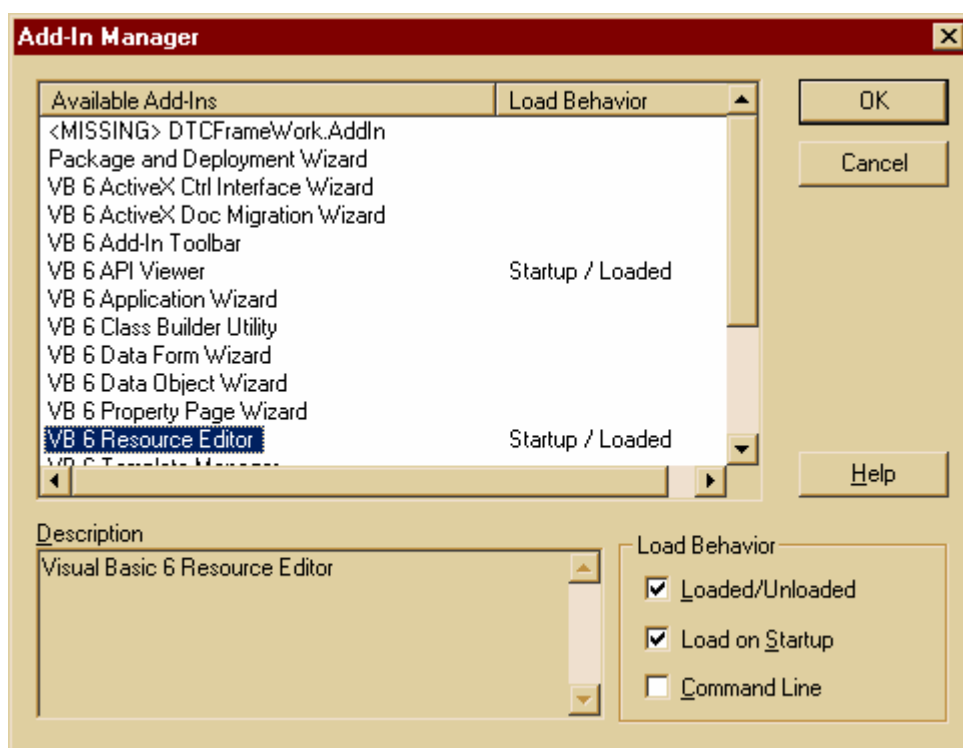


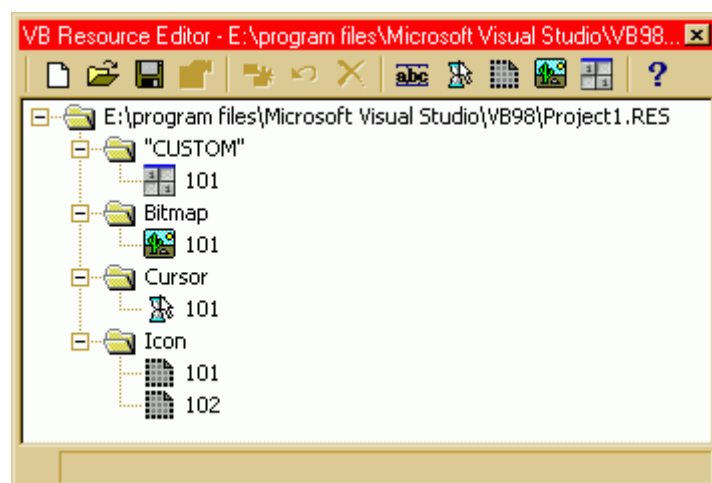
استفاده از فایل‌های منبع در ویژوال بیسیک

۱- بارگذاری فایل‌های تصویری از فایل منبع

با استفاده از فایل‌های منبع می‌توان چندین تصویر، کرسر، فایل‌های صوتی و سایر فایل‌ها را در یک فایل ذخیره کرد و این فایل هنگام ایجاد فایل اجرایی برنامه جزئی از آن خواهد شد. با استفاده از **VB Resource Editor** می‌توان اینکار را انجام داد. این ویرایشگر در قسمت منوی **Add-Ins** قرار دارد. (به شکل‌های زیر مراجعه شود).



یک جعبه تصویر به فرم جاری برنامه اضافه کنید. برای اضافه کردن فایل‌های BMP, Ico, Cur به سادگی می‌توانید به نوار ابزار این ویرایشگر رجوع کرده و این فایل‌ها را اضافه نمایید. به صورت پیش فرض آنها از شماره ۱۰۱ به بعد ذخیره خواهند شد. برای اضافه کردن فایل‌های GIF/JPG از نوار ابزار گزینه **Add Custom Resource** را انتخاب کنید. سپس فایل منبع را با نامی دلخواه ذخیره کنید. اگر خواستید این فایل را به برنامه دیگری نیز اضافه کنید به منوی **Project->Add new resource file** مراجعه نمایید. البته این گزینه تنها هنگامی به منوی پروژه اضافه خواهد شد که ویرایشگر فایل منبع را به روشی که گفته شد به برنامه اضافه کرده باشید.



برای بارگذاری فایل های BMP,Ico,Cur به درون جعبه تصویریا فرم به یکی از سه روش زیر عمل کنید:

```
Picture1.Picture = LoadResPicture(101, vbResBitmap)
```

```
Picture1.Picture = LoadResPicture(101, vbResIcon)
```

```
Form1.MouseIcon = LoadResPicture(101, vbResCursor)
& MousePointer -> 99-Custom
```

و برای بارگذاری فایل های GIF/JPG از فایل منبع متاسفانه تابعی وجود ندارد. اما می توان از روش زیر استفاده نمود:

```
Private Sub Form_Load()
'Replace '101' with your File ID.
'Replace 'c:\tmp\mytmp' with the name of the
'Temporary file you want to save the image to.
LoadDataIntoFile 101, "c:\tmp\mytmp"
Picture1.Picture = LoadPicture("c:\tmp\mytmp")
End Sub

Public Sub LoadDataIntoFile(DataName As Integer, _
                             FileName As String)
    Dim myArray() As Byte
    Dim myFile As Long
    If Dir(FileName) = "" Then
        myArray = LoadResData(DataName, "CUSTOM")
        myFile = FreeFile
        Open FileName For Binary Access Write As #myFile
        Put #myFile, , myArray
        Close #myFile
    End If
End Sub
```

۲- ذخیره سازی و فراخوانی رشته های منبع با توجه به نوع زبان نصب شده

برای اینکار یک پروژه جدید **ActiveX DLL** را شروع کنید. همانگونه که گفته شد ویرایشگر

فایل منبع را آغاز نمایید. روی آیکن 'abc' آن کلیک کرده و نوع زبان دلخواه را از صفحه

ظاهر شده انتخاب کنید. و سپس متن دلخواه خود را وارد نمایید. فایل منبع را با نام

res1009 و برنامه را با نام **resdll1009** ذخیره و کامپایل کنید کنید. نام فایل **DLL** مهم

است.

سپس یک پروژه معمولی باز کنید و کد زیر را در فرم آن وارد کنید:

```
1. Option Explicit
2.
3. Private Declare Function GetThreadLocale Lib "kernel32" () As Long
4. Private Declare Function LoadLibrary Lib "kernel32" Alias "LoadLibraryA" _
5.     (ByVal lpLibFileName As String) As Long
6. Private Declare Function LoadString Lib "user32" Alias "LoadStringA" _
7.     (ByVal hInstance As Long, ByVal wID As Long, ByVal lpBuffer As String, _
8.     ByVal nBufferMax As Long) As Long
9. Private Declare Function GetLocaleInfo Lib "kernel32" Alias "GetLocaleInfoA" _
10.    (ByVal Locale As Long, ByVal LCType As Long, ByVal lpLCData As String, _
11.    ByVal cchData As Long) As Long
12.
13. Private Const LOCALE_IDEFAULTANSICODEPAGE = &H1004&
14. Private Const LOCALE_SENCGCOUNTRY = &H1002
15. Private Const LOCALE_SNATIVECTRYNAME = &H8
16. Private Const LOCALE_SCOUNTRY = &H6
17.
18. Function GetCharSet(sCdp As String) As Integer
19.     Select Case sCdp
20.         Case "932" ' Japanese
21.             GetCharSet = 128
22.         Case "936" ' Simplified Chinese
23.             GetCharSet = 134
24.         Case "949" ' Korean
25.             GetCharSet = 129
26.         Case "950" ' Traditional Chinese
27.             GetCharSet = 136
28.         Case "1250" ' Eastern Europe
29.             GetCharSet = 238
30.         Case "1251" ' Russian
31.             GetCharSet = 204
32.         Case "1252" ' Western European Languages
33.             GetCharSet = 0
34.         Case "1253" ' Greek
35.             GetCharSet = 161
36.         Case "1254" ' Turkish
37.             GetCharSet = 162
38.         Case "1255" ' Hebrew
39.             GetCharSet = 177
40.         Case "1256" ' Arabic
```

```

41.         GetCharSet = 178
42.         Case "1257" ' Baltic
43.             GetCharSet = 186
44.         Case Else
45.             GetCharSet = 0
46.         End Select
47.     End Function
48.     Function StripNullTerminator(sCP As String)
49.         Dim posNull As Long
50.         posNull = InStr(sCP, Chr$(0))
51.         StripNullTerminator = Left$(sCP, posNull - 1)
52.     End Function
53.     Private Sub Form_Load()
54.         Dim hInst As Long, lResult As Long, x As Long
55.         Dim LCID As Long, sLcid As String
56.         Dim resString As String * 255
57.         Dim sCodePage As String
58.
59.         sCodePage = String$(16, " ")
60.         Label1.AutoSize = True
61.         Label1.Caption = ""
62.
63.         LCID = GetThreadLocale() 'Get Current locale
64.         sLcid = Hex$(Trim$(CStr(LCID))) 'Convert to Hex
65.         ' Display decimal value of the LCID (Hex in Parentheses)
66.         Form1.Caption = "LCID " & LCID & " (" & sLcid & ")"
67.         x = GetLocaleInfo(LCID, LOCALE_IDEFAULTANSICODEPAGE, _
68.             sCodePage, Len(sCodePage)) 'Get code page
69.         sCodePage = StripNullTerminator(sCodePage)
70.         ' Load dll with string table resource.
71.         ' Might need to change path for the resdll.
72.         hInst = LoadLibrary("resdll" & sLcid & ".dll")
73.
74.         ' Get string with ID 101.
75.         lResult = LoadString(hInst, 101, resString, 255)
76.         With Label1.Font
77.             .Name = "Lucida Sans Unicode"
78.             .Size = 14
79.             .Charset = GetCharSet(sCodePage) 'Convert code page to charset
80.         End With
81.         ' Display the localized string.
82.         Label1.Caption = resString
83.     End Sub

```

با تغییر **Regional Settings** به یکی از زبانهای فوق می توانید رشته های مربوط به آن

به آن زبان را صحیح نمایش دهید.