



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](#)

# DosClose

This call closes a handle to a file, pipe, or device.

## Syntax

```
DosClose (FileHandle)
```

## Parameters

- FileHandle ([HFILE](#)) - input : Handle returned by a previous [DosOpen](#), [DosMakeNmPipe](#), or [DosMakePipe](#) call.

## Return Code

rc ([USHORT](#)) - return

Return code descriptions are:

- 0 NO\_ERROR
- 2 ERROR\_FILE\_NOT\_FOUND
- 5 ERROR\_ACCESS\_DENIED
- 6 ERROR\_INVALID\_HANDLE

## Remarks

Issuing DosClose with the handle to a file closes a handle to a file, pipe, or device.

If one or more additional handles to a file have been created with [DosDupHandle](#), the directory is not updated and all internal buffers are not written to the medium until DosClose has been issued for the duplicated handles.

Closing a handle to a device causes the device to be notified of the close, if appropriate.

DosClose closes a named pipe by handle. When all handles referencing one end of a pipe are closed, the pipe is considered broken.

If the server end closes when the pipe is already broken, it is deallocated immediately; otherwise, the pipe is not deallocated until the last client handle is closed.

## Example Code

## C Binding

```
#define INCL_DOSFILEMGR

USHORT rc = DosClose(FileHandle);

HFILE FileHandle; /* File handle */
USHORT rc; /* return code */
```

This example opens a file, then closes it.

```
#define INCL_DOSFILEMGR

#define OPEN_FILE 0x01
#define CREATE_FILE 0x10
#define FILE_ARCHIVE 0x20
#define FILE_EXISTS OPEN_FILE
#define FILE_NOEXISTS CREATE_FILE
#define DASD_FLAG 0
#define INHERIT 0x80
#define WRITE_THRU 0
#define FAIL_FLAG 0
#define SHARE_FLAG 0x10
#define ACCESS_FLAG 0x02


#define FILE_NAME "test.dat"
#define FILE_SIZE 800L
#define FILE_ATTRIBUTE FILE_ARCHIVE
#define RESERVED 0L


HFILE      FileHandle;
USHORT     Wrote;
USHORT     Action;
PSZ        FileData[100];
USHORT     rc;
```

```
Action = 2;
strcpy(FileData, "Data...");
if(!DosOpen(FILE_NAME, /* File path name */
              &FileHandle, /* File handle */
              &Action, /* Action taken */
              FILE_SIZE, /* File primary allocation */
              FILE_ATTRIBUTE, /* File attribute */
              FILE_EXISTS | FILE_NOEXISTS, /* Open function type */
              DASD_FLAG | INHERIT | /* Open mode of the file */
              WRITE_THRU | FAIL_FLAG |
              SHARE_FLAG | ACCESS_FLAG,
              RESERVED)) /* Reserved (must be zero) */
    rc = DosClose(FileHandle); /* File Handle */
```

MASM Binding

```
EXTRN  DosClose:FAR
INCL_DOSFILEMGR      EQU 1

PUSH    WORD    FileHandle    ;File handle
CALL    DosClose
```

Returns WORD

Note

Text based on [http://www.edm2.com/index.php/DosClose\\_\(FAP\)](http://www.edm2.com/index.php/DosClose_(FAP))

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
	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

Family API	
VIO	<a href="#">VioGetBuf</a> <a href="#">VioGetConfig</a> <a href="#">VioGetCurPos</a> <a href="#">VioGetCurType</a> <a href="#">VioGetPhysBuf</a> <a href="#">VioReadCellStr</a> <a href="#">VioReadCharStr</a> <a href="#">VioScrollUp</a> <a href="#">VioScrollDn</a> <a href="#">VioScrollLf</a> <a href="#">VioScrollRt</a> <a href="#">VioScrUnLock</a> <a href="#">VioSetCurPos</a> <a href="#">VioSetCurType</a> <a href="#">VioSetMode</a> <a href="#">VioGetMode</a> <a href="#">VioShowBuf</a> <a href="#">VioWrtCellStr</a> <a href="#">VioWrtCharStr</a> <a href="#">VioWrtCharStrAtt</a> <a href="#">VioWrtNAttr</a> <a href="#">VioWrtNCell</a> <a href="#">VioWrtNChar</a> <a href="#">VioWrtTTY</a> <a href="#">VioScrLock</a> <a href="#">VioPopUp</a>
Tools	<a href="#">BIND</a>
Modules	<a href="#">DOSCALLS.DLL</a> <a href="#">VIOCALLS.DLL</a> <a href="#">KBDCALLS.DLL</a> <a href="#">MSG.DLL</a>
Libraries	<a href="#">API.LIB</a> <a href="#">OS2386.LIB</a> <a href="#">FAPI.LIB</a> <a href="#">DOSCALLS.LIB</a> <a href="#">SUBCALLS.LIB</a>

2018/08/25 15:05 · [prokushev](#) · [0 Comments](#)

From:

<https://cocorico.osfree.org/doku/> - **osFree wiki**

Permanent link:

<https://cocorico.osfree.org/doku/doku.php?id=en:docs:fapi:dosclose>

Last update:

**2021/12/05 06:37**

