

'RIC Mobile Course Catalog
 'By Lianne Elsner -> 2005 - 2006 Computer Science Honors Project
 '""""""""""Version Tracking Info""""""""""
 'Version 0.1 - 8/11/2005
 '--tested interface -> a simple 'hello world' program, no database functionality
 'Version 0.2 - 8/11/2005
 '--tested database connectivity -> program displayed a "sucess" message if database
 connection could be established
 'Version 0.3 - 8/23/2005
 '--add additional error checking
 'version 0.4 - 8/23/2005
 '--create record sets to pull data from the course database, create root of tree, lock
 treeview from label editing
 'version 1.0 - 9/1/2005
 '--populate majors
 'version 1.1 - 9/4/2005
 '--populate subcategories, added additional error checking
 'version 1.2 - 9/6/2005
 '--populate courses
 '-----Alpha Versions:-----
 'version 1.3 - 9/15/2005
 '--display course info for selected course, fixed bug in error checking
 'version 1.4 - 9/16/2005
 '--adds error checking for course info display - FIRST STABLE VERSION
 '-----Beta Versions:-----
 'version 1.5 - 9/23/2005
 '--neaten course info panel, changed labels to textboxes to
 'show outlines (due to a bug in eVB compiler)
 'version 1.6 - 9/30/2005
 '--fix cut-off text in display panel, added splash screen
 '-----Final Release:-----
 'version 1.7 - 10/1/2005
 'sorted items in the tree, fixed error messages

Option Explicit

Public conndb As ADOCE.Connection

Private Sub PopulateTree(ok As String)

On Error Resume Next 'prevents a crash (v1.1)

'MsgBox ok 'for testing v0.4

Dim strKeyVal As String 'uses primary key as node key

Dim strParentKeyVal As String 'holds node key of parent

Dim astrCourseParentVals As String 'will hold a string array

'create record sets (v0.4)

Dim rsRoot As ADOCE.Recordset

Dim rsMajors As ADOCE.Recordset

```

Dim rsSubCat As ADOCE.Recordset
Dim rsCourses As ADOCE.Recordset
'populate record sets (v0.4)
Dim sql As String
Dim strNodeText As String
'---Root---
sql = "SELECT * FROM Title"
Set rsRoot = CreateObject("ADOCE.RecordSet.3.0")
rsRoot.Open sql, conndb, adOpenForwardOnly, adLockReadOnly
'---Majors---
sql = "SELECT * FROM Majors ORDER BY Major"
Set rsMajors = CreateObject("ADOCE.RecordSet.3.0")
rsMajors.Open sql, conndb, adOpenForwardOnly, adLockReadOnly
'---Subcategories---
sql = "SELECT * FROM Subcategories ORDER BY SubCatCode"
Set rsSubCat = CreateObject("ADOCE.RecordSet.3.0")
rsSubCat.Open sql, conndb, adOpenForwardOnly, adLockReadOnly
'---Courses---
sql = "SELECT * FROM Courses ORDER BY Course_Code"
Set rsCourses = CreateObject("ADOCE.RecordSet.3.0")
rsCourses.Open sql, conndb, adOpenForwardOnly, adLockReadOnly
'---Populate tree
If rsRoot.RecordCount > 0 Then
    strNodeText = rsRoot.Fields(0)
    tvCatalog.Nodes.Add , , "Root", strNodeText
End If
rsRoot.Close
Set rsRoot = Nothing
'---Populate tree with majors (v1.0)---
If rsMajors.RecordCount > 0 Then
    Do While Not rsMajors.EOF
        strKeyVal = "M" & CStr(rsMajors.Fields(0))
        strNodeText = rsMajors.Fields(1)
        tvCatalog.Nodes.Add "Root", 4, strKeyVal, strNodeText
        rsMajors.MoveNext
    Loop
End If
rsMajors.Close
'---Add subcategories to corresponding major (v1.1)---
If rsSubCat.RecordCount > 0 Then
    Do While Not rsSubCat.EOF
        strKeyVal = "S" & CStr(rsSubCat.Fields(0))
        strNodeText = rsSubCat.Fields(1)
        strParentKeyVal = "M" & CStr(rsSubCat.Fields(2))
        tvCatalog.Nodes.Add strParentKeyVal, 4, strKeyVal, strNodeText
        rsSubCat.MoveNext
    Loop
End If

```

```

    Loop
End If
rsSubCat.Close
'---Add courses to corresponding subcategories (v1.2)---
Dim strParentHolder As String
Dim intParentStringLength As Integer
Dim intCounter As Integer
If rsCourses.RecordCount > 0 Then
    Do While Not rsCourses.EOF
        intCounter = 0
        strNodeText = rsCourses.Fields(1)
        strParentHolder = rsCourses.Fields(7)
        astrCourseParentVals = Split(strParentHolder, ",", -1, vbTextCompare) 'split string
on the ','
        'MsgBox astrCourseParentVals(0) 'for testing (v1.2)
        'MsgBox astrCourseParentVals(1) 'for testing (v1.2)
        intParentStringLength = UBound(astrCourseParentVals) 'determine the upper bound
of the array
        'MsgBox intParentStringLength 'for testing (v1.2)
        Do While intCounter <= intParentStringLength 'add each course to the correct
subcategories
            strParentKeyVal = "S" & CStr(astrCourseParentVals(intCounter))
            tvCatalog.Nodes.Add strParentKeyVal, 4, , strNodeText
            intCounter = intCounter + 1
        Loop
        rsCourses.MoveNext
    Loop
End If
rsCourses.Close
tvCatalog.Visible = True
End Sub

```

```

Private Sub Form_Load()
'Splash screen (v1.6)
fraSplash.Top = 0
fraSplash.Left = 0
fraSplash.Visible = True
imgSplash.Picture = "\My Documents\catalog.bmp"
Me.Refresh
'lblDescription.Caption = "Hello World!" 'testing line from v0.1
""""Connect to database"""" - added in v0.2
On Error Resume Next 'prevents crash if database not found (v0.3)
Dim strOK As String 'a dummy variable to pass (v0.4)
Set conndb = CreateObject("ADOCE.Connection.3.0")
conndb.ConnectionString = "data source = \My Documents\CourseCatalog.cdb"

```

```

conndb.Open
""""Test connection""""added in v0.2
Dim strMsg As String
Dim intState As Integer
intState = conndb.State
If intState = 0 Then
    strMsg = "Database error! Please refer the the help file included on the program CD."
    MsgBox strMsg, vbOKOnly, "RIC Mobile Course Catalog"
    App.End 'close the program
'this clause has been replaced by the splash screen (v1.6)
'ElseIf intState = 1 Then
'    strMsg = "Welcome to the RIC Mobile Course Catalog!"
'    MsgBox strMsg
End If
strOK = "ok" 'eVB requires that parameter is passed (v0.4)
Call PopulateTree(strOK)
conndb.Close
Set conndb = Nothing
End Sub

```

```

Private Sub Form_OKClick()
    App.End
End Sub
'Splash screen (v1.6)
Private Sub tmrSplash_Timer()
    fraSplash.Visible = False
End Sub

```

```

'-----Display course info - version 1.3-----
Private Sub tvCatalog_NodeClick(ByVal Index As Long)
On Error Resume Next 'prevents crash if database not found
Set conndb = CreateObject("ADOCE.Connection.3.0")
conndb.ConnectionString = "data source = \My Documents\CourseCatalog.cdb"
conndb.Open
Dim strCourseToDisplay As String
Dim sql As String
'----Error checking for course info display (v1.4)-----
Dim strTypeCheck As String
strTypeCheck = CStr(tvCatalog.Nodes(Index).Key)
'weed out the root, majors, and subcategories
If InStr(strTypeCheck, "M") Or InStr(strTypeCheck, "S") Or InStr(strTypeCheck, "R")
Then
    Exit Sub 'escape if the selected item is NOT a course
End If
'----Pull course info from database and display (v1.3)-----
strCourseToDisplay = CStr(tvCatalog.Nodes(Index).Text)

```

```
Dim strMsg As String
Dim intState As Integer
intState = conndb.State
If intState = 0 Then
    strMsg = "Database error! Please refer the the help file included on the program CD."
    MsgBox strMsg, vbOKOnly, "RIC Mobile Course Catalog"
    App.End 'close the program
End If
Dim rsCourses As ADOCE.Recordset
sql = "SELECT * FROM Courses WHERE (Course_Code = '' & strCourseToDisplay &
'')"
'MsgBox sql 'for testing - v1.3
Set rsCourses = CreateObject("ADOCE.RecordSet.3.0")
rsCourses.Open sql, conndb, adOpenForwardOnly, adLockReadOnly
'---Populate the text boxes----
txtName.Text = CStr(rsCourses.Fields(2))
txtCredits.Text = CStr(rsCourses.Fields(3))
txtOffered.Text = CStr(rsCourses.Fields(4))
txtPrereq.Text = CStr(rsCourses.Fields(5))
txtDescription.Text = CStr(rsCourses.Fields(6))
rsCourses.Close
conndb.Close
Set conndb = Nothing
End Sub
```