

```

Rem =====
Rem A_Query_Account Analysis OoBasic Program (Copyright:Shuichi Sunaga)
Rem (2005/10/31:301)Division into PstBKAmts(), PstWHTax(), PstOTAmts()
Rem Added CF_Account Analysis(2006/04/10:415)
Rem Discarded PstBKAmts()Adopted FindALL(2006/05/15:400)FindCode() (2009/09/22:524)
Rem LenseJump(2011/12/10:465)BkChrg(2012/07/06:555)English Translation(2012/08/12:509)
Rem -----
Rem Sub DirectANL()
Rem Sub AcctAnlys(ByVal iStrtMo%)
Rem Sub FindPost(ByVal selActCD$, ByVal iStrtMo%)
Rem Sub PstOTAmts(WkWk As Object, iStrtMo, FKAcct$, oSearchDesc, oSRange, Msigma, fWHC$)
Rem Function ColSum&(ByVal iStrtMo%, ByVal smCol%)
Rem Sub PstTaxes(WkWk As Object,iStrtMo%,StartR%, Mword$, Msigma%)
Rem Sub PstBegBal(ByVal iStrtMo%, FKAcct$) (Added 2007/04/12 with DbClick_Account Analysis)
Rem Sub PstWHTax(ByVal iStrtRow&, ByVal iEndRow&, ByVal iCol%)
Rem Sub PstDptAmt(ByVal WkWk As object, ByVal iRow&, ByVal Msigma%.pWHC$)
Rem Sub ANNextLn(ByVal fLnRow&)
Rem Sub PutTitle()
Rem Function BankChrg(ByVal iStrtMo%) as long
Rem Function WithHTax(ByVal iStrtMo%) as long
Rem Sub DptTitle()
Rem Sub DirectLJump()
Rem =====
Option Explicit
Sub DirectAcctAnlys()
    Dim oSelection As Object, oCellRangeAddress As Object
    Dim sAddr$, sCol%, eCol%, sMonth%, eMonth% 'addr.Sheet addr.Row
    Dim sm$
    IntroTo3D()
    setCol4All()
    oSelection = ThisComponent.CurrentController.Selection
    oCellRangeAddress = ThisComponent.createInstance("com.sun.star.sheet.SheetCellRanges")
    oCellRangeAddress.InsertByName( "", oSelection )
    sAddr$ = oCellRangeAddress.getRangeAddressesAsString()
    sAddr$= Right(sAddr$, len(sAddr$)-lnStr(1, sAddr$, ".", 1))
    sCol% = CNumero(sAddr$)
    sAddr$= Right(sAddr$, len(sAddr$)-lnStr(1, sAddr$, ".", 1))
    eCol% = CNumero(sAddr$)
    If StopOrGo("Acct_Analysis") then
        If sCol% = CFCol then
            If StrtMo% <> GetDate("Beg","FYM") Then
                sm$ = CStr(StrtMo% - Int(StrtMo%/100)*100)
                MsgTxt$ = "The starting month of Account Analysis differs from "& chr(13)
                MsgTxt$ = MsgTxt$ & "the beginning day of the fiscal year."& chr(13)
                MsgTxt$ = MsgTxt$ & "OK to execute Analysis with this starting month ("& sm$ & ") ?"
                MsgTtl$ = "Confirmation of starting month of Analysis"
                resp% = MsgBox(MsgTxt$, 289, MsgTtl$)
                If resp% = 2 Then
                    End
                End If
            End If
            AcctAnlys StrtMo%, cMonth%
        Elseif sCol% > RMCOL And sCol% <= CBS_YB%-13 Then
            MsgTxt$ = "The starting month of Account Analysis falls within " & chr(13)
            MsgTxt$ = MsgTxt$ & "the fiscal year preceding the prior one."
            MsgTtl$ = "Test of starting month of Account Analysis"
            resp% = MsgBox(MsgTxt$, 0, MsgTtl$)
            End
        Elseif sCol% > CBS_YB%-13 Then
            If sCol% < CBS_YB% Then
                MsgTxt$ = "The Account Analysis with he starting month which "&chr(13)
                MsgTxt$ = MsgTxt$ & "falls within the prior fiscal year could be incomplete."&chr(13)
                MsgTxt$ = MsgTxt$ & "OK to try this command for some information ?"
                MsgTtl$ = "Account Analysis including prior fiscal period"
                resp% = MsgBox(MsgTxt$, 289, MsgTtl$)
                If resp% = 2 Then
                    End
                End If
            End If
        End If
        eCol%=myMin(eCol%, CBS_YE%+3)
    End Sub

```

```

1 Rem =====
2 Rem A_Query_Account Analysis Excel VBA Program (Copyright:Shuichi Sunaga)
3 Rem
4 Rem BkChrg(2012/07/06:555)
5 Rem LenseJump(2012/07/07)(473))
6 Rem Translation into English(2012/08/12:555)
7 Rem -----
8 Rem Sub DirectANL() '分析
9 Rem Sub AcctAnlys(ByVal iStrtMo%)
10 Rem Sub FindPost(ByVal selActCD$, ByVal iStrtMo%)
11 Rem Sub PstOTAmts(WkWk As Object, iStrtMo, FKAcct$, oSearchDesc, oSRange, Msigma, fWHC$)
12 Rem Function ColSum&(ByVal iStrtMo%, ByVal smCol%)
13 Rem Sub PstTaxes(WkWk As Object,iStrtMo%,StartR%, Mword$, Msigma%)
14 Rem Sub PstBegBal(ByVal iStrtMo%, FKAcct$) (Added 2007/04/12 with DbClick_Account Analysis)
15 Rem Sub PstWHTax(ByVal iStrtRow&, ByVal iEndRow&, ByVal iCol%)
16 Rem Sub PstDptAmt(ByVal WkWk As object, ByVal iRow&, ByVal Msigma%.pWHC$)
17 Rem Sub ANNextLn(ByVal fLnRow&)
18 Rem Sub PutTitle()
19 Rem Function BankChrg(ByVal iStrtMo%) as long
20 Rem Function WithHTax(ByVal iStrtMo%) as long
21 Rem Sub DptTitle()
22 Rem Sub DirectLJump()
23 Rem =====
24 Option Explicit
25 Sub DirectAcctAnlys()
26     Dim sCol%, eCol%, sMonth%, eMonth%
27     Dim sm$
28
29     IntroTo3D
30     setCol4All
31
32
33
34
35
36     sCol% = Selection.Column
37
38     eCol% = sCol% + Selection.Columns.Count - 1
39     If StopOrGo("Acct_Analysis") Then
40         If sCol% = CFCol Then
41             If StrtMo% <> GetDate("Beg","FYM") Then
42                 sm$ = CStr(StrtMo% - Int(StrtMo% / 100) * 100)
43                 MsgTxt$ = "The starting month of Account Analysis differs from " & Chr(13)
44                 MsgTxt$ = MsgTxt$ & "the beginning day of the fiscal year." & Chr(13)
45                 MsgTxt$ = MsgTxt$ & "OK to execute Analysis with this starting month (" & sm$ & ") ?"
46                 MsgTtl$ = "Confirmation of starting month of Analysis"
47                 resp% = MsgBox(MsgTxt$, 289, MsgTtl$)
48                 If resp% = 2 Then
49                     End
50                 End If
51             End If
52             AcctAnlys StrtMo%, cMonth%
53         Elseif sCol% > RMCOL And sCol% <= CBS_YB% - 13 Then
54             MsgTxt$ = "The starting month of Account Analysis falls within " & Chr(13)
55             MsgTxt$ = MsgTxt$ & "the fiscal year preceding the prior one."
56             MsgTtl$ = "Test of starting month of Account Analysis"
57             resp% = MsgBox(MsgTxt$, 0, MsgTtl$)
58             End
59         Elseif sCol% > CBS_YB% - 13 Then
60             If sCol% < CBS_YB% Then
61                 MsgTxt$ = "The Account Analysis with he starting month which " & Chr(13)
62                 MsgTxt$ = MsgTxt$ & "falls within the prior fiscal year could be incomplete." & Chr(13)
63                 MsgTxt$ = MsgTxt$ & "OK to try this command for some information ?"
64                 MsgTtl$ = "Account Analysis including prior fiscal period"
65                 resp% = MsgBox(MsgTxt$, 289, MsgTtl$)
66                 If resp% = 2 Then
67                     End
68                 End If
69             End If
70             eCol% = myMin(eCol%, CBS_YE% + 3)
        End Sub

```

```

        sMonth% = Val(Right(CellString(WkBS, CNRow%, sCol%), 6))-190000
        eMonth% = Val(Right(CellString(WkBS, CNRow%, eCol%), 6))-190000
        AcctAnlys sMonth%, eMonth%
    Else
        AcctAnlys cMonth%, cMonth%
    End If
Else
    MsgBox("Acct_Analysis")
End If
End Sub
Sub AcctAnlys(ByVal iStrtMo%, ByVal iEndMo%)
    Dim AcCode$, ANLWS$
    ANLWS$ = aSName$
    Rem -----(Preparation)-----
    WkLN.Unprotect ("pwd")
    AcCode$ = CellString(aSheet, SelectionRow(), 2)
    SetCellString WkLN, 1, 2, AcCode$
    SetCellString WkLN, 1, 3, CellString(aSheet, SelectionRow(), 4)
    aController.ActiveSheet = WkLN
    Rem =====
    If CellString(WkLN, 2, 9) <> "JE Amount" Then
        PutTitle()
    End If
    If ANLWS = "CF" then
        CFindPost AcCode$, iStrtMo%, iEndMo%
    Else
        FindPost AcCode$, iStrtMo%, iEndMo%, "ActCD"
    End If
    ThisComponent.calculateAll()
End Sub
Sub FindPost(ByVal selActCD$, ByVal gStrtMo%, gEndMo%)
    Dim qdbMonthTop&, qdbMonthBtm&
    Dim i%, fCol_Act%, BKChrg&, tCTX&, tVal&, mFormula$
    ClearRange WkLN, 3, 1, 3000, 10 + NumDpt
    WkLN.getCellRangeByPosition(6, 0, 6, 0).Columns.IsVisible = False
    LnRow& = 3
    If aSName$ = "BS" And A_VBSCD(selActCD$)Then
        PstBegBal gStrtMo%, selActCD$
    End If
    Rem ----- Dafine Data Range Subject to Account Analysis -----
    SetSrcRange "ValMo", gStrtMo%, gEndMo%, "ActCD",
    Rem ----- From BK -----
    PstOTAMts WkBK, gStrtMo%, gEndMo%, selActCD$, oBKSDesc, BKRange, "BP"
    If selActCD$ = CRTXCD Then
        PstTxs WkBK, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Bank Book", 1
    End If
    Rem ----- From JV -----
    PstOTAMts WkJV, gStrtMo%, gEndMo%, selActCD$, oJVSDesc, JVRRange, "BP"
    If selActCD$ = CRTXCD Then
        LnRow& = LnRow& - 1
        PstTxs WkJV, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Journal Voucher", 1
    End If
    Rem ----- From AP -----
    If APUse then
        PstOTAMts WkAP, gStrtMo%, gEndMo%, selActCD$, oAPSDesc, APRRange, "BP"
        If selActCD$ = CRTXCD Then
            LnRow& = LnRow& - 1
            PstTxs WkAP, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Accounts Payable Register", 1
        End If
    End If
    Rem ----- From AR -----
    If ARUse then
        PstOTAMts WkAR, gStrtMo%, gEndMo%, selActCD$, oARSDesc, ARRange, "BP"
        If selActCD$ = PYTXCD Then
            LnRow& = LnRow& - 1
            PstTxs WkAR, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Accounts Receivable Register"
        End If
    End If
    Rem ----- From MW -----
    If MWUse then

```

```

71         sMonth% = Val(Right(CellString(WkBS, CNRow%, sCol%), 6)) - 190000
72         eMonth% = Val(Right(CellString(WkBS, CNRow%, eCol%), 6)) - 190000
73         AcctAnlys sMonth%, eMonth%
74     Else
75         AcctAnlys cMonth%, cMonth%
76     End If
77 Else
78     MsgBox ("Acct_Analysis")
79 End If
80 End Sub
81 Sub AcctAnlys(ByVal iStrtMo%, ByVal iEndMo%)
82     Dim AcCode$, ANLWS$
83     ANLWS$ = aSName$
84     Rem -----(Preparation)-----
85     WkLN.Unprotect ("pwd")
86     AcCode$ = CellString(aSheet, SelectionRow(), 2)
87     SetCellString WkLN, 1, 2, AcCode$
88     SetCellString WkLN, 1, 3, CellString(aSheet, SelectionRow(), 4)
89     WkLN.Activate
90     Rem =====
91     If CellString(WkLN, 2, 9) <> "JE Amount" Then
92         PutTitle()
93     End If
94     If ANLWS = "CF" Then
95         CFindPost AcCode$, iStrtMo%, iEndMo%
96     Else
97         FindPost AcCode$, iStrtMo%, iEndMo%, "ActCD"
98     End If
99     Calculate
100 End Sub
101 Sub FindPost(ByVal selActCD$, ByVal gStrtMo%, gEndMo%, colNM$)
102     Dim qdbMonthTop&, qdbMonthBtm&
103     Dim i%, fCol_Act%, BKChrg&, tCTX&, tVal&, mFormula$
104     ClearRange WkLN, 3, 1, 3000, 10 + NumDpt
105     WkLN.Columns(7).Hidden = True
106     LnRow& = 3
107     If aSName$ = "BS" And A_VBSCD(selActCD$) Then
108         PstBegBal gStrtMo%, selActCD$
109     End If
110     Rem ----- Dafine Data Range Subject to Account Analysis -----
111     SetSrcRange "ValMo", gStrtMo%, gEndMo%, "ActCD"
112     Rem ----- From BK -----
113     PstOTAMts WkBK, gStrtMo%, gEndMo%, selActCD$, BKRange, "BP"
114     If selActCD$ = CRTXCD Then
115         PstTxs WkBK, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Bank Book", 1
116     End If
117     Rem ----- From JV -----
118     PstOTAMts WkJV, gStrtMo%, gEndMo%, selActCD$, JVRRange, "BP"
119     If selActCD$ = CRTXCD Then
120         LnRow& = LnRow& - 1
121         PstTxs WkJV, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Journal Voucher", 1
122     End If
123     Rem ----- From AP -----
124     If APUse Then
125         PstOTAMts WkAP, gStrtMo%, gEndMo%, selActCD$, APRRange, "BP"
126         If selActCD$ = CRTXCD Then
127             LnRow& = LnRow& - 1
128             PstTxs WkAP, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Accounts Payable Register", 1
129         End If
130     End If
131     Rem ----- From AR -----
132     If ARUse Then
133         PstOTAMts WkAR, gStrtMo%, gEndMo%, selActCD$, ARRange, "BP"
134         If selActCD$ = PYTXCD Then
135             LnRow& = LnRow& - 1
136             PstTxs WkAR, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Accounts Receivable Register", -1
137         End If
138     End If
139     Rem ----- From MW -----
140     If MWUse Then

```

```

PstOTAMts WkMW, gStrtMo%, gEndMo%, selActCD$, oMWSDesc, MWRRange, "BP"
If selActCD$ = PYTXCD Then
    LnRow& = LnRow& -1
    PstTass WkMW, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Prepayments Register", -1
End If
End If
Rem -----(Post analysis procedure)-----
If selActCD$ = WHTaxCD then
    WkLN.getCellRangeByPosition(6, 0, 6, 0).Columns.IsVisible = True
    SetSrcRange "ValMo", gStrtMo%, gEndMo%, "WHT" 'BKのみ
    PstWHTax gStrtMo%, gEndMo%, "BP"
    Rem -----(The starting row of sorting is 4
    SortLN(4)
End If
If selActCD$ = BKChgCD Then
    PostBKChrg gStrtMo%, gEndMo%, "BP"
End If
SelectCell LnRow&, 6
SetCellString WkLN, LnRow&, 5, "Total"
If LnRow& > 3 Then
    For i% = 6 To 10 + NumDpt
        mFormula$ = "=SUM(" & CAlpha(i%) & "3:" & CAlpha(i%) & Cstr(LnRow& -1) & ")"
        SetCellFormula WkLN, LnRow&, i%, mFormula$
    Next i%
End If
End Sub
Sub PstOTAMts(ByVal WkWk As Object, iStrtMo%, iEndMo%, FkAcct$, oSearchDesc as object, oSRange as ob
    Dim mSigma%, sgm0%, Col_WhcMo%, pSName$
    Dim LnAcct$, LnMonth%, tmpVal&, mFormula$
    Dim oFound as object, oFLength%, i%, j%, xRow&
    pSName$ = WkWk.getname()
    getColArray(pSName$)
    mSigma% = 1
    Col_WhcMo% = Col_ValMo%
    If fWHC$ = "CF" then
        mSigma% = -1
        If pSName$ = "AP" Or pSName$ = "AR" Or pSName$ = "MW" then
            Col_WhcMo% = Col_PaidMo%
        end if
    End If
    oSearchDesc.SearchString = FkAcct$
    oSearchDesc.SearchBackwards = False
    Set oFound = oSRange.findAll(oSearchDesc)
    If IsNull(oFound) then
        Exit Sub
    Endif
    oFLength% = oFound.getCount()
    For i% = 0 to oFLength%-1
        xRow& = oFound.getByIndex(i%).RangeAddress.StartRow + 1
        for j% = 1 to oFound.getByIndex(i%).Rows.Count
            LnMonth% = Val(CellString(WkWk, xRow&, Col_WhcMo%))
            LnAcct$ = CellString(WkWk, xRow&, Col_ActCD%)
            If fWHC$ = "CF" And pSName$="JV" And CellString(WkWk, xRow&, Col_LineNo%+1) <> "Y" Then
                Rem JV entries are reflected in CF statement only when marked Y next to Line No (2011/12/
            Elseif LnAcct$ = FkAcct$ And LnMonth% >= iStrtMo% And LnMonth% <= iEndMo% Then
                If aSName$ = "ANL" Then
                    ANNextLn(LnRow&)
                end if
                SelectCell LnRow&, 1
                SetCellString WkLN, LnRow&, 1, WkWk.Name + "-" + CStr(xRow&)
                SetCellString WkLN, LnRow&, 2, CellString(WkWk, xRow&, Col_ValMo%)
                SetCellString WkLN, LnRow&, 4, CellString(WkWk, xRow&, Col_Payer%)
                SetCellString WkLN, LnRow&, 5, CellString(WkWk, xRow&, Col_Descr%)
                If WkWk.Name = "BK" Then
                    tmpVal& = -CellValue(WkBK, xRow&, Col_Gross% - 1) + CellValue(WkBK, xRow&, Col
                    SetCellValue WkLN, LnRow&, 6, tmpVal& * mSigma%
                    If Col_CXCr% <> Col_Border% then
                        tmpVal& = CellValue(WkBK, xRow&, Col_CXCr%) + CellValue(WkBK, xRow&, Col
                    Else
                        tmpVal& = CellValue(WkBK, xRow&, Col_CXDr%)

```

```

PstOTAMts WkMW, gStrtMo%, gEndMo%, selActCD$, MWRRange, "BP"
If selActCD$ = PYTXCD Then
    LnRow& = LnRow& - 1
    PstTass WkMW, gStrtMo%, gEndMo%, qdbMonthTop&, "Total from Prepayments Register", -1
End If
End If
Rem -----(Post analysis procedure)-----
If selActCD$ = WHTaxCD Then
    WkLN.Columns(7).Hidden = False
    SetSrcRange "ValMo", gStrtMo%, gEndMo%, "WHT" 'BKのみ
    PstWHTax gStrtMo%, gEndMo%, "BP"
    Rem -----(The starting row of sorting is 4
    SortLN (4)
End If
If selActCD$ = BKChgCD Then
    PostBKChrg gStrtMo%, gEndMo%, "BP"
End If
SelectCell LnRow&, 6
SetCellString WkLN, LnRow&, 5, "Total"
If LnRow& > 3 Then
    For i% = 6 To 10 + NumDpt
        mFormula$ = "=SUM(R[-" & Cstr(LnRow& - 3) & "]C:R[-1]C)"
        SetCellFormula WkLN, LnRow&, i%, mFormula$
    Next i%
End If
End Sub
Sub PstOTAMts(ByVal WkWk As Worksheet, iStrtMo%, iEndMo%, FkAcct$, oSRange As Object, fWHC$)
    Dim mSigma%, sgm0%, Col_WhcMo%, pSName$
    Dim LnAcct$, LnMonth%, tmpVal&, mFormula$
    Dim oFound As Object, oFLength%, i%, j%, xRow&, firstAddress
    pSName$ = WkWk.Name
    getColArray (pSName$)
    mSigma% = 1
    Col_WhcMo% = Col_ValMo%
    If fWHC$ = "CF" Then
        mSigma% = -1
        If pSName$ = "AP" Or pSName$ = "AR" Or pSName$ = "MW" Then
            Col_WhcMo% = Col_PaidMo%
        End If
    End If
    Set oFound = oSRange.Find(FkAcct$, Lookat:=xlWhole, SearchDirection:=xlNext)
    If oFound Is Nothing Then
        Exit Sub
    End If
    firstAddress = oFound.Address
    Do
        xRow& = oFound.Row
        LnMonth% = Val(CellString(WkWk, xRow&, Col_WhcMo%))
        LnAcct$ = CellString(WkWk, xRow&, Col_ActCD%)
        If fWHC$ = "CF" And pSName$ = "JV" And CellString(WkWk, xRow&, Col_LineNo% + 1) <> "Y" Then
            Rem JV entries are reflected in CF statement only when marked Y next to Line No (2011/12/17)
        Elseif LnAcct$ = FkAcct$ And LnMonth% >= iStrtMo% And LnMonth% <= iEndMo% Then
            If aSName$ = "ANL" Then
                ANNextLn (LnRow&)
            End If
            SelectCell LnRow&, 1
            SetCellString WkLN, LnRow&, 1, WkWk.Name + "-" + CStr(xRow&)
            SetCellString WkLN, LnRow&, 2, CellString(WkWk, xRow&, Col_ValMo%)
            SetCellString WkLN, LnRow&, 4, CellString(WkWk, xRow&, Col_Payer%)
            SetCellString WkLN, LnRow&, 5, CellString(WkWk, xRow&, Col_Descr%)
            If WkWk.Name = "BK" Then
                tmpVal& = -CellValue(WkBK, xRow&, Col_Gross% - 1) + CellValue(WkBK, xRow&, Col_Gross%)
                SetCellValue WkLN, LnRow&, 6, tmpVal& * mSigma%
                If Col_CXCr% <> Col_Border% Then
                    tmpVal& = CellValue(WkBK, xRow&, Col_CXCr%) + CellValue(WkBK, xRow&, Col_CXDr%)
                Else
                    tmpVal& = CellValue(WkBK, xRow&, Col_CXDr%)

```

```

End If
SetCellValue WkLN, LnRow&, 8, tmpVal& * mSigma%
SetCellValue WkLN, LnRow&, 9, CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
Elseif WkWK.Name = "JV" Then
SetCellValue WkLN, LnRow&, 6, CellValue(WkJV, xRow&, Col_Gross%)* mSigma%
tmpVal& = CellValue(WkJV, xRow&, Col_CXCr%)+ CellValue(WkJV, xRow&, Col_CXDr%)* mSigma%
SetCellValue WkLN, LnRow&, 8, tmpVal& * mSigma%
SetCellValue WkLN, LnRow&, 9, CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
Elseif WkWK.Name = "AP" Then
SetCellString WkLN, LnRow&, 3, CellString(WkWK, xRow&, Col_PaidMo%)
sgm0% = myIIF(CStr(FKAcct) = CStr(APCD), -1, 1)
If AP_UQP = True Then
SetCellValue WkLN, LnRow&, 6, sgm0% * CellValue(WkAP, xRow&, Col_Gross%)* mSigma%
Else
SetCellValue WkLN, LnRow&, 6, sgm0% * CellValue(WkAP, xRow&, Col_Gross%)* mSigma%
End If
SetCellValue WkLN, LnRow&, 8, sgm0% * CellValue(WkAP, xRow&, Col_CXDr%)* mSigma%
SetCellValue WkLN, LnRow&, 9, sgm0% * CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
Elseif WkWK.Name = "AR" Then
SetCellString WkLN, LnRow&, 3, CellString(WkWK, xRow&, Col_PaidMo%)
sgm0% = myIIF(CStr(FKAcct) = CStr(ARCD), 1, -1)
If AR_UQP = True Then
SetCellValue WkLN, LnRow&, 6, sgm0% * CellValue(WkAR, xRow&, Col_Gross%)* mSigma%
Else
SetCellValue WkLN, LnRow&, 6, sgm0% * CellValue(WkAR, xRow&, Col_Gross%)* mSigma%
End If
SetCellValue WkLN, LnRow&, 8, sgm0% * CellValue(WkAR, xRow&, Col_CXCr%)* mSigma%
SetCellValue WkLN, LnRow&, 9, sgm0% * CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
Elseif WkWK.Name = "MW" Then
SetCellString WkLN, LnRow&, 3, CellString(WkWK, xRow&, Col_PaidMo%)
SetCellValue WkLN, LnRow&, 6, CellValue(WkMW, xRow&, Col_Amount%)* mSigma%
SetCellValue WkLN, LnRow&, 9, CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
End If
If bAllocDpt and A_VallocCD(FkAcct) Then
PstDptAmt(WkWK, xRow&, fWHC$)
End If
mFormula$ = "=H" & CStr(LnRow&) & "-int(F" & CStr(LnRow&) & ""*5/105)"
SetCellFormula WkLN, LnRow&, 10 + NumDpt, mFormula$
LnRow& = LnRow& + 1
End If
xRow& = xRow& + 1

next j%
Next i%
End Sub
Sub PstTaxes(ByVal WkWK As Object, ByVal iStrtMo%, ByVal StartR%, Mword$, Msigma%)
dim tvMonth%, fRow&, Col_CTax%
dim CTXamt&, BKChrg&
Col_CTax% = myIIF(WkWK.Name="AR", Col_CXCr%, Col_CXDr%)
fRow& = StartR
CTXamt& = 0
Do While fRow& < dbBtmBorder&
tvMonth = CellValue(WkWK, fRow&, Col_ValMo%)
If tvMonth >= iStrtMo% And tvMonth <= cMonth% Then
CTXamt& = CTXamt& + CellValue(WkWK, fRow&, Col_CTax%)
End If
fRow& = fRow& + 1
Loop
If aSName$ = "ANL" Then
ANNNextLn(LnRow&)
End If
SetCellString WkLN, LnRow&, 1, WkWK.Name
SetCellString WkLN, LnRow&, 2, cMonth%
SetCellString WkLN, LnRow&, 5, Mword
SetCellValue WkLN, LnRow&, 6, mSigma% * CTXamt&
SetCellValue WkLN, LnRow&, 9, mSigma% * CTXamt&
LnRow& = LnRow& + 1
If WkWK.Name = "BK" then

```

```

End If
SetCellValue WkLN, LnRow&, 8, tmpVal& * mSigma%
SetCellValue WkLN, LnRow&, 9, CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
Elseif WkWK.Name = "JV" Then
SetCellValue WkLN, LnRow&, 6, CellValue(WkJV, xRow&, Col_Gross%)* mSigma%
tmpVal& = CellValue(WkJV, xRow&, Col_CXCr%) + CellValue(WkJV, xRow&, Col_CXDr%)* mSigma%
SetCellValue WkLN, LnRow&, 8, tmpVal& * mSigma%
SetCellValue WkLN, LnRow&, 9, CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
Elseif WkWK.Name = "AP" Then
SetCellString WkLN, LnRow&, 3, CellString(WkWK, xRow&, Col_PaidMo%)
sgm0% = myIIF(CStr(FkAcct) = CStr(APCD), -1, 1)
If AP_UQP = True Then
SetCellValue WkLN, LnRow&, 6, sgm0% * CellValue(WkAP, xRow&, Col_Gross%)* mSigma%
Else
SetCellValue WkLN, LnRow&, 6, sgm0% * CellValue(WkAP, xRow&, Col_Gross%)* mSigma%
End If
SetCellValue WkLN, LnRow&, 8, sgm0% * CellValue(WkAP, xRow&, Col_CXDr%)* mSigma%
SetCellValue WkLN, LnRow&, 9, sgm0% * CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
Elseif WkWK.Name = "AR" Then
SetCellString WkLN, LnRow&, 3, CellString(WkWK, xRow&, Col_PaidMo%)
sgm0% = myIIF(CStr(FkAcct) = CStr(ARCD), 1, -1)
If AR_UQP = True Then
SetCellValue WkLN, LnRow&, 6, sgm0% * CellValue(WkAR, xRow&, Col_Gross%)* mSigma%
Else
SetCellValue WkLN, LnRow&, 6, sgm0% * CellValue(WkAR, xRow&, Col_Gross%)* mSigma%
End If
SetCellValue WkLN, LnRow&, 8, sgm0% * CellValue(WkAR, xRow&, Col_CXCr%)* mSigma%
SetCellValue WkLN, LnRow&, 9, sgm0% * CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
Elseif WkWK.Name = "MW" Then
SetCellString WkLN, LnRow&, 3, CellString(WkWK, xRow&, Col_PaidMo%)
SetCellValue WkLN, LnRow&, 6, CellValue(WkMW, xRow&, Col_Amount%)* mSigma%
SetCellValue WkLN, LnRow&, 9, CellValue(WkWK, xRow&, Col_Amount%)* mSigma%
End If
If bAllocDpt And A_VallocCD(FkAcct) Then
PstDptAmt WkWK, xRow&, fWHC$
End If
mFormula$ = "=RC[-" & CStr(NumDpt + 2)
mFormula$ = mFormula$ & "]"-Int(RC[-" & CStr(NumDpt + 4) & "]"/1.05*0.05)"
SetCellFormula WkLN, LnRow&, 10 + NumDpt, mFormula$
LnRow& = LnRow& + 1
End If
Set oFound = oSRange.FindNext(oFound)
If oFound Is Nothing Then
Exit Do
End If
Loop While oFound.Address <> firstAddress
End Sub
Sub PstTaxes(ByVal WkWK As Object, iStrtMo%, iEndMo%, StartR%, Mword$, mSigma%)
dim tvMonth%, fRow&, Col_CTax%
dim CTXamt&, BKChrg&
Col_CTax% = myIIF(WkWK.Name = "AR", Col_CXCr%, Col_CXDr%)
fRow& = StartR
CTXamt& = 0
Do While fRow& < dbBtmBorder&
tvMonth = CellValue(WkWK, fRow&, Col_ValMo%)
If tvMonth >= iStrtMo% And tvMonth <= iEndMo% Then
CTXamt& = CTXamt& + CellValue(WkWK, fRow&, Col_CTax%)
End If
fRow& = fRow& + 1
Loop
If aSName$ = "ANL" Then
ANNNextLn (LnRow&)
End If
SetCellString WkLN, LnRow&, 1, WkWK.Name
SetCellString WkLN, LnRow&, 2, cMonth%
SetCellString WkLN, LnRow&, 5, Mword
SetCellValue WkLN, LnRow&, 6, mSigma% * CTXamt&
SetCellValue WkLN, LnRow&, 9, mSigma% * CTXamt&
LnRow& = LnRow& + 1
If WkWK.Name = "BK" Then

```

```

if aSName$ = "ANL" Then
    ANNextLn(LnRow&)
end if
BKChrg& = RangeSum&(WkBK, Col_BKChg%, dbMonthTop&, Col_BKChg%, dbMonthBtm&)
CTXAmt& = Int(BKChrg& / 105 * 5)
SetCellString WkLn, LnRow&, 1, WkWk.Name
SetCellString WkLn, LnRow&, 2, cMonth%
MsgTxt$ = "Bank Transfer Charges"
SetCellString WkLn, LnRow&, 4, MsgTxt$
MsgTxt$ = "VAT including rounding difference"
SetCellString WkLn, LnRow&, 5, MsgTxt$
SetCellValue WkLn, LnRow&, 6, mSigma% * CTxAmt&
SetCellValue WkLn, LnRow&, 9, mSigma% * CTxAmt&
LnRow& = LnRow& + 1
End If
End Sub
Sub PstBegBal(ByVal iStrtMo%, FKAcct$)
    Dim tVal&, Col_RB$, fCol%, mWD$
    Col_RB$ = "Col_" & CStr(iStrtMo%+190000)
    fCol% = Col_InFR%("BS", Col_RB$) - 1
    tVal& = CellValue(WkBK, getFSRow$(WkBK, FKAcct$, ActCol), fCol%)
    mWD$ = "Beginning Balance"
    SetCellString WkLn, LnRow&, 5, mWD$
    SetCellValue WkLn, LnRow&, 9, tVal&
    LnRow& = LnRow& + 1
End Sub
Sub PstWHTax(ByVal iStrtMo%, iEndMo%, fWHC$)
    Dim i%, j%, xRow&, mSigma%, oFLength%, LnMonth%
    Dim LnAcct$, tmpVal&, mFormula$, WHTaxAmt&
    Dim oFound as object
    mSigma% = myIIF(fWHC$ = "CF", -1, 1)
    oBKSDesc.SearchRegularExpression = true
    Rem regular expression for other than space,tab,CR,LF,double size space
    oBKSDesc.SearchString = "[^ \t\r\n000d\r000a\r3000\r] +"
    Set oFound = BKRRange.findAll(oBKSDesc)
    If IsNull(oFound) Then
        Exit Sub
    End If
    getColArray("BK")
    oFLength% = oFound.getCount()
    For i% = 0 to oFLength% - 1
        xRow& = oFound.getByIndex(i%).RangeAddress.StartRow + 1
        for j% = 1 to oFound.getByIndex(i%).Rows.Count
            LnMonth% = Val(CellString(WkBK, xRow&, Col_ValMo%))
            WHTaxAmt& = CellValue(WkBK, xRow&, Col_WHT%)
            If WHTaxAmt& <> 0 And LnMonth% >= iStrtMo% And LnMonth% <= iEndMo% Then
                if aSName$ = "ANL" Then
                    ANNextLn(LnRow&)
                end if
                SelectCell LnRow&, 1
                SetCellString WkLn, LnRow&, 1, "BK" & "-" & CStr(xRow&)
                SetCellString WkLn, LnRow&, 2, CellString(WkBK, xRow&, Col_ValMo%)
                SetCellString WkLn, LnRow&, 4, CellString(WkBK, xRow&, Col_Descr% - 1)
                SetCellString WkLn, LnRow&, 5, CellString(WkBK, xRow&, Col_Descr%)
                if fWHC$ = "CF" then
                    SetCellValue WkLn, LnRow&, 6, WHTaxAmt& * mSigma%
                else
                    tmpVal& = -CellValue(WkBK, xRow&, Col_Gross% - 1) + CellValue(WkBK, xRow&, Col_
                    SetCellValue WkLn, LnRow&, 6, tmpVal&
                    If Col_CXCr% <> Col_Border% then
                        tmpVal& = mSigma% * CellValue(WkBK, xRow&, Col_CXCr%)
                        tmpVal& = tmpVal& + CellValue(WkBK, xRow&, Col_CXDr%)
                    Else
                        tmpVal& = CellValue(WkBK, xRow&, Col_CXDr%)
                    End If
                    SetCellValue WkLn, LnRow&, 8, tmpVal&
                end if
                SetCellValue WkLn, LnRow&, 7, WHTaxAmt& * mSigma%
                SetCellValue WkLn, LnRow&, 9, WHTaxAmt& * mSigma%
                mFormula$ = "=" & CAlpha(8)& Cstr(LnRow&)& "-int(" & CAlpha(6)& Cstr(LnRow&)& ""5/105)

```

```

281 If aSName$ = "ANL" Then
282     ANNextLn (LnRow&)
283 End If
284 BKChrg& = RangeSum&(WkBK, Col_BKChg%, dbMonthTop&, Col_BKChg%, dbMonthBtm&)
285 CTxAmt& = Int(BKChrg& / 105 * 5)
286 SetCellString WkLn, LnRow&, 1, WkWk.Name
287 SetCellString WkLn, LnRow&, 2, cMonth%
288 MsgTxt$ = "Bank Transfer Charges"
289 SetCellString WkLn, LnRow&, 4, MsgTxt$
290 MsgTxt$ = "VAT including rounding difference"
291 SetCellString WkLn, LnRow&, 5, MsgTxt$
292 SetCellValue WkLn, LnRow&, 6, mSigma% * CTxAmt&
293 SetCellValue WkLn, LnRow&, 9, mSigma% * CTxAmt&
294 LnRow& = LnRow& + 1
295 End If
296 End Sub
297 Sub PstBegBal(ByVal iStrtMo%, FkAcct$)
298     Dim tVal&, Col_RB$, fCol%, mWD$
299     Col_RB$ = "Col_" & CStr(iStrtMo% + 190000)
300     fCol% = Col_InFR%("BS", Col_RB$) - 1
301     tVal& = CellValue(WkBK, getFSRow$(WkBK, FkAcct$, ActCol), fCol%)
302     mWD$ = "Beginning Balance"
303     SetCellString WkLn, LnRow&, 5, mWD$
304     SetCellValue WkLn, LnRow&, 9, tVal&
305     LnRow& = LnRow& + 1
306 End Sub
307 Sub PstWHTax(ByVal iStrtMo%, iEndMo%, fWHC$)
308     Dim i%, j%, xRow&, mSigma%, Col_WHTax%, oFLength%, LnMonth%
309     Dim LnAcct$, tmpVal&, mFormula$, WHTaxAmt&
310     mSigma% = myIIF(fWHC$ = "CF", -1, 1)
311     SeekTBRow& WkBK, "ValMo", iStrtMo%, iEndMo%, "Top"
312
313
314
315
316
317
318
319 getColArray ("BK")
320 xRow& = dbMonthTop&
321 Do While xRow& <= dbMonthBtm&
322     LnMonth% = Val(CellString(WkBK, xRow&, Col_ValMo%))
323     WHTaxAmt& = CellValue(WkBK, xRow&, Col_WHT%)
324
325
326 If WHTaxAmt& <> 0 And LnMonth% >= iStrtMo% And LnMonth% <= iEndMo% Then
327     If aSName$ = "ANL" Then
328         ANNextLn (LnRow&)
329     End If
330     SelectCell LnRow&, 1
331     SetCellString WkLn, LnRow&, 1, "BK" & "-" & CStr(xRow&)
332     SetCellString WkLn, LnRow&, 2, CellString(WkBK, xRow&, Col_ValMo%)
333     SetCellString WkLn, LnRow&, 4, CellString(WkBK, xRow&, Col_Descr% - 1)
334     SetCellString WkLn, LnRow&, 5, CellString(WkBK, xRow&, Col_Descr%)
335     If fWHC$ = "CF" Then
336         SetCellValue WkLn, LnRow&, 6, WHTaxAmt& * mSigma%
337     Else
338         tmpVal& = -CellValue(WkBK, xRow&, Col_Gross% - 1) + CellValue(WkBK, xRow&, Col_Gross%)
339         SetCellValue WkLn, LnRow&, 6, tmpVal&
340         If Col_CXCr% <> Col_Border% Then
341             tmpVal& = mSigma% * CellValue(WkBK, xRow&, Col_CXCr%)
342             tmpVal& = tmpVal& + CellValue(WkBK, xRow&, Col_CXDr%)
343         Else
344             tmpVal& = CellValue(WkBK, xRow&, Col_CXDr%)
345         End If
346         SetCellValue WkLn, LnRow&, 8, tmpVal&
347     End If
348     SetCellValue WkLn, LnRow&, 7, WHTaxAmt& * mSigma%
349     SetCellValue WkLn, LnRow&, 9, WHTaxAmt& * mSigma%
350     mFormula$ = "=RC[-" & CStr(NumDpt + 2) & "]" - Int(RC[-"

```



```

        SetCellFormula WkLN, LnRow&, 10 + NumDpt, mFormula$
        LnRow& = LnRow& + 1
    End If
    xRow& = xRow& + 1
next j%
Next i%
End Sub
Sub PostBKChrg(ByVal fStrtMo%, fEndMo%, fWHC$)
    Rem ----- Current month summation of bank transaction and consumption tax ----
    Dim aArgs(0) As Variant
    Dim oFound As Object, tBKRRange As Object, oRange As Object
    Dim MsgTxt1$, MsgTxt2$, mSigma%
    Dim iStrRow&, iEndRow&, fMonth%
    Dim BKChgAmt&, CTXAmt&
    Dim MBKC%, ColName$, mFormula$
    Rem ----- Deletion of bank charges row -----
    mSigma% = myIIF(fWHC$ = "CF", -1, 1)
    MsgTxt1$ = "Bank transfer charges - automatic journal entry"
    MsgTxt2$ = "Currnet month BK Charges"
    Rem ----- Calculation of BK Charges by bank -----
    Col_BKChg% = Col_InArray("BK", "BKChg")
    Col_BKCD% = Col_InArray("BK", "BKCD")
    fMonth% = fStrtMo%
    Do While fMonth% <= fEndMo%
        targetRow& = SeekTbRow&(WkBk, "ValMo", fMonth%, fMonth%, "Top")
        RNM$ = CAlpha(Col_BKCD%) & Cstr(dbMonthTop%) & cjc$
        RNM$ = RNM$ & CAlpha(Col_BKCD%) & Cstr(dbMonthBtm%)
        Set tBKRRange = WkBk.getCellRangeByName(RNM$)
        iEndRow& = dbMonthTop&-1
        Do While True
            MBKC% = CellValue(WkBk, iEndRow& + 1, Col_BKCD%)
            iStrRow& = getFirstRow&(tBKRRange, Cstr(MBKC%))
            iEndRow& = getLastRow&(tBKRRange, Cstr(MBKC%))
            SelectCell LnRow&, 1
            Rem ===== (Bank transfer charges by Bank)=====
            ColName$ = CAlpha(Col_BKChg%)
            RNM$ = ColName$ & Cstr(iStrRow&) & cjc$ & ColName$ & Cstr(iEndRow&)
            aArgs(0) = WkBk.getCellRangeByName( RNM$)
            BKChgAmt& = oFunctionAccess.callFunction("SUM", aArgs() )
            CTXAmt& = Int((BKChgAmt& / (100 + CTXRate)) * CTXRate)
            If BKChgAmt& <> 0 Then
                SetCellString WkLN, LnRow&, 1, "BK" & "-" & Cstr(iStrRow&)
                SetCellString WkLN, LnRow&, 2, Cstr(fMonth%)
                SetCellString WkLN, LnRow&, 4, MsgTxt1$
                SetCellString WkLN, LnRow&, 5, MsgTxt2$ & "(" & Cstr(MBKC%) & ")"
                SetCellValue WkLN, LnRow&, 6, BKChgAmt& * mSigma%
                SetCellValue WkLN, LnRow&, 8, CTXAmt& * mSigma
                SetCellValue WkLN, LnRow&, 9, (BKChgAmt& - CTXAmt&) * mSigma%
            If IsMultiDP Then
                SetCellValue WkLN, LnRow&, 9+NumDpt, (BKChgAmt& - CTXAmt&) * mSigma%
            End If
            mFormula$ = "=" & CAlpha(8)& Cstr(LnRow&)
            mFormula$ = mFormula$ & "-int(" & CAlpha(6)& Cstr(LnRow&)& "5/105)"
            SetCellFormula WkLN, LnRow&, 10 + NumDpt, mFormula$
            LnRow& = LnRow& + 1
        End If
        If iEndRow& + 1 > dbMonthBtm& Then
            Exit Do
        End If
    Loop
    fMonth% = NxtMonth(fMonth%)
Loop
End Sub
Sub PstDptAmt(ByVal WkWk As object, ByVal iRow&, pWHC$)
    dim i%, mSigma%, cfSigma%
    mSigma% = IIF(WkWk.Name = "AR", -1, 1)
    cfSigma% = IIF(pWHC$ = "CF", -1, 1)
    For i% = 1 To NumDpt
        If CellValue(WkWk, iRow&, Col_Amount% + i%) = 0 Then
            SetCellValue WkLN, LnRow&, 9 + i%, ""

```

```

351         mFormula$ = mFormula$ & Cstr(NumDpt + 4) & "/1.05*0.05)"
352         SetCellFormula WkLN, LnRow&, 10 + NumDpt, mFormula$
353         LnRow& = LnRow& + 1
354     End If
355     xRow& = xRow& + 1
356 Loop
357 End Sub
358 Sub PostBKChrg(ByVal fStrtMo%, fEndMo%, fWHC$)
359     Rem ----- Current month summation of bank transaction and consumption tax ----
360
361     Dim oFound As Object, tBKRRange As Range, oRange As Range
362     Dim MsgTxt1$, MsgTxt2$, mSigma%
363     Dim iStrRow&, iEndRow&, fMonth%
364     Dim BKChgAmt&, CTXAmt&
365     Dim MBKC%, ColName$, mFormula$
366     Rem ----- Deletion of bank charges row -----
367     mSigma% = myIIF(fWHC$ = "CF", -1, 1)
368     MsgTxt1$ = "Bank transfer charges - automatic journal entry"
369     MsgTxt2$ = "Currnet month BK Charges"
370     Rem ----- Calculation of BK Charges by bank -----
371     Col_BKChg% = Col_InArray("BK", "BKChg")
372     Col_BKCD% = Col_InArray("BK", "BKCD")
373     fMonth% = fStrtMo%
374     Do While fMonth% <= fEndMo%
375         targetRow& = SeekTbRow&(WkBk, "ValMo", fMonth%, fMonth%, "Top")
376         RNM$ = CAlpha(Col_BKCD%) & Cstr(dbMonthTop%) & "-"
377         RNM$ = RNM$ & CAlpha(Col_BKCD%) & Cstr(dbMonthBtm%)
378         Set tBKRRange = WkBk.Range(RNM$)
379         iEndRow& = dbMonthTop& - 1
380         Do While True
381             MBKC% = CellValue(WkBk, iEndRow& + 1, Col_BKCD%)
382             iStrRow& = getFirstRow&(tBKRRange, Cstr(MBKC%))
383             iEndRow& = getLastRow&(tBKRRange, Cstr(MBKC%))
384             SelectCell LnRow&, 1
385             Rem ===== (Bank transfer charges by Bank)=====
386             ColName$ = CAlpha(Col_BKChg%)
387             RNM$ = ColName$ & Cstr(iStrRow&) & "-" & ColName$ & Cstr(iEndRow&)
388             Set oRange = WkBk.Range(RNM$)
389             BKChgAmt& = Application.WorksheetFunction.Sum(oRange)
390             CTXAmt& = Int((BKChgAmt& / (100 + CTXRate)) * CTXRate)
391             If BKChgAmt& <> 0 Then
392                 SetCellString WkLN, LnRow&, 1, "BK" & "-" & Cstr(iStrRow&)
393                 SetCellString WkLN, LnRow&, 2, Cstr(fMonth%)
394                 SetCellString WkLN, LnRow&, 4, MsgTxt1$
395                 SetCellString WkLN, LnRow&, 5, MsgTxt2$ & "(" & Cstr(MBKC%) & ")"
396                 SetCellValue WkLN, LnRow&, 6, BKChgAmt& * mSigma%
397                 SetCellValue WkLN, LnRow&, 8, CTXAmt& * mSigma
398                 SetCellValue WkLN, LnRow&, 9, (BKChgAmt& - CTXAmt&) * mSigma%
399             If IsMultiDP Then
400                 SetCellValue WkLN, LnRow&, 9 + NumDpt, (BKChgAmt& - CTXAmt&) * mSigma%
401             End If
402             mFormula$ = "=RC[-" & Cstr(NumDpt + 2) & "]"-Int(RC[-"
403             mFormula$ = mFormula$ & Cstr(NumDpt + 4) & "/1.05*0.05)"
404             SetCellFormula WkLN, LnRow&, 10 + NumDpt, mFormula$
405             LnRow& = LnRow& + 1
406         End If
407         If iEndRow& + 1 > dbMonthBtm& Then
408             Exit Do
409         End If
410     Loop
411     fMonth% = NxtMonth(fMonth%)
412 Loop
413 End Sub
414 Sub PstDptAmt(ByVal WkWk As Object, ByVal iRow&, pWHC$)
415     Dim i%, mSigma%, cfSigma%
416     mSigma% = IIF(WkWk.Name = "AR", -1, 1)
417     cfSigma% = IIF(pWHC$ = "CF", -1, 1)
418     For i% = 1 To NumDpt
419         If CellValue(WkWk, iRow&, Col_Amount% + i%) = 0 Then
420             SetCellValue WkLN, LnRow&, 9 + i%, 0

```

```

Else
    SetCellValue WkLN, LnRow&, 9+i%, CellValue(WkWk,iRow&,Col_Amount%+i%)*mSigma%*cfSig
End If
Next i%
End Sub
Sub ANNextLn(ByVal fLnRow&)
    Dim oCellRange as object, mFormula$
    Do While CellValue(WkAN, fLnRow&, 2) <> cMonth%
        If IsEmpty(CellString(WkAN, fLnRow&, 2)) Then
            Exit Do
        End If
        fLnRow& = fLnRow& + 1
    Loop
    If CellValue(WkAN, fLnRow&, 2) <> cMonth% Or IsEmpty(CellString(WkAN, fLnRow&, 2)) Then
        RNM$ = cna$ & Cstr(fLnRow&) & cna$ & Cstr(fLnRow&)
        Set oCellRange=WkAN.getCellRangeByPosition(RNM$)
        InsertRange(oCellRange, Rows)
        mFormula$ = "=Sum(G" & Cstr(fLnRow&) & ":I" & Cstr(fLnRow&) & ")"
        SetCellFormula WkAN, fLnRow&, 10, mFormula$
    Else
        ClearCell fLnRow&, 7
        ClearCell fLnRow&, 9
    End If
    SelectCell fLnRow&, 1
End Sub
Function CNumero%(fWD$)
    Dim i%, iLen%, rWD$
    If InStr(1, fWD$, ".", 1) > 0 Then
        fWD$ = Right(fWD$, len(fWD$)-InStr(1, fWD$, ".", 1))
    End If
    iLen% = Len(fWD$)
    For i% = 0 To iLen%-1
        If IsNumeric(Left(Right(fWD$, iLen%-i%),1)) Then
            Exit For
        End If
    Next i%
    rWD$ = left(fWD$,i%)
    iLen% = Len(rWD$)
    If iLen%=1 Then
        CNumero% =Asc(rWD$)-64
    ElseIf iLen%=2 Then
        CNumero% =(Asc(Left(rWD$,1))-64)*26+Asc(Right(rWD$,1))-64
    End If
End Function
Sub PutTitle()
    Dim oRangeOrg as object, oCellCpy as object
    SetCellString WkLN, 2, 1, "Source"
    SetCellString WkLN, 2, 2, "Month Accrued"
    SetCellString WkLN, 2, 3, "Month Settled"
    SetCellString WkLN, 2, 4, "Vendor/Customer"
    SetCellString WkLN, 2, 5, "Description"
    WkLN.getCellRangeByPosition(0, 1, 8, 1).CellBackColor = "&HFFFC"
    SetCellString WkLN, 2, 6, "With-VAT Amount"
    SetCellString WkLN, 2, 7, "Withholding Tax"
    SetCellString WkLN, 2, 8, "VAT Amount"
    SetCellString WkLN, 2, 9, "JE Amount"
    If IsMultiDP Then
        Col_Amount% = Col_InArray("BK", "Amount")
        Col_Border% = Col_InArray("BK", "Border")
        Set oRangeOrg = WkBK.getCellRangeByPosition(Col_Amount%, 1, Col_Border%-1, 1).RangeAddress
        Set oCellCpy = WkLN.getCellByPosition(9, 1).CellAddress ' insert position
        WkLN.CopyRange(oCellCpy, oRangeOrg) ' copy
    End If
    SetCellString WkLN, 2, 10+NumDpt, "VAT Audit"
End Sub
Rem =====(Jump to BK/JV/AP/AR indicated on Lense sheet)=====
Sub DirectLJump()
    Dim i%, mLen%, iLnRow&
    Dim cAddress$, trgSName$
    IntroTo3D()

```

```

Else
    SetCellValue WkLN, LnRow&, 9 + i%, CellValue(WkWk, iRow&, Col_Amount% + i%) * mSigma% * cfSig
End If
Next i%
End Sub
Sub ANNextLn(ByVal fLnRow&)
    Dim oCellRange As Range, mFormula$
    Do While CellValue(WkAN, fLnRow&, 2) <> cMonth%
        If IsEmpty(CellString(WkAN, fLnRow&, 2)) Then
            Exit Do
        End If
        fLnRow& = fLnRow& + 1
    Loop
    If CellValue(WkAN, fLnRow&, 2) <> cMonth% Or IsEmpty(CellString(WkAN, fLnRow&, 2)) Then
        RNM$ = "$A$" & CStr(fLnRow&) & ":$A$" & CStr(fLnRow&)
        Set oCellRange = WkAN.Range(RNM$)
        oCellRange.EntireRow.Insert
        mFormula$ = "=SUM(RC[-3]:RC[-1])"
        SetCellFormula WkAN, fLnRow&, 10, mFormula$
    Else
        ClearCell fLnRow&, 7
        ClearCell fLnRow&, 9
    End If
    SelectCell fLnRow&, 1
End Sub
Sub PutTitle()
    Dim oRangeOrg As Range, oRangeDst As Range
    SetCellString WkLN, 2, 1, "Source"
    SetCellString WkLN, 2, 2, "Month Accrued"
    SetCellString WkLN, 2, 3, "Month Settled"
    SetCellString WkLN, 2, 4, "Vendor/Customer"
    SetCellString WkLN, 2, 5, "Description"
    WkLN.Range("$A$2:$I$2").Interior.ColorIndex = 2
    SetCellString WkLN, 2, 6, "With-VAT Amount"
    SetCellString WkLN, 2, 7, "Withholding Tax"
    SetCellString WkLN, 2, 8, "VAT Amount"
    SetCellString WkLN, 2, 9, "JE Amount"
    If IsMultiDP Then
        Col_Amount% = Col_InArray("BK", "Amount")
        Col_Border% = Col_InArray("BK", "Border")
        RNM$ = CAlpha(Col_Amount% + 1) & "2:" & CAlpha(Col_Border% - 1) & CStr(2)
        WkBK.Range(RNM$).Copy
        WkLN.Paste Destination:=Cells("$J$2")
    End If
    SetCellString WkLN, 2, 10 + NumDpt, "C_Tax Audit"
End Sub
Rem =====(Jump to BK/JV/AP/AR indicated on Lense sheet)=====
Sub DirectLJump()
    Dim i%, mLen%, iLnRow&, a%
    Dim cAddress$, trgSName$
    IntroTo3D()

```

```

If Not StopOrGo("LNJump") then
    MsgTxt$ = "This command of Lense Jump is unusable on this "& aSName$ & " sheet"
    MsgTtl$ = "Test of sheet for applicability of the command"
    msgbox(MsgTxt$, 16 , MsgTtl$)
End
End If
iLnRow& = SelectionRow()
cAddress$ = CellString(aSheet, iLnRow&, 1)
If Len(cAddress$) > 0 Then
    mLen% = Len(cAddress$)
    For i% = 0 To mLen%
        If Mid(cAddress$, i + 1, 1) = "-" Then
            trgSName$ = Left(cAddress$, i%)
            targetRow& = Val(Right(cAddress$, mLen% - i - 1))
            Exit For
        End If
    Next i%
    Col_ActCD% = Col_InArray(trgSName$, "ActCD")
    aController.ActiveSheet = oSheets.getByname(trgSName$)
    SelectCell targetRow&, Col_ActCD%
End If
End Sub

```

```

491 If Not StopOrGo("LNJump") Then
492     MsgTxt$ = "This command of Lense Jump is unusable on this " & aSName$ & " sheet"
493     MsgTtl$ = "Test of sheet for applicability of the command"
494     resp% = MsgBox(MsgTxt$, 16, MsgTtl$)
495     End
496 End If
497 iLnRow& = SelectionRow()
498 cAddress$ = CellString(aSheet, iLnRow&, 1)
499 If Len(cAddress$) > 0 Then
500     mLen% = Len(cAddress$)
501     For i% = 0 To mLen%
502         If Mid(cAddress$, i + 1, 1) = "-" Then
503             trgSName$ = Left(cAddress$, i%)
504             targetRow& = Val(Right(cAddress$, mLen% - i - 1))
505             Exit For
506         End If
507     Next i%
508     Col_ActCD% = Col_InArray(trgSName$, "ActCD")
509     Workbooks(GLName).Worksheets(trgSName$).Activate
510     SelectCell targetRow&, Col_ActCD%
511     End If
512 End Sub

```