 This is part of **Family API** which allow to create dual-os version of program runs under OS/2 and DOS

Note: This is legacy API call. It is recommended to use 32-bit equivalent

2021/09/17 04:47 · prokushev · 0 Comments
2021/08/20 03:18 · prokushev · 0 Comments

This call loads a dynamic link module and returns a handle for the module.

Syntax

DosLoadModule (ObjNameBuf, ObjNameBufL, ModuleName, ModuleHandle)

Parameters

;ObjNameBuf (PSZ) - output: Address of the name of an object that contributed to the failure of DosLoadModule. ;ObjNameBufL (USHORT) - input: Length, in bytes, of the buffer described by ObjNameBuf. ;ModuleName (PSZ) - input: Address of the ASCIIZ name string containing the dynamic link module name or the pathname string. The filename extension used for dynamic link libraries is .DLL. When a pathname is provided, the module name may have any extension. The internal module name (the name given in the LIBRARY statement in the .DEF file) must be the same as the filename without the extension. ;ModuleHandle (PHMODULE) - output: Address of the handle for the dynamic link module.

Return Code

;rc (USHORT) - return:Return code descriptions are: * 0 NO_ERROR * 2 ERROR_FILE_NOT_FOUND * 3 ERROR_PATH_NOT_FOUND * 4 ERROR_TOO_MANY_OPEN_FILES * 5 ERROR_ACCESS_DENIED * 8 ERROR_NOT_ENOUGH_MEMORY * 11 ERROR_BAD_FORMAT * 26 ERROR_NOT_DOS_DISK * 32 ERROR_SHARING_VIOLATION * 33 ERROR_LOCK_VIOLATION * 36 ERROR_SHARING_BUFFER_EXCEEDED * 95 ERROR_INTERRUPT * 108 ERROR_DRIVE_LOCKED * 123 ERROR_INVALID_NAME * 127 ERROR_PROC_NOT_FOUND * 180 ERROR_INVALID_SEGMENT_NUMBER * 182 ERROR_INVALID_ORDINAL * 190 ERROR_INVALID_MODULETYPE * 191 ERROR_INVALID_EXE_SIGNATURE * 192 ERROR_EXE_MARKED_INVALID * 194 ERROR_ITERATED_DATA_EXCEEDS_64k * 195 ERROR_INVALID_MINALLOCSIZE * 196 ERROR_DYNLINK_FROM_INVALID_RING * 198 ERROR_INVALID_SEGDPL * 199 ERROR_AUTODATASEG_EXCEEDS_64k * 201 ERROR_RELOC_CHAIN_XEEDS_SEGLIM * 206 ERROR_FILENAME_EXCED_RANGE

Remarks

If the file is an OS/2 dynamic link module, it is loaded, and a handle is returned. This handle is used for freeing the dynamic link module with a DosFreeModule request, getting procedure addresses with DosGetProcAddr requests, and getting the fully qualified file name with a DosGetModName request.

DosLoadModule may not be called from a dynamic library initialization routine if the module to be loaded or any module referenced by it contains a dynamic link library initialization routing.

Bindings

C

<PRE> #define INCL_DOSMODULEMGR

USHORT rc = DosLoadModule(ObjNameBuf, ObjNameBufL, ModuleName, ModuleHandle);

PSZ ObjNameBuf; /* Address of object name buffer */ USHORT ObjNameBufL; /* Length of object name buffer */ PSZ ModuleName; /* Address of module name string */ PHMODULE ModuleHandle; /* Address of module handle (returned) */

USHORT rc; /* return code */ </PRE>

MASM

<PRE> EXTRN DosLoadModule:FAR INCL_DOSMODULEMGR EQU 1

PUSH@ OTHER ObjNameBuf ;Object name buffer (returned) PUSH WORD ObjNameBufL ;Length of object name buffer PUSH@ ASCIIZ ModuleName ;Module name string PUSH@ WORD ModuleHandle ;Module handle (returned) CALL DosLoadModule

Returns WORD </PRE>

Example

This example loads a module. <PRE> #define INCL_DOSMODULEMGR

#define MODULE_NAME “abcd” #define FULL_MODULE_NAME “\\nifty\\abcd.dll”

CHAR LoadError[100]; HMODULE ModuleHandle; USHORT rc;

```
if (DosLoadModule(LoadError,            /* Object name buffer */
                  sizeof(LoadError),    /* Length of object name buffer */
                  MODULE_NAME,          /* Module name string */
                  &ModuleHandle) == 2)  /* Module handle */
```

</PRE>

Family API		
DOS	Process Manager	DosBeep DosExit DosSleep DosExecPgm
	File Manager	DosChDir DosChgFilePtr DosClose DosDelete DosDupHandle DosMkDir DosMove DosQCurDir DosQCurDisk DosSetFileMode DosOpen DosQFileInfo DosRead DosQFileMode DosQFSInfo DosQVerify DosRmDir DosSelectDisk DosFindClose DosFindFirst DosFindNext DosSetFileInfo DosSetVerify DosWrite DosFileLocks DosSetFHandState DosNewSize DosBufReset DosQFHandState DosSetFSinfo DosShutdown
	Memory Manager	DosFreeSeg DosSubAlloc DosSubFree DosSubSet DosAllocHuge DosAllocSeg DosReallocHuge DosReallocSeg DosGetHugeShift DosCreateCSAlias
	NLS	DosCaseMap DosGetCtryInfo DosGetDBCSEv DosSetCtryCode DosGetCollate DosGetMessage DosInsMessage DosPutMessage
	Date and Time	DosSetDateTime DosGetDateTime
	Devices	DosDevConfig DosDevIOCtl DosDevIOCtl2
	Signals	DosHoldSignal DosSetSigHandler
	Misc	BadDynLink DosGetEnv DosGetMachineMode DosGetVersion DosError DosErrClass DosSetVec
KBD		KbdCharIn KbdFlushBuffer KbdGetStatus KbdSetStatus KbdStringIn KbdPeek
VIO		VioGetBuf VioGetConfig VioGetCurPos VioGetCurType VioGetPhysBuf VioReadCellStr VioReadCharStr VioScrollUp VioScrollDn VioScrollLf VioScrollRt VioScrUnLock VioSetCurPos VioSetCurType VioSetMode VioGetMode VioShowBuf VioWrtCellStr VioWrtCharStr VioWrtCharStrAtt VioWrtNAttr VioWrtNCell VioWrtNChar VioWrtTTY VioScrLock VioPopUp
Tools		BIND
Modules		DOSCALLS.DLL VIOCALLS.DLL KBDCALLS.DLL MSG.DLL
Libraries		API.LIB OS2386.LIB FAPI.LIB DOSCALLS.LIB SUBCALLS.LIB

2018/08/25 15:05 · prokushev · 0 Comments