

Win32 Window Handles - Creating a Handle Child Windows viewer

Starting with a simpler variation of the [Util - Windows Handles - View Handle Screenshot.h2](#) script

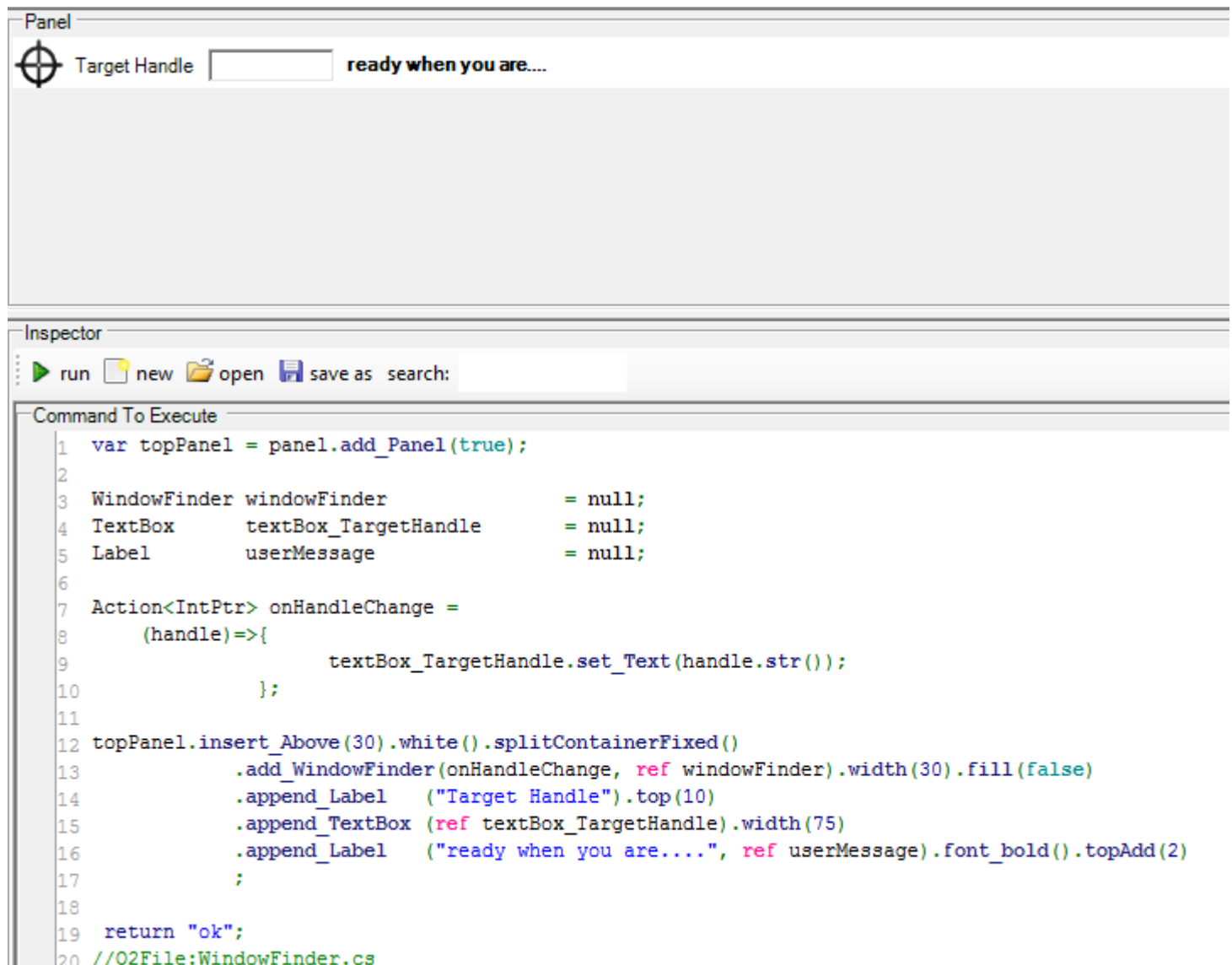
```
var topPanel = panel.add_Panel(true);

WindowFinder windowFinder          = null;
TextBox      textBox_TargetHandle  = null;
Label        userMessage          = null;

Action<IntPtr> onHandleChange =
    (handle)=>{
        textBox_TargetHandle.set_Text(handle.str());
    };

topPanel.insert_Above(30).white().splitContainerFixed()
    .add_WindowFinder(onHandleChange, ref windowFinder).width(30).fill(false)
    .append_Label("Target Handle").top(10)
    .append_TextBox(ref textBox_TargetHandle).width(75)
    .append_Label("ready when you are...", ref userMessage).font_bold().topAdd(2)
    ;

return "ok";
//O2File:WindowFinder.cs
```



adding a TreeView (and some refactoring)

```
var topPanel = panel.add_Panel(true);

WindowFinder windowFinder      = null;
TextBox        handleValue     = null;
Label          userMessage     = null;
TreeView       handleTree      = null;

Action<IntPtr> onHandleChange =
    (handle)=>{
        handleValue.set_Text(handle.str());
    };

topPanel.insert_Above(30).splitContainerFixed().white()
    .add_WindowFinder(onHandleChange, ref windowFinder)
    .append_Label    ("Target Handle:",10)
    .append_TextBox  (ref handleValue).width(75)
    .append_Label    ("ready when you are...", 10, ref userMessage).bold();

topPanel.add_TreeView(ref handleTree);

return "done";
//O2File:WindowFinder.cs
```

The screenshot displays a software development environment with two main components: a GUI window and a code editor.

GUI Window: The window has a title bar labeled "Panel". Inside, there is a label "Target Handle:" followed by a text input field. To the right of the input field is the text "ready when you are....".

Code Editor: The editor shows the C# code from the previous block, with line numbers 1 through 22. The code is syntax-highlighted. The file path at the bottom is "//O2File:WindowFinder.cs".

Adding a TreeNode with the selected handle (and its className and text)

```

var topPanel = panel.add_Panel(true);

WindowFinder windowFinder      = null;
TextBox        handleValue     = null;
Label          userMessage     = null;
TreeView       handleTree      = null;

Action<IntPtr> onHandleChange =
    (handle)=>{
        handleValue.set_Text(handle.str());
        var handleClass = handle.className();
        var handleText  = handle.get_ControlText();
        var nodeText = "{0} : {1} : {2}".format(handle,
        handleClass,handleText);
        handleTree.clear()
                    .add_Node(nodeText);
    };

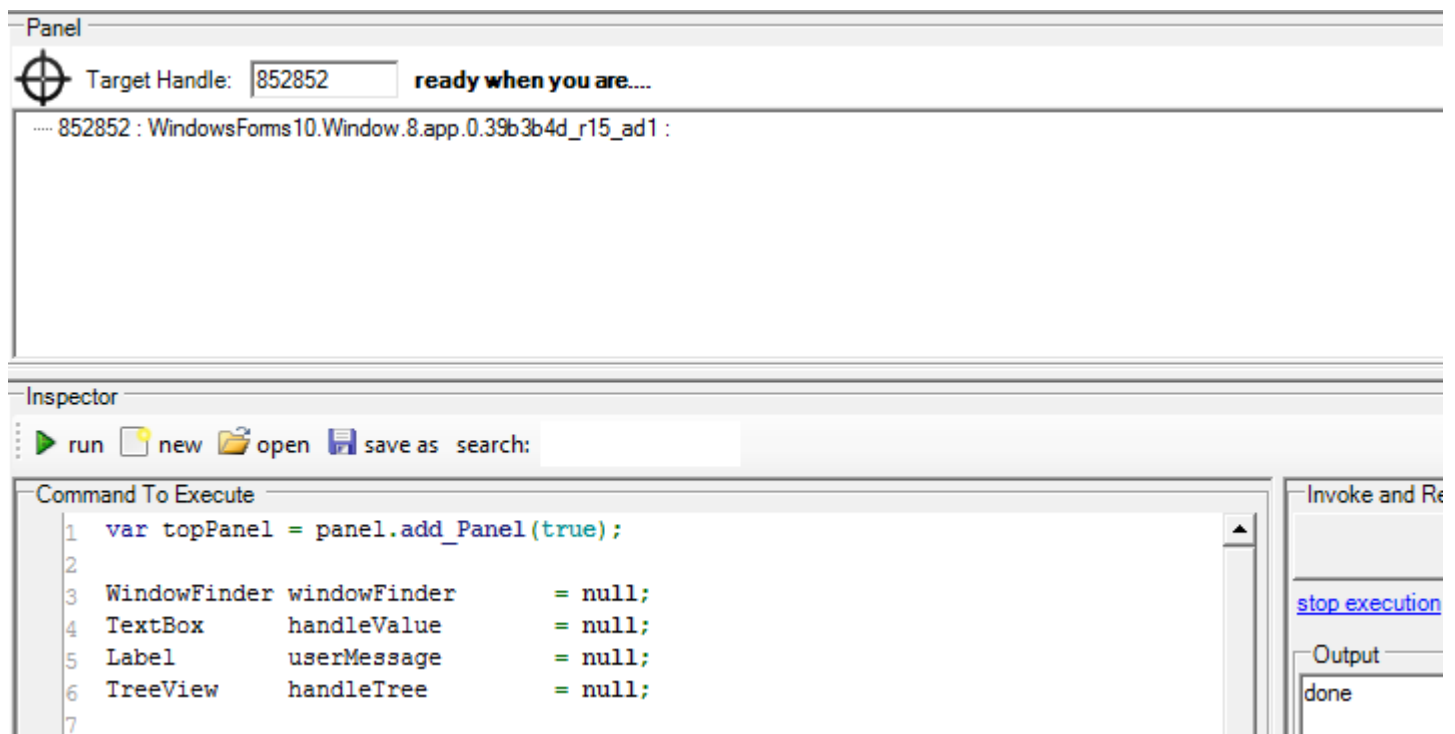
topPanel.insert_Above(30).splitContainerFixed().white()
    .add_WindowFinder(onHandleChange, ref windowFinder)
    .append_Label("Target Handle:",10)
    .append_TextBox (ref handleValue).width(75)
    .append_Label("ready when you are...", 10, ref userMessage).bold();

topPanel.add_TreeView(ref handleTree);

return "done";
//O2File:WindowFinder.cs

//O2File:_Extra_methods_To_Add_to_Main_CodeBase.cs

```



Adding a TreeView with the child handles

```

//var topPanel = panel.add_Panel(true);
var topPanel = "Util - Windows Handle Viewer (with Child Windows)".popupWindow(400,300);

WindowFinder windowFinder      = null;
TextBox        handleValue     = null;
Label          userMessage     = null;
TreeView       handleTree      = null;
IntPtr        handleInThreeView = IntPtr.Zero;

Action<IntPtr,TreeNode> addHandleToNode =
    (handle, treeNode) =>{
        handleValue.set_Text(handle.str());

```

```

(handleClass,handle, handleText);

handle.hasChilds());

};

Action<IntPtr> onHandleChange =
    (handle)=>{
        handleTree.clear();
        addHandleToNode(handle, handleTree.rootNode());
        handleTree.selectFirst().expand();
    };

topPanel.insert_Above(30).splitContainerFixed().white()
    .add_WindowFinder(onHandleChange, ref windowFinder)
    .append_Label("Target Handle:",10)
    .append_TextBox(ref handleValue).width(75)
    .append_Label("ready when you are...", 10, ref userMessage).bold();

topPanel.add_TreeView(ref handleTree);

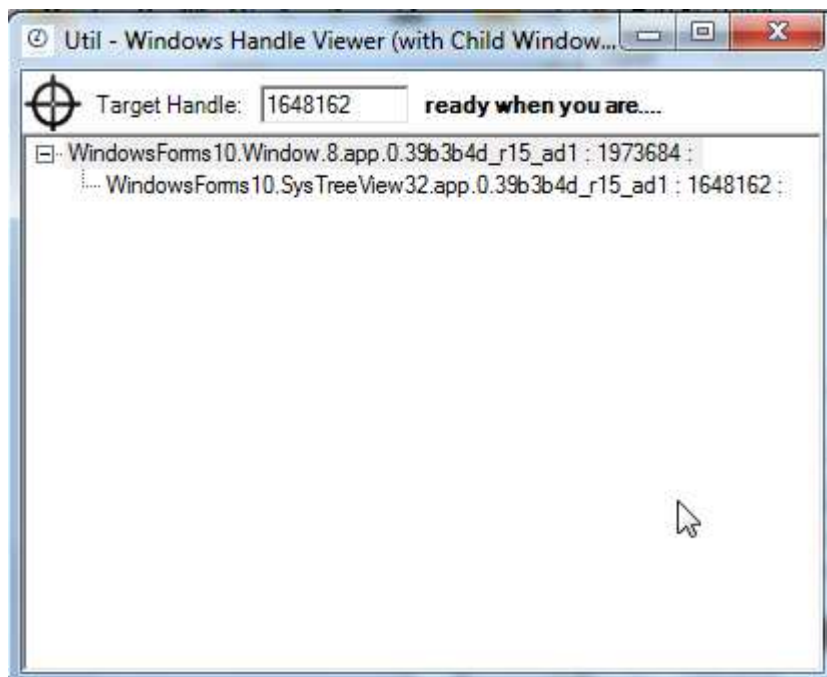
handleTree.beforeExpand<IntPtr>(
    (treeNode,handle)=> {
        foreach(var childHandle in handle.child_Windows())
            addHandleToNode(childHandle, treeNode);
    });

handleTree.afterSelect<IntPtr>(
    (handle) => {
        handleInTreeView.window_Redraw();
        handleInTreeView = handle.window_Highlight();
    });

onHandleChange(topPanel.handle());

return "done";
//O2File:WindowFinder.cs

```



Viewing the handles in Write.exe

