

PlaySound

The **PlaySound** function plays a sound specified by the given filename, resource, or system event. (A system event may be associated with a sound in the registry or in the WIN.INI file.)

```
BOOL PlaySound(  
    LPCSTR pszSound,  
    HMODULE hmod,  
    DWORD fdwSound  
);
```

Parameters

pszSound

A string that specifies the sound to play. If this parameter is NULL, any currently playing waveform sound is stopped. To stop a non-waveform sound, specify SND_PURGE in the *fdwSound* parameter.

Three flags in *fdwSound* (SND_ALIAS, SND_FILENAME, and SND_RESOURCE) determine whether the name is interpreted as an alias for a system event, a filename, or a resource identifier. If none of these flags are specified, **PlaySound** searches the registry or the WIN.INI file for an association with the specified sound name. If an association is found, the sound event is played. If no association is found in the registry, the name is interpreted as a filename.

hmod

Handle of the executable file that contains the resource to be loaded. This parameter must be NULL unless SND_RESOURCE is specified in *fdwSound*.

fdwSound

Flags for playing the sound. The following values are defined:

SND_APPLICATION

The sound is played using an application-specific association.

SND_ALIAS

The *pszSound* parameter is a system-event alias in the registry or the WIN.INI file. Do not use with either SND_FILENAME or SND_RESOURCE.

SND_ALIAS_ID

The *pszSound* parameter is a predefined sound identifier.

SND_ASYNC

The sound is played asynchronously and **PlaySound** returns immediately after beginning the sound. To terminate an asynchronously played waveform sound, call **PlaySound** with *pszSound* set to NULL.

SND_FILENAME

The *pszSound* parameter is a filename.

SND_LOOP

The sound plays repeatedly until **PlaySound** is called again with the *pszSound* parameter set to NULL. You must also specify the SND_ASYNC flag to indicate an asynchronous sound event.

SND_MEMORY

A sound event's file is loaded in RAM. The parameter specified by *pszSound* must point to an image of a sound in memory.

SND_NODEFAULT

No default sound event is used. If the sound cannot be found, **PlaySound** returns silently without playing the default sound.

SND_NOSTOP

The specified sound event will yield to another sound event that is already playing. If a sound cannot be played because the resource needed to generate that sound is busy playing another sound, the function immediately returns FALSE without playing the requested sound.

If this flag is not specified, **PlaySound** attempts to stop the currently playing sound so that the device can be used to play the new sound.

SND_NOWAIT

If the driver is busy, return immediately without playing the sound.

SND_PURGE

Sounds are to be stopped for the calling task. If *pszSound* is not NULL, all instances of the specified sound are stopped. If *pszSound* is NULL, all sounds that are playing on behalf of the calling task are stopped.

You must also specify the instance handle to stop SND_RESOURCE events.

SND_RESOURCE

The *pszSound* parameter is a resource identifier; *hmod* must identify the instance that contains the resource.

SND_SYNC

Synchronous playback of a sound event. **PlaySound** returns after the sound event completes.

Return Values

Returns TRUE if successful or FALSE otherwise.

Remarks

The sound specified by *pszSound* must fit into available physical memory and be playable by an installed waveform-audio device driver. **PlaySound** searches the following directories for sound files: the current directory; the Windows directory; the Windows system directory; directories listed in the PATH environment variable; and the list of directories mapped in a network. For more information about the directory search order, see the documentation for the **OpenFile** function.

If it cannot find the specified sound, **PlaySound** uses the default system event sound entry instead. If the function can find neither the system default entry nor the default sound, it makes no sound and returns FALSE.

QuickInfo

Windows NT: Requires version 3.1 or later.

Windows: Requires Windows 95 or later.

Windows CE: Unsupported.

Header: Declared in mmsystem.h.

Import Library: Use winmm.lib.

Unicode: Implemented as Unicode and ANSI versions on Windows NT.

See Also

[Waveform Audio Overview](#), [Waveform Functions](#)