1 Develop a calculator using Button, Textbox and Label

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        MsgBox("The sum is " & Val(TextBox1.Text) + Val(TextBox2.Text))
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        MsgBox("The sum is " & Val(TextBox1.Text) - Val(TextBox2.Text))
    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles
Button3.Click
        MsgBox("The sum is " & Val(TextBox1.Text) * Val(TextBox2.Text))
    End Sub

    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles
Button4.Click
        MsgBox("The sum is " & Val(TextBox1.Text) / Val(TextBox2.Text))
    End Sub
End Class
```

2 Write a Program using select case statement using Console

```
Imports System
Module Module1
    Sub Main()
        Console.WriteLine("Please enter a number between 1 and 5:")
        Dim userInput As Integer = Console.ReadLine()

        Select Case userInput
            Case 1
                Console.WriteLine("You entered 1.")
            Case 2
                Console.WriteLine("You entered 2.")
            Case 3
                Console.WriteLine("You entered 3.")
            Case 4
                Console.WriteLine("You entered 4.")
            Case 5
                Console.WriteLine("You entered 5.")
            Case Else
                Console.WriteLine("Invalid input.")
        End Select

        Console.ReadLine()
    End Sub
End Module
```

3. Write a program using While statement to print the prime numbers between 1 to 100 using Console

```
Module Module1

    Sub Main()
        Dim num As Integer = 2
        While num <= 100
            Dim isPrime As Boolean = True
            Dim divisor As Integer = 2

            While divisor <= Math.Sqrt(num)
                If num Mod divisor = 0 Then
                    isPrime = False
                    Exit While
                End If
                divisor += 1
            End While

            If isPrime AndAlso num > 1 Then
                Console.Write(num & " ")
            End If

            num += 1
        End While

        Console.ReadLine()
    End Sub
End Module
```

4 Write program using For Next loop statement to find the Armstrong numbers between 1 to 500 using Console

```
Module Module1

    Sub Main()

        Dim num, temp, r, s, t As Integer

        Console.WriteLine(" Armstrong Number Between 1 to 500: ")

        For num = 1 To 500

            temp = num
            s = 0

            For t = 0 To num
                r = temp Mod 10
                s = (r * r * r) + s
                temp = temp \ 10
            Next

            If num = s Then
                Console.WriteLine(num)
            End If

        Next
    Console.ReadKey()

    End Sub

End Module
```

5) Implement the program using if-else statement to find the number is even or odd using Button, Textbox and Label

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click

        Dim input As Integer = Integer.Parse(TextBox1.Text)

            If input Mod 2 = 0 Then
                Label1.Text = input & " is even."
            Else
                Label1.Text = input & " is odd."
            End If
        End Sub
    End Class
```

6) Write a program to find the Factorial using Button, Textbox and Label.

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Dim input As Integer = Integer.Parse(TextBox1.Text)
        Dim factorial As Integer = 1

        For i As Integer = 1 To input
            factorial *= i
        Next

        Label1.Text = "Factorial of " & input & " is " & factorial
    End Sub
End Class
```

7) Write the program using RadioButton to change the bulb state ON/OFF.

```
Public Class Form1
    Private Sub RadioButton1_CheckedChanged(sender As Object, e As EventArgs)
Handles RadioButton1.CheckedChanged
        PictureBox2.Show()
        PictureBox1.Hide()
    End Sub
    Private Sub RadioButton2_CheckedChanged(sender As Object, e As EventArgs)
Handles RadioButton2.CheckedChanged
        PictureBox1.Show()
        PictureBox2.Hide()
    End Sub
End Class
```

8) Write the program to select colleges using SINngle ComboBox.

```
Public Class Form1
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        ComboBox1.Items.Add("MET")
        ComboBox1.Items.Add("GGSP")
        ComboBox1.Items.Add("KKW")
        ComboBox1.Items.Add("GP")
    End Sub
    Private Sub ComboBox1_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles ComboBox1.SelectedIndexChanged
        MsgBox(ComboBox1.SelectedItem.ToString)
    End Sub
End Class
```

9) Write a program to display the traffic signal using timer control.

Public Class Form1

```
Public Class Form1
    Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
        If PictureBox1.Visible Then
            PictureBox1.Visible = False
            PictureBox2.Visible = True
            PictureBox3.Visible = False
        ElseIf PictureBox2.Visible Then
            PictureBox1.Visible = False
            PictureBox2.Visible = False
            PictureBox3.Visible = True
        ElseIf PictureBox3.Visible Then
            PictureBox1.Visible = True
            PictureBox2.Visible = False
            PictureBox3.Visible = False
        End If
    End Sub
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Timer1.Enabled = True
        Timer1.Interval = 1600
        PictureBox1.Visible = True
        PictureBox2.Visible = False
        PictureBox3.Visible = False
    End Sub
End Class
```

10) Develop a program to print the reverse of any number using Sub procedure.

```
Module Module1

    Sub Main()
        Dim number As Integer = 0
        Dim remainder As Integer = 0
        Dim reverse As Integer = 0

        Console.Write("Enter the number: ")
        number = Integer.Parse(Console.ReadLine())

        While (number > 0)
            remainder = number Mod 10
            reverse = reverse * 10 + remainder
            number = number / 10
        End While

        Console.WriteLine("Reverse: {0}", reverse)
    End Sub

End Module
```

11) Write a program to identify maximum number using parameterized function. (Use two Textbox for input a integer number and display output in Message Box)

```
Public Class Form1
    Private Function GetMax(ByVal num1 As Integer, ByVal num2 As Integer) As Integer
        If num1 > num2 Then
            Return num1
        Else
            Return num2
        End If
    End Function

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Dim num1 As Integer = Integer.Parse(TextBox1.Text)
        Dim num2 As Integer = Integer.Parse(TextBox2.Text)

        Dim max As Integer = GetMax(num1, num2)

        MessageBox.Show("The maximum number is " & max)
    End Sub
End Class
```

12) Implement a program to accept values from Combo Box and Display average of this inmessage box using class

```
Public Class Form1
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
ComboBox1.Items.Add(5)
ComboBox1.Items.Add(8)
ComboBox1.Items.Add(12)
ComboBox1.Items.Add(20)
ComboBox1.Items.Add(32)
ComboBox2.Items.Add(6)
ComboBox2.Items.Add(11)
ComboBox2.Items.Add(17)
ComboBox2.Items.Add(24)
ComboBox2.Items.Add(36)
End Sub
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
Dim average As Single
average = (Val(ComboBox1.Text) + Val(ComboBox2.Text)) / 2 MsgBox("Average = " & average)
End Sub
 End Class
```

13) Implement a program to calculate area of circle using parameterized constructor.

```
Module Module1 Sub Main()
Dim obj As New circle(2)
obj.area()
Console.ReadLine()
End Sub
Class circle
Dim p As Double = 3.14
Dim r, a As Double
Public Sub New(ByVal i As Integer)
r = i
End Sub
Sub area()
a = p * r * r
Console.WriteLine("Area of Circle = " & a)
End Sub
End Class
End Module
```

14) Write a program using data adapter to connect to the database and display data on datagridview(Use Employee table)

```
Imports System.Data.SqlClient

Public Class Form1
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        ' Replace the values below with your own database connection details
        Dim connectionString As String = "Data Source=myServerAddress;Initial
Catalog=myDataBase;User Id=myUsername;Password=myPassword;"

        ' Create a new SqlConnection object using the connection string
        Using connection As New SqlConnection(connectionString)
            ' Open the connection
            connection.Open()

            ' Create a new SqlDataAdapter object to retrieve data from the
Employee table
            Dim adapter As New SqlDataAdapter("SELECT * FROM Employee",
connection)

            ' Create a new DataTable object to hold the data
            Dim table As New DataTable()

            ' Fill the DataTable with data from the SqlDataAdapter
            adapter.Fill(table)

            ' Set the DataSource property of the DataGridView to the DataTable
            DataGridView1.DataSource = table
        End Using
    End Sub
End Class
```

15) Write a program to insert data into the database.(Use Student Table)

```
Imports System.Data.SqlClient

Public Class Form1
    Private Sub btnInsert_Click(sender As Object, e As EventArgs) Handles btnInsert.Click
        ' Replace the values below with your own database connection details
        Dim connectionString As String = "Data Source=myServerAddress;Initial
Catalog=myDataBase;User Id=myUsername;Password=myPassword;"

        ' Create a new SqlConnection object using the connection string
        Using connection As New SqlConnection(connectionString)
            ' Open the connection
            connection.Open()

            ' Create a new SqlCommand object to execute the INSERT statement
            Dim command As New SqlCommand("INSERT INTO Student (Name, Age, Grade) VALUES
(@Name, @Age, @Grade)", connection)

            ' Add parameters to the SqlCommand object
            command.Parameters.AddWithValue("@Name", txtName.Text)
            command.Parameters.AddWithValue("@Age", txtAge.Text)
            command.Parameters.AddWithValue("@Grade", txtGrade.Text)

            ' Execute the INSERT statement
            command.ExecuteNonQuery()

            ' Display a message box to indicate that the data has been inserted
            MessageBox.Show("Data inserted successfully.")
        End Using
    End Sub
End Class
```

16) Write the program to validate Email ID, Mobile No. using Regex.

```
Imports System.Text.RegularExpressions

Module Module1

    Sub Main()

        ' Validate email address
        Dim email As String = "john.doe@example.com"
        If Regex.IsMatch(email, "^\w+([\.-]?\w+)*@\w+([\.-]?\w+)*(\.\w{2,3})+$") Then
            Console.WriteLine("Email is valid")
        Else
            Console.WriteLine("Email is not valid")
        End If

        ' Validate mobile number
        Dim mobile As String = "9876543210"
        If Regex.IsMatch(mobile, "^[1-9]\d{9}$") Then
            Console.WriteLine("Mobile number is valid")
        Else
            Console.WriteLine("Mobile number is not valid")
        End If

        Console.ReadLine()

    End Sub

End Module
```