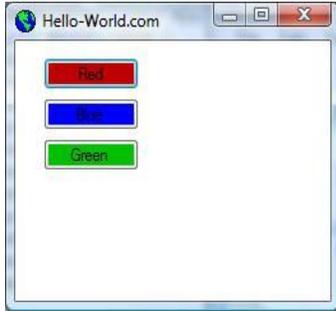


Sender

We have looked at the values of the argument `e`. Now we will look at the value of `Sender` and use it to improve our code.



Start a new project and name it **Sender**. Build the form as shown with the following buttons:

- **BtnRed** has a red background and the text is "Red"
- **BtnBlue** has a blue background and the text is "Blue"
- **BtnGreen** has a green background and the text is "Green"

Double click on `BtnRed` to open the code view window with `BtnRed_Click` selected. Change the name of the procedure to `Button_Click` and write the code as shown below:

```
Private Sub Button_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) _
    Handles BtnRed.Click, BtnGreen.Click, BtnBlue.Click
    Me.BackColor = sender.backcolor
    Me.Text = sender.text
End Sub
```

Run the program and click each button. The backcolor and text changes to match the button that was clicked.

Let's look at what is going on. First of all, notice the **Handles** clause after the arguments: instead of writing a procedure for each button, we are using one procedure and letting it handle the click event for each of the three buttons. The value of `sender` will be the button that was clicked.

Second, when we typed the code and typed `sender dot`, the list of properties for `sender` did not include `backcolor` and `text`. We had to type in the words `backcolor` and `text` ourselves.

If you know that **Sender** will be a button, you can change the type of `sender` to `button` by placing the cursor after **As** in **ByVal sender As** and type a space. The pop-up window will let you select **Button** as the type. Now when you type the dot after `sender`, the list will include all the properties of a button.