This powerpoint was done using a previous version of VB.

In the general area, I dimensioned a variable for the minor total and the final total. I also set up a hold area for the dept so that I could compare and determine when a break has happened.

In the general area, I am establishing a minor total accumulator, a final total accumulator and an area to hold the department number for comparison purposes. These are established in the general area so they can be used in all subroutines.

Private Sub Form_Load()
    ...
    holddept = ""
    End Sub

When the form is loaded I set the holddept = ""

Break processing calls for setting up accumulators to hold the totals you need to accumulate and a hold area to keep the data you are comparing to in order to determine if a break has happened. That hold area should be initialized so it is empty.
Minor Break

Private Sub Form_Load()
    If Right(App.Path, 1) <> "\" Then
        Open App.Path & "\Minor.txt" For Input As #1
    Else
        Open App.Path & "\Minor.txt" For Input As #1
    End If
    If holddept = "" Then
    End Sub

The variable holddept which will be used with the break processing is set to null as defined in the general area so it is available in all Subs.

When the form is loaded, I am going to check the right most character of the path to see if it is not equal to a slash. If the slash is not there then the path to the text file being read is opened as the current path concatenated with a slash and the name of the file. If the slash is already there then the file is opened as the application path including the slash followed by the name of the file.

For the App object, Path specifies the path of the project .VBP file when running the application from the development environment or the path of the exe file when running the application as an executable file. (Quote from the help file with Visual Basic.)

The file that is being read contains a dept number and an amount. It was created in notepad. The two fields were surrounded with quotes as text fields and they are separated with commas. Each record is on a separate line.
This shows all of the code. The general area is in blue and each sub is in a separate color. We have discussed the form load, on the next slide we will look at frmProcess_Click.
This shows all of the code. The general area is in blue and each sub is in a separate color. We have discussed the form load, on the next slide we will look at fnProcClick().
Minor Total Processing

Input/Output

This is the input data - each record on the file contains a dept # and an amount.

<table>
<thead>
<tr>
<th>Dept</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>500</td>
</tr>
<tr>
<td>12</td>
<td>100</td>
</tr>
<tr>
<td>12</td>
<td>50</td>
</tr>
<tr>
<td>15</td>
<td>200</td>
</tr>
<tr>
<td>15</td>
<td>100</td>
</tr>
<tr>
<td>15</td>
<td>300</td>
</tr>
<tr>
<td>15</td>
<td>50</td>
</tr>
<tr>
<td>17</td>
<td>20</td>
</tr>
<tr>
<td>17</td>
<td>240</td>
</tr>
<tr>
<td>17</td>
<td>20</td>
</tr>
<tr>
<td>17</td>
<td>240</td>
</tr>
</tbody>
</table>

Total 15: 660
Output report that is produced contains the input records and a total each time the department number changes.

Final total for all departments: 1590

Records from department 12:
Total 12: 650
Total for department 12:

The input file is in order by department number. When the department number changes, we want to print out a total for that department before starting to process the next department.

When all of the records have been processed, we want to print out a sum of the data in all of the records. This is the same as a sum of the
This shows all of the code. The general area is in blue and each sub is in a separate color. We have discussed the form load, on the next slide we will look at frmProcess_Click().
Minor Total Processing

The input file is in order by department number. When the department number changes, we want to print out a total for that department before starting to process the next department.

When all of the records have been processed, we want to print out a sum of the data in all of the records. This is the same as a sum of the output report that is produced containing the input records and a total each time the department number changed.

<table>
<thead>
<tr>
<th>Dept #</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>500</td>
</tr>
<tr>
<td>12</td>
<td>100</td>
</tr>
<tr>
<td>12</td>
<td>50</td>
</tr>
<tr>
<td>15</td>
<td>200</td>
</tr>
<tr>
<td>15</td>
<td>100</td>
</tr>
<tr>
<td>15</td>
<td>100</td>
</tr>
<tr>
<td>15</td>
<td>50</td>
</tr>
<tr>
<td>17</td>
<td>20</td>
</tr>
<tr>
<td>17</td>
<td>240</td>
</tr>
<tr>
<td>17</td>
<td>20</td>
</tr>
<tr>
<td>17</td>
<td>240</td>
</tr>
</tbody>
</table>

Total 12: 650
Total for department 12:

Total 15: 660

Total 17: 260
Final total for all departments: 1,590

This is the input data - each record on the file contains a dept # and an amount.
Title: Mar 1-12:52 PM (8 of 29)

This shows one way that minor break processing could be handled. There are other approaches especially involved with deciding whether a break occurred or this is the first record.
Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnProcess.Click

    Dim wkdeptNo As String, wkamt As String
    If Not EOF(1) Then
        Input(1, wkdeptNo)
        Input(1, wkamt)
        If wkHoldDept <> wkDeptNo Then
            wkHoldDept = wkDeptNo
            wkMinTot = 0
        Else
            wkHoldDept = wkDeptNo
        End If
    End If
    lstMinor.Items.Add(wkdeptNo & " " & FormatCurrency(wkAmt))
    wkFinTot = wkFinTot + wkAmt
    wkFinTot = wkFinTot + wkAmt
    Else
        MsgBox("End of File", vbOKOnly, "EOF")
        lstMinor.Items.Add("Final Total: " & FormatCurrency(wkFinTot))
        FileClose(1)
    End If
End Sub

Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
Private Sub frmMinor_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles sender.Load
    FileOpen(1, "Minor.txt", OpenMode.Input)
    wkHoldDept = ""
End Sub

Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Dim wkDeptNo As String, wkAmt As String
    If Not EOF(1) Then
        Input(1, wkDeptNo)
        Input(1, wkAmt)
        If wkHoldDept <> wkDeptNo Then
            wkHoldDept = wkDeptNo
            wkMinTot = 0
        Else
            wkHoldDept = wkDeptNo
        End If
    End If
    lstMinor.Items.Add(wkDeptNo & " " & FormatCurrency(wkAmt))
    wkMinTot = wkMinTot + wkAmt
    wkFinTot = wkFinTot + wkAmt
Else
    MsgBox("End of File", vbOKOnly, "EOF")
    lstMinor.Items.Add("Final Total: " & FormatCurrency(wkFinTot))
    FileClose(1)
End If
End Sub

Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnExit.Click
End
Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Dim wkDeptNo As String, wkAmt As String
    If Not EOF(1) Then
        Input(1, wkDeptNo)
        If wkHoldDept <> wkDeptNo Then
            wkHoldDept = wkDeptNo
            wkMinTot = 0
        Else
            wkHoldDept = wkDeptNo
        End If
    Else
        lstMinor.Items.Add(wkDeptNo & "  " & FormatCurrency(wkAmt))
        wkMinTot = wkMinTot + wkAmt
        wkFinTot = wkFinTot + wkAmt
    Else
        MsgBox("End of File", vbOKOnly, "EOF")
        lstMinor.Items.Add("Final Total:  " & FormatCurrency(wkFinTot))
    End If
End Sub
<table>
<thead>
<tr>
<th>Dept</th>
<th>Process</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td></td>
<td>$500</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td>$100</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td>$200</td>
</tr>
<tr>
<td>28</td>
<td></td>
<td>$150</td>
</tr>
<tr>
<td>28</td>
<td></td>
<td>$275</td>
</tr>
</tbody>
</table>

```vbnet
wkHoldDept = wkDeptNo
End If
If wkHoldDiv <> wkDivNo Then
    lstXIM.Item.Add("Dept Total: " & vbCrLf & walletDept & vbCrLf & "Branch Total: " & vbCrLf & walletDiv & vbCrLf & "Division Total: " & vbCrLf & walletXIM & vbCrLf & "Inter-Dept Total = 0" & vbCrLf & "Major-Dept Total = 0"
Else
    ' Code for processing
End If
```
Private Sub frmMinor_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    FileOpen1("NM.txt", OpenMode.Input)
    vkHoldDept = ""
    vkHoldBr = ""
    vkHoldDiv = ""
End Sub

Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnProcess.Click
    Dim wkDivNo As String, wkBrNo As String, wkDeptNo As String, wkAmnt As String
    If Not EOF(1) Then
        Input1, wkDivNo
        Input1, wkBrNo
        Input1, wkDeptNo
        Input1, wkAmnt
        If wkHoldDiv = "" Then
            wkHoldDiv = wkDivNo
            wkHoldBr = wkBrNo
            wkHoldDept = wkDeptNo
        End If
        If wkHoldDiv <> wkDivNo Then
            lstNM.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
            lstNM.Items.Add("Branch Total: " & FormatCurrency(wkInterTot))
            lstNM.Items.Add("Division Total: " & FormatCurrency(wkMajorTot))
            wkHoldDept = wkDeptNo
            wkHoldBr = wkBrNo
            wkHoldDiv = wkDivNo
            wkMinTot = 0
            wkInterTot = 0
            wkMajorTot = 0
            Else
        End If
    End If
End Sub
Public Class frmMM

    Inherits System.Windows.Forms.Form

    Dim wkMinTot As Integer
    Dim wkInterTot As Integer
    Dim wkMajorTot As Integer
    Dim wkFinTot As Integer
    Dim wkHoldDept As String
    Dim wkHolder As String
    Dim wkHoldDiv As String

    Windows Form Designer generated code

    Private Sub frmMM_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        FileOpen(1, "MM.txt", OpenMode.Input)
        wkHoldDept = ""
        wkHolder = ""
        wkHoldDiv = ""
        End Sub

    Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnProcess.Click
        Dim wkDivNo As String, wkStkNo As String, wkDeptNo As String, wkName As String
        If Not EOF(1) Then
            Input(1, wkDivNo)
            Input(1, wkStkNo)
            Input(1, wkDeptNo)
            Input(1, wkName)
        If wkHoldDiv = "" Then
            wkHoldDiv = wkDivNo
            wkHoldDr = wkStkNo
            wkHoldDept = wkDeptNo
        End If
        If wkHoldDiv <> wkDivNo Then

Start
Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Dim wkidNo As String, wkstrNo As String, wkdeptNo As String, wkamt As String
    If Not EOF(1) Then
        Input 1, wkkidNo
        Input 1, wkstrNo
        Input 1, wkdeptNo
        Input 1, wkamt
        If wkstrDiv = "" Then
            wkstrDiv = wkstrNo
            wkstrDept = wkdeptNo
        End If
        If wkstrDiv <> wkstrNo Then
            lstKIM.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
            lstKIM.Items.Add("Branch Total: " & FormatCurrency(wkInterTot))
            lstKIM.Items.Add("Division Total: " & FormatCurrency(wkMajorTot))
            wkstrDept = wkdeptNo
            wkstrDiv = wkstrNo
        End If
        If wkstrBr <> wkstrNo Then
            lstKIM.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
            lstKIM.Items.Add("Branch Total: " & FormatCurrency(wkInterTot))
            lstKIM.Items.Add("Division Total: " & FormatCurrency(wkMajorTot))
            wkstrDept = wkdeptNo
            wkstrBr = wkstrNo
            wkminTot = 0
            wkinterTot = 0
        Else
            Else
        End If
    End If
End Sub
If wHoldDiv <> wkDivNo Then
    lstMIN.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
    lstMIN.Items.Add("Branch Total: " & FormatCurrency(wkIntTot))
    lstMIN.Items.Add("Division Total: " & FormatCurrency(wkMajorTot))
    wkHoldDept = wkDeptNo
    wkHoldBr = wkBrNo
    wkHoldDiv = wkDivNo
    wkMinTot = 0
    wkIntTot = 0
    wkMajorTot = 0
Else
    If wkHoldBr <> wkBrNo Then
        lstMIN.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
        lstMIN.Items.Add("Branch Total: " & FormatCurrency(wkIntTot))
        wkHoldDept = wkDeptNo
        wkHoldBr = wkBrNo
        wkHoldDiv = wkDivNo
        wkMinTot = 0
        wkIntTot = 0
    Else
        If wkHoldDept <> wkDeptNo Then
            lstMIN.Items.Add("Dept Total: " & FormatCurrency(wkMinTot))
            wkHoldDept = wkDeptNo
            wkMinTot = 0
        End If
    End If
Else
    End If
End If
End If
Let MIN.Items.Add(wkDivNo & " & wkBrNo & " & wkDeptNo & " & FormatCurrency(wkMinTot) + wName
wkIntTot = wkIntTot + wCount
wkMajorTot = wkMajorTot + wCount
wkFinTot = wkFinTot + wCount
Error on flowchart - missing department break decision.
Public Class frmMIM
    Inherits System.Windows.Forms.Form
    Dim wkMinTot As Integer
    Dim wkInterTot As Integer
    Dim wkMajorTot As Integer
    Dim wkFinTot As Integer
    Dim wkHoldDept As String
    Dim wkHoldDiv As String
    Dim wkHoldBr As String

    Windows Form Designer generated code

    Private Sub frmMIM_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        FileOpen(1, "MIM.txt", OpenMode.Input)
        wkHoldBr = "" "wkHoldDiv = ""
    End Sub

    Private Sub btnProcess_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnProcess.Click
        Dim wkDivNo As String, wkBrNo As String, wkDeptNo As String, wkamt /
        If Not EOF(1) Then
            Input(1, wkDivNo)
            Input(1, wkBrNo)
            Input(1, wkDeptNo)
        End If
        IF wkHoldDiv = "" Then
            wkHoldDiv = wkDivNo
            wkHoldBr = wkBrNo
            wkHoldDept = wkDeptNo
    End If