Imports System

Imports System.Globalization

Imports System.Data

Imports System.Data.SqlClient

Imports System.IO

Imports System.Configuration

Imports DAL

Normally used when names need to be listed in a drop drown combobox control.

1. Return a set of only the active students and clinician, or return all of clinicians, or all of the students.
2. Also, displays all the guardians.

Public Interface IPopulateAllNames

    Function DisplayStudents(ByVal returnAllStudentsFlag As Boolean) As DataSet

    Function DisplayGuardians() As DataSet

    Function DisplayClinician(ByVal returnAllClinciansFlag As Boolean) As DataSet

End Interface

Public Class IPopulateNames

    Inherits ScheduleTemplate

    Implements IPopulateAllNames

    Dim parseApostrophe As New nameOperation

Retrieve all active students only, or all of the student from the database .

Public Function DisplayStudents(ByVal returnAllStudentsFlag As Boolean) As DataSet Implements IPopulateAllNames.DisplayStudents

        Dim scheduleResults As New returnStudentData

        Dim StudentAttributes(,) As String

        Dim x As Integer

Return all the students from the database

        StudentAttributes = scheduleResults.ReturnStudentInfo()

        Dim liststudent As Boolean = True

        Dim p = UBound(StudentAttributes)

        Dim ds1 As New DataSet

        ds1 = ScheduleFooter()

        Dim dt1 As DataTable = ds1.Tables("ScheduleFooter")

        Dim dr As DataRow

Include an empty row with the returned names'

        dr = dt1.NewRow()

        dt1.Rows.Add(dr)

        Dim clinicianfirstName As String = String.Empty

        Dim clinicianlastName As String = String.Empty

        Dim studentFirstName As String = String.Empty

        Dim studentLastName As String = String.Empty

        For x = 0 To p

            studentFirstName = StudentAttributes(x, 0)

            studentLastName = StudentAttributes(x, 1)

        Dim studentFullName As String = studentLastName.Trim & ", " & studentFirstName.Trim

Determine if the user wants to return active students only, or all of the students

            If returnAllStudentsFlag = False Then

Check to see if student is marked as active

                If StudentAttributes(x, 6) = True Then

                    dr = dt1.NewRow()

                    dr("FullName") = studentFullName

                    dt1.Rows.Add(dr)

                End If

            Else

                dr = dt1.NewRow()

                dr("FullName") = studentFullName

                dt1.Rows.Add(dr)

            End If

        Next

        Return ds1

    End Function

Retrieve all of the guardians from the database.

 Public Function DisplayGuardians() As DataSet Implements IPopulateAllNames.DisplayGuardians

        Dim GuardianName As New returnGuardianInfo

        Dim Firstname As String = Nothing

        Dim Lastname As String = Nothing

        Dim ds As DataSet

        ds = GuardianName.GetGuardianInfo

        Dim dt As DataTable = ds.Tables("GuardianProfile")

        Dim ds1 As New DataSet

        ds1 = ScheduleFooter()

        Dim dt1 As DataTable = ds1.Tables("ScheduleFooter")

        Dim dr1 As DataRow

        Dim dr As DataRow

        For Each dr In dt.Rows

            dr1 = dt1.NewRow()

Populate dataset with all the names of the guardians.

            Firstname = dr("First Name")

            Lastname = dr("Last Name")

            dr1 = dt1.NewRow()

            dr1("FirstName") = Firstname.Trim

            dr1("LastName") = Lastname.Trim

            dr1("FullName") = Lastname.Trim & ", " & Firstname.Trim

            dt1.Rows.Add(dr1)

        Next

        Return ds1

    End Function

Retrieve all the clinicians or only the active clinicians from the database

Public Function displayClinician(ByVal returnAllClinciansFlag As Boolean) As DataSet Implements IPopulateAllNames.DisplayClinician

        Dim clinicianinfo As New Clinicians

        Dim ClinicianAttributes(,) As String

        Dim clinicianLastName As String = String.Empty

        Dim clinicianFirstName As String = String.Empty

        ClinicianAttributes = clinicianinfo.GetClinicianInfo()

        Dim ds As DataSet = Nothing

        ds = ClinicianTable()

        Dim dt As DataTable = ds.Tables("Cliniciandata")

        Dim dr As DataRow

        Dim x As Integer

        Dim FinalClinician As Integer = UBound(ClinicianAttributes)

Determine if the user wants to return active students only, or all of the students

        For x = 0 To FinalClinician

            dr = dt.NewRow()

            If x < 1 Then

                If returnAllClinciansFlag = True Then

                    dr("CL\_FullName") = "AUTO SELECT"

                ElseIf returnAllClinciansFlag = False Then

                    dr("CL\_FullName") = String.Empty

                End If

            Else

                clinicianFirstName = ClinicianAttributes(x, 2).Trim

                clinicianLastName = ClinicianAttributes(x, 1).Trim

   Dim clinicianFullName As String = clinicianLastName.Trim & ", " & clinicianFirstName.Trim

                clinicianFullName = parseApostrophe.executeName(clinicianFullName, 0)

                dr("CL\_FullName") = clinicianFullName

            End If

            dt.Rows.Add(dr)

        Next

        Return ds

    End Function

 End Class