

Ligne : 1028

Connexion à la base de Saari via ODBC

```
OptOD.Value = True = True
If cmbExercice.Text = "" Then
    nRep = MsgBox("L'Exercice ne doit pas être vide", vbInformation + vbOKOnly, "AGETIPA")
    GoTo 1015
End If
cmbPERIODE.Text = "---"
If cmbPERIODE.Text = "" Then
    nRep = MsgBox("La Période ne doit pas être vide", vbInformation + vbOKOnly, "AGETIPA")
    GoTo 1015
End If
If OptTresorerie.Value = True Then
    strModele = "ASSOCIATION"
    strConnect = "DSN=Comptabilite; "
    nImp = 2
End If
If OptOD.Value = True Then
    strModele = "ASSOCIATION"
    strConnect = "DSN=Comptabilite; "
    nImp = 0
End If
If optProjet.Value = True Then
    strModele = "PROJET"
    strConnect = "DSN=ComptabiliteProjet; "
    nImp = 1
End If
MousePointer = 11
sPeriode = cmbPERIODE.Text
sExercice = cmbExercice.Text
lblIdentification.Visible = True
aNOMBRE = 0
Refresh
```

'Construction de la chaîne de connexion.

'Les différents paramètres de la chaîne de connexion sont séparés par un point virgule (;)

'DSN identifie le nom de la source de données

'Il est possible de renseigner le nom et le mot de passe à utiliser à l'aide des commandes UID et PWD

'++++++ strConnect = "DSN=Comptabilite;"

'Création de l'objet ActiveX

Set cnxCPTA = New adodb.Connection

cnxCPTA.ConnectionTimeout = 15 'Définition du TimeOut de connexion

cnxCPTA.CommandTimeout = 30 'Définition du TimeOut d'exécution de requêtes

'Connexion à la source

cnxCPTA.Open strConnect, "<Admin>", ""

'Code Visual Basic

Ligne : 1164

Récupération des écritures analyriques able ECRITUREA

```
'-----
' Max(EC_NO)
strSQL = "SELECT EC_NO From F_ECRITUREA_E0 ORDER BY EC_NO"
Set dsEcriture = dbFiche.OpenRecordset(strSQL)
dsEcriture.MoveLast
nEC_NO_MAX = dsEcriture("EC_NO")

If cmbExercice.Text = 2008 Then
    cmdSQL.CommandText = "SELECT * FROM F_ECRITUREA WHERE EC_NO > " & 37800
Else
    If strModele = "ASSOCIATION" Then
        cmdSQL.CommandText = "SELECT * FROM F_ECRITUREA WHERE EC_NO > " & (nEC_NO_MAX - 10000)
    End If
    If strModele = "PROJET" Then
        cmdSQL.CommandText = "SELECT * FROM F_ECRITUREA"
    End If
End If
```

```

cmdSQL.CommandType = adCmdText
Set cmdSQL.ActiveConnection = cnxCPTA
Set rsEcriture = cmdSQL.Execute

strsQl1 = "delete * from IMPUTATION_ASSOCIATION_NOUVELLE"
dbFicheLocal.Execute strsQl1

strsQl1 = "delete * from IMPUTATION_ASSOCIATION_NOUVELLE_II"
dbFicheLocal.Execute strsQl1

strsQl1 = "delete * from F_ECRITUREC_TEMP"
dbFicheLocal.Execute strsQl1

strsQl1 = "delete * from F_ECRITUREA_TEMP"
dbFicheLocal.Execute strsQl1
'Set dsEcritureC_TEMP = dbFicheLocal.OpenRecordset("F_ECRITUREC_TEMP", dbOpenTable)

Set dsNouvelleImputation = dbFicheLocal.OpenRecordset("IMPUTATION_ASSOCIATION_NOUVELLE", dbOpenTable)

'-----
Set dsEcritureA_E0 = dbFicheLocal.OpenRecordset("F_ECRITUREA_E0", dbOpenTable)
'--- BASE - ACCES DISTANT
Set dsEcritureA_E0 = dbFiche.OpenRecordset("F_ECRITUREA_E0", dbOpenTable)

Set dsEcritureA_TEMP = dbFicheLocal.OpenRecordset("F_ECRITUREA_TEMP", dbOpenTable)

dsEcritureA_E0.Index = "EC_NO_LIGNE_MODELE"
n_ = rsEcriture.Fields.Count
ij = 1
'PLAN BUDGETAIRE -----
Set dsPBServer = dbFiche.OpenRecordset("PLAN_BUDGETAIRE", dbOpenTable)
Set dsPBLocal = dbFicheLocal.OpenRecordset("PLAN_BUDGETAIRE", dbOpenTable)
dsPBLocal.Index = "CODE_MODELE"
Do While Not dsPBServer.EOF
    dsPBLocal.Seek "=", dsPBServer("CODE"), dsPBServer("MODELE")
    If dsPBLocal.NoMatch Then
        dsPBLocal.AddNew

```

```

For ijh = 1 To dsPBServer.Fields.Count - 1
    dsPBL local.Fields(ijh) = dsPBServer.Fields(ijh)
Next ijh
dsPBL local.Update
End If
dsPBServer.MoveNext
Loop

'+++++ REC NUMERO -----
'strS = "delete * from REC_NUMERO_ANALYTIQUE"
'dbFicheLocal.Execute strS
'Set dsREC_A = dbFicheLocal.OpenRecordset("REC_NUMERO_ANALYTIQUE", dbOpenTable)
'strS = "delete * from REC_NUMERO_GENERAL"
'dbFicheLocal.Execute strS
'Set dsREC_G = dbFicheLocal.OpenRecordset("REC_NUMERO_GENERAL", dbOpenTable)
'+++++ REC NUMERO -----
'----- INTIALIASATION DE LA BASE DE DEPART - ECRITURE ANALYTIQUE

While Not rsEcriture.EOF
    dsEcritureA_E0.Seek "=", rsEcriture("EC_NO"), rsEcriture("EA_LIGNE"), strModele
    '+dsREC_A.AddNew
    '+dsREC_A("NO_REC") = rsEcriture("EC_NO")
    '+dsREC_A("ANALYTIQUE") = rsEcriture("CA_NUM")
    '+dsREC_A.Update
    If dsEcritureA_E0.NoMatch Then
        'dsEcritureA_E0.AddNew
        dsEcritureA_TEMP.AddNew
        For i = 0 To n_ - 1
            '+dsEcritureA_E0.Fields(i) = rsEcriture.Fields(i)
            dsEcritureA_TEMP.Fields(i) = rsEcriture.Fields(i)
        Next i
        If rsEcriture("N_ANALYTIQUE") = 5 Or rsEcriture("N_ANALYTIQUE") = 9 Or rsEcriture("N_ANALYTIQUE") = 10 Then
            '+dsEcritureA_E0("CA_NUM") = rsEcriture("CA_NUM")
            dsEcritureA_TEMP("CA_NUM") = rsEcriture("CA_NUM")
        End If
        '+dsEcritureA_E0.Update
        dsEcritureA_TEMP.Update
    End If

```

```
End If
```

```
'Debug.Print ij  
ij = ij + 1  
rsEcriture.MoveNext  
Wend
```

Ligne : 1348

Récupération des écritures générales de Saari, table ECRITUREC

```
cmdSQL.CommandText = "SELECT * FROM F_ECRITUREC WHERE year(JM_Date)=2022 or year(JM_Date)=2023 ORDER BY JO_NUM,EC_PIECE,EC_NO"
```

```
'rsEcriture ("EC_JOUR") & Mid(rsEcriture("JM_DATE"), 3, 8)
```

```
cmdSQL.CommandType = adCmdText  
Set cmdSQL.ActiveConnection = cnxCPTA  
Set rsEcriture = cmdSQL.Execute  
rsEcriture.MoveFirst
```

```
strSQL = "delete * from F_ECRITUREC_TEMP"
```

```
dbFicheLocal.Execute strSQL
```

```
Set dsEcritureC_TEMP = dbFicheLocal.OpenTable("F_ECRITUREC_TEMP", dbOpenTable)
```

```
While Not rsEcriture.EOF  
    dsEcritureC_TEMP.AddNew
```

```
    For iC = 0 To 38
```

```
        dsEcritureC_TEMP.Fields(iC) = rsEcriture.Fields(iC)
```

```
    Next iC
```

```
'For iC = 0 To nC_ - 1
```

```
'    Debug.Print "iC = " & iC & " Champ = " & rsEcriture.Fields(iC).Name; " ---> " & rsEcriture.Fields(iC)
```

```
'Next iC
```

```

dsEcritureC_TEMP("EC_SENS") = rsEcriture("EC_SENS")
dsEcritureC_TEMP("SEQUENCE") = nSequence
dsEcritureC_TEMP("MODELE") = strModele
dsEcritureC_TEMP("EC_NO") = rsEcriture("EC_NO")

dsEcritureC_TEMP("DATE_PIECE") = rsEcriture("EC_JOUR") & Mid(rsEcriture("JM_DATE"), 3, 8)
dsEcritureC_TEMP("DATE_SAISIE") = rsEcriture("EC_Date")
dsEcritureC_TEMP("EC_SENS") = rsEcriture("EC_SENS")

dsEcritureC_TEMP.Update
rsEcriture.MoveNext
Wend
-----
```

Ligne : 1453

La colonne **EC_NO** est la jointure entre les tables **ECRITUREC** et **ECRITUREA**

```

strsQ11 = "SELECT F_ECRITUREA_TEMP.EC_NO, F_ECRITUREC_TEMP.JO_NUM, F_ECRITUREC_TEMP.EC_DATE,
F_ECRITUREC_TEMP.EC_PIECE, F_ECRITUREC_TEMP.EC_INTITULE, F_ECRITUREA_TEMP.N_ANALYTIQUE,
F_ECRITUREA_TEMP.CA_NUM, F_ECRITUREC_TEMP.DEBIT, F_ECRITUREC_TEMP.CREDIT, F_ECRITUREC_TEMP.EC_SENS,
F_ECRITUREA_TEMP.EA_MONTANT, COMPTE_ANALYTIQUE.NEW INTO ANALYTIQUE_ASSO_EXPORT"
& " FROM (F_ECRITUREA_TEMP INNER JOIN F_ECRITUREC_TEMP ON F_ECRITUREA_TEMP.EC_NO =
F_ECRITUREC_TEMP.EC_NO) INNER JOIN COMPTE_ANALYTIQUE ON F_ECRITUREA_TEMP.CA_NUM =
COMPTE_ANALYTIQUE.ANALYTIQUE_CODE"
& " WHERE (((F_ECRITUREC_TEMP.EC_DATE)>="#" & D1 & "# And (F_ECRITUREC_TEMP.EC_DATE)<="#" & D2 & "#) AND
((F_ECRITUREA_TEMP.N_ANALYTIQUE)=7) AND ((COMPTE_ANALYTIQUE.NEW)=1));"
```