

```
Option Explicit
```

```
Const FILENAME = "names.dat"
```

```
Dim employee As namedata
```

```
Dim recordnum As Integer
```

```
Private Sub PurgeFile()
```

```
    Dim counter As Long
```

```
    If Dir(App.Path & "\temp.dat") <> "" Then
```

```
        Kill App.Path & "\temp.dat"
```

```
    End If
```

```
    Open App.Path & "\temp.dat" For Random As #2 Len = Len(employee)
```

```
    recordnum = 0
```

```
    Close #1
```

```
    Open App.Path & "\" & FILENAME For Random As #1 Len = Len(employee)
```

```
    For counter = 1 To LOF(1) / Len(employee)
```

```
        Get #1, , employee
```

```
        If employee.delflag <> True Then
```

```
            Put #2, , employee
```

```
            recordnum = recordnum + 1
```

```
        End If
```

```
    Next counter
```

```
    Close #1
```

```
    Close #2
```

```
    Kill App.Path & "\" & FILENAME
```

```
    Name App.Path & "\temp.dat" As App.Path & "\" & FILENAME
```

```
    Open App.Path & "\" & FILENAME For Random As #1 Len = Len(employee)
```

```
    Call DisplayResults
```

```
End Sub
```

```
Private Sub DisplayResults()
```

```
    Dim counter As Long
```

```
    picFileContents.Cls
```

```
    picFileContents.Print "Rec#", "First", "Last", "Number"
```

```
    picFileContents.Print
```

```
    For counter = 1 To recordnum
```

```
        Get #1, counter, employee
```

```
        If employee.delflag <> True Then
```

```
            With employee
```

```
                picFileContents.Print counter, RTrim(.first), RTrim(.last), employee.number
```

```
            End With
```

```
        End If
```

```
    Next counter
```

```
End Sub
```

```
Private Sub AddToFile()
```

```
    With employee
```

```
        .first = InputBox("Enter first name", "Add Info")
```

```
        .last = InputBox("Enter last name", "Add Info")
```

```
        .number = InputBox("Enter number", "Add Info")
```

```
        .delflag = False
```

```
        recordnum = recordnum + 1
```

```
        Put #1, recordnum, employee
```

```
    End With
```

```
End Sub
```

```
Private Sub About_Click()
```

```
    frmAbout.Show
```

```
End Sub
```

```
Private Sub cmdAddFile_Click()
```

```
    Call AddToFile
```

```
Call DisplayResults
```

```
End Sub
```

```
Private Sub cmdDelete_Click()
```

```
Dim recnum As Integer
```

```
Dim i As Integer
```

```
recnum = Val(InputBox("Enter record to delete.", "Delete"))
```

```
Get #1, recnum, employee
```

```
With employee
```

```
.first = ""
```

```
.last = ""
```

```
.number = 0
```

```
.delflag = True
```

```
End With
```

```
Put #1, recnum, employee
```

```
Call DisplayResults
```

```
End Sub
```

```
Private Sub cmdDisplay_Click()
```

```
Call DisplayResults
```

```
End Sub
```

```
Private Sub cmdExit_Click()
```

```
Close #1
```

```
End
```

```
End Sub
```

```
Private Sub cmdUpdate_Click()
```

```
Dim recnum As Integer
```

```
Dim i As Integer
```

```
recnum = Val(InputBox("Enter record to update.", "Update"))
```

```
With employee
```

```
.first = InputBox("Enter first name", "Add Info")
```

```
.last = InputBox("Enter last name", "Add Info")
```

```
.number = InputBox("Enter number", "Add Info")
```

```
End With
```

```
Put #1, recnum, employee
```

```
Call DisplayResults
```

```
End Sub
```

```
Private Sub Exit_Click()
```

```
End
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
Open App.Path & "\" & FILENAME For Random As #1 Len = Len(employee)
```

```
recordnum = LOF(1) / Len(employee)
```

```
End Sub
```

```
Private Sub Purge_Click()
```

```
Call PurgeFile
```

```
End Sub
```

frmAbout - 1

Option Explicit

' Reg Key Security Options...

Const READ_CONTROL = &H20000

Const KEY_QUERY_VALUE = &H1

Const KEY_SET_VALUE = &H2

Const KEY_CREATE_SUB_KEY = &H4

Const KEY_ENUMERATE_SUB_KEYS = &H8

Const KEY_NOTIFY = &H10

Const KEY_CREATE_LINK = &H20

Const KEY_ALL_ACCESS = KEY_QUERY_VALUE + KEY_SET_VALUE + _
KEY_CREATE_SUB_KEY + KEY_ENUMERATE_SUB_KEYS + _
KEY_NOTIFY + KEY_CREATE_LINK + READ_CONTROL

' Reg Key ROOT Types...

Const HKEY_LOCAL_MACHINE = &H80000002

Const ERROR_SUCCESS = 0

Const REG_SZ = 1 ' Unicode nul terminated string

Const REG_DWORD = 4 ' 32-bit number

Const gREGKEYSYSINFOLOC = "SOFTWARE\Microsoft\Shared Tools Location"

Const gREGVALSYSINFOLOC = "MSINFO"

Const gREGKEYSYSINFO = "SOFTWARE\Microsoft\Shared Tools\MSINFO"

Const gREGVALSYSINFO = "PATH"

Private Declare Function RegOpenKeyEx Lib "advapi32" Alias "RegOpenKeyExA" (ByVal hKey As Long, ByVal lpSubKey As String, ByVal ulOptions As Long, ByVal samDesired As Long, ByRef phkResult As Long) As Long

Private Declare Function RegQueryValueEx Lib "advapi32" Alias "RegQueryValueExA" (ByVal hKey As Long, ByVal lpValueName As String, ByVal lpReserved As Long, ByRef lpType As Long, ByVal lpData As String, ByRef lpcbData As Long) As Long

Private Declare Function RegCloseKey Lib "advapi32" (ByVal hKey As Long) As Long

Private Sub cmdSysInfo_Click()

Call StartSysInfo

End Sub

Private Sub cmdOK_Click()

Unload Me

End Sub

Private Sub Form_Load()

Me.Caption = "About " & App.Title

lblVersion.Caption = "Version " & App.Major & "." & App.Minor & "." & App.Revision

lblTitle.Caption = App.Title

End Sub

Public Sub StartSysInfo()

On Error GoTo SysInfoErr

Dim rc As Long

Dim SysInfoPath As String

' Try To Get System Info Program Path\Name From Registry...

If GetKeyValue(HKEY_LOCAL_MACHINE, gREGKEYSYSINFO, gREGVALSYSINFO, SysInfoPath) Then

' Try To Get System Info Program Path Only From Registry...

ElseIf GetKeyValue(HKEY_LOCAL_MACHINE, gREGKEYSYSINFOLOC, gREGVALSYSINFOLOC, SysInfoPath) Th

en

' Validate Existance Of Known 32 Bit File Version

If (Dir(SysInfoPath & "\MSINFO32.EXE") <> "") Then

```

        SysInfoPath = SysInfoPath & "\MSINFO32.EXE"

        ' Error - File Can Not Be Found...
    Else
        GoTo SysInfoErr
    End If
' Error - Registry Entry Can Not Be Found...
Else
    GoTo SysInfoErr
End If

Call Shell(SysInfoPath, vbNormalFocus)

Exit Sub
SysInfoErr:
    MsgBox "System Information Is Unavailable At This Time", vbOKOnly
End Sub

Public Function GetKeyValue(KeyRoot As Long, KeyName As String, SubKeyRef As String, ByRef KeyVal
l As String) As Boolean
    Dim i As Long                ' Loop Counter
    Dim rc As Long              ' Return Code
    Dim hKey As Long            ' Handle To An Open Registry Key
    Dim hDepth As Long         '
    Dim KeyValType As Long      ' Data Type Of A Registry Key
    Dim tmpVal As String        ' Tempory Storage For A Registry Key
    Value
    Dim KeyValSize As Long      ' Size Of Registry Key Variable
    '-----
    ' Open RegKey Under KeyRoot {HKEY_LOCAL_MACHINE...}
    '-----
    rc = RegOpenKeyEx(KeyRoot, KeyName, 0, KEY_ALL_ACCESS, hKey) ' Open Registry Key

    If (rc <> ERROR_SUCCESS) Then GoTo GetKeyError                ' Handle Error...

    tmpVal = String$(1024, 0)                                     ' Allocate Variable Space
    KeyValSize = 1024                                           ' Mark Variable Size

    '-----
    ' Retrieve Registry Key Value...
    '-----
    rc = RegQueryValueEx(hKey, SubKeyRef, 0, _
        KeyValType, tmpVal, KeyValSize) ' Get/Create Key Value

    If (rc <> ERROR_SUCCESS) Then GoTo GetKeyError                ' Handle Errors

    If (Asc(Mid(tmpVal, KeyValSize, 1)) = 0) Then                 ' Win95 Adds Null Terminated String.
..
        tmpVal = Left(tmpVal, KeyValSize - 1)                   ' Null Found, Extract From String
    Else                                                         ' WinNT Does NOT Null Terminate Stri
ng...
        tmpVal = Left(tmpVal, KeyValSize)                       ' Null Not Found, Extract String Onl
y
    End If

    '-----
    ' Determine Key Value Type For Conversion...
    '-----
    Select Case KeyValType
    Case REG_SZ
        KeyVal = tmpVal
    Case REG_DWORD
        ' Search Data Types...
        ' String Registry Key Data Type
        ' Copy String Value
        ' Double Word Registry Key Data Type

```

frmAbout - 3

```
    For i = Len(tmpVal) To 1 Step -1           ' Convert Each Bit
        KeyVal = KeyVal + Hex(Asc(Mid(tmpVal, i, 1))) ' Build Value Char. By Char.
    Next
    KeyVal = Format$("&h" + KeyVal)           ' Convert Double Word To String
End Select

GetKeyValue = True                           ' Return Success
rc = RegCloseKey(hKey)                       ' Close Registry Key
Exit Function                                ' Exit

GetKeyError:                                ' Cleanup After An Error Has Occured...
    KeyVal = ""                               ' Set Return Val To Empty String
    GetKeyValue = False                      ' Return Failure
    rc = RegCloseKey(hKey)                   ' Close Registry Key
End Function
```

Module1 - 1

```
Public Type namedata
    first As String * 15
    last As String * 20
    number As Long
    delflag As Boolean
End Type
```