

Navigation

- [index](#)
- [modules](#) |
- [next](#) |
- [previous](#) |
- [Python v2.6.4 documentation](#) »
- [The Python Standard Library](#) »
- [35. MS Windows Specific Services](#) »

35.4. winsound — Sound-playing interface for Windows¶

Platforms: Windows

New in version 1.5.2.

The `winsound` module provides access to the basic sound-playing machinery provided by Windows platforms. It includes functions and several constants.

`winsound.Beep(frequency, duration)`¶

Beep the PC's speaker. The *frequency* parameter specifies frequency, in hertz, of the sound, and must be in the range 37 through 32,767. The *duration* parameter specifies the number of milliseconds the sound should last. If the system is not able to beep the speaker, `RuntimeError` is raised.

New in version 1.6.

`winsound.PlaySound(sound, flags)`¶

Call the underlying `PlaySound()` function from the Platform API. The *sound* parameter may be a filename, audio data as a string, or `None`. Its interpretation depends on the value of *flags*, which can be a bitwise ORed combination of the constants described below. If the *sound* parameter is `None`, any currently playing waveform sound is stopped. If the system indicates an error, `RuntimeError` is raised.

`winsound.MessageBeep([type=MB_OK])`¶

Call the underlying `MessageBeep()` function from the Platform API. This plays a sound as specified in the registry. The *type* argument specifies which sound to play; possible values are `-1`, `MB_ICONASTERISK`, `MB_ICONEXCLAMATION`, `MB_ICONHAND`, `MB_ICONQUESTION`, and `MB_OK`, all described below. The value `-1` produces a "simple beep"; this is the final fallback if a sound cannot be played otherwise.

New in version 2.3.

`winsound.SND_FILENAME`¶

The *sound* parameter is the name of a WAV file. Do not use with `SND_ALIAS`.

`winsound.SND_ALIAS`¶

The *sound* parameter is a sound association name from the registry. If the registry contains no such name, play the system default sound unless `SND_NODEFAULT` is also specified. If no default sound is registered, raise `RuntimeError`. Do not use with `SND_FILENAME`.

All Win32 systems support at least the following; most systems support many more:

<code>PlaySound()</code> name	Corresponding Control Panel Sound name
'SystemAsterisk'	Asterisk
'SystemExclamation'	Exclamation
'SystemExit'	Exit Windows
'SystemHand'	Critical Stop
'SystemQuestion'	Question

For example:

```
import winsound
# Play Windows exit sound.
winsound.PlaySound("SystemExit", winsound.SND_ALIAS)

# Probably play Windows default sound, if any is registered (because
# "" probably isn't the registered name of any sound).
winsound.PlaySound("", winsound.SND_ALIAS)
```

`winsound.SND_LOOP`¶

Play the sound repeatedly. The `SND_ASYNC` flag must also be used to avoid blocking. Cannot be used with `SND_MEMORY`.

`winsound.SND_MEMORY`¶

The *sound* parameter to `PlaySound()` is a memory image of a WAV file, as a string.

Note

This module does not support playing from a memory image asynchronously, so a combination of this flag and [SND_ASYNC](#) will raise [RuntimeError](#).

`winsound.SND_PURGE`

Stop playing all instances of the specified sound.

Note

This flag is not supported on modern Windows platforms.

`winsound.SND_ASYNC`

Return immediately, allowing sounds to play asynchronously.

`winsound.SND_NODEFAULT`

If the specified sound cannot be found, do not play the system default sound.

`winsound.SND_NOSTOP`

Do not interrupt sounds currently playing.

`winsound.SND_NOWAIT`

Return immediately if the sound driver is busy.

`winsound.MB_ICONASTERISK`

Play the `SystemDefault` sound.

`winsound.MB_ICONEXCLAMATION`

Play the `SystemExclamation` sound.

`winsound.MB_ICONHAND`

Play the `SystemHand` sound.

`winsound.MB_ICONQUESTION`

Play the `SystemQuestion` sound.

`winsound.MB_OK`

Play the `SystemDefault` sound.

Previous topic

[35.3. _winreg – Windows registry access](#)

Next topic

[36. Unix Specific Services](#)

This Page

- [Show Source](#)

Navigation

- [index](#)
- [modules](#) |
- [next](#) |
- [previous](#) |
- [Python v2.6.4 documentation](#) »
- [The Python Standard Library](#) »
- [35. MS Windows Specific Services](#) »

© [Copyright](#) 1990-2010, Python Software Foundation.

The Python Software Foundation is a non-profit corporation. [Please donate.](#)

Last updated on Feb 26, 2010. Created using [Sphinx](#) 0.6.3.