

PBCreate Create an empty file (both forks)

#include <Files.h>

File Manager (PBxxx)

OSErr **PBCreate**(*pb*, *async*);
ParmBlkPtr *pb* ; address of a 80-byte FileParam structure
Boolean *async* ; 0=await completion; 1=immediate return
 returns Error Code; 0=no error

PBCreate creates a 0-length file (both the data and resource forks), setting its a creation and modification time to the current date and time. Use **PBSetFInfo** to supply other descriptive information about the file. The file is NOT opened by this call.

pb is the address of an 80-byte FileParam structure. The relevant fields are as follows:

| Out-In Name | Type | Size | Offset | Description |
|-----------------|-------------------|------|--------|--|
| -> ioCompletion | <u>ProcPtr</u> | 4 | 12 | Completion routine address (if <i>async</i> =TRUE) |
| -> ioNamePtr | <u>StringPtr</u> | 4 | 18 | Address of full or partial path/filename |
| -> ioVRefNum | <u>short</u> | 2 | 22 | Volume or working directory reference |
| -> ioFVersNum | <u>SignedByte</u> | 1 | 26 | Version (best to use 0, must be 0 for HFS) |
| -> ioFIVersNum | <u>char</u> | 1 | 31 | File version (always set to 0) |
| <- ioResult | <u>OSErr</u> | 2 | 16 | Error Code (0=no error, 1=not done yet) |

async is a Boolean value. Use FALSE for normal (synchronous) operation or TRUE to enqueue the request and resume control immediately. See Async I/O.

Returns: an operating system Error Code. It will be one of:

| | | |
|-----------|-------|-----------------------------------|
| noErr | (0) | No error |
| dirFulErr | (-33) | Directory full |
| dupFNErr | (-48) | Duplicate filename already exists |
| extFSErr | (-58) | External file system |
| ioErr | (-36) | I/O error |
| nsvErr | (-35) | No such volume |
| vLckdErr | (-46) | Volume is locked |
| wPrErr | (-44) | Diskette is write-protected |

Notes: If an existing file has the specified name and volume/directory, then **PBCreate** fails, returning dupFNErr. If you use **SFGetFile** to obtain the file name, you can assume that the user intended to overwrite the existing file. You can call **PBDelete** before calling **PBCreate**.

The file's creation date and time are set correctly, but its file type and creator ID are set to 0. The Finder displays such a file as the standard document icon, placed in the first-available "clean-up grid" position. Unless you intend to delete the file before terminating, you should use **PBSetFInfo** to override these defaults.

The higher-level **Create** function lets you set the document type and creator ID in the call.

Be sure to set ioFVersNum to 0 - Standard File and the latest File Manager functions tend to assume a version number of 0 and your new file may not be accessible. If you use **PBHCreate** (and other **PBHxxx** calls), you can

ignore version numbers altogether.

The ioFIVersNum field should always be set to 0 for normal Macintosh files. There are a number of parts of the Macintosh System that will not deal correctly with files that have the wrong version: the **Standard File Package** will not display any file with a non-zero ioFIVersNum; the **Segment Loader** and **Resource Manager** cannot open files that have a non-zero ioFIVersNum. This is not actually a problem unless a file is being created on an MFS disk. However, it is better safe than confused.

Don't start filenames with a period '.' since that is the convention used for device names (e.g., ".Sony"). You should not allow the user to specify such names.

Desk Accessories should create files only in the directory containing the system folder. Just use ioVRefNum=BootDrive.