

***Polymorphism* Sample Help**

Sample Description: [Polymorphism](#)

Points of Interest

[Creating Polymorphic Methods](#)

[Executing Polymorphic Methods](#)

Control

Controls

For Help on Help, Press F1

Polymorphism

In object-oriented programming, polymorphism allows the same function name to be used by more than one object. The main benefits of polymorphism are that for a developer using an object library, it makes objects more independent of each other and allows new objects to be added with minimal changes to existing objects.

In the Polymorphism sample, there are several controls on the form. Each control contains a Demonstrate and Reset method. When the Demonstrate method is clicked, each control runs its own Demonstrate code. When the File/Reset menu entry is clicked, each control resets itself by running its corresponding Reset code.

Creating Polymorphic Methods

To create a Polymorphic method on an object, you must select the control, then create a method on that control. The following example shows how this was done for the Label control in the upper left hand corner of the form.

```
Sub Demonstrate
    Caption = "I am a Label control."
    ForeColor = 255
End Sub
```

As you can see, this method appears only on the object itself and is not associated with the form. The setting of the Caption property is `Caption = "I am a Label control."` instead of `PolymorphismForm.LblControl.Caption = "I am a Label control."` The Reset method on the Label control is defined as follows:

```
Sub Reset
    Caption = "Label"
    ForeColor = -1
End Sub
```

Executing Polymorphic Methods

In Envelop it is very easy to execute a method across all controls on a form. The Demonstrate button on the form runs the following program code.

```
Sub BtnDemonstrate_Click()  
    Controls.Demonstrate  
End Sub
```

Using "Controls" is a simplified way to send all controls on the form some type of instruction to execute. If the control has a corresponding method, the method is automatically executed, otherwise it is ignored. Because of this, each control is capable of handling its own code execution.

When you click the Demonstrate button, each of the controls on the form will execute some type of Demonstrate method and provide some type of unique demonstration of its capabilities. When you click the File/Reset menu entry, each control on the form will execute its corresponding Reset method to clear and reset itself.

