

DOUBLE DATA TYPE

1. Assign result to both an integer and a double

Imports System

Public Class MainClass

Shared Sub Main(ByVal args **As** String())

Dim int1 **As** Integer = 12

Dim int2 **As** Integer = 5

Dim intAnswer **As** Integer

Dim doubleAnswer **As** Double

Console.WriteLine("{0} + {1} = {2}", int1, int2, int1 + int2)

Console.WriteLine("{0} - {1} = {2}", int1, int2, int1 - int2)

Console.WriteLine("{0} * {1} = {2}", int1, int2, int1 * int2)

' integer division

intAnswer = int1 \ int2

doubleAnswer = int1 \ int2

Console.WriteLine("{0} \ {1} = [integer] {2} [double] {3}", _
int1, int2, intAnswer, doubleAnswer)

' division. Assign result to both an integer and a **double**

' note, option strict must be off!

intAnswer = int1 / int2

doubleAnswer = int1 / int2

Console.WriteLine("{0} / {1} = [integer] {2} [double] {3}", _
int1, int2, intAnswer, doubleAnswer)

End Sub

End Class



2. Double: Set number, multiply numbers, and display results

Imports System

Public Class MainClass

Shared Sub Main()

'Declare variable

Dim dblNumber **As** Double

```
'Set number, multiply numbers, and display results
dblNumber = 45.34
dblNumber *= 4.333
System.Console.WriteLine("Multiplication test... " & dblNumber)
```

End Sub

End Class



3.Double: Set number, divide numbers, and display results

Imports System

Public Class MainClass

Shared Sub Main()

'Declare variable

Dim dblNumber **As** Double

'Set number, divide numbers, and display results

dblNumber = 45.34

dblNumber /= 4.333

System.Console.WriteLine("Division test... " & dblNumber)

End Sub

End Class



4.Double number format: 0:n3

Imports System

Public Class MainClass

Shared Sub Main()

'Declare variable

Dim dblNumber **As** Double

'Set number, divide numbers, and display results

dblNumber = 45.34

dblNumber /= 4.333

System.Console.WriteLine("Division test... " & dblNumber)

System.Console.WriteLine("With formatting: " & String.Format("{0:n3}", dblNumber))

End Sub

End Class



5. Double value calculation

Imports System

Public Class MainClass

Shared Sub Main()

Dim dblRadiusSquared **As** Double

Dim dblResult **As** Double

Dim radius **As** Double

radius = 10.01

dblRadiusSquared = radius * radius

dblResult = dblRadiusSquared * Math.PI

System.Console.WriteLine(dblResult)

End Sub

End Class

```
314.787898048963
```

6. Append Double data type values to a StringBuilder object.

Imports System.Text

Class Sample

Public Shared Sub Main()

Dim sb **As** New StringBuilder()

Dim xDouble **As** Double = 7.7

sb = sb.Append(xDouble)

Dim str **As** [String] = sb.ToString()

Console.WriteLine("The appended string is:")

Console.WriteLine(str)

End Sub

End Class

7. Generic and nongeneric versions of the CompareTo method for Double value

Imports System

Class Sample

```
Public Shared Sub Main()
```

```
    Try
```

```
        Dim a1 As [Double] = 7.7, a2 As [Double] = -7.7  
        Show(a1, a2, a1.CompareTo(a2), a1.CompareTo(CObj(a2)))
```

```
    Catch e As Exception
```

```
        Console.WriteLine(e)
```

```
    End Try
```

```
End Sub
```

```
Public Shared Sub Show(var1 As [Object], var2 As [Object], resultGeneric As Integer, resultNonGeneric As Integer)
```

```
    Console.WriteLine(var1)
```

```
    Console.WriteLine(var2)
```

```
    If resultGeneric = resultNonGeneric Then
```

```
        If resultGeneric < 0 Then
```

```
            Console.WriteLine("less than")
```

```
        ElseIf resultGeneric > 0 Then
```

```
            Console.WriteLine("greater than")
```

```
        Else
```

```
            Console.WriteLine("equal to")
```

```
        End If
```

```
    End If
```

```
End Sub
```

```
End Class
```

8. Use the Sign(Double) method to determine the sign of a Double value and display it to the console.

```
Class Sample
```

```
    Public Shared Sub Main()
```

```
        Dim xDouble1 As Double = 6.0
```

```
        Console.WriteLine(xDouble1)
```

```
        Console.WriteLine(Test(Math.Sign(xDouble1)))
```

```
    End Sub
```

```
    Public Shared Function Test([compare] As Integer) As [String]
```

```
        If [compare] = 0 Then
```

```
            Return "equal to"
```

```
        ElseIf [compare] < 0 Then
```

```
            Return "less than"
```

```
        Else
```

```
            Return "greater than"
```

```
        End If
```

```
    End Function
```

```
End Class
```

9. Use Double.Parse to parse double string value

```
Public Class Example
```

```
    Public Shared Sub Main()
```

```
        Dim value As String
```

```
value = Double.MinValue.ToString()  
Try  
    Console.WriteLine(Double.Parse(value))  
Catch e As OverflowException  
    Console.WriteLine("{0} is outside the range of the Double type.", value)  
End Try  
End Sub  
End Class
```

Convert Double.MinValue to string

```
Public Class Example  
    Public Shared Sub Main()  
  
        Dim value As String  
  
        value = Double.MinValue.ToString()  
        Try  
            Console.WriteLine(Double.Parse(value))  
        Catch e As OverflowException  
            Console.WriteLine("{0} is outside the range of the Double type.", value)  
        End Try  
    End Sub  
End Class
```