

آموزش کار با کنترل TreeView

بخش اول: تعاریف مربوط به درخت

نام	توضیح
گره (Node)	بهترین ایده این است که یک گره را مانند یک شاخه منشعب در درخت فرض کنیم. هر گره نشاندهنده یک عضو است. یک گره می تواند هر تعداد زیر گره را نیز داشته باشد.
برادر (Sibling)	یک برادر گره دیگری است که در همان انشعاب گره مذکور قرار دارد
فرزندان (Children)	فرزندان به زیر گره های گره جاری گفته می شود. بعضی مواقع به آنها فرزند هم گفته می شود.
پدر (Parent)	پدر گره ای است که در بالای گره جاری قرار دارد. همه گره ها به جز گره ریشه دارای گره پدر می باشند.

بخش دوم: اضافه کردن گره به درخت

در این بخش شما به کنترل های زیر نیاز خواهید داشت:

نوع شی	نام شی	توضیحات
TreeView	TreeView1	این کنترل TreeView در همه مثالها استفاده خواهد شد
ImageList	ImageList1	این کنترل تصاویری برای استفاده در گره ها را در خود نگهداری می کند. ۳ تصویر را مطابق جدول زیر در ImageList قرار دهید.

کلید	تصویر
Closed	تصویر Folder بسته
Open	تصویر یک Folder باز شده
Leaf	تصویر یک فایل

کنترل Treeview تعدادی گره را در خود نگه می دارد بنابراین برای اضافه کردن گره به درخت باید یک گره جدید را ابتدا تعریف کنید و سپس آن گره را به مجموعه Treeview اضافه کنید. مانند این مثال:

```
Dim nodx As Node
```

```
    اضافه کردن گره ریشه  
Set nodx = TreeView1.Nodes.Add(, , "Root", "Root Node")
```

آن را اجرا کنید. شما باید فقط یک عضو را در درخت بنام "Root Node" ببینید. دقت کنید که خاصیت Parent در مثال فوق خالی بود. این مسئله برای این است که می خواستیم گره مذکور بالاترین گره یا ریشه باشد، و بنابراین نباید هیچ پدری داشته باشد. اجازه دهید تعدادی گره به درختمان اضافه کنیم، امالین دفعه تصاویری را نیز به گره هایمان اضافه خواهیم کرد:

```
Dim nodx As Node
```

```
Set TreeView1.ImageList = ImageList1
```

```
'Add root Node  
Set nodx = TreeView1.Nodes.Add(, , "Root", "Root Node","Closed")
```

```
'Expand root node so we can see what's under it
```

```

nodx.ExpandedImage = "Open"
nodx.Expanded = True

'Create a child node under the root node
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child1", "Child node
1", "Closed")

'Expand this node so we can see what's under it
nodx.ExpandedImage = "Open"
nodx.Expanded = True

'Create several more children
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child2", _
    "Child node 2", "Leaf")
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child3", _
    "Child node 3", "Leaf")
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child4", _
    "Child node 4", "Leaf")
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child5", _
    "Child node 5", "Leaf")

'Create two child nodes under the first child node of root
Set nodx = TreeView1.Nodes.Add("Child1", tvwChild, "Child1A", _
    "Child node 1 A", "Leaf")
Set nodx = TreeView1.Nodes.Add("Child1", tvwChild, "Child1B", _
    "Child node 1 B", "Leaf")

```

حالا اگر به کد دقت کنید می بینید ما یک ارجاع داریم که گره را در خودش ذخیره می کند و nodx نام دارد. شما می توانید این ارجاع را برای اصلاح خواص آن گره استفاده کنید. مانند خطوط که بصورت زیر آمده اند:

nodex.expanded=True

کاری که این خط انجام می دهد باز کردن یک گره است بنابراین ما می توانیم گره های فرزند زیر آن را مشاهده کنیم. آن مانند وقتی است که کاربر روی یک گره کلیک می کند تا خودشان را باز کنند. این کار بوسیله این کد قابل انجام است.

ادامه دهید و آن را اجرا کنید. شما خواهید دید که گره ریشه اصلی حالا ۵ فرزند در زیر دارد و اولین آنها دو فرزند در زیر دارد.

بخش سوم: پیمایش درخت

در اینجا نمونه هایی برای پیمایش درخت برای پیدا کردن همه گره های زیر یک گره خاص را آورده ایم. مثال زیر مثال ساده ای است که به شما نشان می دهد چگونه همه گره های زیر یک گره خاص را بدست آورید و متن برجسب آنها را نمایش دهید:

```

Dim nodx As Node

Set TreeView1.ImageList = ImageList1

'Add root Node
Set nodx = TreeView1.Nodes.Add(, , "Root", "Root Node", "Closed")

'Expand root node so we can see what's under it
nodx.ExpandedImage = "Open"
nodx.Expanded = True

```

```

'Create a child node under the root node
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child1", _
    "Child node 1", "Closed")

'Expand this node so we can see what's under it
nodx.ExpandedImage = "Open"
nodx.Expanded = True

'Create several more children
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child2", _
    "Child node 2", "Leaf")
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child3", _
    "Child node 3", "Leaf")
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child4", _
    "Child node 4", "Leaf")
Set nodx = TreeView1.Nodes.Add("Root", tvwChild, "Child5", _
    "Child node 5", "Leaf")

'Create two child nodes under the first child node of root
Set nodx = TreeView1.Nodes.Add("Child1", tvwChild, "Child1A", _
    "Child node 1 A", "Leaf")
Set nodx = TreeView1.Nodes.Add("Child1", tvwChild, "Child1B", _
    "Child node 1 B", "Leaf")

'Loop though each child of the root node
Dim i As Long

'Set nodx to the first child node of root.
Set nodx = TreeView1.Nodes("Root").Child

'Loop though each child nod assigning it to nodx
For i = 1 To TreeView1.Nodes("Root").Children
    MsgBox nodx.Text
    Set nodx = nodx.Next
Next

```

فقط بلوک آخر کد در این مثال جدید است. شما می توانید خاصیت Children را برای فهمیدن تعداد گره های فرزند یک گره خاص بکار ببرید. خاصیت Child به اولین گره فرزند اشاره می کند و خاصیت Next به گره بعدی نسبت به گره جاری اشاره می کند.

بخش چهارم: رویدادها

الان باید درباره چیزهای جدیدی صحبت کنیم. برای این مثال ما می خواهیم فقط دو گره را که مانند دایرکتوری در سیستم شما هستند به آن اضافه کنیم. وقتی شما روی گره کلیک کنید، مسیر واقعی فایل برای آن فولدر را به شما می دهد:

```

Private Sub Form_Load()
    Dim nodx As Node

    Set TreeView1.ImageList = ImageList1

    'Add Drive
    Set nodx = TreeView1.Nodes.Add(, , , "c:", "Closed")
    nodx.ExpandedImage = "Open"
    nodx.Expanded = True

    'Add Folder
    Set nodx = TreeView1.Nodes.Add(nodx, tvwChild, , "Windows", "Closed")
    nodx.ExpandedImage = "Open"

```

```

nodx.Expanded = True

'Add Another Folder
Set nodx = TreeView1.Nodes.Add(nodx, tvwChild, , "System", "Closed")
nodx.ExpandedImage = "Open"
nodx.Expanded = True

End Sub

Private Sub TreeView1_NodeClick(ByVal Node As MSComctlLib.Node)
    MsgBox Node.FullPath
End Sub

```

وقتی ما روی یک گره در درخت کلیک کنیم، رویداد NodeClick روی می دهد و آن گره ای را که کلیک شده بود را به ما برمی گرداند، بنابراین دستکاری آن بسیار آسان است. به نظر شما چیز خطرناکی در مثال فوق وجود دارد؟ این صحیح است در اینجا هیچ کلیدی برای هر کدام از گره ها وجود ندارد. کلید اختیاری است. دقت کنید که چون من کلیدی ندارم بجای فقط کلید گره که در تابع add برای اضافه کردن یک گره به درخت استفاده می شود، خود گره را بدست می آورم.

این علت ارسال شی nodx به تابع add می باشد، چون آن واقعاً فقط یک اشاره گر به گره قبلی است. شما باید قادر باشید که در مثال فوق ببینید که کنترل Treeview بخوبی برای نمایش سیستم فایل کار می کند، مانند اکسپلورر در ویندوز، خاصیت FullPath هر چیزی را که ما درباره محل فایل نیاز داریم را به ما خواهد داد. در اینجا یک مثال از رویدادهای NodeClick، Expand و Collapse است:

```

Private Sub Form_Load()
    Dim nodx As Node
    Dim nodr As Node

    'Show Root Lines
    TreeView1.LineStyle = tvwRootLines

    'Display Checkboxes
    TreeView1.Checkboxes = True

    'Add Items
    Set nodx = TreeView1.Nodes.Add(, , , "Item 1")
    Set nodx = TreeView1.Nodes.Add(, , , "Item 2")
    Set nodx = TreeView1.Nodes.Add(, , , "Item 3")
    Set nodx = TreeView1.Nodes.Add(, , , "Item 4")
    Set nodx = TreeView1.Nodes.Add(, , , "Item 5")
    nodx.Expanded = True

    Set nodr = TreeView1.Nodes.Add(nodx, tvwChild, , "Item 6")
    nodr.Expanded = True

    Set nodx = TreeView1.Nodes.Add(nodr, tvwChild, , "Item 7")
    Set nodx = TreeView1.Nodes.Add(nodr, tvwChild, , "Item 8")
    Set nodx = TreeView1.Nodes.Add(nodr, tvwChild, , "Item 9")
    Set nodx = TreeView1.Nodes.Add(nodr, tvwChild, , "Item 10")
    Set nodx = TreeView1.Nodes.Add(nodr, tvwChild, , "Item 11")

    Set nodx = TreeView1.Nodes.Add(, , , "Item 12")
End Sub

Private Sub TreeView1_Collapse(ByVal Node As MSComctlLib.Node)
    MsgBox "Collapsing node: " & Node.Text
End Sub

```

```
Private Sub TreeView1_Expand(ByVal Node As MSComctlLib.Node)
    MsgBox "Expanding node: " & Node.Text
End Sub

Private Sub TreeView1_NodeCheck(ByVal Node As MSComctlLib.Node)
    If Node.Checked Then
        MsgBox "Node " & Node.Text & " was checked"
    Else
        MsgBox "Node " & Node.Text & " was Unchecked"
    End If
End Sub
```

آن را اجرا کنید و روی چک باکسها کلیک کنید. شما باید پیغامی را دریافت کنید که به شما می گوید که این گره شما فقط چک خورده است. حالا سعی کنید بعضی از گره ها را جمع کنید و یا باز کنید و در این حالتها شما پیغامی را از رویدادهای Expand و Collapse دریافت خواهید کرد.