Catch will catch VB problems as opposed to crashing.
Crash - specifically InvalidCastException

Conversion from string "abc" to type 'Integer' is not valid.

Troubleshooting tips:
- When casting from a number, the value must be a number, not less than infinity.
- Make sure the source type is convertible to the destination type.
- General help for this exception.

InvalidCastException: Make sure your method arguments are in the right format.

Actions:
- View Details...
- Copy exception detail to the clipboard
Data not numeric error will come up instead of the answer.
Note that this actually shows Infinity.
Back up to crash to show specific error.

InvalidCastException was unhandled

Conversion from string 'abc' to type Integer is not valid.

Troubleshooting tips:
- When casting a number, the value must be a number less than infinity.
- Make sure the source type is convertible to the destination type.

Get general help for this exception.
- IntegerException: When converting a string to a DateTime, parse the string to take the data before pulling each variable into the DateTime object.
- Search for more help online...

Actions:
- View Details...
- Copy exception detail to the clipboard
Form_Load event done once when form is loaded

Private Sub usedGroupForm_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    MessageBox.Show("Please enter item Name, Price and Cost")
End Sub

Private Sub btnCalcProfit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCalcProfit.Click
    Dim dblCost, dblPrice As Double
    dblPrice = CDbl(txtPrice.Text)
    dblCost = CDbl(txtCost.Text)
    txtProfit.Text = dblPrice - dblCost
End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click
    txtPrice.Clear()
    txtCost.Clear()
    txtProfit.Clear()
End Sub

Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
    Me.Close()
End Sub
End Class
If there is a default property setting it will be overridden.

Reset to basic colors in Clear.
Public Class frmIntroFF

    Private Sub btnCheckFF_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCheckFF.Click
        If txtFirst.Text = "CI" Then
            txtResult.ForeColor = Color.AntiqueWhite
            txtResult.BackColor = Color.Red
            txtResult.Text = "CI Major"
        Else
            txtResult.ForeColor = Color.Beige
            txtResult.BackColor = Color.Brown
            txtResult.Text = "Not a CI Major"
        End If
    End Sub

    Private Sub btnClearFF_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClearFF.Click
        txtResult.Clear()
        txtFirst.Clear()
        txtSecond.Clear()
        txtResult.ForeColor = Color.White
        txtResult.BackColor = Color.Black
    End Sub

    Private Sub btnAnd_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAnd.Click
        If txtFirst.Text = "CI" Then
            If txtSecond.Text > 3.5 Then
                txtResult.Text = "Honors"
            End If
        End If
        If txtFirst.Text = "CI" And txtSecond.Text > 3.6 Then
            txtResult.Text = "Honors"
        End If
    End Sub

    Private Sub btnAndElse_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAndElse.Click
        If txtFirst.Text = "CI" Then
            If txtSecond.Text > 3.5 Then
                txtResult.Text = "Honors"
            Else
                txtResult.Text = "Not honors"
            End If
        Else
            If txtFirst.Text = "CI" And txtSecond.Text > 3.6 Then
                txtResult.Text = "Honors"
            Else
                txtResult.Text = "Not honors"
            End If
        End If
    End Sub
End Class
Can be written as compound.
When the NO processing is the same you can use compound. When the NO processing is different, you need simple IF statements rather than a compound.
Should have said
Not either to be consistent
'If txtFirst.Text = "CI" And (txtSecond.Text > 3.5 Or txtThird.Text > 30) Then
   txtResult.Text = "Meets Criteria"
'End If

'If txtFirst.Text = "CI" And txtSecond.Text > 3.5 Or txtThird.Text > 30 Then
   txtResult.Text = "Meets Criteria"
'End If

Order of operation says that ANDs are resolved before ORs. You can use parenthesis to change the order. Inside parenthesis is resolved first.
If the things circled as same are different then you cannot use compound.