Sub FormatExcel()

'Deletes blank rows

On Error Resume Next

Columns("B").SpecialCells(xlCellTypeBlanks).EntireRow.Delete

a = Cells(Rows.Count, 1).End(xlUp).Row

'Copies broken data due to formatting issues, places them in correct cells

For i = 3 To a

For j = 10 To 15

If Cells(i, 1) <> Cells(i - 1, 1) And IsEmpty(Cells(i, 8)) Then

Cells(i, j - 8).Copy Cells(i - 1, j)

End If

Next j

Next i

Application.CutCopyMode = False

Cells(1, 1).Select

'Deletes the copied rows

For i = 3 To a

If Cells(i, 1) <> Cells(i - 1, 1) And IsEmpty(Cells(i - 1, 8)) Then

Cells(i, 1).EntireRow.Delete

End If

Next i

'Counts the number of rows and deletes Unavailable assignments from the last row to the first

a = Cells(Rows.Count, 1).End(xlUp).Row

For i = a To 2 Step -1

If Cells(i, 14) = "Unavailable" Or IsEmpty(Cells(i, 10)) Then

Cells(i, 1).EntireRow.Delete

End If

Next i

'Deletes unnecessary columns

Columns(15).EntireColumn.Delete

Columns(13).EntireColumn.Delete

Columns(12).EntireColumn.Delete

Columns(10).EntireColumn.Delete

Columns(8).EntireColumn.Delete

Columns(7).EntireColumn.Delete

Columns(6).EntireColumn.Delete

Columns(5).EntireColumn.Delete

Columns(4).EntireColumn.Delete

'Creates proper column headers, declares variables

a = Cells(Rows.Count, 1).End(xlUp).Row

Dim TextStrng As String

Dim Result() As String

Dim TimeResult As Variant

Dim Invoice As Variant

Cells(1, 7) = "StartDate"

Cells(1, 8) = "StartTime"

Cells(1, 9) = "EndDate"

Cells(1, 10) = "EndTime"

Cells(1, 11) = "ElapsedTime(hours)"

Cells(1, 12) = "InvoiceAmount"

Cells(1, 13) = "OriginalInfo"

Cells(1, 14) = "Summary"

'Formats and fills in the Start Date

For i = 2 To a

TextStrng = Cells(i, 2)

Result() = Split(TextStrng)

Cells(i, 7) = Result()

Next i

'Formats and fills in the Start Time

For j = 2 To a

TextStrng = Cells(j, 2)

TimeResult = Format(Right(TextStrng, 11), "Short Time")

Cells(j, 8) = TimeResult

Next j

'Formats and fills in the End Date

For k = 2 To a

TextStrng = Cells(k, 3)

Result() = Split(TextStrng)

Cells(k, 9) = Result()

Next k

'Formats and fills in the End Time

For m = 2 To a

TextStrng = Cells(m, 3)

TimeResult = Format(Right(TextStrng, 11), "Short Time")

Cells(m, 10) = TimeResult

Next m

'Calculates elapsed time between start time and end time

For n = 2 To a

Cells(n, 11) = DateDiff("s", Cells(n, 2), Cells(n, 3)) / 3600

Next n

'Grabs the invoice amount from the description, usually formatted correctly, but formats the irregular ones into a dollar amount

For p = 2 To a

If Left(Cells(p, 4), 1) <> "$" Then

Invoice = FormatCurrency(Cells(p, 4))

Cells(p, 12) = Invoice

Else

TextStrng = Cells(p, 4)

Result() = Split(TextStrng)

Cells(p, 12) = Result()

TextStrng = Cells(p, 12)

Invoice = Right(TextStrng, Len(TextStrng) - 1)

Cells(p, 12) = FormatCurrency(CDec(Invoice))

End If

Next p

'Copies over original info

For q = 2 To a

Cells(q, 13) = Cells(q, 4)

Next q

'Copies over summary info

For r = 2 To a

Cells(r, 14) = Cells(r, 6)

Next r

'Deletes unnecessary columns

Columns(6).EntireColumn.Delete

Columns(5).EntireColumn.Delete

Columns(4).EntireColumn.Delete

Columns(3).EntireColumn.Delete

Columns(2).EntireColumn.Delete

'Sets up admin hours for part 2

Columns("H:H").Insert Shift:=xlToRight, \_

CopyOrigin:=xlFormatFromLeftOrAbove

Cells(1, 8) = "AdminHours?"

'Invoice number to be manually input later, per request of client

Columns("H:H").Insert Shift:=xlToRight, \_

CopyOrigin:=xlFormatFromLeftOrAbove

Cells(1, 8) = "InvoiceNumber"

'Formats all dates to US standard

a = Cells(Rows.Count, 1).End(xlUp).Row

For i = 2 To a

Cells(i, 2) = Format(Cells(i, 2), "mm/dd/yyyy")

Cells(i, 4) = Format(Cells(i, 4), "mm/dd/yyyy")

Next i

'Sorts sheet by name, then by date

With ActiveSheet.Sort

.SortFields.Add Key:=Range("A1"), Order:=xlAscending

.SortFields.Add Key:=Range("B1"), Order:=xlAscending

.SetRange ActiveSheet.UsedRange

.Header = xlYes

.Apply

End With

'Gets a count of all the unique calendars

a = Cells(Rows.Count, 1).End(xlUp).Row

b = Application.Sheets.Count

n = 1 'Unique calendars

For i = 3 To a

If Cells(i, 1) <> Cells(i - 1, 1) Then

n = n + 1

End If

Next i

'Creates a new worksheet for every calendar and copies over header info

For b = Application.Sheets.Count To n - 1

Sheets.Add After:=Sheets(Sheets.Count)

Sheets(b).Range("A1:K1").Copy Sheets(b + 1).Range("A1")

Next b

Sheets(1).Activate

'On each new worksheet, copies over each row that does not match the topmost entry, deletes that row from original worksheet

For Z = 1 To b - 1

For i = a To 2 Step -1

If Sheets(Z).Cells(i, 1) <> Sheets(Z).Cells(2, 1) Then

Cells(i, 1).EntireRow.Copy Destination:=Sheets(Z + 1).Range("A" & Rows.Count).End(xlUp).Offset(1)

Sheets(Z).Activate

Cells(i, 1).EntireRow.Delete

End If

Next i

Next Z

'Sorts new worksheet and formats it attractively

For Z = 1 To b

Sheets(Z).Activate

With ActiveSheet.Sort

.SortFields.Add Key:=Range("A1"), Order:=xlAscending

.SortFields.Add Key:=Range("B1"), Order:=xlAscending

.SetRange ActiveSheet.UsedRange

.Header = xlYes

.Apply

End With

Columns("A:K").AutoFit

Range("A1", Range("A1").End(xlDown)).RowHeight = 15

Range("J:J").ColumnWidth = 25

ActiveSheet.Name = Range("a2")

Next Z

'Lets user check for errors based on most common errors

MsgBox "Please double check for and correct the following:" & vbNewLine & "Non-dollar or negative amounts in Column G" & vbNewLine & "Strange symbols in Column J" & vbNewLine & "If the amount shown in Column G does not match its corresponding value in Column J" & vbNewLine & "Finally, put a lowercase 'x' next in Column I for any task that should be designated as administrative hours"

End Sub

Sub summary()

'Adds new worksheet before the first worksheet, adds headers

Worksheets.Add before:=Worksheets(1)

Cells(1, 1) = "Summary Data"

Cells(2, 1) = "Employee Name"

Cells(2, 2) = "Commission (Enter as Decimal)"

Cells(2, 3) = "Admin Hours"

Cells(2, 4) = "Walking/Sitting Hours"

Cells(2, 5) = "Total Hours Worked"

Cells(2, 6) = "Total Pay"

'Creates a new row for every sheet

For i = 2 To Sheets.Count

Cells(i + 1, 1) = Sheets(i).Name

'Asks user input for commission rate, adds commission next to Username; if they opt to exit, quits sub

answer = InputBox("What is the Commission for " & Cells(i + 1, 1) & "? (Please enter as a decimal i.e. 0.67)" & vbNewLine & vbNewLine & "If you do not want to run calculations, type 'exit'.")

If IsNumeric(answer) Then

Cells(i + 1, 2) = answer

Else: ActiveSheet.Delete

Exit Sub

End If

'Calculates Admin hours based on x mark

Dim ws As Worksheet

Set ws = Sheets(i)

Cells(i + 1, 3) = Application.WorksheetFunction.SumIf(ws.Range("I:I"), "<>", ws.Range("F:F"))

'Calculates total hours

Cells(i + 1, 5) = Application.WorksheetFunction.Sum(ws.Range("F:F"))

'Calculates walking hours by subtracting admin from total

Cells(i + 1, 4) = Cells(i + 1, 5) - Cells(i + 1, 3)

'Calculates total pay by multiplying admin hours by 8.25 and all other invoices without x by commission

Cells(i + 1, 6) = (Cells(i + 1, 3) \* 8.25) + Application.WorksheetFunction.SumIf(ws.Range("I:I"), "", ws.Range("G:G")) \* Cells(i + 1, 2)

'Formats pay into currency

Cells(i + 1, 6) = CCur(Cells(i + 1, 6))

Next i

'Formats whole sheet

Columns("A:F").AutoFit

Range("A1:F1").Merge

Range("A1:F1").HorizontalAlignment = xlCenter

Range("A1").RowHeight = 30

Range("A1").Font.Size = 20

Range("A1:F1").Style = "Heading 1"

ActiveSheet.Name = Range("a1")

End Sub