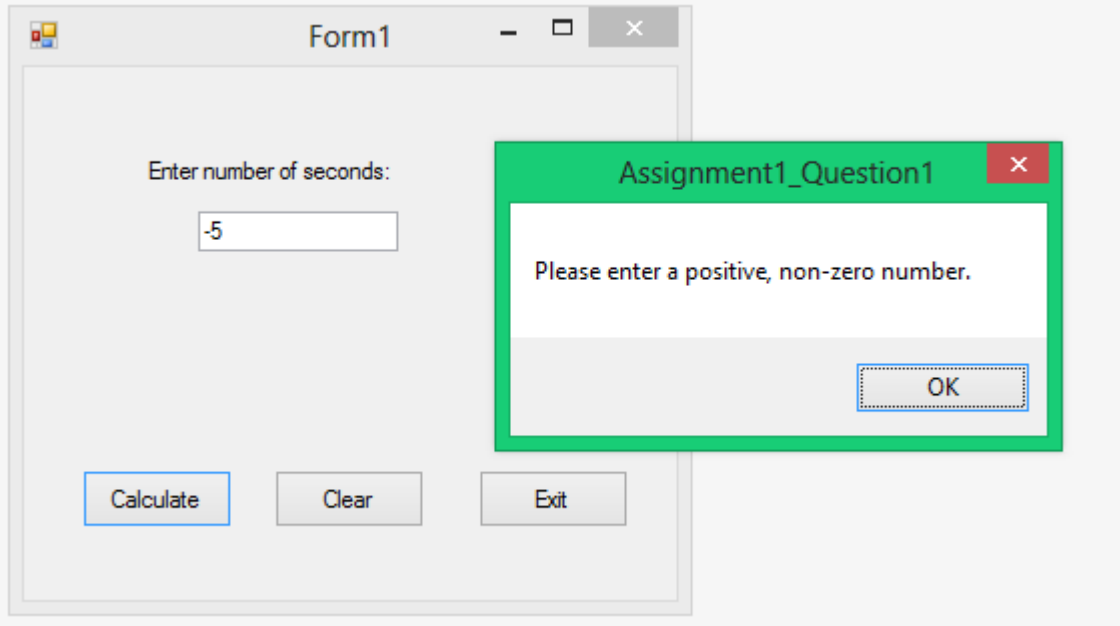
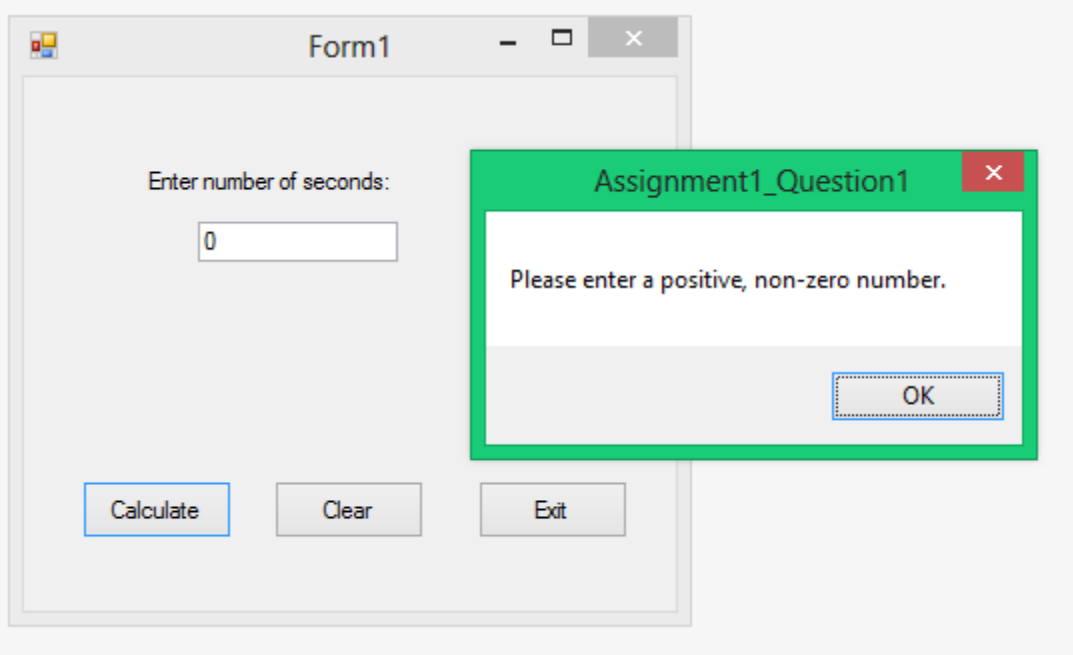
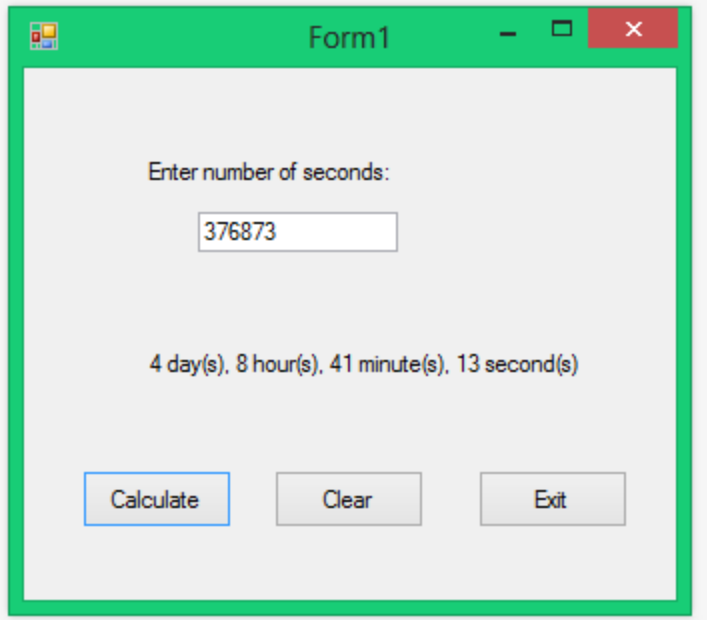
Question 1:

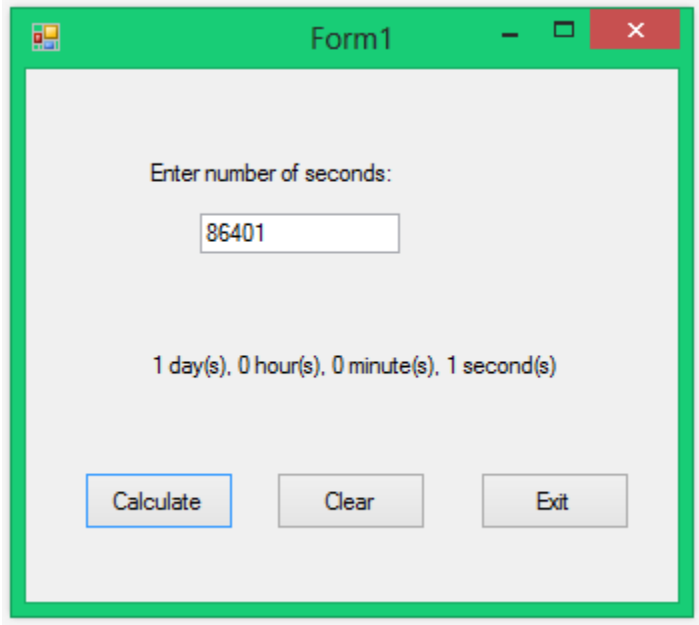
```
Public Class Form1    'Ilker Hadzhalaran

    Dim intSecondsEntered As Integer

    Dim intSeconds As Integer
    Dim intMinutes As Integer
    Dim intHours As Integer
    Dim intDays As Integer

    Dim intMinutesRemainder As Integer
    Dim intHoursRemainder As Integer
    Dim intDaysRemainder As Integer

    Const SECONDS_IN_MINUTE As Integer = 60
    Const SECONDS_IN_HOUR As Integer = 3600
    Const SECONDS_IN_DAY As Integer = 86400


    Private Sub btnExit_Click(sender As Object, e As EventArgs) Handles btnExit.Click
        Me.Close()
    End Sub

    Private Sub btnCalculate_Click(sender As Object, e As EventArgs) Handles
btnCalculate.Click

        If Not (Integer.TryParse(txtEnterSeconds.Text, intSecondsEntered) Or
Double.TryParse(txtEnterSeconds.Text, intSecondsEntered)) Then
            MsgBox("Invalid value. Please enter a number only.")
            txtEnterSeconds.Clear()
            txtEnterSeconds.Focus()
            Exit Sub

        End If

        intSecondsEntered = CInt(txtEnterSeconds.Text)

        If (intSecondsEntered <= 0) Then
            MsgBox("Please enter a positive, non-zero number.")
            Exit Sub
        End If

        intDays = intSecondsEntered \ SECONDS_IN_DAY
        intDaysRemainder = intSecondsEntered Mod SECONDS_IN_DAY

        intHours = intDaysRemainder \ SECONDS_IN_HOUR
        intHoursRemainder = intDaysRemainder Mod SECONDS_IN_HOUR

        intMinutes = intHoursRemainder \ SECONDS_IN_MINUTE
        intMinutesRemainder = intDaysRemainder Mod SECONDS_IN_MINUTE

        intSeconds = intMinutesRemainder
```

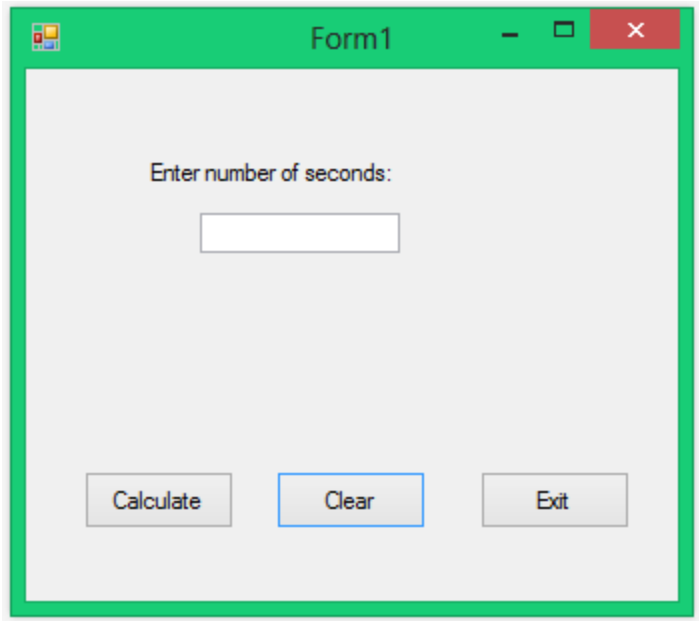
```
        lblDisplayResults.Text = (intDays).ToString & " day(s), " & (intHours).ToString &
" hour(s), " & (intMinutes).ToString & " minute(s), " & (intSeconds).ToString & "
second(s)"

    End Sub

    Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles btnClear.Click
        lblDisplayResults.Text = String.Empty
        txtEnterSeconds.Clear()

    End Sub

End Class
```




Form1
Enter number of seconds:
0
Assignment1_Question1
Please enter a positive, non-zero number.
OK
Calculate
Clear
Exit

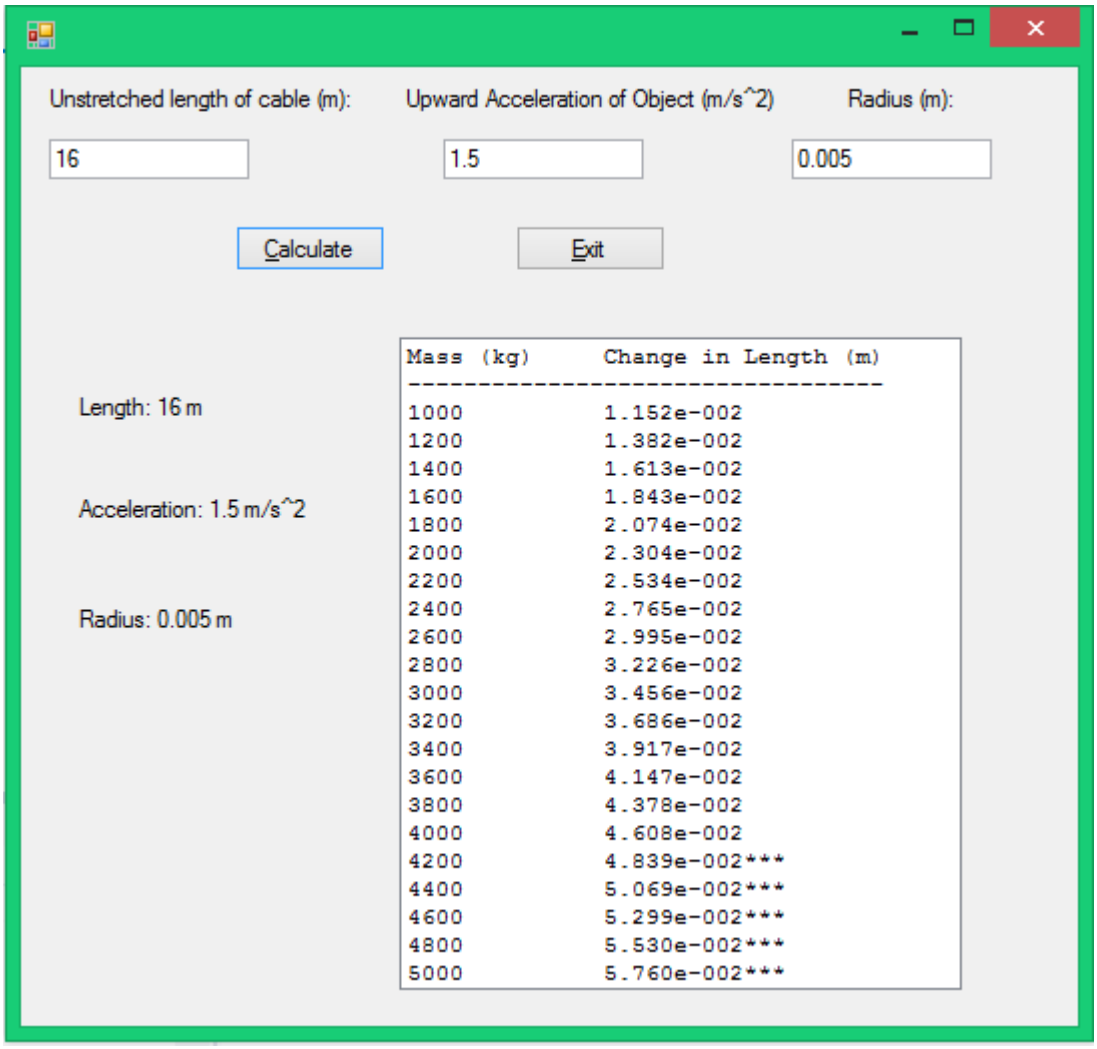

Form1
Enter number of seconds:
376873
4 day(s), 8 hour(s), 41 minute(s), 13 second(s)
Calculate
Clear
Exit


Form1
Enter number of seconds:
86401
1 day(s), 0 hour(s), 0 minute(s), 1 second(s)
Calculate
Clear
Exit


Form1
Enter number of seconds:
Calculate
Clear
Exit

Question 2:

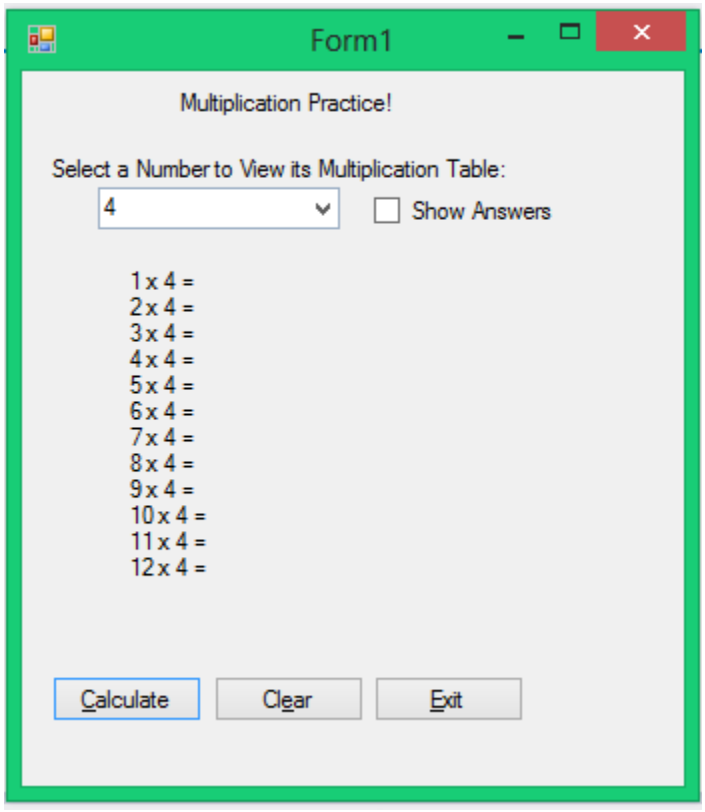
```
Public Class Form1       'Ilker Hadzhalaran

    Dim dblUnstretchedLength As Double
    Dim dblUpwardAcceleration As Double
    Dim dblRadius As Double
    Dim dblChangeInLength As Double
    Const YOUNGS_MODULUS As Double = (2.0 * 10 ^ (11))
    Const GRAVITY_CONSTANT As Double = 9.81

    Private Sub btnExit_Click(sender As Object, e As EventArgs) Handles btnExit.Click
        Me.Close()
    End Sub

    Private Sub btnCalculate_Click(sender As Object, e As EventArgs) Handles
btnCalculate.Click

        lstTableofResults.Items.Clear()

        dblUnstretchedLength = CDbl(txtInputUnstretchedLength.Text)
        dblUpwardAcceleration = CDbl(txtInputAccelerationObject.Text)
        dblRadius = CDbl(txtInputRadius.Text)

        lblShowInputLength.Text = "Length: " & dblUnstretchedLength.ToString & " m"
        lblShowInputAcceleration.Text = "Acceleration: " & dblUpwardAcceleration.ToString
& " m/s^2"
        lblShowInputRadius.Text = "Radius: " & dblRadius.ToString & " m"

        lstTableofResults.Items.Add("Mass (kg)" & Space(5) & "Change in Length (m)")
        lstTableofResults.Items.Add("-----------------------------------")

        For intMass As Integer = 1000 To 5000 Step 200

            dblChangeInLength = intMass * (dblUpwardAcceleration + GRAVITY_CONSTANT) *
dblUnstretchedLength / (YOUNGS_MODULUS * (Math.PI * dblRadius ^ 2))

            If (dblChangeInLength >= (0.003 * dblUnstretchedLength)) Then
                lstTableofResults.Items.Add((intMass).ToString & Space(10) &
(dblChangeInLength).ToString("e3") & "***")

            Else
                lstTableofResults.Items.Add((intMass).ToString & Space(10) &
(dblChangeInLength).ToString("e3"))

            End If


        Next


    End Sub



End Class
```

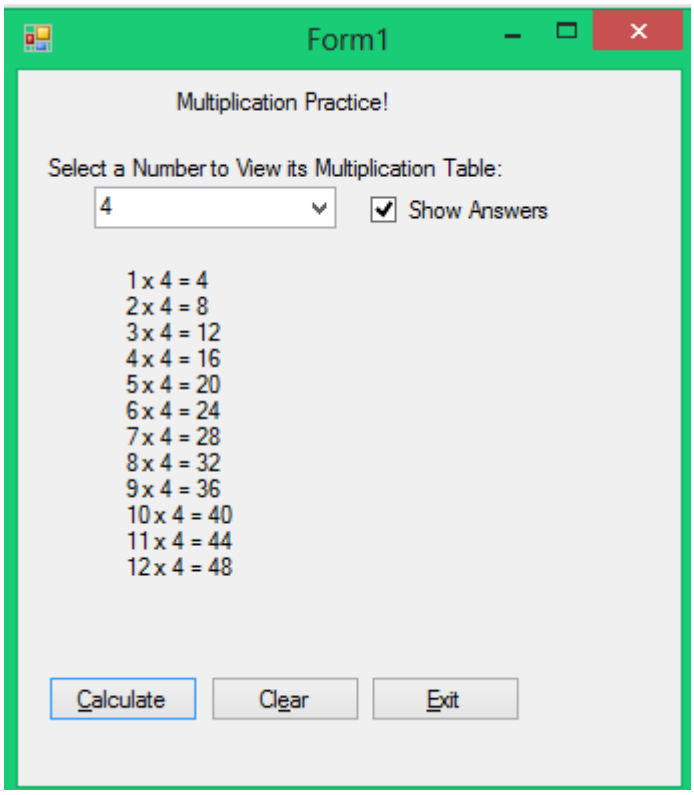

Unstretched length of cable (m):
Upward Acceleration of Object (m/s^2)
Radius (m):
16
1.5
0.005
Calculate
Exit
Length: 16 m
Acceleration: 1.5 m/s^2
Radius: 0.005 m
Mass (kg) Change in Length (m)
1000 1.152e-002
1200 1.382e-002
1400 1.613e-002
1600 1.843e-002
1800 2.074e-002
2000 2.304e-002
2200 2.534e-002
2400 2.765e-002
2600 2.995e-002
2800 3.226e-002
3000 3.456e-002
3200 3.686e-002
3400 3.917e-002
3600 4.147e-002
3800 4.378e-002
4000 4.608e-002
4200 4.839e-002***
4400 5.069e-002***
4600 5.299e-002***
4800 5.530e-002***
5000 5.760e-002***

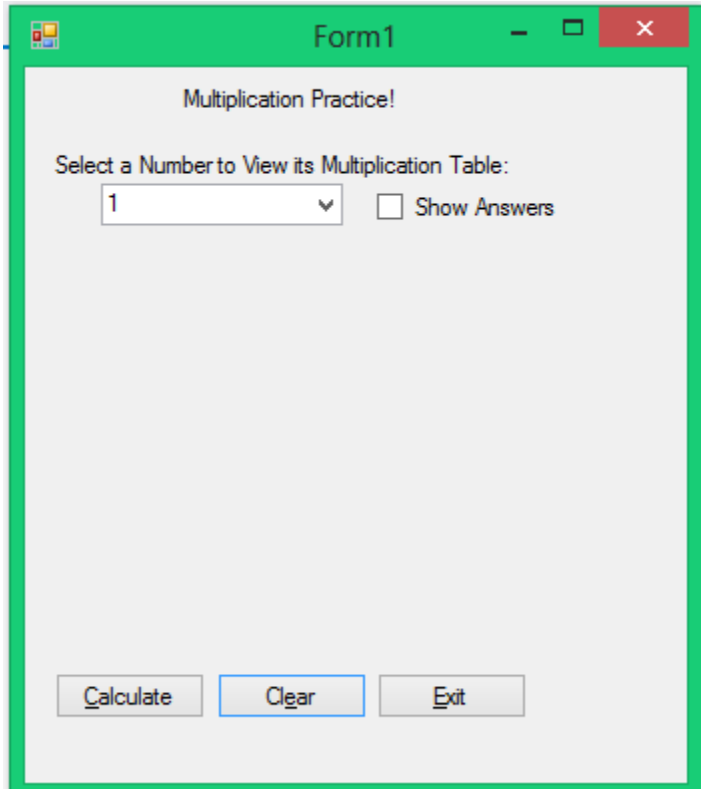
Question 3:

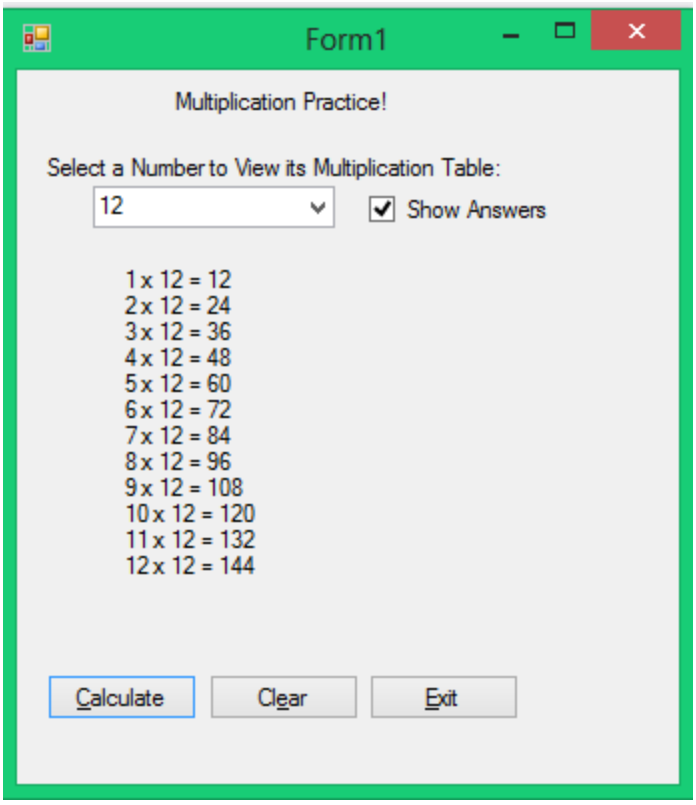
```
Public Class Form1          'Ilker Hadzhalaran

    Dim intCounter As Integer
    Dim strResults As String



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

        For intCounter As Integer = 1 To 12 Step 1
            cboMultiplicationTable.Items.Add(intCounter)
        Next

        chkShowAnswers.Checked = False

    End Sub

    Private Sub btnExit_Click(sender As Object, e As EventArgs) Handles btnExit.Click
        Me.Close()
    End Sub

    Private Sub btnReset_Click(sender As Object, e As EventArgs) Handles btnClear.Click

        lblResults.Text = String.Empty
        chkShowAnswers.Checked = False
        cboMultiplicationTable.SelectedIndex = 0

    End Sub

    Private Sub btnCalculate_Click(sender As Object, e As EventArgs) Handles
btnCalculate.Click



        intCounter = 1



        If chkShowAnswers.Checked Then

            strResults = String.Empty
            lblResults.Text = String.Empty

            Do
                strResults &= intCounter.ToString & " x " &
(cboMultiplicationTable.SelectedItem).ToString & " = " & (intCounter *
cboMultiplicationTable.SelectedItem).ToString & vbCr
                lblResults.Text = strResults
                intCounter = intCounter + 1

            Loop While (intCounter <= 12)
```

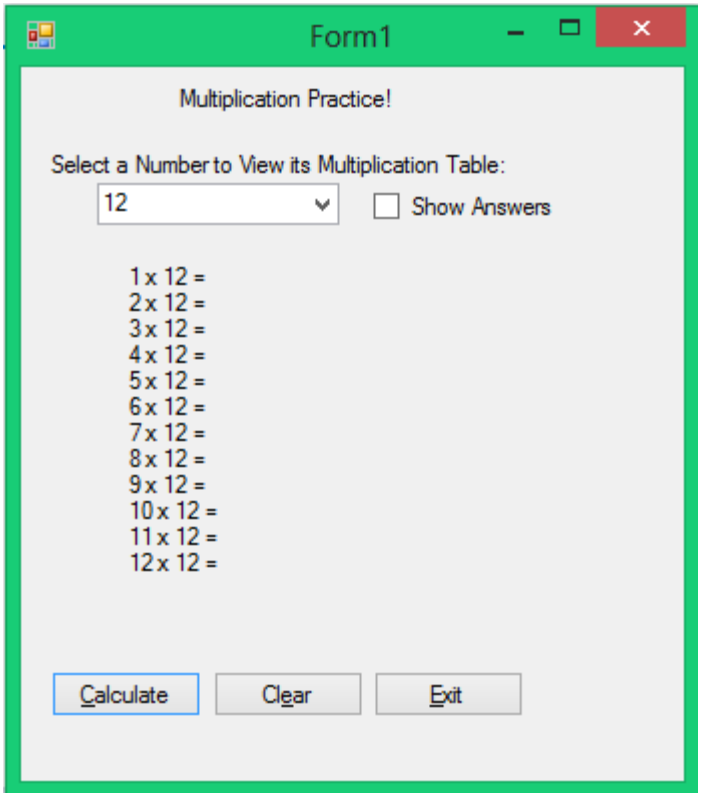
```
        Else

            strResults = String.Empty
            lblResults.Text = String.Empty

            Do
                strResults &= intCounter.ToString & " x " &
(cboMultiplicationTable.SelectedItem).ToString & " = " & vbCr
                lblResults.Text = strResults
                intCounter = intCounter + 1


            Loop While (intCounter <= 12)

        End If


    End Sub




End Class
```




Form1
Multiplication Practice!
Select a Number to View its Multiplication Table:
4
Show Answers
1 x 4 = 4
2 x 4 = 8
3 x 4 = 12
4 x 4 = 16
5 x 4 = 20
6 x 4 = 24
7 x 4 = 28
8 x 4 = 32
9 x 4 = 36
10 x 4 = 40
11 x 4 = 44
12 x 4 = 48
Calculate
Clear
Exit


Form1
Multiplication Practice!
Select a Number to View its Multiplication Table:
1
Show Answers
Calculate
Clear
Exit


Form1
Multiplication Practice!
Select a Number to View its Multiplication Table:
12
Show Answers
1 x 12 = 12
2 x 12 = 24
3 x 12 = 36
4 x 12 = 48
5 x 12 = 60
6 x 12 = 72
7 x 12 = 84
8 x 12 = 96
9 x 12 = 108
10 x 12 = 120
11 x 12 = 132
12 x 12 = 144
Calculate
Clear
Exit


Form1
Multiplication Practice!
Select a Number to View its Multiplication Table:
12
Show Answers
1 x 12 =
2 x 12 =
3 x 12 =
4 x 12 =
5 x 12 =
6 x 12 =
7 x 12 =
8 x 12 =
9 x 12 =
10 x 12 =
11 x 12 =
12 x 12 =
Calculate
Clear
Exit