'Stores the path of a Calendar

'Returns the path location of a Calendar

'Makes use of Factory Design Pattern

Public Interface IfileStoragePath

    Function path(newPath As String, newModule As String) As String

End Interface

Public Class saveStorageModules

    Implements IfileStoragePath

    Public Function path(newPath As String, scheduleType As String) As String Implements IfileStoragePath.path

        Dim selectedModule As IStorageModule = Nothing

        Select Case (scheduleType)

            Case "student"

                selectedModule = New storeStudentScheduleFilePath

                selectedModule.storeFilePath(newPath.Trim())

                Exit Select

            Case "office"

                selectedModule = New storeOfficeScheduleFilePath

                selectedModule.storeFilePath(newPath.Trim())

                Exit Select

            Case Else

                Exit Select

        End Select

        Return newPath

    End Function

End Class

Public Class returnStorageModules

    Implements IfileStoragePath

    Public Function path(newPath As String, scheduleType As String) As String Implements IfileStoragePath.path

        Dim selectedModule As IreturnStorageModule = Nothing

        Dim defaultLocation As String = String.Empty

        Select Case (scheduleType)

            Case "student"

                selectedModule = New StudentCalendarFilelocation

                defaultLocation = selectedModule.locateFilePath(newPath.Trim())

                Exit Select

            Case "office"

                selectedModule = New OfficeCalendarFilelocation

                defaultLocation = selectedModule.locateFilePath(newPath.Trim())

                Exit Select

            Case Else

                Exit Select

        End Select

        Return defaultLocation

    End Function

End Class

Public Interface IStorageModule

    Function storeFilePath(path As String)

End Interface

Public Interface IreturnStorageModule

    Function locateFilePath(path As String) As String

End Interface

Class storeStudentScheduleFilePath

    Implements IstorageModule

    Dim connectionString As Object

    Public Sub New()

        Try

            connectionString = ConfigurationManager.ConnectionStrings("Familyliteracy").ConnectionString

        Catch ex As Exception

            MsgBox("Cannot connect to Datasource")

        End Try

    End Sub

    Function storeFilePath(path As String) Implements IStorageModule.storeFilePath

        Dim query As String = "Update DefaultScheduleFileLocations SET [Student\_Schedule\_Location]=@storagePath,[StorageDate]=@storageDate"

        Dim conn As New SqlConnection(connectionString)

        Dim cmd As New SqlCommand(query, conn)

        cmd.Parameters.AddWithValue("@storagePath", path)

        cmd.Parameters.AddWithValue("@storageDate", Now)

        Dim updated As Integer = 0

        conn.Open()

        updated = cmd.ExecuteNonQuery()

        conn.Close()

        Return Nothing

    End Function

End Class

Public Class storeOfficeScheduleFilePath

    Implements IstorageModule

    Dim connectionString As Object

    Public Sub New()

        Try

            connectionString = ConfigurationManager.ConnectionStrings("Familyliteracy").ConnectionString

        Catch ex As Exception

            MsgBox("Cannot connect to Datasource")

        End Try

    End Sub

    Function storeFilePath(path As String) Implements IstorageModule.storeFilePath

        Dim query As String = "Update DefaultScheduleFileLocations SET [Office\_Schedule\_Location]=@storagePath,[StorageDate]=@storageDate"

        Dim conn As New SqlConnection(connectionString)

        Dim cmd As New SqlCommand(query, conn)

        cmd.Parameters.AddWithValue("@storagePath", path)

        cmd.Parameters.AddWithValue("@storageDate", Now)

        Dim updated As Integer = 0

        conn.Open()

        updated = cmd.ExecuteNonQuery()

        conn.Close()

        Return Nothing

    End Function

End Class

Public Class StudentCalendarFilelocation

    Implements IreturnStorageModule

    Dim connectionString As Object

    Public Sub New()

        Try

            connectionString = ConfigurationManager.ConnectionStrings("Familyliteracy").ConnectionString

        Catch ex As Exception

            MsgBox("Cannot connect to Datasource")

        End Try

    End Sub

    Function locateFilePath(path As String) As String Implements IreturnStorageModule.locateFilePath

        Dim query As String = "SELECT \* FROM DefaultScheduleFileLocations "

        Dim conn As New SqlConnection(connectionString)

        Dim cmd As New SqlCommand(query, conn)

        Dim da As New SqlDataAdapter(cmd)

        Dim ds As New DataSet

        conn.Open()

        da.Fill(ds, "DefaultScheduleFileLocations")

        conn.Close()

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

        Dim rw As DataRow

        Dim defaultPath As String = String.Empty

        For Each rw In dt.Rows

            defaultPath = rw("Student\_Schedule\_Location")

        Next

        Dim objCommandBuilder As New SqlCommandBuilder(da)

        da.Update(ds, "DefaultScheduleFileLocations")

        Return defaultPath

    End Function

End Class

Public Class OfficeCalendarFilelocation

    Implements IreturnStorageModule

    Dim connectionString As Object

    Public Sub New()

        Try

            connectionString = ConfigurationManager.ConnectionStrings("Familyliteracy").ConnectionString

        Catch ex As Exception

            MsgBox("Cannot connect to Datasource")

        End Try

    End Sub

    Function locateFilePath(path As String) As String Implements IreturnStorageModule.locateFilePath

        Dim query As String = "SELECT \* FROM DefaultScheduleFileLocations"

        Dim conn As New SqlConnection(connectionString)

        Dim cmd As New SqlCommand(query, conn)

        Dim da As New SqlDataAdapter(cmd)

        Dim ds As New DataSet

        conn.Open()

        da.Fill(ds, "DefaultScheduleFileLocations")

        conn.Close()

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

        Dim rw As DataRow

        Dim defaultPath As String = String.Empty

        For Each rw In dt.Rows

            defaultPath = rw("Office\_Schedule\_Location")

        Next

        Dim objCommandBuilder As New SqlCommandBuilder(da)

        da.Update(ds, "DefaultScheduleFileLocations")

        Return defaultPath

    End Function

End Class