

```

Catch QuantityException As FormatException
    ' Handle a quantity exception.
    MessageBox.Show("Quantity must be numeric.", "Data Entry Error",
        MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
With QuantityTextBox
    .Focus()
    .SelectAll()
End With

Catch AnException As Exception
    ' Handle any other exception.
    MessageBox.Show("Error: " & AnException.Message)
End Try
End Sub

Private Sub ClearButton_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles ClearButton.Click
    ' Clear previous amounts from the form.

    TitleTextBox.Clear()
    PriceTextBox.Clear()
    ExtendedPriceTextBox.Clear()
    DiscountTextBox.Clear()
    DiscountedPriceTextBox.Clear()
    With QuantityTextBox
        .Clear()
        .Focus()
    End With
End Sub

Private Sub ExitButton_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles ExitButton.Click
    ' Exit the project.

    Me.Close()
End Sub
End Class

```

## Summary

1. Variables are temporary memory locations that have a name (called an *identifier*), a data type, and a scope. A constant also has a name, data type, and scope, but it also must have a value assigned to it when it is declared. The value stored in a variable can be changed during the execution of the project; the values stored in constants cannot change.
2. The data type determines what type of values may be assigned to a variable or constant. The most common data types are String, Integer, Decimal, Single, and Boolean.
3. Identifiers for variables and constants must follow the Visual Basic naming rules and should follow good naming standards, called *conventions*. An identifier should be meaningful and have the data type appended at the