

RE: Get parameter dates in heading

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.externaldata/2006-10/msg00186.html>

- *From:* TKM <TKM@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Thu, 19 Oct 2006 14:03:02 -0700
-

Hey it works pretty good! Now if I want that code to run everytime a user wants to see a report (the macro I did was formatting) where do I put it so it will format the exported report automatically when the spreadsheet opens? Does it go into Access or Excel? And if it goes in Excel where do you put it. Again I want to thank you for all your help. You have given me a great advance on this problem project! Thanks again

"Klatuu" wrote:

The code I posted is in a form. It is the form from which the user runs the report. She has the option to Print, Preview, or Export to Excel. This is how I produce the Excel version.

As I said, it is a lot of code, but it sure makes pretty Excel reports. You don't put anything in Excel. This code, in fact does not start with an existing workbook. It creates it.

The two biggest hurdles to get across are managing the xobjects and understanding the Excel Object Model. If you use the techniques for creating xobjects in Access, closing and releasing them, you wont have to worry about that. The main problem you will have is that if you don't handle the xl objectst correctly, you can leave an instance of Excel in memory. The way you will know it is there is if you try to open Excel or an Excel spreadsheet, it will hang. You can open Task Manager, select the Processes tab and see Excel.exe. You will have to delete it. Then Excel will run. What happens is either you did not correctly destroy the object references correctly or you did not establish the references correctly. If you reference an xl object and Access can't figure out what the parent object is, it will create an additional instance of Excel. That instance will not be closed when you quite Excel in your code.

The other is the Excel object model. There is a lot of it. Two tricks that will help. Use the Object browser in VBA to explore it and you will find a lot about what can be done and how to reference it. The coolest trick, however, is if you are not sure how to do it in code, open Excel, start recording a Macro, do what you want to do, stop the recording. Then open the Macro for editing and copy the code it wrote. You will have to modify it a

RE: Get parameter dates in heading

little to use in Access, but it is a good way to learn and sometimes a good short cut when there is a lot to do.

"TKM" wrote:

wHolly smokes. Do I put this in Excel or the Access code and for either one where would it go? Thanks again for your help!

"Klatuu" wrote:

It is a lot of VBA code and you have to learn the Excel object model. There is a bit of a learning curve, but once you get it down, it alllows you to make some really nice Excel based reports. I hope it all fits. If the last line is not End Sub, tell me how much you got and I'll send the rest.

```
Sub Build_XL_Report(strOutPut As String)
Const conLightGray As Long = 12632256
Const conLightBlue As Long = 16777164
Const conLightYellow As Long = 10092543
```

```
Dim xlApp As Object 'Application Object
Dim xlBook As Object 'Workbook Object
Dim xlSheet As Object 'Worksheet Object
Dim varGetFileName As Variant 'File Name with Full Path
Dim rstSCCB As Recordset 'Recordset to load data from
Dim rstItms As Recordset 'Recordset to load ITM Name in
Header
Dim qdf As QueryDef 'Query def to load data
Dim lngItmCount As Long 'Number of ITMs in the
RecordSet
Dim lngDetailCount As Long 'Number of Detail Data rows
in the recordset
Dim intX As Integer 'Loop Counter
Dim strMonth As String 'Used to create a Short month name
ie
January to Jan
Dim strCurrItm As String 'Hold the ITM Name to format
Total cell
Dim lngRowCount As Long 'A loop counter that gives the
current row
reference
Dim lngTotalPos As Long 'Used to format ITM Total cells
Dim strPrintArea As String 'Defines the print area for the
sheet
Dim strTitleRows As String 'Defines the rows to print at the
```

RE: Get parameter dates in heading

```
top of
each page
Dim strLeftRange As String 'Used to format range references
Dim strRightRange As String 'Used to format range
references
Dim lngFirstDataRow As Long 'The first row with detail
data
Dim lngLastDataRow As Long 'The last row with detail data
Dim blnExcelWasNotRunning As Boolean
Dim strDefaultDir 'Where to save spreadsheet
Dim strDefaultFileName 'Name to Save as
Dim lngFlags As Long 'Flags for common dialog
Dim strFilter As String 'File Display for Common Dialog
Dim strCurrMonth As String 'To create directory name for
save
Dim strCurrYear As String 'To create directory name for
save
Dim blnStopXl As Boolean 'Leave Open for Spreadsheet
Version

On Error GoTo Build_XL_Report_ERR

DoCmd.Hourglass (True)
Me.txtStatus = "Updating Queries"
Me.txtStatus.Visible = True

'Set up the necessary objects
On Error Resume Next ' Defer error trapping.
Set xlApp = GetObject(, "Excel.Application")
If Err.Number <> 0 Then
blnExcelWasNotRunning = True
Set xlApp = CreateObject("excel.application")
Else
DetectExcel
End If
Err.Clear ' Clear Err object in case error occurred.
On Error GoTo Build_XL_Report_ERR
xlApp.DisplayAlerts = False
xlApp.Interactive = False
xlApp.ScreenUpdating = False
Set xlBook = xlApp.Workbooks.Add

Me.txtStatus = "Building Workbook"
Me.Repaint

'Remove excess worksheets
Do While xlBook.Worksheets.Count > 1
xlApp.Worksheets(xlApp.Worksheets.Count).Delete
Loop
Set xlSheet = xlBook.ActiveSheet
```

RE: Get parameter dates in heading

```
'Build The Spreadsheet
'Build The Headers
Me.txtStatus = "Creating Headers"
Me.Repaint

strMonth = Left(Me.cboPeriod.Column(1), 3)
xlSheet.Name = Me.cboResource & " Hours " & strMonth &
" YTD"
With xlSheet
.Cells(1, 1) = "ITM"
.Cells(1, 2) = Me.txtCurrYear & _
" Activity # Description"
.Cells(1, 3) = "Budget " & Me.txtCurrYear
.Cells(1, 4).Value = Me.txtCurrYear & " YTD Budget"
.Cells(1, 5) = "Actuals YTD"
.Cells(1, 6) = "Variance YTD"
.Cells(1, 7) = "TO GO"
.Cells(1, 8) = Iif(Me.cboPeriod >= 1, "JAN ACT", "JAN
ETC")
.Cells(1, 9) = Iif(Me.cboPeriod >= 2, "FEB ACT", "FEB
ETC")
.Cells(1, 10) = Iif(Me.cboPeriod >= 3, "MAR ACT", "MAR
ETC")
.Cells(1, 11) = Iif(Me.cboPeriod >= 4, "APR ACT", "APR
ETC")
.Cells(1, 12) = Iif(Me.cboPeriod >= 5, "MAY ACT", "MAY
ETC")
.Cells(1, 13) = Iif(Me.cboPeriod >= 6, "JUN ACT", "JUN
ETC")
.Cells(1, 14) = Iif(Me.cboPeriod >= 7, "JUL ACT", "JUL
ETC")
.Cells(1, 15) = Iif(Me.cboPeriod >= 8, "AUG ACT", "AUG
ETC")
.Cells(1, 16) = Iif(Me.cboPeriod >= 9, "SEP ACT", "SEP
ETC")
.Cells(1, 17) = Iif(Me.cboPeriod >= 10, "OCT ACT", "OCT
ETC")
.Cells(1, 18) = Iif(Me.cboPeriod >= 11, "NOV ACT", "NOV
ETC")
.Cells(1, 19) = Iif(Me.cboPeriod >= 12, "DEC ACT", "DEC
ETC")
End With
'Format Row 1
With xlSheet
For Each cell In xlSheet.Range("A1", "S1")
cell.Font.Size = 10
cell.Font.Name = "Arial"
cell.Font.Bold = True
cell.Interior.Color = conLightGray
cell.HorizontalAlignment = xlHAlignCenter
cell.WrapText = True
```

RE: Get parameter dates in heading

```
Next
.Cells(1, 2).HorizontalAlignment = xlHAlignLeft
.Columns("A").ColumnWidth = 9
.Columns("B").ColumnWidth = 39
.Columns("C:S").ColumnWidth = 9
.Rows(1).RowHeight = 25.5
End With

'Set Up Recordset for ITM Header data
Me.txtStatus = "Loading ITM Data"
Me.Repaint

Set qdf = CurrentDb.QueryDefs("qselSCCBhdr")
qdf.Parameters(0) = Me.cboResource
qdf.Parameters(1) = Me.cboPeriod
Set rstItms = qdf.OpenRecordset(dbOpenSnapshot,
dbReadOnly)
'Be sure there are records to process
rstItms.MoveLast
rstItms.MoveFirst
lngItmCount = rstItms.RecordCount
If lngItmCount = 0 Then
MsgBox "No Data Found For This Report", vbInformation +
vbOKOnly,
"Data Error"
GoTo Build_XL_Report_Exit
End If

'Load Header Data
xlSheet.Cells(2, 1).CopyFromRecordset rstItms
rstItms.Close
Set rstItms = Nothing
Set qdf = Nothing

'Format the ITM Name Cells
Me.txtStatus = "Formatting Headers"
Me.Repaint

With xlSheet
For Each cell In xlSheet.Range("A2", "A" &
Trim(str(lngItmCount + 2)))
cell.Font.Size = 10
cell.Font.Name = "Arial"
cell.Font.Bold = True
cell.Interior.Color = conLightGray
cell.HorizontalAlignment = xlHAlignLeft
cell.WrapText = False
Next
End With

'Merge the ITM Cells
```

RE: Get parameter dates in heading

```
For intX = 2 To lngItmCount + 2
strLeftRange = "A" & Trim(str(intX)) & ":B" &
Trim(str(intX))
xlSheet.Range(strLeftRange).MergeCells = True
Next intX

'Size the Blank Row
xlSheet.Rows(lngItmCount + 3).RowHeight = 30

'Format Header Area and put in formulas
With xlSheet
For intX = 2 To lngItmCount + 1
strLeftRange = "C" & Trim(str(intX))
strRightRange = "S" & Trim(str(intX))
For Each cell In xlSheet.Range(strLeftRange, strRightRange)
cell.Font.Size = 10
cell.Font.Name = "Arial"
cell.Font.Bold = True
cell.Interior.Color = conLightBlue
cell.NumberFormat = "##,###,##0_);[Red](##,###,##0)"
Next
Next intX
'Do The Grand Total Row
strLeftRange = "C" & Trim(str(intX))
strRightRange = "S" & Trim(str(intX))
For Each cell In xlSheet.Range(strLeftRange, strRightRange)
cell.Font.Size = 10
cell.Font.Name = "Arial"
cell.Font.Bold = True
cell.Interior.Color = conLightYellow
cell.Formula = "= Grand"
cell.NumberFormat = "##,###,##0_);[Red](##,###,##0)"
Next
End With

'Put Borders around the Header Area
With xlSheet.Range("A1", "S" & Trim(str(lngItmCount +
2)))
.Borders(xlTop).LineStyle = xlContinuous
.Borders(xlTop).Weight = xlThin
.Borders(xlBottom).LineStyle = xlContinuous
.Borders(xlBottom).Weight = xlThin
.Borders(xlLeft).LineStyle = xlContinuous
.Borders(xlLeft).Weight = xlThin
.Borders(xlRight).LineStyle = xlContinuous
.Borders(xlRight).Weight = xlThin
End With

'Add Total to ITM Names
For intX = 2 To lngItmCount + 1
xlSheet.Cells(intX, 1) = "Grand Total " &
```

RE: Get parameter dates in heading

```
xlSheet.Cells(intX, 1)
Next intX
xlSheet.Cells(intX, 1) = "Grand Total " & _
Me.cboResource & " HOURS"

'Copy the Header Row to the top of the Data Area
xlSheet.Range("A1:S1").Copy _
Destination:=xlSheet.Range("A" & Trim(str(intX + 2)))

'Load the Data
Me.txtStatus = "Loading Detail Data"
Me.Repaint

Set qdf = CurrentDb.QueryDefs("qselSCCB rpt")
qdf.Parameters(0) = Me.cboResource
qdf.Parameters(1) = Me.cboPeriod
Set rstSCCB = qdf.OpenRecordset(dbOpenSnapshot,
dbReadOnly)
xlSheet.Cells(intX + 3, 1).CopyFromRecordset rstSCCB
lngDetailCount = rstSCCB.RecordCount
rstSCCB.Close
Set rstSCCB = Nothing
Set qdf = Nothing

'Put in the SubTotals
Me.txtStatus = "Creating Subtotals"
Me.Repaint

lngFirstDataRow = intX + 3
lngLastDataRow = lngFirstDataRow + lngItmCount +
lngDetailCount
With xlSheet
.Range(.Cells(lngFirstDataRow - 1, 1), _
.Cells(lngLastDataRow, 19)).Subtotal groupBy:=1,
Function:=xlSum, _
totalList:=Array(3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,
16, 17, 18, 19)
End With

'Create Formulas and range names
For lngRowCount = lngFirstDataRow To lngLastDataRow
lngTotalPos = InStr(xlSheet.Cells(lngRowCount, 1), "Total")
If lngTotalPos = 0 Then 'Column S needs to be light yellow
if not a
total row
xlSheet.Cells(lngRowCount, 5).Interior.Color =
conLightYellow
xlSheet.Cells(lngRowCount, 6).Interior.Color =
conLightYellow
Else
strCurrItm = Left(xlSheet.Cells(lngRowCount, 1),
```

RE: Get parameter dates in heading

```
lngTotalPos - 2)
With xlSheet
.Range("C" & Trim(str(lngRowCount)) & ":S" & _
Trim(str(lngRowCount))).Name = strCurrItm
.Range("A" & Trim(str(lngRowCount)) & ":S" & _
Trim(str(lngRowCount))).Interior.Color = conLightGray
End With
End If
Next lngRowCount
```