

Introduction

In Visual Basic, adding menus at runtime is handled in the exact same way as adding any control at runtime. First you have to create a control array at design time, then at runtime you just load the new controls into the array. In this tutorial, I will show you all the steps to handle this. I will first show you how to set up a menu control array at design time. Next I will show you how to add more menus to your array at runtime. Finally, I will show you how to handle the click events of all the menus you add at runtime.

I encourage you to open up a project in Visual Basic, and follow along with me as I introduce you to this topic. Also, if my explanations just aren't making sense, please download the code from the link at the right. This should help you understand everything I am trying to say.

Design Time

So, how do you create a menu control array at design time? Simple, first right click any place on your form where there are no controls. Then choose Menu Editor... Now in the editor add a main menu, give it a caption such as Favorites and a name such as mnuFav. Then click next, and then the right arrow button so that the new menu will appear under it. Now give this menu a fake caption for now like My Caption and give it a name such as mnuFavLinks. Finally set its index property to 0. By setting its index property, you tell VB that you want this menu to be the first element in the control array. That's all you have to do at design time.

Run Time

Now, at runtime to add a new menu to the menu control array you simply use the command: `Load mnuFavsLinks(index)` where index is the next number in the control array. For example I want three favorites under my Favorites menu. I want them to show up as Favorite 1, Favorite 2, and Favorite 3. This is the code I would use in the form load event:

```
Private Sub Form_Load()  
    mnuFavLinks(0).caption = "Favorite 1"  
    Load mnuFavLinks(1)  
    Load mnuFavLinks(2)  
    MnuFavLinks(1).Caption = "Favorite 2"  
    mnuFavLinks(2).Caption = "Favorite 3"  
End Sub
```

Handling Clicks

Incase you don't know how control arrays work; all three of my favorites above will have the same event procedure associated with them. So in my example above the mnuFavLinks_Click event would fire for all three Favorites, and the only way to tell them apart is to check their index, which gets passed into the event procedure. The example below will show you what I mean. All it does is when you click on one of the favorites, a message box will appear telling you its index and its caption.

```
Private Sub mnuFavLinks_Click(Index As Integer)
    MsgBox Index & ": " & mnuFavLinks(Index).Caption
End Sub
```