

A Study on Teaching Methods of Scope in VB6.0

Xiaoping Guo

School of Energy and Power Engineering, Dalian
University of Technology
Dalian, Liaoning Province, China
gxp190@aliyun.com

Yichao Wu

School of Energy and Power Engineering, Dalian
University of Technology
Dalian, Liaoning Province, China
15242606605@163.com

Abstract--VB6.0 is an object-oriented programming language. Scope is a part of the important concepts in programming. Treating variables declared in the universal declaration section of the form with different programming angle is the key is to elaborate the knowledge of variable scope in VB6.0. Better learning outcomes can be achieved by analyzing from OOP perspective. In addition, from the perspective of OOP, the nature of form-level public variables and standard controls which can be cited by the code are the same, they are visible throughout the application.

Keywords--OOP, VB6.0, variables, scope.

I. INTRODUCTION

Visual Basic 6.0 (VB6.0) is the more popular version of Visual Basic. So far, the majority of universities and colleges in China take VB programming courses for non-computer majors on VB6.0 platform.

The scope is an important concept of programming in VB learning. Usually we illustrate the knowledge points of scope with a variable as an example^[1-3]. For VB6.0, variables are divided into four parts by scope: local variables (process-level variables), module-level variables and global variables. This article will discuss the scope of variables.

II. THE BASIC CONCEPT OF OOP

In OOP (Object Oriented Programming), the things studied are called the objects. The objects which have similar characteristics and behaviors are combined into classes. Thus, in OOP, class is an abstract object and the object is an instance of class.

Object has characteristics and behaviors. In OOP, the characteristics of an object called its properties, described by data, the behaviors of an object called its methods, process (function) to represent. For visual programming, another technique to describe the object behavior is the event. The event is what the object corresponded to the user's operation.

The three characteristics of OOP are encapsulation, inheritance and polymorphism, however, VB6.0 does not have the inheritance features. To illustrate the scope, this article concerns the encapsulation of OOP.

According to OOP's encapsulation characteristic, object's properties and methods are always defined as private or public (this article does not consider other access levels). Private properties and methods can only be seen in the class, while public properties and methods are visible

outside it, that is to say, anywhere in the application we can access public properties or methods of the object. The syntaxes are:

Object name. Attribute name (1)

Object name. Method name (2)

1. VB6.0 PROGRAMMING FEATURES

VB6.0 is object-oriented programming language. All entities appear in the form of object and comply with the rules of encapsulation.

For non-computer professionals we generally do not introduce custom class [1-3]. So in VB6.0, subjects are divided into two categories: form classes and control classes.

A Standard controls

Controls in the toolbox provided by the system are called the standard control class, the resulting control objects belong to standard control objects, referred to as controls. Syntax used to access the properties of the control in the program is:

Controls name. Attribute name (3)

Syntax used to call control is:

Object name. Method name (4)

Events to call the standard control are usually selected from the event set in the code window.

In fact, (3) is (1), (4) is (2). But in VB6.0 learning, we generally only describe how to use them without explaining why such use. Because what can access properties, methods or events of controls are not concerned. Obviously, we are outside the control to gain access to the properties and methods. Therefore, for the control class itself, these attributes, methods are public.

B Form

Form is the basis to create a VB application. Each form established is an instance of the form class, a form class object. Since the form is an object of the form class, so the form object has properties, methods and events.

VB6.0 not only provides a form of default properties, methods and events, and allows users to add properties and methods to the form.

Method to add properties to the form: defining variables in the universal declaration section of the form.

The way of adding methods to the form: defining process (a process or sub-process of function) in the statement area of the form.

While defining variables and process methods, users can separately define them as private and public:

Using keywords Public to define, methods or process attributes variables are public and can be accessed anywhere in the application.

Using keywords Private (or Dim) to define, methods or process attributes variables are private and can be accessed only within the class.

As a form is an object, so if users want to access the public properties and methods of the form on the outside, the syntax is exactly the same as the syntax which used to access the properties and methods of controls, follow (3) and (4). That means accessing the custom attributes of a form, accessing the caption property and access attributes of a control "Text1.text" are of the same form, because they are public properties of the object. However, most VB6.0 textbooks didn't elaborate this essential content, and separate custom attributes of forms for discussion from the knowledge points of scope. It seems that the custom public properties and the default properties are two different things, it will mislead learners.

III. PROPERTIES AND SHARED DATA OF OBJECTS

Most VB6.0 textbooks emphasized defining variables in the universal declaration section of the form, the purpose is to share data various processes in the form, but they did not come forth from the perspective of OOP to elaborate that this variable is a property of the object.

For example, now there is a 5×5 matrix, using picture box P1 to display the matrix, picture box P2 to display its upper triangular matrix and picture box P3 to display its lower triangular matrix.

The following is the idea of shared data. As P1, P2 and P3 display data with a same matrix, so state a two-dimensional array of 5×5 in the universal declaration section of the form shared by P1, P2 and P3. For example, the following program:

```
Private a(1 to 5, 1 to 5) As integer
'Declare form module-level variables
Dim a(1 to 5, 1 to 5) As integer
Private Sub P1_Click()      'Sub-process P1 of the
form module
Print matrix "a"          'Shared form-level variables
End Sub
Private Sub P2_Click()      'Sub-process P2 of the
form module
'Shared form-level variables
Print upper triangular matrix of "a"
End Sub
Private Sub P3_Click()      'Sub-process P3 of the
form module
'Shared form-level variables
Print lower triangular matrix of "a"
End Sub
```

This way of thinking belongs to process-oriented programming method.

Analyze this problem from the perspective of OOP. This form has a two-dimensional array attribute and three acts: printing matrix, printing upper triangular matrix and printing lower triangular matrix. Therefore, program design idea is that:

```
Class Form1
'Declared property of the form
Private a(1 to 5, 1 to 5) As integer
Dim a(1 to 5, 1 to 5) As integer
'Event of Picture box P1 complete the form
behavior1
Private Sub P1_Click()
Print matrix "a"      'Access its own properties
End Sub
'Event of Picture box P2 complete the form
behavior2
Private Sub P2_Click()
'Access its own properties
Print upper triangular matrix of "a"
End Sub
'Event of Picture box P3 complete the form
behavior3
Private Sub P3_Click()
'Access its own properties
Print lower triangular matrix of "a"
End Sub
End Class
```

VB is an object-oriented programming language. This paper argues that analyzing problems from the perspective of OOP can reflect the VB program design features better.

IV. SCOPE

Scope means the area that can be accessed. Elaborate knowledge points of scope with the example of variable. In VB6.0, global variables have two main points. First, the scope of a global variable is the entire application and can be accessed by any process of the application. Second, the value of a global variable will not disappear or be re-initialized throughout the application unless the entire application execution ends.

A. Define variables in standard modules with Public

Variables defined in the universal declarations section of a standard module by using Public are global variables. It conforms to two characteristics of global variables: users are able to access the variable at any location. The variable of the application and the variable will not be initialized before the end of the application.

B. Public Properties of Objects

Attributes are data. Attribute name is the name of the object's variable. Almost all standard control properties are public and visible in any place of the application.

But the lifetime of an object's property is the lifetime of the object itself. When the object is destroyed, the attribute value will disappear.

In this paper, several small program segments will be used to illustrate the characteristics of object properties.

1) Start VB6.0, the system creates Form1 automatically, this form is the default startup form.

2) Create Form2 and create a command button Command1 and a text box control Text1.

a) Define a public variable in the universal declaration section of Form2

Public number1 As Integer

b) The following events in Form2

Private Sub Form_Load()

Number1=234 'Initialize the properties

of Form2

'Initialize the properties of the control in

Form2

Text1.Text= " "

End Sub

C. Set up a command button named "Display Data" in Form1. Design events for the command button:

- Event One

Private Sub Display Data_Click()

Analysis: Cite the public property number1 of Form2 in Form1, because we have loaded Form2 before cited, number1 is assigned, so the value of number1 displayed is 234.

- Event Three

Add a statement in Event Two to destroy Form2:

Private Sub Display Data_Click()

Form2.Show 'Load and show Form2

Set Form2 = Nothing 'Destroy Form2

'Quote the public properties of Form2 in Form1

Print Form2.number1

End Sub

Run the program, click on the "Display Data" button in Form1, Form2 shows and Form1 shows: 0.

Analysis: Compare Event Two of Event Three, steps are exactly the same: After Form2 is loaded and displayed, destroy Form2 object, so the property values of Form2 object are all emptied.

Conclusion: The scope of the public property of the object is the entire application. However, when the application is running, there can be a variety of factors may re-initialize the value of the public property of the object.

V. LOAD FORMS THROUGH ACCESSION CONTROLS

Multiple forms are one of the main contents of VB programming. There are usually two methods to load the specified form:

3) *Method One: Use the Load statement.* The syntax is: Load Form name

4) *Method Two: Use syntax Show.* The syntax is: Form Name.Show.

The purpose of using the show method is to display the form. However, the form must be loaded before being displayed.

The scope of the control property is the entire application. So users are able to load a form through accessing the properties of controls in it. For example:

'Quote the public properties of Form2 in Form1

Print Form2.number1

End Sub

Run the program, click on the "Display Data" button in Form1, form shows: 0

Analysis: Cite the public property number1 of Form2 in Form1, as number1 is not assigned before, so it shows the default value of number1.

- Event Two

Load code of Form2 in event One:

Private Sub Display Data_Click()

'Load and display Form2 or use "Load Form2" only to load

Form2.Show

'Quote the public properties of Form2 in Form1

Print Form2.number1

End Sub

Run the program, click on the "Display Data" button in Form1, form shows: 234.

- Event Four

Private Sub Display Data_Click()

'Assign properties of the control in Form2

Form2.Text1.Text= "good"

Print Form2.number1

End Sub

Run the program, click on the "Display Data" button in Form1, Form1 displays: 234.

Analysis: The function of the statement: Form2.Text1.Text = "good" is to operate the control in Form2. Form2 must be loaded before the operation and then you can perform the specified statement. The key is that loading Form2 is performing the Load event of it, initializing the variable of number1.

VI. CONCLUSION

VB6.0 is an object-oriented programming language. Scope is an important concept in programming. But in VB6.0 learning, process-oriented methods are often used when elaborating the knowledge points. This may not be clear enough. This paper argues that the understanding of OOP techniques should be emphasized to solve this problem.

Creating a form is implementing an instantiation of the form class. We can add properties and methods to the form. Declaring a variable in the form module is adding a property for the form. If the property is declared by syntax Public, the scope and default attributes of it are the same as those of standard controls, they can be seen in the whole application.

Whether a property is public or private, it can be seen everywhere within the object.

Global variables can only be declared in the universal declarations section of a standard module with syntax Public.

REFERENCES

- [1] Tan, H.Q., Yuan, M., Xue, S.B.: Visual Basic Programming Design, 3rd ed. Tsinghua University Press, Beijing, 2012.
- [2] Gong, P.Z.: An Introduction to Visual Programming, 4th ed. Higher Education Press, Beijing, 2013.
- [3] Zhang, F.X., Zhang, W.M.: Visual Basic Programming Design. China Electric Power Press, 2013.