

Chapter 12

Mathematical Functions

❖ Learning How to use the Mathematical functions

We have learned how to write code to perform mathematical calculations using standard mathematical operators. However, we need to use the built-in Math functions in VB2010 to handle complex mathematical calculations. Math functions are methods that belong to the Math Class of the .Net framework. They are similar to the math functions in Visual Basic 6. The Math functions in VB2010 are Abs, Exp, Fix, Int, Log, Rnd(), Round and the trigonometric functions.

12.1 The Abs function

The Abs returns the absolute value of a given number.

The syntax is

Math. Abs (number)

The **Math** keyword here indicates that the Abs function belongs to the Math class. However, not all mathematical functions belong to the Math class.

12.2 The Exp function

The Exp of a number x is the exponential value of x, i.e. e^x .

For example, $\text{Exp}(1)=e=2.71828182$

The syntax is

Math.Exp (number)

Example 12.2

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button1.Click
```

```
Dim num1, num2 As Single
```

```
num1 = TextBox1.Text
```

```
num2 = Math.Exp(num1)
```

```
Label1.Text = num2
```

```
End Sub
```

12.3 The Fix Function

The Fix function truncates the decimal part of a positive number and returns the largest integer smaller than the number. However, when the number is negative, it will return smallest integer larger than the number. For example, Fix (9.2)=9 but Fix(-9.4)=-9

Example 12.3

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button1.Click
```

```
Dim num1, num2 As Single
```

```
num1 = TextBox1.Text
```

```
num2 = Fix(num1)
```

```
Label1.Text = num2
```

```
End Sub
```

12.4 The Int Function

The Int is a function that converts a number into an integer by truncating its decimal part and the resulting integer is the largest integer that is smaller than the number. For example

$\text{Int}(2.4)=2$, $\text{Int}(6.9)=6$, $\text{Int}(-5.7)=-6$, $\text{Int}(-99.8)=-100$

12.5 The Log Function

The Log function is the function that returns the natural logarithm of a number. For example, $\text{Log}(10)=2.302585$

Example 12.4

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
```

```
Dim num1, num2 As Single
```

```
num1 = TextBox1.Text
```

```
num2 = Math.Log(num1)
```

```
Label1.Text = num2
```

```
End Sub
```

* Label1 shows the value of logarithm of num1.

12.6 The Rnd() Function

The Rnd is very useful when we deal with the concept of chance and probability. The Rnd function returns a random value between 0 and 1. Random numbers in their original form are not very useful in programming until we convert them to integers. For example, if we need to obtain a random output of 6 integers ranging from 1 to 6, which makes the program behave like a virtual dice, we need to convert the random numbers to integers using the formula $\text{Int}(\text{Rnd}*6)+1$.

Example 12.5

```

Private Sub Button1_Click (ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
Dim num as integer
Randomize ( )
Num=Int(Rnd()*6)+1

Label1.Text=Num

End Sub

```

In this example, `Int(Rnd*6)` will generate a random integer between 0 and 5 because the function **Int** truncates the decimal part of the random number and returns an integer. After adding 1, you will get a random number between 1 and 6 every time you click the command button. For example, let say the random number generated is 0.98, after multiplying it by 6, it becomes 5.88, and using the integer function `Int(5.88)` will convert the number to 5; and after adding 1 you will get 6.

12.7 The Round Function

The **Round** function rounds up a number to a certain number of decimal places. The Format is `Round (n, m)` which means to round a number `n` to `m` decimal places. For example, `Math.Round (7.2567, 2) =7.26`

Example 12.6

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
Dim num1, num2 As Single
num1 = TextBox1.Text
num2 = Math.Round(num1, 2)

```

```
Label1.Text = num2
```

```
End Sub
```

12.8 The Sqrt Function

The Sqrt function returns the square root of a number. For example, Sqrt(400) will return a value of 20. You can use this function to solve problems related to Pythagoras theorem. For exam, you may want to find the length of the hypotenuse given the length of the adjacent side and the length of the opposite side of a triangle. The code in VB2010 is:

```
c=Math.Sqrt(a^2+b^2)
```

*As Sqrt is a function that belongs to the Math class, we need to use the Math keyword.

The following code computes the hypotenuse c given the length of adjacent side and the length of the opposite side of triangle.

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
```

```
    Dim a, b, c As Single
```

```
    a = Val(TxtA.Text)
```

```
    b = Val(TxtB.Text)
```

```
    c = Math.Sqrt(a ^ 2 + b ^ 2)
```

```
    LblC.Text = c.ToString("F")
```

```
End Sub
```

The above project requires two text boxes and five label controls. One of the label controls is for displaying the results of the calculation.

The output image looks something like in Figure 12.1

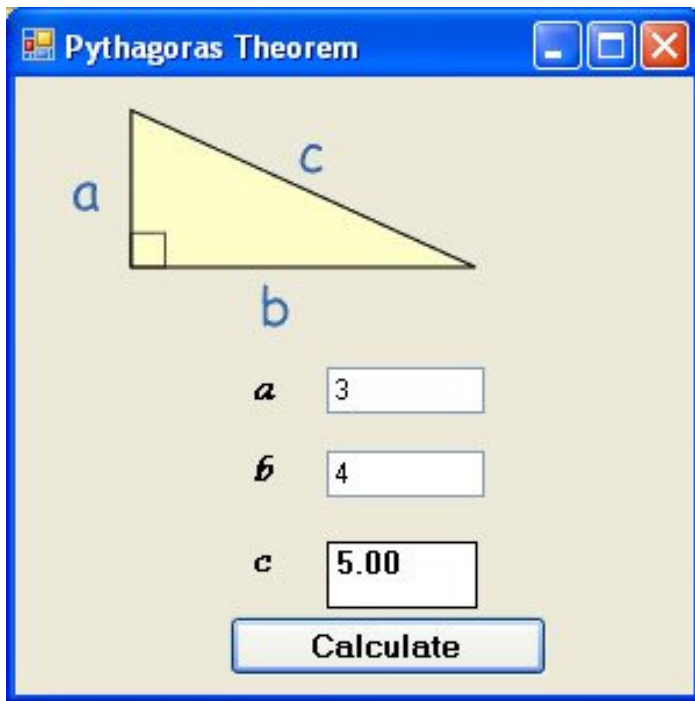


Figure 12.1

12.9 Trigonometric Functions

Trigonometric functions handle problems involving angles. The basic trigonometric functions in Visual Basic 2010 are Sin, Cos, Tan and Atan. Sin is the function that returns the value of sine of an angle in radian, Cos returns the value of cosine of an angle in radian and Tan returns the value of tangent of an angle in radian. Atan returns the value of Arc tangent, which represents the value of the angle in radian given the value of tangent of this angle. Arc tangent is expressed as $\tan^{-1}(x) = y$, which means $\tan(y) = x$. For example, $\tan^{-1}(1) = \frac{\pi}{4}$.

If you wish to accept input in degree from the user, you need to convert degree to radian using the following formula:

$$1 \text{ degree} = \frac{\pi}{180}$$

The first code you should write before you can values of trigometric functions is the function to compute the value of π , or Pi. We use the fact that $\tan^{-1}(1) = \frac{\pi}{4}$, which is

$\text{Atan}(1) = \frac{\text{Pi}}{4}$ in VB language to obtain the formula $\text{Pi} = 4 \times \text{Atan}(1)$. Therefore, the code to get value of Pi is as follows:

```
Public Function Pi( ) As Double
    Return 4.0 * Math.Atan(1.0)
End Function
```

We use the keyword Public as we wish to use the value of Pi throughout the module.

In the following example, we will show you how to obtain the values of Sine, Cosine and Tangent of an angle.

Example 12.7

In this example, the program allows the user to enter an angle in degree and calculate the values of sine, cosine and tangent of this angle. Start a new project and name it Trigo Functions. Next, insert one text box into the form and name it as TxtAngle. The purpose of the text box is allowing the user to enter an angle in degree. You also add three labels and name them as LblSin, LblCos and LblTan to display the values of sine, cosine and tangent of the angle. Insert four other labels for labeling purpose. Lastly, add one button and name it as BtnCal.

First, under the statement Public Class Form1 enter the code to compute the value of Pi as follows:

```
Public Function Pi() As Double
    ' Calculate the value of pi.
    Return 4.0 * Math.Atan(1.0)
End Function
```

Next, click on the button and enter the following code:

```
Private Sub BtnCal_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles BtnCal.Click
```

```
    Dim degree As Single, angle As Double
```

```
    degree = TxtAngle.Text
```

```
    angle = degree * (Pi()) / 180)
```

```
    LblSin.Text = Math.Sin(angle).ToString("F")
```

```
    LblCos.Text = Math.Cos(angle).ToString("F")
```

```
    LblTan.Text = Math.Tan(angle).ToString("F")
```

```
End Sub
```

The output interface is shown in Figure 12.2

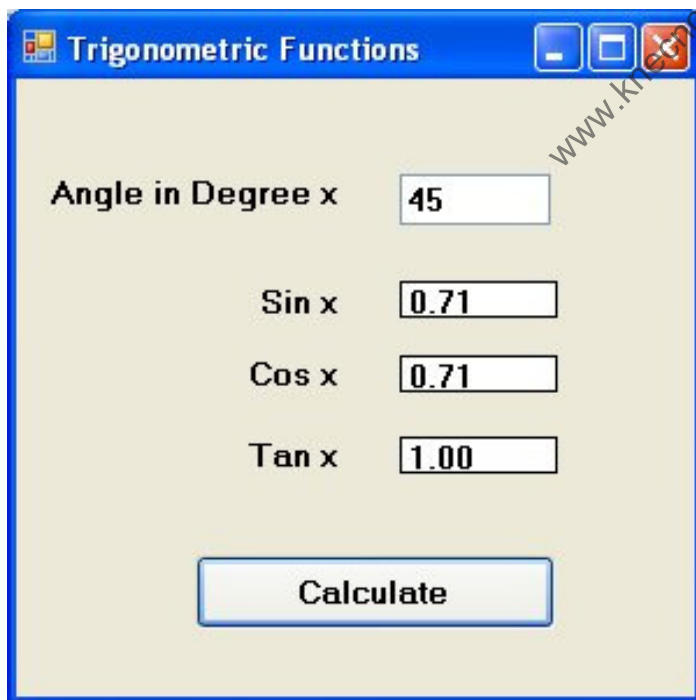


Figure 12.2

Example 12.8

This example computes the area and circumference of a circle. The formula of area of circle is πr^2 and the formula of circumference is $2\pi r$. In this program, you insert a text box to allow the user to enter the value of radius of the circle. Add two labels to display the value of Area and the value of circumference. Use the ToString method to specify number of decimal places with the F specifier. F3 means three decimal places and F alone means two decimal places.

```
Public Function Pi() As Double
```

```
    Return 4.0 * Math.Atan(1.0)
```

```
End Function
```

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles MyBase.Load
```

```
End Sub
```

```
Private Sub TxtRadius_TextChanged(ByVal sender As System.Object, ByVal e  
As System.EventArgs) Handles TxtRadius.TextChanged
```

```
    Dim r, l, Area As Double
```

```
    r = Val(TxtRadius.Text)
```

```
    l = 2 * Pi() * r
```

```
    Area = Pi() * r ^ 2
```

```
    LblCirCumF.Text = l.ToString("F")
```

```
    LblArea.Text = Area.ToString("F3")
```

```
End Sub
```

The runtime interface is shown in Figure 12.3

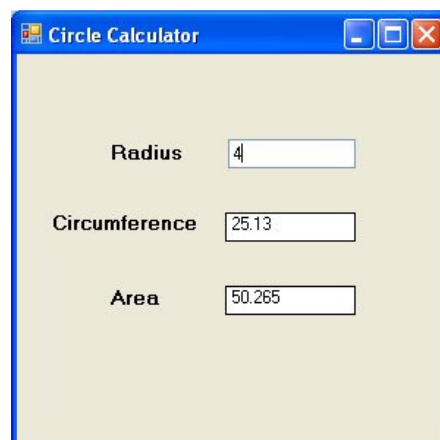


Figure 12.2

Summary

In this chapter, you learned how to write code to perform calculations using various built-in mathematical functions in VB2010.

- In section 12.1, you learned how to use the Abs function to return the absolute value of a given number.
- In section 12.2, you learned how to use the Exp function to express a number in exponential form.
- In section 12.3, you learned how to use the Fix function to truncate the decimal part of a positive number and return the largest integer smaller than the number.
- In section 12.4, you learned how to use the Int function to convert a number into an integer by truncating its decimal part and the resulting integer is the largest integer that is smaller than the number.
- In section 12.5, you learned how to use the Log function to return value of the natural logarithm of a number
- In section 12.6, you learned how to use the Rnd() function to generate a random value between 0 and 1.
- In section 12.7, you learned how to use Round function to round up a number to a certain number of decimal places.
- In Section 12.8, you learned how to use the Sqrt function to obtain the square root of a number.
- In section 12.9, you learned how to use the trigonometric functions Sin, Cos and Tan to compute the values of sine, cosine and tangent of an angle in radian. You also learned how to obtain the value of Pi using the formula $Pi=4 \times \text{Atan}(1)$