

# پخش فایل‌های صوتی و تصویری

---

- سوال ۱ - چگونه می‌توان فایل‌های WAV را در VB نواخت ؟
- سوال ۲ - چگونه می‌توان فایل‌های MID را در VB نواخت ؟
- سوال ۳ - چگونه می‌توان فایل‌های AVI, MPG را در VB نمایش داد؟
- سوال ۴ - چگونه می‌توان فایل‌های چند رسانه‌ای را در فایل EXE ذخیره نمود و سپس آنها را در VB پخش نمود؟
- سوال ۵ - چگونه می‌توان سی‌دی موسیقی را در VB نواخت ؟
- سوال ۶ - چگونه می‌توان وجود کارت صوتی را تشخیص داد؟
- سوال ۷ - چگونه می‌توان فایل صوتی را در حافظه بار کرد و سپس آنرا در VB نواخت ؟
- 

سوال ۱ - چگونه می‌توان فایل‌های WAV را در VB نواخت ؟

## پاسخ :

اینگونه فایل‌ها می‌توانند حاوی هر نوع صوتی باشند و آنها را با یک میکروفون به صورت دیجیتالی می‌توان ذخیره نمود. حجم آنها بسیار زیاد است و فرمت استاندارد فایل‌های صوتی تحت ویندوز می‌باشند. تعداد زیادی از آنها را می‌توان در دایرکتوری C:\WINDOWS\MEDIA یافت .

برای پخش نمودن این فایل‌ها باید از توابع ویندوز کمک گرفت. در زیر دو نمونه از توابع مربوطه ذکر می‌شوند:

## ۱- تابع `sndPlaySound` :

اولین آرگومان آن مسیر و نام فایل `WAV` می باشد و دومین آرگومان آن ثوابتی هستند که در زیر ارائه می شوند و طریقه نواخته شدن فایل صوتی را معین می کنند . برای ترکیب آنها می توان از عملگر `OR` استفاده کرد :

**`SND_SYNC`** : در صورت انتخاب این ثابت به برنامه تا تمام شدن پخش فایل صوتی اجازه ادامه یافتن داده نمی شود.

**`SND_ASYNC`** : با انتخاب این ثابت همزمان با پخش فایل صوتی کارهای عادی برنامه نیز می توانند صورت گیرند.

**`SND_NODEFAULT`** : این ثابت باعث می شود که اگر فایل صوتی مورد نظر یافت نشد صدایی پخش نشود ، در غیر اینصورت صدای پیش فرض سیستم پخش خواهد شد.

**`SND_LOOP`** : این ثابت سبب پخش مکرر فایل صوتی مورد نظر شده و تا در جای دیگر برنامه آرگومان اول برابر `0` قرار نگیرد به پخش مکرر ادامه می دهد.

همچنین این ثابت باید با ثابت `SND_ASYNC` بطور همزمان بکار گرفته شود.

**`SND_NOSTOP`** : اگر فایل صوتی در حال نواخته شدن باشد بکار گیری این تابع سبب خواهد شد که بدون نواخته شدن فایل جدید تابع سریعاً مقدار `False` را بازگشت دهد.

**`SND_MEMORY`** : اشاره گری است به صدای بار شده در حافظه و بیشتر هنگامی مورد استفاده قرار می گیرد که فایل صوتی درون فایل اجرایی برنامه ذخیره شده باشد.

**`SND_ALIAS`** : صدای رخدادهای ویندوز که نام آنها در محضر خانه آن ذخیره شده پخش خواهد شد.

**`SND_FILENAME`** : معین می کند که فایل از روی سخت دیسک خوانده می شود و سپس در حافظه بار خواهد شد.

شرکت مایکروسافت توصیه کرده است که از این تابع برای نواختن فایل های بیش از **100 KB** استفاده نشود. اما توابع دیگری که در این فصل معرفی می شوند این مشکل را ندارند. ولی برخلاف این تابع قدرت پخش چندین فایل را بطور همزمان دارا نیستند.

برای استفاده ساده تر از این توابع می توان از توابع زیر استفاده کرد. برای استفاده از آنها باید یک ماژول جدید باز کنید و توابع زیر را در آنها بنویسید:

---

```
Declare Function sndPlaySound Lib "winmm.dll" Alias _  
"sndPlaySoundA" ( _  
  ByVal lpszSoundName As String, _  
  ByVal uFlags As Long) As Long
```

```
Public Declare Function PlaySound Lib "winmm.dll" Alias _  
"PlaySoundA" ( _  
  ByVal lpszName As String, _  
  ByVal hModule As Long, _  
  ByVal dwFlags As Long) As Long
```

```
Enum SND_CONSTANTS  
  SND_ALIAS = &H10000  
  SND_ASYNC = &H1  
  SND_FILENAME = &H20000  
  SND_LOOP = &H8  
  SND_NODEFAULT = &H2  
  SND_NOSTOP = &H10  
  SND_NOWAIT = &H2000  
  SND_SYNC = &H0  
  SND_MEMORY = &H4
```

```
End Enum
```

---

```
Sub PlayWav02(FileName As String, wFlags As SND_CONSTANTS)  
  PlaySound FileName, 0, wFlags  
End Sub
```

---

```
Sub PlayWav(SoundFileName As String, wFlags As SND_CONSTANTS)  
  Dim x  
  x = sndPlaySound(SoundFileName, wFlags)  
End Sub
```

---

```
Sub EndPlayWav()  
  Dim ret As Variant  
  ret = sndPlaySound(0&, 0&)  
End Sub
```

---

دو مثال در مورد چگونگی استفاده از توابع فوق :

```
PlayWav "d:\myWav.wav", SND_ASYNC Or SND_NODEFAULT
```

```
'PlayWav02 "d:\myWav.wav", SND_ASYNC + SND_NODEFAULT
```

---

سوال ۲ - چگونه می توان فایل های **MID**. را در **VB** نواخت ؟

**پاسخ :**

با استفاده از تابع **mciExecute** می توان فایل های متنوعی را در **VB** نواخت .

```
Public Declare Function mciExecute Lib "winmm.dll" _  
  (ByVal lpstrCommand As String) As Long
```

---

```
Sub PlayMIDIorMP3orWAV(MIDIorMP3orWAV_FileName As String)  
  mciExecute "play " & MIDIorMP3orWAV_FileName  
End Sub
```

---

```
Sub StopMIDIorMP3orWAV(MIDIorMP3orWAV_FileName As String)  
  mciExecute "stop " & MIDIorMP3orWAV_FileName  
End Sub
```

---

```
PlayMIDIorMP3orWAV "d:\mymidi.mid"
```

```
StopMIDIorMP3orWAV "d:\mymidi.mid"
```

```
PlayMIDIorMP3orWAV "d:\mymp3.mp3"
```

```
StopMIDIorMP3orWAV "d:\mymp3.mp3"
```

---

سوال ۳ - چگونه می توان فایل های **AVI**، **MPG**. را در **VB** نمایش داد؟

**پاسخ :**

با استفاده از تابع **mciSendString** می توان فایل های صوتی و همچنین تصویری را نمایش داد .  
فایل های **AVI**. فرمت استاندارد فایل های ویدیویی تحت ویندوز هستند اما بسیار حجیم می باشند.  
و برای فشرده سازی آنها از فرمت **MPG**. استفاده می شود. اغلب **VCD** ها با این فرمت ذخیره  
می گردند.

تابع `mciSendString` توانایی پخش فایل‌های بالای صد کیلو بایت را به سادگی دارا است اما در هر لحظه فقط یک فایل را می‌تواند پخش نماید. مهمترین آرگومان آن `lpstrCommand` می‌باشد و عملی را که تابع باید انجام دهد معین می‌نماید و در زیر طرز استفاده از آن بصورت مثال ارائه شده است.

---

```
Public Declare Function mciSendString Lib "winmm.dll" Alias _  
"mciSendStringA" ( _  
  ByVal lpstrCommand As String, _  
  ByVal lpstrReturnString As String, _  
  ByVal uReturnLength As Long, _  
  ByVal hwndCallback As Long) As Long
```

---

```
Sub PlayAVIorMPG(AVIorMPG_FileName As String, FullScreen As Boolean)
```

```
  Dim ReturnVal
```

```
  If FullScreen = False Then
```

```
    ReturnVal = mciSendString("play " & AVIorMPG_FileName, 0&, 0, 0&)
```

```
  Else
```

```
    ReturnVal = mciSendString("play " & AVIorMPG_FileName & " fullscreen", 0&, 0, 0&)
```

```
  End If
```

```
End Sub
```

---

```
PlayAVIorMPG "d:\myavi.avi", False
```

```
PlayAVIorMPG "d:\mympg.mpg", False
```

و برای پخش فایل‌های ویدیویی در یک پنجره خاص می‌توان از کد زیر کمک گرفت:

```
Declare Function mciGetErrorString Lib "winmm" Alias _  
"mciGetErrorStringA" ( _  
  ByVal dwError As Long, ByVal lpstrBuffer As String, _  
  ByVal uLength As Long) As Long
```

```
Declare Function GetShortPathName Lib "kernel32" Alias _  
"GetShortPathNameA" (ByVal lpszLongPath As String, _  
  ByVal lpszShortPath As String, ByVal cchBuffer As Long) _  
  As Long
```

```
Public Const WS_CHILD = &H40000000
```

```

Sub PlayAVIPictureBox( _
  FileName As String, _
  ByVal Window As PictureBox _
)
  Dim RetVal As Long
  Dim CommandString As String
  Dim ShortFileName As String * 260
  Dim deviceIsOpen As Boolean

  'Retrieve short file name format
  RetVal = GetShortPathName(FileName, ShortFileName, _
    Len(ShortFileName))
  FileName = Left$(ShortFileName, RetVal)

  'Open the device
  CommandString = "Open " & FileName & _
    " type AVIVideo alias AVIFile Parent " _
    & CStr(Window.hWnd) & _
    " style " & CStr(WS_CHILD)

  RetVal = mciSendString(CommandString, vbNullString, 0, 0&)
  If RetVal Then GoTo Error

  'remember that the device is now open
  deviceIsOpen = True

  'Resize the movie to PictureBox size
  CommandString = "put AVIFile window at 0 0 " & CStr(Window.ScaleWidth / _
    Screen.TwipsPerPixelX) & " " & CStr(Window.ScaleHeight / _
    Screen.TwipsPerPixelY)
  RetVal = mciSendString(CommandString, vbNullString, 0, 0&)
  If RetVal <> 0 Then GoTo Error

  'Play the file
  CommandString = "Play AVIFile wait"
  RetVal = mciSendString(CommandString, vbNullString, 0, 0&)
  If RetVal <> 0 Then GoTo Error

  'Close the device
  CommandString = "Close AVIFile"
  RetVal = mciSendString(CommandString, vbNullString, 0, 0)

Exit Sub
Error:
  MsgBox "error...", vbExclamation
End Sub

```

---

```

Private Sub Command1_Click()
  PlayAVIPictureBox App.Path + "\myavi.avi", Picture1
End Sub

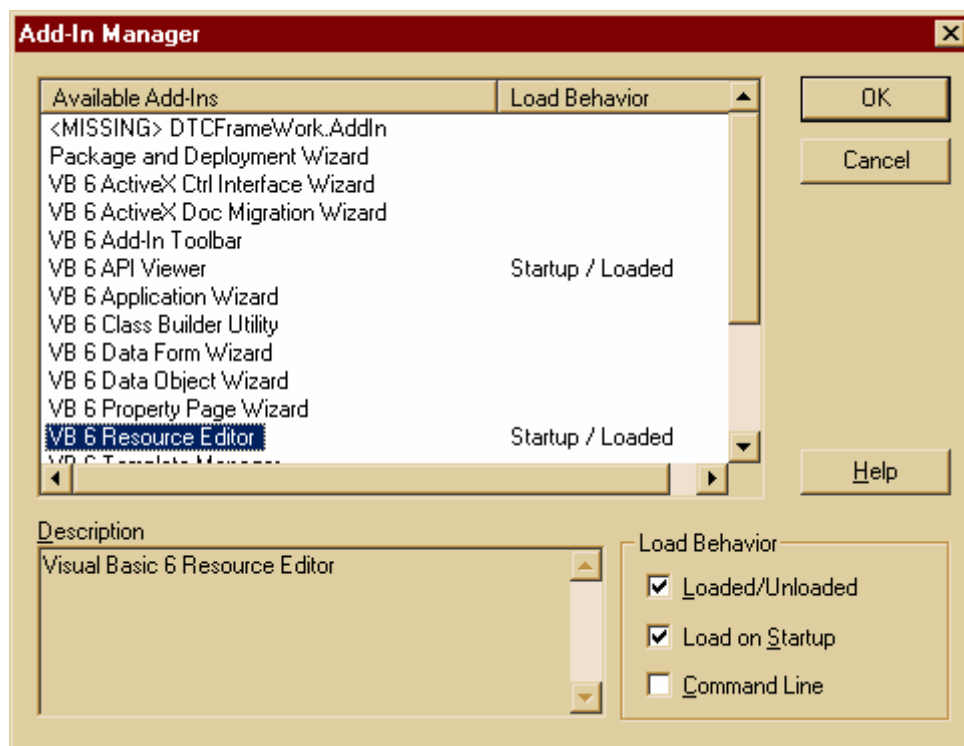
```

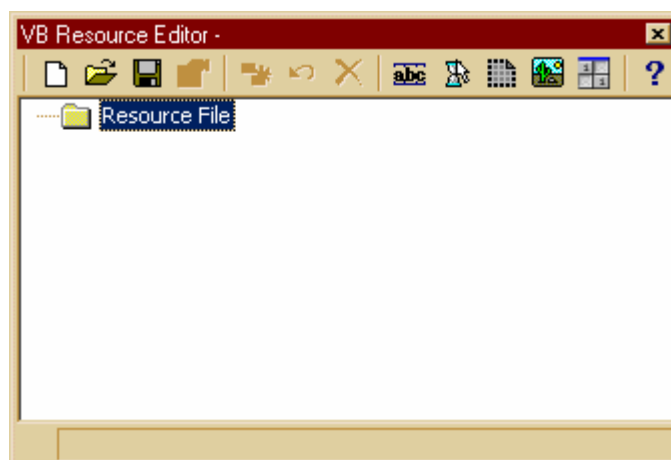
---

سوال ۱۴ - چگونه می توان فایل های چند رسانه ای را در فایل EXE ذخیره نمود و سپس آنها را در VB پفش نمود؟

**پاسخ :**

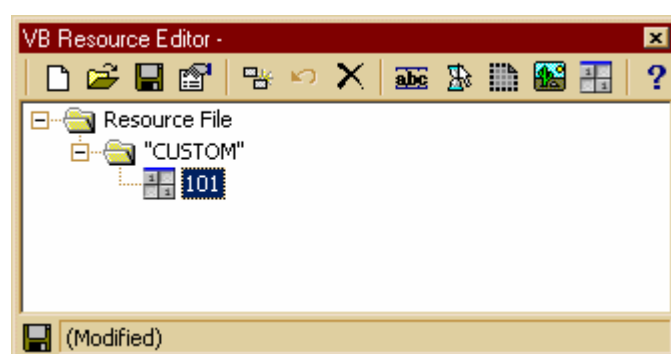
با استفاده از **VB Resource Editor** می توان انواع و اقسام فایل ها مانند آیکونها ، تصاویر، متون ، و همچنین فایل های چند رسانه ای را درون فایل اجرایی برنامه مخفی کرد. این ویرایشگر در قسمت منوی **Add-Ins** قرار دارد.(به شکل های زیر مراجعه شود).





برای اینکه بتوانید یک فایل **WAVE** را به این ویرایشگر اضافه کنید فقط کافی است روی آیکن **Add Custom Resource...** کلیک نموده و فایل صوتی دلخواه خود را اضافه نمایید. سپس با استفاده از روال ارائه شده در زیر می توان این فایل را نواخت. لازم به ذکر است که فایل منبع نخیره شده (که با پسوند **.Res** نخیره می گردد) را لزومی ندارد همراه برنامه خود ارائه نمایید. زیرا هنگام کامپایل شدن برنامه این فایل جزئی از فایل اجرایی برنامه می گردد.

در شکل زیر یک فایل صوتی با شماره ۱۰۱ و از نوع **Custom** به فایل منبع اضافه شده است .




---

**Sub BeginPlaySound\_From\_Res\_File**(ByVal ResourceId **As Integer**, RES\_Type **As String**)

**Dim** SoundBuffer **As String**

SoundBuffer = StrConv(LoadResData(ResourceId, RES\_Type), **vbUnicode**)



```
sndPlaySound SoundBuffer, SND_ASYNC Or SND_NODEFAULT Or SND_MEMORY
' Important: This function is necessary for playing sound asynchronously
DoEvents
End Sub
```

---

```
BeginPlaySound_From_Res_File 101, "custom"
```

---

سوال ۵ - چگونه می توان سی دی موسیقی را در VB نواخت ؟

پاسخ :

با استفاده از

---

```
Sub PlayCDAudio(ByVal nCurrentTrack As Long)
Dim lRet As Long
```

```
'Open the device
```

```
lRet = mciSendString("open cdaudio alias cd wait", 0&, 0, 0)
```

```
'Set the time format to Tracks (default is milliseconds)
```

```
lRet = mciSendString("set cd time format tmsf", 0&, 0, 0)
```

```
'play from a specific track
```

```
lRet = mciSendString("play cd from" & Str(nCurrentTrack), 0&, 0, 0)
```

```
End Sub
```

---

```
Sub CloseCDAudio()
```

```
Dim lRet As Long
```

```
'Stop the playback
```

```
lRet = mciSendString("stop cd wait", 0&, 0, 0)
```

```
DoEvents 'Let Windows process the event
```

```
'Close the device
```

```
lRet = mciSendString("close cd", 0&, 0, 0)
```

```
End Sub
```

---

سوال ۶ - چگونه می توان وجود کارت صوتی را تشخیص داد؟

پاسخ :

با استفاده از

```
Public Declare Function auxGetNumDevs Lib "winmm.dll" () As Long
```

---

```
Function DetectingSoundCard() As Boolean
```

```
Dim i
```

```
    i = auxGetNumDevs()
```

```
If i > 0 Then ' There is at least one sound card on the system
```

```
    DetectingSoundCard = True
```

```
Else ' auxGetNumDevs returns a 0 if there is no sound card
```

```
    DetectingSoundCard = False
```

```
End If
```

```
End Function
```

---

```
If DetectingSoundCard = True Then
```

```
    MsgBox "A Sound Card has been detected.", vbInformation
```

```
Else
```

```
    MsgBox "There is no Sound Card on this system.", vbExclamation
```

```
End If
```

---

```
Public Enum CanSystemPlaySound
```

```
    AUDIO_NONE = 0
```

```
    AUDIO_WAVE = 1
```

```
    AUDIO_MIDI = 2
```

```
    Both = 3
```

```
End Enum
```

```
Private Declare Function waveOutGetNumDevs Lib "winmm" () _  
As Long
```

```
Private Declare Function midiOutGetNumDevs Lib "winmm" () _  
As Integer
```

```
' Returns 1 if wave output
```

```
' Returns 2 if midi output
```

```
' Returns 3 if both
```

```
'
```

```
Public Function CanPlaySound() As CanSystemPlaySound  
  Dim i As Integer
```

```
  i = AUDIO_NONE
```

```
  If waveOutGetNumDevs > 0 Then
```

```
    i = AUDIO_WAVE
```

```
  End If
```

```
  If midiOutGetNumDevs > 0 Then
```

```
    i = i + AUDIO_MIDI
```

```
  End If
```

```
  CanPlaySound = I
```

```
End Function
```

---

```
If CanPlaySound = AUDIO_NONE Then
```

```
  MsgBox "CanPlaySound = FALSE ", vbInformation
```

```
Else
```

```
  MsgBox "CanPlaySound = TRUE ", vbInformation
```

```
End If
```

---

سوال ۷- چگونه می توان فایل صوتی را در حافظه بار کرد و سپس آنرا در VB نواخت ؟  
پاسخ :

با استفاده از پرچم SND\_MEMORY می توان فایلی را که هم اکنون در حافظه بار شده نواخت. اینکار به شدت روی کارآیی برنامه نسبت به حالتی که فایل باید از روی سخت دیسک بار شود تاثیر مثبتی دارد.

برای اینکار با توجه به اینکه این تابع نام فایل را به صورت رشته دریافت می کند فقط کافی است فایل را درون یک رشته بار کنیم و بدیهی است که در این حالت فایل درون حافظه مستقر شده است .

تابع زیر روش انجام این کار را بیان می کند.

---

'Loads a sound from file into memory

```
Private Function LoadSound(FileName As String) As String
```

**On Error GoTo** Error\_Handler

**Dim** FreeFileNumber **As Integer**

**Dim** SoundBuffer **As String**

FreeFileNumber = FreeFile

SoundBuffer = Space\$(FileLen(FileName)) **'Make room for sound file**

**Open** FileName **For Binary As** #FreeFileNumber

**Get** #FreeFileNumber, , SoundBuffer

**Close** FreeFileNumber

**'remove wasted spaces**

LoadSound = Trim\$(SoundBuffer)

Error\_Handler:

**Select Case** Err

**Case 0** 'No error

**Case Else**

**MsgBox** Err.Description & vbCrLf & "Error Number: " & Err.Number

            Err.Clear

            LoadSound = ""

**End Select**

**End Function**

---

متغیر رشته ای soundBuffer بعنوان یک بافر موقتی برای فایل صوتی عمل خواهد نمود.

**Private Sub** cmdFile\_Click()

    sndPlaySound SoundFile, SND\_ASYNC **Or** SND\_FILENAME

**End Sub**

پرچم SND\_FILENAME فایل را از روی دیسک خوانده و آنرا در حافظه بار می کند.

---

و پرچم SND\_MEMORY فایل جای گرفته در حافظه را پخش می نماید.

**Private Sub** cmdMemory\_Click()

    sndPlaySound SoundInMemory, SND\_ASYNC **Or** SND\_MEMORY

**End Sub**

---

و با استفاده از کد رخداد زیر می توان پخش فایل را متوقف نمود.

**Private Sub** cmdStopFile\_Click()

    sndPlaySound 0, 0

**End Sub**

البته در این حالت در هر بار فقط یک فایل را می توان نواخت و توصیه صد کیلو بایتی مایکرو  
سافت در اینجا بیش از پیش رخ می نماید!

---