Program to swap two numbers

```
Imports System
Module Program
    Sub Main(args As String())
        Dim x, y As Integer
        Console.WriteLine("Enter the value of x")
        x = Console.ReadLine()
        Console.WriteLine("Enter the value of y")
        y = Console.ReadLine()
        x = x + y
        y = x - y
        x = x - y
        Console.WriteLine("After swapping values of x and y are{0}", x & y)
        'Console.WriteLine(x)
        'Console.WriteLine(y)
    End Sub
End Module
```

Output:

Program to sort an array

```
Imports System
Module Program
    Sub Main(args As String())
        Dim names(3) As String
        Console.WriteLine("Enter names")
        For name As Integer = 0 To names.Length - 1
            names(name) = Console.ReadLine()
        Next
        Array.Sort(names)
```

```
        Console.WriteLine(" Sorted Names are: “)
        For name As Integer = 0 To names.Length - 1
            Console.WriteLine(names(name))
        Next
    End Sub
End Module
```

Output:

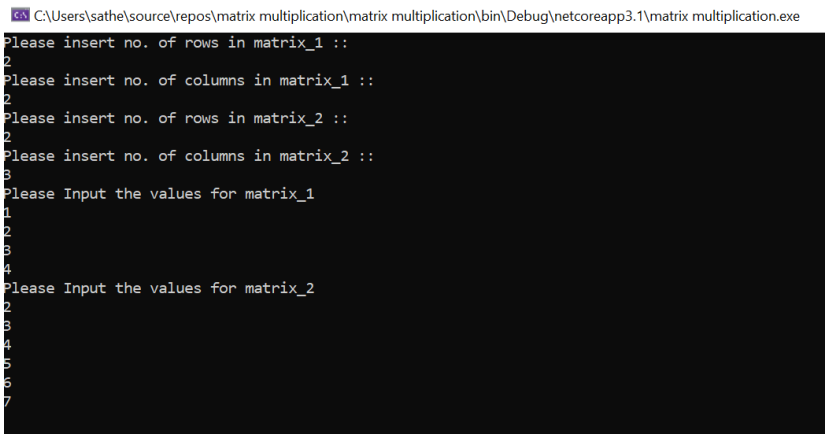
Program to multiply two matrices

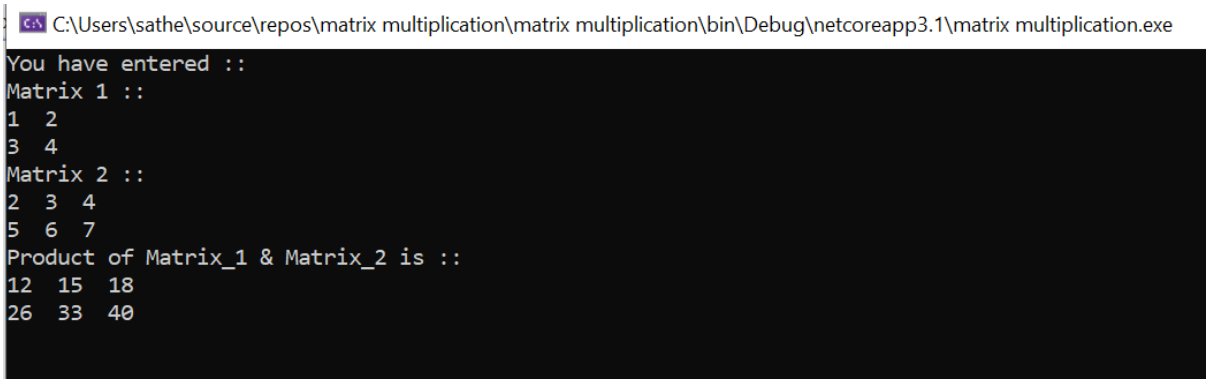
```
Imports System
Module Module1
    Sub Main()
        Dim temp As Integer = 0
        Dim matrix1 As Integer(,), matrix2 As Integer(,)
        Dim row1 As Integer, col1 As Integer, row2 As Integer, col2 As Integer, i As Integer, j As Integer,
k As Integer
        Console.WriteLine("Please insert no. of rows in matrix_1 :: ")
        row1 = Integer.Parse(Console.ReadLine())
        Console.WriteLine("Please insert no. of columns in matrix_1 :: ")
        col1 = Integer.Parse(Console.ReadLine())
        matrix1 = New Integer(row1 - 1, col1 - 1) {}
        Console.WriteLine("Please insert no. of rows in matrix_2 :: ")
        row2 = Integer.Parse(Console.ReadLine())
        Console.WriteLine("Please insert no. of columns in matrix_2 :: ")
        col2 = Integer.Parse(Console.ReadLine())
        matrix2 = New Integer(row2 - 1, col2 - 1) {}

        If col1 <> row2 Then
```

```
        Console.WriteLine("Multiplication is not applicable!!!")
        Console.WriteLine("Note : Number of columns of matrix_1 must be equal to Number of rows of matrix_2.")
    Else

        Console.WriteLine("Please Input the values for matrix_1")
        For i = 0 To row1 - 1
            For j = 0 To col1 - 1
                matrix1(i, j) = Integer.Parse(Console.ReadLine())
            Next
        Next

        Console.WriteLine("Please Input the values for matrix_2")
        For i = 0 To row2 - 1
            For j = 0 To col2 - 1
                matrix2(i, j) = Integer.Parse(Console.ReadLine())
            Next
        Next
        Console.Clear()
        Console.WriteLine("You have entered :: ")
        Console.WriteLine("Matrix 1 ::")
        For i = 0 To row1 - 1
            For j = 0 To col1 - 1
                Console.Write(matrix1(i, j))
                Console.Write("  ")
            Next
            Console.WriteLine()
        Next

        Console.WriteLine("Matrix 2 ::")
        For i = 0 To row2 - 1
```

```
            For j = 0 To col2 - 1
                Console.Write(matrix2(i, j))
                Console.Write("  ")
            Next
            Console.WriteLine()
        Next

        Console.WriteLine("Product of Matrix_1 & Matrix_2 is ::")
        For i = 0 To row1 - 1
            For j = 0 To col2 - 1
                For k = 0 To row2 - 1
                    temp = temp + (matrix1(i, k) * matrix2(k, j))
                Next
                Console.Write(temp & "  ")
                temp = 0
            Next
            Console.WriteLine()
        Next
    End If
    Console.Read()
  End Sub
End Module
```

Output:

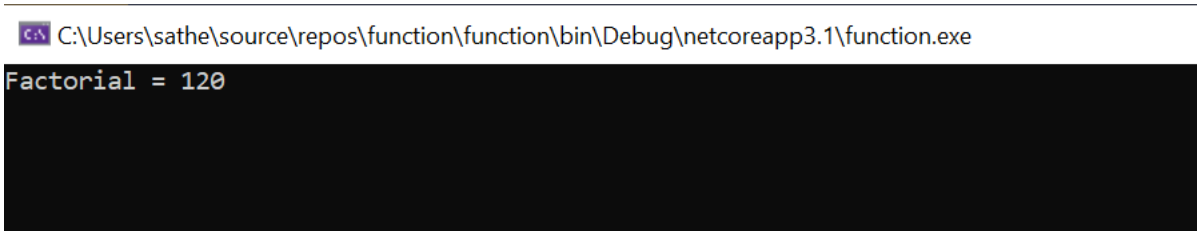




Program to implement function

```
Imports System

Module Module1

    Function Factorial(ByVal num As Integer)

        Dim result As Integer

        If num = 1 Then

            Return 1

        Else

            result = Factorial(num - 1) * num

            Return result

        End If

    End Function


    Sub Main()

        Console.WriteLine("Factorial = {0} ", Factorial(5))

        Console.ReadKey()

    End Sub

End Module
```

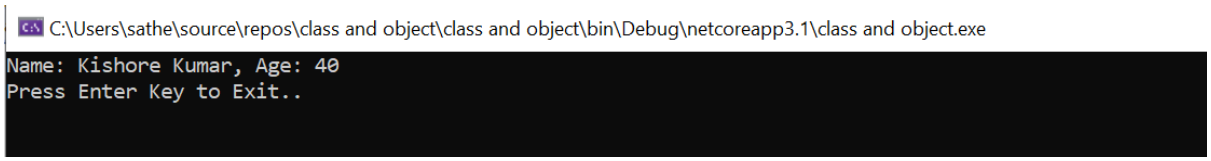
Output:



Program to implement class

```
Imports System

Module Module1
    Sub Main()
        Dim user As Users = New Users("Kishore Kumar", 40)
        user.GetUserDetails()
        Console.WriteLine("Press Enter Key to Exit..")
        Console.ReadLine()
    End Sub
    Public Class Users
        Public Property UName As String
        Public Property UAge As Integer
        Public Sub New(ByVal name As String, ByVal age As Integer)
            UName = name
            UAge = age
        End Sub
        Public Sub GetUserDetails()
            Console.WriteLine("Name: {0}, Age: {1}", UName, UAge)
        End Sub
    End Class
End Module
```

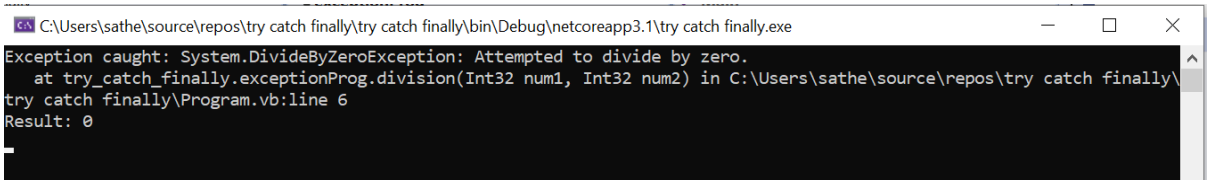
Output:



Program to implement try,catch...finally

```
Imports System.IO
Module exceptionProg
    Sub division(ByVal num1 As Integer, ByVal num2 As Integer)
        Dim result As Integer
        Try
            result = num1 \ num2
        Catch e As DivideByZeroException
            Console.WriteLine("Exception caught: {0}", e)
        Finally
            Console.WriteLine("Result: {0}", result)
        End Try
    End Sub
    Sub Main()
        division(25, 0)
        Console.ReadKey()
    End Sub
End Module
```

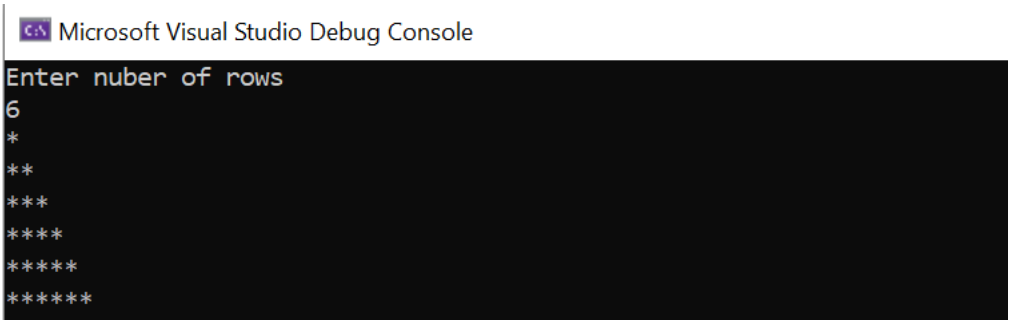
Output:



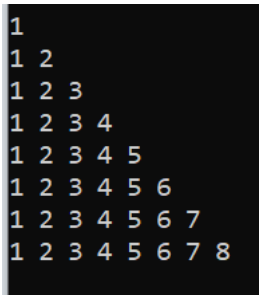
Program to print


```
Imports System

Module Module1
    Sub Main()
        Dim row, col, noofrow As Integer
        Console.WriteLine("Enter number of rows")
        noofrow = Console.ReadLine()
        For row = 1 To noofrow Step 1
            For col = 1 To row Step 1
                Console.Write("*")
            Next
            Console.Write(vbNewLine)
        Next
    End Sub
End Module
```

Output



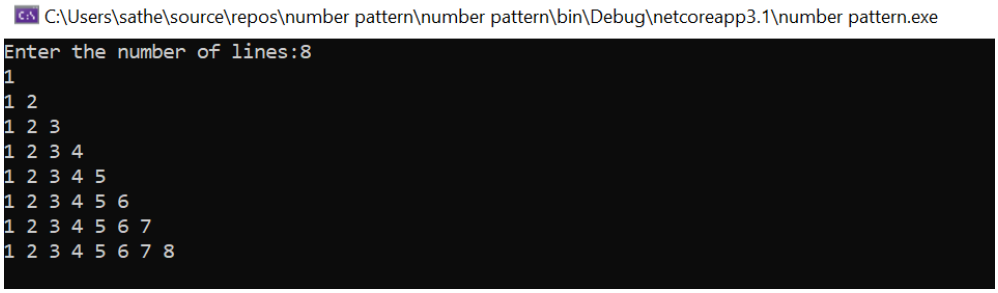
Program to print




```
Imports System

Module Program
    Sub Main(args As String())

        Dim i, j, n As Integer
        Console.Write("Enter the number of lines:")
        n = Console.ReadLine()
        For i = 1 To n
```

```
            For j = 1 To i
                Console.Write(j & " ")
            Next
            Console.WriteLine()
        Next
        Console.ReadLine()

    End Sub
End Module
```

Output:



Program to implement message box (YES/NO type) on button click event.

```
Public Class Form1
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    End Sub
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Dim answer As Integer
        answer = MsgBox("Text", vbQuestion + vbYesNo + vbDefaultButton1, "Message Box")
        If answer = vbYes Then
            MsgBox("You clicked Yes Button")
        Else
            MsgBox("You clicked No button")
        End If
    End Sub
End Class
```

Output: