Changing the Text of a Menu Command

原文

https://docs.microsoft.com/zh-cn/visualstudio/extensibility/changing-the-text-of-a-menu-command

本文内容

Changing a menu command label with the IMenuCommandService

The following steps show how to change the text label of a menu command by using the [IMenuCommandService](https://msdn.microsoft.com/library/system.componentmodel.design.imenucommandservice(v=vs.110).aspx) service.

Changing a menu command label with the IMenuCommandService

**1.** Create a VSIX project named MenuText with a menu command named **ChangeMenuText**. For more information, see [Creating an Extension with a Menu Command](https://docs.microsoft.com/zh-cn/visualstudio/extensibility/creating-an-extension-with-a-menu-command).

**2.** In the .vstc file, add the TextChanges flag to your menu command, as shown in the following example.

<Button guid="guidChangeMenuTextPackageCmdSet" id="ChangeMenuTextId" priority="0x0100" type="Button">

<Parent guid="guidChangeMenuTextPackageCmdSet" id="MyMenuGroup" />

<Icon guid="guidImages" id="bmpPic1" />

<CommandFlag>TextChanges</CommandFlag>

<Strings>

<ButtonText>Invoke ChangeMenuText</ButtonText>

</Strings>

</Button>

**3.** In the ChangeMenuText.cs file, create an event handler that will be called before the menu command is displayed.

private void OnBeforeQueryStatus(object sender, EventArgs e)

{

var myCommand = sender as OleMenuCommand;

if (null != myCommand)

{

myCommand.Text = "New Text";

}

}

You can also update the status of the menu command in this method by changing the Visible, Checked, and Enabled properties on the OleMenuCommand object.

**4.** In the ChangeMenuText constructor, replace the original command initialization and placement code with code that creates a [OleMenuCommand](https://msdn.microsoft.com/en-us/library/microsoft.visualstudio.shell.olemenucommand(v=vs.140).aspx) (rather than a MenuCommand) that represents the menu command, adds the [BeforeQueryStatus](https://msdn.microsoft.com/en-us/library/bb130694(v=vs.140).aspx) event handler, and gives the menu command to the menu command service.

Here is what it should look like:

private ChangeMenuText(Package package)

{

if (package == null)

{

throw new ArgumentNullException(nameof(package));

}

this.package = package;

OleMenuCommandService commandService = this.ServiceProvider.GetService(typeof(IMenuCommandService)) as OleMenuCommandService;

if (commandService != null)

{

CommandID menuCommandID = new CommandID(MenuGroup, CommandId);

EventHandler eventHandler = this.ShowMessageBox;

OleMenuCommand menuItem = new OleMenuCommand(ShowMessageBox, menuCommandID);

menuItem.BeforeQueryStatus +=

new EventHandler(OnBeforeQueryStatus);

commandService.AddCommand(menuItem);

}

}

**5.** Build the project and start debugging. The experimental instance of Visual Studio appears.

**6.** On the **Tools** menu you should see a command named **Invoke ChangeMenuText**.

**7.** Click the command. You should see the message box announcing that MenuItemCallback has been called. When you dismiss the message box, you should see that the name of the command on the Tools menu is now **New Text**.