32.dll", "EmptyClipboard", 公开, 清除当前 Windows 系统剪辑板中的所有数据

.DLL 命令 **恢复窗口_**, 整数型, "user32.dll", "OpenIcon", 公开, 恢复被最小化或最大化的窗口
.参数 窗口句柄, 整数型, , hwnd, 窗口句柄

.DLL 命令 **打开剪贴板_**, 整数型, "user32.dll", "OpenClipboard", 公开, (0 为失败)
.参数 窗口句柄, 整数型, , hwnd, 窗口句柄

.DLL 命令 **OpenClipboard**, 整数型, "user32.dll", "OpenClipboard", 公开, (0 为失败)
.参数 窗口句柄, 整数型, , hwnd, 窗口句柄

.DLL 命令 **确定格式是否可用_**, 整数型, "user32", "IsClipboardFormatAvailable", 公开, (0 为失败)
.参数 wFormat, 整数型, , CF_TEXT

.DLL 命令 **取剪辑版序号_**, 整数型, "user32.dll", "GetClipboardSequenceNumber", 公开

.DLL 命令 **取剪贴板数据句柄_**, 整数型, "user32.dll", "GetClipboardData", 公开, 返回剪贴板数据句柄
.参数 剪贴板格式, 整数型, , wFormat

.DLL 命令 **API_CallWindowProc**, 整数型, "user32", "CallWindowProcA", , 将消息传答窗口函数
.参数 lpPrevWndFunc, 整数型

.参数 hwnd, 整数型

.参数 msg, 整数型

.参数 wParam, 整数型

.参数 lParam, 整数型

.DLL 命令 API_SetWindowLong, 整数型, "user32", "SetWindowLongA", , 在窗口结构中为指定的窗口设置信息指定数据的前一个值

.参数 hwnd, 整数型, , 欲为其取得信息的窗口的句柄

.参数 nIndex, 整数型, , 请参考 GetWindowLong 函数的 nIndex参数的说明

.参数 dwNewLong, 整数型, , 由 nIndex 指定的窗口信息的新值;

.DLL 命令 锁住全局内存块_, 文本型, "kernel32.dll", "GlobalLock", 公开

.参数 内存句柄, 整数型, , hMem

.DLL 命令 解锁全局内存块_, 整数型, "kernel32.dll", "GlobalUnlock", 公开

.参数 内存句柄, 整数型, , hMem

.DLL 命令 关闭剪贴板_, 整数型, "user32.dll", "CloseClipboard", 公开, 0 为失败

.DLL 命令 CloseClipboard, 整数型, "user32.dll", "CloseClipboard", 公开

.DLL 命令 OpenIcon, 整数型, "user32.dll", "OpenIcon", 公开, 恢复被最小化或最大化的窗口

.参数 剪贴板句柄, 整数型

.参数 窗口句柄, 整数型, , hwnd, 窗口句柄

.DLL 命令 启动 Internet控制面板_, 整数型, "inetctl.cpl", "LaunchInternetControlPanel", 公开

.参数 父窗口句柄, 整数型, , hwndParent

.DLL 命令 LaunchInternetControlPanel, 整数型, "inetctl.cpl", "LaunchInternetControlPanel", 公开

.参数 父窗口句柄, 整数型, , hwndParent

.DLL 命令 启动连接对话框_, 整数型, "inetctl.cpl", "LaunchConnectionDialog", 公开

.参数 父窗口句柄, 整数型, , hwndParent

.DLL 命令 LaunchConnectionDialog, 整数型, "inetctl.cpl", "LaunchConnectionDialog", 公开

.参数 父窗口句柄, 整数型, , hwndParent

.DLL 命令 启动安全对话框_, 整数型, "inetctl.cpl", "LaunchSecurityDialog", 公开

.参数 父窗口句柄, 整数型, , hwndParent

.参数 保留参数, 整数型, , 一般填写 0即可

.DLL 命令 LaunchSecurityDialog, 整数型, "inetctl.cpl", "LaunchSecurityDialog", 公开

.参数 父窗口句柄, 整数型, , hwndParent

.参数 保留参数, 整数型, , 一般填写 0即可

.DLL 命令 启动证书对话框_, 整数型, "inetctl.cpl", "LaunchSiteCertDialog", 公开

.参数 父窗口句柄, 整数型, , hwndParent

.DLL 命令 LaunchSiteCertDialog, 整数型, "inetctl.cpl", "LaunchSiteCertDialog", 公开

.参数 父窗口句柄, 整数型, , hwndParent

.DLL 命令 寻找子窗口_, 整数型, "user32", "FindWindowExA", 公开, , 在窗口列表中寻找与指定条件相符的第一个子窗口 找到的窗口的句柄。如未找到相符窗口, 则返回零。会设置 GetLastError

.参数 窗口句柄 1, 整数型, , 在其中查找子的父窗口。如设为零, 表示使用桌面窗口(通常说的顶级窗口都被认为是桌面的子窗口, 所以也会对它们进行查找)

.参数 窗口句柄 2, 整数型, , 一般可以设置为 0 从这个窗口后开始查找。这样便可利用对 FindWindowEx 的多次调用找到符合条件的所有子窗口。如设为零, 表示从第一个子

窗口开始搜索

.参数 欲搜索的类名 1, 文本型, 欲搜索的类名。 零表示忽略 易语言代码: 字符(0)

.参数 欲搜索的类名 2, 文本型, 指向一个指定了窗口名(窗口标题)的空结束字符串 "按钮"

.DLL 命令 FindWindowExA_ 整数型, "user32", "FindWindowExA", 公开, 在窗口列表中寻找与指定条件相符的第一个子窗口 找到的窗口的句柄。如未找到相符窗口, 则返回零。会设置 GetLastError

.参数 窗口句柄 1, 整数型, 在其中查找子的父窗口。如设为零, 表示使用桌面窗口(通常说的顶级窗口都被认为是桌面的子窗口, 所以也会对它们进行查找)

.参数 窗口句柄 2, 整数型, 一般可以设置为 0 从这个窗口后开始查找。这样便可利用对 FindWindowEx 的多次调用找到符合条件的所有子窗口。如设为零, 表示从第一个子窗口开始搜索

.参数 欲搜索的类名 1, 文本型, 欲搜索的类名。 零表示忽略 易语言代码: 字符(0)

.参数 欲搜索的类名 2, 文本型, 指向一个指定了窗口名(窗口标题)的空结束字符串 "按钮"

.DLL 命令 发送消息文本_ 整数型, "user32", "SendMessageA", 调用一个窗口的窗口函数, 将一条消息发给那个窗口。除非消息处理完毕, 否则该函数不会返回。SendMessageBynum, ; SendMessageByString 是该函数的“类型安全”声明形式 由具体的消息决定

.参数 窗口句柄, 整数型, 要接收消息的那个窗口的句柄

.参数 信息值, 整数型, 消息的标识符

.参数 参数一, 整数型, 具体取决于消息

.参数 参数二, 文本型, 具体取决于消息;

.DLL 命令 取得类名称_ 整数型, "user32", "GetClassNameA", 公开

.参数 窗口句柄, 整数型

.参数 缓冲区, 文本型, 传址

.参数 缓冲区长度, 整数型

.DLL 命令 取窗口类名_ 整数型, "user32", "GetClassNameA", 公开

.参数 窗口句柄, 整数型

.参数 缓冲区, 文本型, 传址

.参数 缓冲区长度, 整数型

.DLL 命令 GetClassNameA 整数型, "user32", "GetClassNameA", 公开

.参数 窗口句柄, 整数型

.参数 缓冲区, 文本型, 传址

.参数 缓冲区长度, 整数型

.DLL 命令 MessageBeep 整数型, "user32.dll", "MessageBeep", 公开, 非零表示成功 零表示不成功

.参数 声音常量, 整数型, wType, 参见相关帮助 16、#错误图标; 32、#询问图标; 48、#警告图标; 64、#信息图标

.DLL 命令 SetCursorPos 整数型, "user32.dll", "SetCursorPos", 公开, 非零表示成功, 零表示失败

.参数 水平位置, 整数型, 鼠标指针在屏幕像素坐标系统中的水平位置

.参数 垂直位置, 整数型, 鼠标指针在屏幕像素坐标系统中的垂直位置

.DLL 命令 设置鼠标坐标_ 整数型, "user32.dll", "SetCursorPos", 公开, 非零表示成功, 零表示失败

.参数 水平位置, 整数型, , 鼠标指针在屏幕像素坐标系统中的水平位置

.参数 垂直位置, 整数型, , 鼠标指针在屏幕像素坐标系统中的垂直位置

.DLL 命令 系统状态栏信息_, 整数型, "shell32.dll", "SHAppBarMessage", 公开, SHAppBarMessage

.参数 信息值, 整数型, , dwMessage

.参数 数据结构, 系统状态栏信息_, , pData

.DLL 命令 SHAppBarMessage, 整数型, "shell32.dll", "SHAppBarMessage", 公开, SHAppBarMessage

.参数 信息值, 整数型, , dwMessage

.参数 数据结构, 系统状态栏信息_, , pData

.DLL 命令 延迟时间_, 整数型, "kernel32.dll", "Sleep", 公开, , Sleep

.参数 延迟毫秒数, 整数型, , dwMilliseconds

.DLL 命令 判断窗口可见性_, 整数型, "user32.dll", "IsWindowVisible", 公开, , 判断窗口是否可见

.参数 窗口句柄, 整数型, , hwnd, 要测试的那个窗口的句柄

.DLL 命令 窗口是否可见, 整数型, "user32.dll", "IsWindowVisible", 公开, , 判断窗口是否可见 0 为隐藏 1 为显示

.参数 窗口句柄, 整数型, , hwnd, 要测试的那个窗口的句柄

.DLL 命令 IsWindowVisible, 整数型, "user32.dll", "IsWindowVisible", 公开, , 判断窗口是否可见 0 为隐藏 1 为显示

.参数 窗口句柄, 整数型, , hwnd, 要测试的那个窗口的句柄

.DLL 命令 Sleep, 整数型, "kernel32.dll", "Sleep", 公开, , Sleep

.参数 延迟毫秒数, 整数型, , dwMilliseconds

.DLL 命令 SHEmptyRecycleBinA, 整数型, "shell32.dll", "SHEmptyRecycleBinA", 公开, SHEmptyRecycleBin

.参数 窗口句柄, 整数型, , hwnd

.参数 根路径, 文本型, , pszRootPath 可以为整数 如果类型是文本设置为空即可 如果类型是整数 填写 0 即可

.参数 标志, 整数型, , dwFlags 表示为 0 则提示是否清空回收站 7 则不提示直接清空回收站

.DLL 命令 激活禁止窗口, 整数型, "user32.dll", "EnableWindow", 公开, , 在指定的窗口里允许或禁止所有鼠标及键盘输入

.参数 窗口句柄, 整数型, , hwnd, 窗口句柄

.参数 标识符, 逻辑型, , fEnable, 非零允许, 零禁止

.DLL 命令 EnableWindow, 整数型, "user32.dll", "EnableWindow", 公开, , 在指定的窗口里允许或禁止所有鼠标及键盘输入

.参数 窗口句柄, 整数型, , hwnd, 窗口句柄

.参数 标识符, 逻辑型, , fEnable, 非零允许, 零禁止

.DLL 命令 显示隐藏鼠标, 整数型, "ShowCursor", 公开, 控制鼠标指针的可视性(返回值:显示计数(参考注解)) 当设为显示鼠标返回值为正数相加 0 1 2 3 4..... 当设为隐藏鼠标 返回值为负数相减 -1 -2 -3 -4.....

.参数 标识符, 逻辑型, , (非零)显示指针, 零 隐藏 真为显示 假为隐藏

.DLL 命令 ShowCursor, 整数型, "ShowCursor", 公开, 控制鼠标指针的可视性(返回值:显示计数(参考注解)) 当设为显示鼠标返回值为正数相加 0 1 2 3 4..... 当设为隐藏鼠标 返回值

为负数相减-1 -2 -3 -4

.参数 标示符, 整数型, (非零)显示指针, 零 隐藏

.DLL 命令 是否为活动窗口, 整数型, "user32", "IsWindowEnabled", 公开, 0 为非活动窗体
非 0 则为活动窗体

.参数 窗口句柄, 整数型

.DLL 命令 IsWindowEnabled, 整数型, "user32", "IsWindowEnabled", 公开, 0 为非活动窗体
非 0 则为活动窗体

.参数 窗口句柄, 整数型

.DLL 命令 枚举所有子窗体, 整数型, "user32", "EnumChildWindows", 公开

.参数 父窗口句柄, 整数型

.参数 子窗口调用指针, 子程序指针

.参数 枚举期间传递的操作值, 整数型, 传址

.DLL 命令 当前是否为调试版_, 整数型, "kernel32.dll", "IsDebuggerPresent", 公开, 非 0 则当前
进程正在调试器的描述表中运行 否则 当前进程没有在调试器的描述表中运行

.DLL 命令 文件是否存在_, 整数型, "shell32.dll", "#45", , #45

.参数 路径, 文本型, , szPath

.DLL 命令 IsDebuggerPresent, 整数型, "kernel32.dll", "IsDebuggerPresent", 公开, 为 0 则当前
进程正在调试器的描述表中运行 否则 当前进程没有在调试器的描述表中运行

.DLL 命令 检查 Internet连接_, 整数型, "wininet.dll", "InternetCheckConnectionA", 公开,
InternetCheckConnectionA' 连接网站, 成功返回非0

.参数 Url 地址, 文本型, , lpszUrl

.参数 标志, 整数型, , dwFlags 一般为 1

.参数 返回值, 整数型, , dwReserved 一般为 0

.DLL 命令 改变系统编辑框_, , "shlwapi.dll", "SHAutoComplete", 公开, SHAutoComplete
较大缓存时有效

.参数 编辑框句柄, 整数型, , hwndEdit

.参数 标志, 整数型, , dwFlags 一般为 2

.DLL 命令 SHAutoComplete, , "shlwapi.dll", "SHAutoComplete", 公开, SHAutoComplete
较大缓存时有效

.参数 编辑框句柄, 整数型, , hwndEdit

.参数 标志, 整数型, , dwFlags 一般为 2

.DLL 命令 InternetCheckConnectionA, 整数型, "wininet.dll", "InternetCheckConnectionA", 公
开, InternetCheckConnectionA' 连接网站, 成功返回非0

.参数 Url 地址, 文本型, , lpszUrl

.参数 标志, 整数型, , dwFlags 一般为 1

.参数 返回值, 整数型, , dwReserved 一般为 0

.DLL 命令 乘除计算_, 整数型, "kernel32.dll", "MulDiv", 公开, MulDiv 的作用是计算两数
的积再被一个数除 ' 执行成功返回结果, 失败返回-1。 ' 例如: MulDiv(100,2,50),返回 4。
※如出现不整除现象, 将四舍五入。

.参数 乘数 1, 整数型, , nNumber

.参数 乘数 2, 整数型, , nNumerator

.参数 除数, 整数型, , nDenominator

.DLL 命令 十六进制转十进制增强_, 整数型, "shlwapi.dll", "StrToInt64ExA", 公开,
StrToInt64ExA 成功返回 1(非 0) 失败返回 0 可以判断是不是输入的不是 0 到 9 A 到 F

.参数 转换文本, 文本型, , pszString 一般为 如果标志为0 的话 则直接填写 转换文本
如果标志为1 的话 转换文本="0x"+转换文本

.参数 标志, 整数型, , dwFlags 标志为 0 则为 10 进制 标志为 1 则为十六进制

.参数 返回值, 整数型, 传址, piRet

.DLL 命令 延迟时间扩展_, 整数型, "kernel32.dll", "SleepEx", 公开, SleepEx

.参数 延迟毫秒数, 整数型, , dwMilliseconds

.参数 特殊参数, 整数型, , bAlertable 一般可以填写 0

.DLL 命令 创建文件, 整数型, "Kernel32.dll", "_lcreat", 公开, _lcreat 失败返回-1 创建一个文件。如文件已经存在, 就会将其缩短成零长度, 并将其打开, 以便读写

.参数 文件路径, 文本型, , lpPathName, 路径加文件名

.参数 文件类型, 整数型, , iAttribute, 0,文件能够读写 1,创建只读文件 2,创建隐藏文件 3,创建系统文件

.DLL 命令 _lcreat, 整数型, "Kernel32.dll", "_lcreat", 公开, _lcreat 失败返回-1 创建一个文件。如文件已经存在, 就会将其缩短成零长度, 并将其打开, 以便读写

.参数 文件路径, 文本型, , lpPathName, 路径加文件名

.参数 文件类型, 整数型, , iAttribute, 0,文件能够读写 1,创建只读文件 2,创建隐藏文件 3,创建系统文件

.DLL 命令 SleepEx_, 整数型, "kernel32.dll", "SleepEx", 公开, SleepEx

.参数 延迟毫秒数, 整数型, , dwMilliseconds

.参数 特殊参数, 整数型, , bAlertable 一般可以填写 0

.DLL 命令 更改文件名的扩展名_, 整数型, "shlwapi.dll", "PathRenameExtensionA", 公开, PathRenameExtensionA 返回 0 则失败

.参数 文件名, 文本型, , pszPath 第一个参数, 提供全路径文件名变量。比如:c:\windows\system32\user32.txt

.参数 扩展名, 文本型, , pszExt 第二个参数, 提供更改后的扩展名。比如: .dll

.DLL 命令 删除路径文件名_, 整数型, "shlwapi.dll", "PathRemoveFileSpecA", 公开, PathRemoveFileSpecA 如: "c:\windows\boot.ini"返回"c:\windows"

.参数 路径文本, 文本型, , pszPath

.DLL 命令 PathRemoveFileSpecA, 整数型, "shlwapi.dll", "PathRemoveFileSpecA", 公开, PathRemoveFileSpecA 如: "c:\windows\boot.ini"返回"c:\windows"

.参数 路径文本, 文本型, , pszPath

.DLL 命令 PathRenameExtensionA, 整数型, "shlwapi.dll", "PathRenameExtensionA", 公开, PathRenameExtensionA 返回 0 则失败

.参数 文件名, 文本型, , pszPath 第一个参数, 提供全路径文件名变量。比如:c:\windows\system32\user32.txt

.参数 扩展名, 文本型, , pszExt 第二个参数, 提供更改后的扩展名。比如: .dll

.DLL 命令 写内存字节集_, 整数型, "kernel32.dll", "WriteProcessMemory", 公开, WriteProcessMemory 写内存 失败返回 0 返回值

.参数 进程的句柄, 整数型, , hProcess, 进程的句柄 使用函数 OpenProcess() 返回。

.参数 内存地址, 整数型, , lpBaseAddress, 内存地址

.参数 写入数据, 字节集, 传址, lpBuffer, 数据指针 本参数的类型也能是“字节”型。

.参数 数据的长度, 整数型, , nSize, 长度 取字节集长度 (数据)

.参数 实际数据的长度, 整数型, , lpNumberOfBytesWritten 实际写出长度 一般填写 WriteProcessMemory 返回值 为 0 则忽略。

.DLL 命令 WriteProcessMemory_ , 整数型, "kernel32.dll", "WriteProcessMemory", 公开, 写内存 失败返回 0 返回值

.参数 进程的句柄, 整数型, hProcess , 进程的句柄

.参数 内存地址, 整数型, lpBaseAddress, 内存地址

.参数 写入数据, 字节集, lpBuffer, 数据指针

.参数 数据的长度, 整数型, nSize, 长度 取字节集长度 (数据)

.参数 实际数据的长度, 整数型, lpNumberOfBytesWritten 实际写出长度 一般填写

WriteProcessMemory 返回值

.DLL 命令 打开进程_ , 整数型, "kernel32.dll", "OpenProcess", 公开, , OpenProcess 将句柄返回给过程对象 如执行成功, 返回进程句柄; 零表示失败

.参数 进程对象, 整数型, dwDesiredAccess PROCESS_ALL_ACCESS: 2035711 最大权限 1024 16 当设置为 1024 16 不能对内存写入

.参数 继承句柄, 整数型, bInheritHandle 一般可以填写 0

.参数 进程标识符, 整数型, dwProcessId

.DLL 命令 OpenProcess, 整数型, "kernel32.dll", "OpenProcess", 公开, , 将句柄返回给过程对象 如执行成功, 返回进程句柄; 零表示失败

.参数 进程对象, 整数型, dwDesiredAccess PROCESS_ALL_ACCESS: 2035711 最大权限 1024 16 当设置为 1024 16 不能对内存写入

.参数 继承句柄, 整数型, bInheritHandle 一般可以填写 0

.参数 进程标识符, 整数型, dwProcessId

.DLL 命令 取路径文件名_ , 整数型, "shlwapi.dll", "PathFindFileNameA", 公开, PathFindFileNameA 成功返回路径文本指针地址 比如 c:\1.txt 返回的是 1.txt

.参数 路径文本, 文本型, pszPath

.DLL 命令 删除路径根目录_ , 整数型, "shlwapi.dll", "PathSkipRootA", 公开, PathSkipRootA 成功返回路径文本指针地址 PathSkipRootA 的作用是删除路径文本串的驱动器部分。

如: "c:\windows\system32\user32.dll"返回"windows\system32\user32.dll"

.参数 路径文本, 文本型, pszPath

.DLL 命令 PathSkipRootA, 整数型, "shlwapi.dll", "PathSkipRootA", 公开, PathSkipRootA 成功返回路径文本指针地址 PathSkipRootA 的作用是删除路径文本串的驱动器部分。

如: "c:\windows\system32\user32.dll"返回"windows\system32\user32.dll" 与删除路径根目录(PathSkipRootA) 取根路径后文本(PathFindNextComponentA) 功能基本相似

.参数 路径文本, 文本型, pszPath

.DLL 命令 取根路径后文本_ , 整数型, "shlwapi.dll", "PathFindNextComponentA", 公开, PathFindNextComponent 成功返回路径文本指针地址 PathFindNextComponentA 的作用是取删除路径文本串的驱动器部分。

如: "c:\windows\system32\user32.dll"返回"windows\system32\user32.dll" 与删除路径根目录(PathSkipRootA) 取根路径后文本(PathFindNextComponentA) 功能基本相似

.参数 路径文本, 文本型, pszPath

.DLL 命令 PathFindNextComponentA, 整数型, "shlwapi.dll", "PathFindNextComponentA", 公开, PathFindNextComponent 成功返回路径文本指针地址 PathFindNextComponentA 的作用是取删除路径文本串的驱动器部分。

如: "c:\windows\system32\user32.dll"返回"windows\system32\user32.dll" 与删除路径根目录(PathSkipRootA) 取根路径后文本(PathFindNextComponentA) 功能基本相似

.参数 路径文本, 文本型, pszPath

- .DLL 命令 PathFindFileNameA, 整数型, "shlwapi.dll", "PathFindFileNameA", 公开, PathFindFileNameA 成功返回路径文本指针地址 比如 c:\1.txt 返回的是 1.txt
 - .参数 路径文本, 文本型, , pszPath
- .DLL 命令 内存清零_, "kernel32.dll", "RtlZeroMemory", 公开, \$(b)ZeroMemory
 - .参数 目的内存, 文本型, , Destination
 - .参数 长度, 整数型, , Length
- .DLL 命令 RtlZeroMemory, "kernel32.dll", "RtlZeroMemory", 公开, \$(b)ZeroMemory
 - .参数 目的内存, 文本型, , Destination
 - .参数 长度, 整数型, , Length
- .DLL 命令 取空格前文本_, "shlwapi.dll", "PathRemoveArgsA", 公开, PathRemoveArgsA
 - .参数 路径文本, 文本型, , pszPath
- .DLL 命令 PathRemoveArgsA, "shlwapi.dll", "PathRemoveArgsA", 公开
 - .参数 路径文本, 文本型, , pszPath
- .DLL 命令 删文本首尾空_, "shlwapi.dll", "PathRemoveBlanksA", 公开, PathRemoveBlanksA 删除文本首部尾部的空格
 - .参数 文本, 文本型, , pszPath
- .DLL 命令 PathRemoveBlanksA, "shlwapi.dll", "PathRemoveBlanksA", 公开, 删除文本首部尾部的空格
 - .参数 文本, 文本型, , pszPath
- .DLL 命令 俘获鼠标_, 整数型, "user32", "SetCapture", 公开, SetCapture
 - .参数 窗口句柄, 整数型, 传址
- .DLL 命令 关闭句柄, 整数型, "CloseHandle", 公开, 关闭进程句柄
 - .参数 进程句柄, 整数型
- .DLL 命令 CloseHandle, 整数型, "kernel32", "CloseHandle", 公开, CloseHandle 非零表示成功, 零表示失败。会设置 GetLastError关闭一个内核对象。其中包括文件、文件映射、进程、线程、安全和同步对象等。涉及文件处理时, 这个函数通常与 vb 的close命令相似。应尽可能的使用 close, 因为它支持vb 的差错控制。注意这个函数使用的文件句柄与 vb 的文件编号是完全不同的 非零表示成功, 零表示失败。会设置 GetLastError
 - .参数 对象句柄, 整数型, , 欲关闭的一个对象的句柄;
- .DLL 命令 CreateProcessA, 逻辑型, "CreateProcessA", 公开
 - .参数 lpApplicationName, 文本型, , 执行程序文件名
 - .参数 lpCommandLine, 整数型, , 参数行
 - .参数 lpProcessAttributes, 整数型, , 进程安全参数
 - .参数 lpThreadAttributes, 整数型, , 线程安全参数
 - .参数 bInheritHandles, 逻辑型, , 继承标记
 - .参数 dwCreationFlags, 整数型, , 创建标记
 - .参数 lpEnvironment, 整数型, , 环境变量
 - .参数 lpCurrentDirectory, 整数型, , 运行该子进程的初始目录
 - .参数 lpStartupInfo, 启动信息数据, , 创建该子进程的相关参数
 - .参数 lpProcessInformation, 创建进程返回数据类型, , 创建后用于被创建子进程的信息
- .DLL 命令 创建进程_, 整数型, "kernel32.dll", "CreateProcessA", 公开
 - .参数 要执行的程序名, 文本型, , lpApplicationName, 参见相关帮助
 - .参数 要执行的命令行, 文本型, , lpCommandLine, 参见相关帮助
 - .参数 安全特性参数 1, 整数型, , lpProcessAttributes, 参见相关帮助

- .参数 安全特性参数 2, 整数型, , lpThreadAttributes, 参见相关帮助
- .参数 允许新进程继承当前进程的句柄, 逻辑型, , bInheritHandles, 参见相关帮助
- .参数 参数 1, 整数型, , dwCreationFlags, 赋 0 即可
- .参数 环境指针, 文本型, , lpEnvironment, 赋 0 即可
- .参数 新进程当前目录路径, 整数型, , lpCurrentDirectory, 参见相关帮助
- .参数 创建进程时附加信息, 启动信息数据, , lpStartupInfo, 创建进程时附加信息
- .参数 新进程的进程和线程标识符, 创建进程返回数据类型, , lpProcessInformation, 参见相关帮助
- .DLL 命令 SetCapture, 整数型, "user32", "SetCapture", 公开, SetCapture
 - .参数 窗口句柄, 整数型, 传址
- .DLL 命令 释放鼠标_, 整数型, "user32", "ReleaseCapture", 公开, ReleaseCapture
- .DLL 命令 ReleaseCapture, 整数型, "user32", "ReleaseCapture", 公开, ReleaseCapture
- .DLL 命令 改变窗口_, 整数型, "user32.dll", "MoveWindow", 公开, MoveWindow
 - .参数 窗口句柄, 整数型, , hwnd, 欲移动窗口的句柄
 - .参数 左侧位置, 整数型, , x, 窗口新的左侧位置
 - .参数 顶部位置, 整数型, , y, 窗口新的顶部位置
 - .参数 新宽度, 整数型, , nWidth, 窗口的新宽度
 - .参数 新高度, 整数型, , nHeight, 窗口的高宽度
 - .参数 是否重画, 整数型, , bRepaint, 如窗口此时应重画, 则设为 TRUE (非零)
- .DLL 命令 MoveWindow, 整数型, "user32.dll", "MoveWindow", 公开, MoveWindow
 - .参数 窗口句柄, 整数型, , hwnd, 欲移动窗口的句柄
 - .参数 左侧位置, 整数型, , x, 窗口新的左侧位置
 - .参数 顶部位置, 整数型, , y, 窗口新的顶部位置
 - .参数 新宽度, 整数型, , nWidth, 窗口的新宽度
 - .参数 新高度, 整数型, , nHeight, 窗口的高宽度
 - .参数 是否重画, 整数型, , bRepaint, 如窗口此时应重画, 则设为 TRUE (非零)
- .DLL 命令 锁定窗口_, 整数型, "user32.dll", "LockWindowUpdate", 公开, LockWindowUpdate 锁定指定窗口, 禁止它更新。同时只能有一个窗口处于锁定状态
 - .参数 窗口句柄, 整数型, , hwndLock, 欲锁定窗口的句柄。如设为零, 则对窗口解锁
- .DLL 命令 LockWindowUpdate, 整数型, "user32.dll", "LockWindowUpdate", 公开, LockWindowUpdate 锁定指定窗口, 禁止它更新。同时只能有一个窗口处于锁定状态
 - .参数 窗口句柄, 整数型, , hwndLock, 欲锁定窗口的句柄。如设为零, 则对窗口解锁
- .DLL 命令 移动文件_, 整数型, "kernel32.dll", "MoveFileA", 公开, , MoveFileA, 移动文件
 - .参数 文件名, 文本型, , lpExistingFileName, 欲移动的文件名
 - .参数 新文件名, 文本型, , lpNewFileName, 新文件名
- .DLL 命令 取函数地址_, 整数型, "kernel32", "GetProcAddress", 公开, GetProcAddress 返回函数地址
 - .参数 模块句柄, 整数型
 - .参数 函数名, 文本型, , MessageBoxA
- .DLL 命令 GetProcAddress, 整数型, "kernel32", "GetProcAddress", 公开, GetProcAddress 返回函数地址
 - .参数 模块句柄, 整数型
 - .参数 函数名, 文本型, , MessageBoxA
- .DLL 命令 GetModuleHandleA, 整数型, "kernel32", "GetModuleHandleA", 公开,

GetModuleHandleA 获取一个应用程序或动态链接库的模块句柄。如执行成功成功，则返回模块句柄。零表示失败。会设置 GetLastError

.参数 模块名, 文本型, , 指定模块名, 这通常是与模块的文件名相同的一个名字。例如, NOTEPAD.EXE 程序的模块文件名就叫作 NOTEPAD; User32.dll

.DLL 命令 GetModuleHandle, 整数型, "kernel32", "GetModuleHandleA", 公开, GetModuleHandleA 获取一个应用程序或动态链接库的模块句柄。如执行成功成功，则返回模块句柄。零表示失败。会设置 GetLastError

.参数 模块名, 文本型, , 指定模块名, 这通常是与模块的文件名相同的一个名字。例如, NOTEPAD.EXE 程序的模块文件名就叫作 NOTEPAD; User32.dll

.DLL 命令 取模块句柄_, 整数型, "kernel32", "GetModuleHandleA", 公开, GetModuleHandleA 获取一个应用程序或动态链接库的模块句柄。如执行成功成功，则返回模块句柄。零表示失败。会设置 GetLastError

.参数 模块名, 文本型, , 指定模块名, 这通常是与模块的文件名相同的一个名字。例如, NOTEPAD.EXE 程序的模块文件名就叫作 NOTEPAD; User32.dll

.DLL 命令 MoveFileA, 整数型, "kernel32.dll", "MoveFileA", 公开, , MoveFileA, 移动文件

.参数 文件名, 文本型, , lpExistingFileName, 欲移动的文件名

.参数 新文件名, 文本型, , lpNewFileName, 新文件名

.DLL 命令 数值到格式化字节扩展_, 整数型, "shlwapi.dll", "StrFormatByteSize64A", 公开, StrFormatByteSize64A StrFormatByteSize64A 比如:1024 返回 1.0KB

.参数 数值, 长整数型, , qdw

.参数 文本缓冲区, 文本型, 传址, szBuf

.参数 文本缓冲区大小, 整数型, 传址, uiBufSize

.DLL 命令 文本连接_, 整数型, "shlwapi.dll", "StrCatBuffA", 公开, StrCatBuffA

.参数 目标文本, 文本型, , pszDest

.参数 源文本, 文本型, , pszSrc

.参数 目标缓冲区大小, 整数型, , cchDestBuffSize

.DLL 命令 StrCatBuffA, 整数型, "shlwapi.dll", "StrCatBuffA", 公开, StrCatBuffA

.参数 目标文本, 文本型, , pszDest

.参数 源文本, 文本型, , pszSrc

.参数 目标缓冲区大小, 整数型, , cchDestBuffSize

.DLL 命令 StrFormatByteSize64A, 整数型, "shlwapi.dll", "StrFormatByteSize64A", 公开, StrFormatByteSize64A

.参数 数值, 长整数型, , qdw

.参数 文本缓冲区, 文本型, 传址, szBuf

.参数 文本缓冲区大小, 整数型, 传址, uiBufSize

.DLL 命令 BlockInput, 整数型, "BlockInput", 公开, BlockInput

.参数 是否锁定, 整数型, , fBlock 1. 锁定, 0. 解除

.DLL 命令 数值到格式化字节_, 整数型, "shlwapi.dll", "StrFormatByteSizeA", 公开, StrFormatByteSizeA 比如:1024 返回 1.0KB

.参数 数值, 整数型, , dw

.参数 文本缓冲区, 文本型, , szBuf

.参数 文本缓冲区大小, 整数型, 传址, uiBufSize

.DLL 命令 StrFormatByteSizeA, 整数型, "shlwapi.dll", "StrFormatByteSizeA", 公开, StrFormatByteSizeA 比如:1024 返回 1.0KB

.参数 数值, 整数型, , dw

.参数 文本缓冲区, 文本型, , szBuf

.参数 文本缓冲区大小, 整数型, 传址, uiBufSize

.DLL 命令 键盘和鼠标锁定_, 整数型, "BlockInput", 公开, BlockInput

.参数 是否锁定, 整数型, , fBlock 1.锁定, 0.解除

.DLL 命令 CreateToolhelp32Snapshot, 整数型, "kernel32.dll", "CreateToolhelp32Snapshot", 公开, 取进程信息 CreateToolhelp32Snapshot 返回快照句柄

.参数 标志, 整数型, , IFlags 一般填写 8即可

.参数 进程标识符, 整数型, , IProcessID

.DLL 命令 寻找第一个32 位模块_, 整数型, "kernel32.dll", "Module32First", 公开, Module32First 返回模块数目

.参数 快照句柄, 整数型, , hSnapshot

.参数 模块进程结构, 模块地址_, 传址, uProcess

.DLL 命令 取创建 32位帮助工具快照_, 整数型, "kernel32.dll", "CreateToolhelp32Snapshot", 公开, 取进程信息 CreateToolhelp32Snapshot 返回快照句柄

.参数 标志, 整数型, , IFlags 一般填写 8即可

.参数 进程标识符, 整数型, , IProcessID

.DLL 命令 寻找下一个32 位模块_, 整数型, "kernel32.dll", "Module32Next", 公开, Module32Next 返回模块数目

.参数 快照句柄, 整数型, , hSnapshot

.参数 模块进程结构, 模块地址_, 传址, uProcess

.DLL 命令 Module32Next, 整数型, "kernel32.dll", "Module32Next", 公开, Module32Next 返回模块数目

.参数 快照句柄, 整数型, , hSnapshot

.参数 模块进程结构, 模块地址_, 传址, uProcess

.DLL 命令 Module32First, 整数型, "kernel32.dll", "Module32First", 公开, Module32First 返回模块数目

.参数 快照句柄, 整数型, , hSnapshot

.参数 模块进程结构, 模块地址_, 传址, uProcess

.DLL 命令 取当前进程标识符_, 整数型, "kernel32.dll", "GetCurrentProcessId", 公开, 获取当前进程一个唯一的标识符 GetCurrentProcessId

.DLL 命令 GetCurrentProcessId, 整数型, "kernel32.dll", "GetCurrentProcessId", 公开, 获取当前进程一个唯一的标识符 GetCurrentProcessId

.DLL 命令 模拟鼠标_, , "user32.dll", "mouse_event", 公开, mouse_event,模拟一次鼠标事件没有返回值

.参数 标志组合, 整数型, , dwFlags, 一系列标志的组合 dwFlags, 一系列标志的组合 3
为按下鼠标 相关标志自己去网上查下

.参数 水平方向, 整数型, , dx, 参见相关帮助

.参数 垂直方向, 整数型, , dy, 参见相关帮助

.参数 数据信息, 整数型, , cButtons, 未使用

.参数 扩展数据信息, 整数型, , dwExtraInfo, 通常未用的一个值

.DLL 命令 mouse_event, , "user32.dll", "mouse_event", 公开, mouse_event,模拟一次鼠标事件没有返回值

.参数 标志组合, 整数型, , dwFlags, 一系列标志的组合 dwFlags, 一系列标志的组合 3

为按下鼠标 相关标志自己去网上查下

.参数 水平方向, 整数型, , dx, 参见相关帮助

.参数 垂直方向, 整数型, , dy, 参见相关帮助

.参数 数据信息, 整数型, , cButtons, 未使用

.参数 扩展数据信息, 整数型, , dwExtraInfo, 通常未用的一个值

.DLL 命令 HTTP 下载_, 整数型, "urlmon.dll", "URLDownloadToFileA", 公开, URLDownloadToFile 成功返回 0 失败返回-2146697211

.参数 参数 1, 整数型, , pCaller 一般为 0

.参数 HTTP 文件地址, 文本型, , szURL 本参数指定欲读取文件在互联网上的 HTTP 地址 (即 URL)。

.参数 保存到本地文件名, 文本型, , szFileName 例如:c:\1.exe 保存在:c:\1.exe

.参数 参数二, 整数型, , dwReserved 一般为 0

.参数 参数三, 整数型, , lpfnCB 一般为 0

.DLL 命令 URLDownloadToFileA, 整数型, "urlmon.dll", "URLDownloadToFileA", 公开, URLDownloadToFile 成功返回 0 失败返回-2146697211

.参数 参数 1, 整数型, , pCaller 一般为 0

.参数 HTTP 文件地址, 文本型, , szURL 本参数指定欲读取文件在互联网上的 HTTP 地址 (即 URL)。

.参数 保存到本地文件名, 文本型, , szFileName 例如:c:\1.exe 保存在:c:\1.exe

.参数 参数二, 整数型, , dwReserved 一般为 0

.参数 参数三, 整数型, , lpfnCB 一般为 0

.DLL 命令 模拟键盘行动_, 整数型, "user32.dll", "keybd_event", 公开, keybd_event 这个函数模拟了键盘行动 注意键按下后要放开键

.DLL 命令 模拟键盘行动_1, 整数型, "user32.dll", "keybd_event", 公开, keybd_event 这个函数模拟了键盘行动 注意键按下后要放开键

.参数 键代码, 整数型, , bVk, 欲模拟的虚拟键码 bVk, 欲模拟的虚拟键码

.参数 键扫描码, 整数型, , bScan, 键的 OEM 扫描码

.参数 标志值, 整数型, , dwFlags, 标志常数 KEYEVENTF_KEYUP =2 放开某键的标志

.参数 特定值, 整数型, , dwExtraInfo, 通常不用的一个值

.DLL 命令 keybd_event, 整数型, "user32.dll", "keybd_event", 公开, keybd_event 这个函数模拟了键盘行动 注意键按下后要放开键

.参数 键代码, 整数型, , bVk, 欲模拟的虚拟键码 bVk, 欲模拟的虚拟键码

.参数 键扫描码, 整数型, , bScan, 键的 OEM 扫描码

.参数 标志值, 整数型, , dwFlags, 标志常数 KEYEVENTF_KEYUP =2 放开某键的标志

.参数 特定值, 整数型, , dwExtraInfo, 通常不用的一个值

.DLL 命令 取进程线程标识符_, 整数型, "user32.dll", "GetWindowThreadProcessId", 公开, Long, 拥有窗口的线程的标识符 GetWindowThreadProcessId, 获取与指定窗口关联在一起的一个线程和进程标识符

.参数 窗口句柄, 整数型, , hwnd, 指定窗口句柄

.参数 进程标识符, 整数型, 传址, lpdwProcessId, 指定一个变量, 用于装载拥有那个窗口的一个进程的标识符

.DLL 命令 GetWindowThreadProcessId, 整数型, "user32.dll", "GetWindowThreadProcessId",

公开, Long, 拥有窗口的线程的标识符 GetWindowThreadProcessId, 获取与指定窗口关联在一起的一个线程和进程标识符

.参数 窗口句柄, 整数型, hwnd, 指定窗口句柄

.参数 进程标识符, 整数型, 传址, lpdwProcessId, 指定一个变量, 用于装载拥有那个窗口的一个进程的标识符

.DLL 命令 取活动窗口句柄_, 整数型, "user32", "GetActiveWindow", 公开, 获得活动窗口的句柄 GetActiveWindow

.DLL 命令 GetActiveWindow, 整数型, "user32", "GetActiveWindow", 公开, 获得活动窗口的句柄 GetActiveWindow

.DLL 命令 填充内存_, "kernel32.dll", "RtlFillMemory", 公开, \$(b) RtlFillMemory

.参数 目的内存, 文本型, Destination

.参数 长度, 整数型, Length

.参数 填充内容, 字节型, Fill 例子:取代码("A",)

.DLL 命令 RtlFillMemory, "kernel32.dll", "RtlFillMemory", 公开, \$(b) RtlFillMemory

.参数 目的内存, 文本型, Destination

.参数 长度, 整数型, Length

.参数 填充内容, 字节型, Fill 例子:取代码("A",)

.DLL 命令 EnumChildWindows, 整数型, "user32", "EnumChildWindows", 公开

.参数 父窗口句柄, 整数型

.参数 子窗口调用指针, 子程序指针

.参数 枚举期间传递的操作值, 整数型, 传址

.DLL 命令 设置指定窗口的显示状态_, 整数型, "user32.dll", "ShowWindow", 公开, ShowWindow返回值: 如果窗口以前可见, 则返回值为非零。如果窗口以前被隐藏, 则返回值为零。 控制窗口的状态(在 vb 里使用: 针对 vb 窗体及控件, 请使用对应的 vb 属性) 成功:返回 TRUE (非零), 失败:返回 FALSE (零)

.参数 窗口句柄, 整数型, hWnd: 窗口句柄。

.参数 窗口如何显示, 整数型, nCmdShow: 指定窗口如何显示 为窗口指定的一个命令。请用下述任何一个常数:0 隐藏取消激活 1 最小化取消激活 2 还原激活 3 显示激活 4 最大化激活 5 最小化激活 6 最小化 7 显示 8 最近状态显示 9 还原 更多查看常量:ShowWindow

.DLL 命令 替换首尾文本_, 整数型, "shlwapi.dll", "StrTrimA", 公开, StrTrimA 替换首尾文本 例子:文本:12341 替换首尾文本:1 结果: 文本:=234

.参数 文本, 文本型, psz 例子:文本:12341 替换首尾文本:1 结果: 文本:=234

.参数 替换首尾文本, 文本型, pszTrimChars 例子:文本:12341 替换首尾文本:1 结果: 文本:=234

.DLL 命令 StrTrimA, 整数型, "shlwapi.dll", "StrTrimA", 公开, StrTrimA 替换首尾文本 例子: 文本:12341 替换首尾文本:1 结果: 文本:=234

.参数 文本, 文本型, psz 例子:文本:12341 替换首尾文本:1 结果: 文本:=234

.参数 替换首尾文本, 文本型, pszTrimChars 例子:文本:12341 替换首尾文本:1 结果: 文本:=234

.DLL 命令 字节数据到文本_, 文本型, "oleaut32.dll", "SysAllocStringByteLen", 公开, SysAllocStringByteLen

.参数 二进制数据, 字节型, 数组, m_pBase

.参数 二进制长度, 整数型, FunctionCall

- .DLL 命令 拷贝内存_ , 整数型, "kernel32", "RtlMoveMemory", 公开, RtlMoveMemory
 - .参数 目标缓冲区, 文本型, , Destination
 - .参数 源缓冲区, 字节型, 传址 数组, Source
 - .参数 复制长度, 整数型, , Length
- .DLL 命令 SysAllocStringByteLen, 文本型, "oleaut32.dll", "SysAllocStringByteLen", 公开, SysAllocStringByteLen
 - .参数 二进制数据, 字节型, 数组, m_pBase
 - .参数 二进制长度, 整数型, , FunctionCall
- .DLL 命令 RtlMoveMemory, 整数型, "kernel32", "RtlMoveMemory", 公开, RtlMoveMemory
 - .参数 目标缓冲区, 文本型, , Destination
 - .参数 源缓冲区, 字节型, 传址 数组, Source
 - .参数 复制长度, 整数型, , Length
- .DLL 命令 读内存数据_ , 整数型, "kernel32.dll", "ReadProcessMemory", 公开, ReadProcessMemory
 - .参数 进程的句柄, 整数型, , hProcess , 进程的句柄
 - .参数 内存地址, 整数型, , lpBaseAddress, 内存地址
 - .参数 读入数据, 字节集, 传址, lpBuffer, 数据指针 可以为整数型
 - .参数 数据的长度, 整数型, , nSize, 长度 取字节集长度 (数据)
 - .参数 实际数据的长度, 整数型, , lpNumberOfBytesWritten 实际写出长度 一般填写
 ReadProcessMemory 返回值
- .DLL 命令 ReadProcessMemory, 整数型, "kernel32.dll", "ReadProcessMemory", 公开, ReadProcessMemory
 - .参数 进程的句柄, 整数型, , hProcess , 进程的句柄
 - .参数 内存地址, 整数型, , lpBaseAddress, 内存地址
 - .参数 读入数据, 字节集, 传址, lpBuffer, 数据指针 可以为整数型
 - .参数 数据的长度, 整数型, , nSize, 长度 取字节集长度 (数据)
 - .参数 实际数据的长度, 整数型, , lpNumberOfBytesWritten 实际写出长度 一般填写
 ReadProcessMemory 返回值
- .DLL 命令 取模板完整路径_ , 整数型, "kernel32.dll", "GetModuleFileNameA", 公开, GetModuleFileNameA

\$(b) 获取一个已装载模板的完整路径名称 如执行成功, 返回复制到lpFileName的实际字符数量; 零表示失败。会设置 GetLastError

 - .参数 模块句柄, 整数型, , 一个模块的句柄。可以是一个DLL 模块, 或者是一个应用程序的实例句柄一般填写 0
 - .参数 文件名缓冲区, 文本型, , 指定一个字串缓冲区, 要在其中容纳文件的用NULL 字符中止的路径名, hModule 模块就是从这个文件装载进来的
 - .参数 缓冲区大小, 整数型, , 装载到缓冲区 lpFileName 的最大字符数量;
- .DLL 命令 GetModuleFileNameA, 整数型, "kernel32.dll", "GetModuleFileNameA", 公开, GetModuleFileNameA

\$(b) 获取一个已装载模板的完整路径名称 如执行成功, 返回复制到lpFileName的实际字符数量; 零表示失败。会设置 GetLastError

 - .参数 模块句柄, 整数型, , 一个模块的句柄。可以是一个DLL 模块, 或者是一个应用程序的实例句柄一般填写 0
 - .参数 文件名缓冲区, 文本型, , 指定一个字串缓冲区, 要在其中容纳文件的用NULL 字符中止的路径名, hModule 模块就是从这个文件装载进来的
 - .参数 缓冲区大小, 整数型, , 装载到缓冲区 lpFileName 的最大字符数量;

.DLL 命令 格式化磁盘_ , 整数型, "shell32.dll", "SHFormatDrive", 公开, SHFormatDrive
.参数 窗口句柄, 整数型, , hwndOwner
.参数 盘符, 整数型, , iDrive 驱动器‘从 0 开始’0 是 A, 1 是 B 2 是 C 3 是 D 4 是 E 以此类推
.参数 容量, 整数型, , iCapacity 0
.参数 功能, 整数型, , iFormatType 弹出界面的功能‘参数为 0-3’XP 下“0、1”

.DLL 命令 SHFormatDrive, 整数型, "shell32.dll", "SHFormatDrive", 公开, SHFormatDrive
.参数 窗口句柄, 整数型, , hwndOwner
.参数 盘符, 整数型, , iDrive 驱动器‘从 0 开始’0 是 A, 1 是 B 2 是 C 3 是 D 4 是 E 以此类推
.参数 容量, 整数型, , iCapacity 0
.参数 功能, 整数型, , iFormatType 弹出界面的功能‘参数为 0-3’XP 下“0、1”

.DLL 命令 取磁盘信息_ , 整数型, "kernel32", "GetVolumeInformationA", 公开, GetVolumeInformationA, Long, 非零表示成功, 零表示失败。
.参数 卷根路径, 文本型, , lpRootPathName String, 欲获取信息的那个卷的根路径 c:\
.参数 卷名, 文本型, 传址, lpVolumeNameBuffer String, 用于装载卷名(卷标)的一个字符串
.参数 卷名字符长度, 整数型, 传址, nVolumeNameSize Long, lpVolumeNameBuffer 字符串的长度
.参数 磁盘卷序列号, 整数型, 传址, lpVolumeSerialNumber Long, 用于装载磁盘卷序列号的变量
.参数 文件夹名长度, 整数型, 传址, lpMaximumComponentLength Long, 指定一个变量, 用于装载文件名每一部分的长度。例如, 在“c:\component1\component2.ext”的情况下, 它就代表 component1 或 component2 名称的长度
.参数 标志, 整数型, , 具体值自己去查找
.参数 文件系统名称, 文本型, 传址, lpFileSystemNameBuffer String, 指定一个缓冲区, 用于装载文件系统的名称(如 FAT, NTFS 以及其他)
.参数 名称长度, 整数型, , nFileSystemNameSize Long, lpFileSystemNameBuffer 字符串的长度

.DLL 命令 GetVolumeInformationA, 整数型, "kernel32", "GetVolumeInformationA", 公开, GetVolumeInformationA, Long, 非零表示成功, 零表示失败。
.参数 卷根路径, 文本型, , lpRootPathName String, 欲获取信息的那个卷的根路径 c:\
.参数 卷名, 文本型, 传址, lpVolumeNameBuffer String, 用于装载卷名(卷标)的一个字符串
.参数 卷名字符长度, 整数型, 传址, nVolumeNameSize Long, lpVolumeNameBuffer 字符串的长度
.参数 磁盘卷序列号, 整数型, 传址, lpVolumeSerialNumber Long, 用于装载磁盘卷序列号的变量
.参数 文件夹名长度, 整数型, 传址, lpMaximumComponentLength Long, 指定一个变量, 用于装载文件名每一部分的长度。例如, 在“c:\component1\component2.ext”的情况下, 它就代表 component1 或 component2 名称的长度
.参数 标志, 整数型, , 具体值自己去查找
.参数 文件系统名称, 文本型, 传址, lpFileSystemNameBuffer String, 指定一个缓冲区, 用于装载文件系统的名称(如 FAT, NTFS 以及其他)

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/425042204122011240>