Cis 232

Author: Cedric Mulumba

Date: 12/12/16

Chap: 6, 7 & 9 Chap

Program/Assignment: 3D

Title: Sports Pro System

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  | 'Name: Cedric L Mulumba  'Date: 12/12/16  'Project Name: 3D  'Project: Add a form that displays all the open incidents for a selected techician  Imports System.Data.OleDb  Public Class frmTechnicianIncidents  Private techList As List(Of Technician)  Private Sub frmTechnicianIncidents\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load  Me.GetTechnicianList()  Me.GetTechnicianData()  End Sub  Private Sub GetTechnicianList()  Try  techList = TechnicianDB.GetTechnicianList  ComboBox1.DataSource = techList  ComboBox1.DisplayMember = "Name"  Catch ex As Exception  MessageBox.Show(ex.Message, ex.GetType.ToString)  End Try  End Sub  Private Sub GetTechnicianData()  Dim techID As Integer = techList(ComboBox1.SelectedIndex).TechID  Try  Dim technician As Technician = TechnicianDB.GetTechnician(techID)  TechnicianBindingSource.Clear()  TechnicianBindingSource.Add(technician)  Dim incidentList As List(Of Incident)  incidentList = IncidentDB.GetOpenTechnicianIncidents(techID)  IncidentDataGridView.DataSource = incidentList  Catch ex As Exception  MessageBox.Show(ex.Message, ex.GetType.ToString)  End Try  End Sub  Private Sub ComboBox1\_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ComboBox1.SelectedIndexChanged  Me.GetTechnicianData()  End Sub  End Class |
|  | ' Class CustomerDB  Imports System.Data.OleDb  Public Class CustomerDB  Public Shared Function GetCustomerName(ByVal customerID As Integer) As String  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String = String.Format(  "SELECT Name FROM Customers WHERE customerID = {0}",  customerID)  Dim selectCommand As New OleDbCommand(selectStatement, connection)  Dim customerName As String = String.Empty  Try  connection.Open()  customerName = CStr(selectCommand.ExecuteScalar())  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return customerName  End Function  Public Shared Function GetCustomerList() As List(Of Customer)  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String =  "SELECT CustomerID, Name " &  "FROM Customers " &  "ORDER BY Name"  Dim selectCommand As New OleDbCommand(selectStatement, connection)  Dim customers As New List(Of Customer)  Try  connection.Open()  Dim reader As OleDbDataReader = selectCommand.ExecuteReader()  Dim customer As Customer  Do While reader.Read  customer = New Customer  With customer  .CustomerID = CInt(reader("CustomerID"))  .Name = CStr(reader("Name"))  End With  customers.Add(customer)  Loop  reader.Close()  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return customers  End Function  End Class |
|  | 'Class Incident  Option Strict On  Public Class Incident  Private m\_IncidentID As Integer  Private m\_CustomerID As Integer  Private m\_ProductCode As String  Private m\_TechID As Nullable(Of Integer)  Private m\_DateOpened As Date  Private m\_DateClosed As Nullable(Of Date)  Private m\_Title As String  Private m\_Description As String  Public Property IncidentID() As Integer  Get  Return m\_IncidentID  End Get  Set(ByVal value As Integer)  m\_IncidentID = value  End Set  End Property  Public Property CustomerID() As Integer  Get  Return m\_CustomerID  End Get  Set(ByVal value As Integer)  m\_CustomerID = value  End Set  End Property  Public Property ProductCode() As String  Get  Return m\_ProductCode  End Get  Set(ByVal value As String)  m\_ProductCode = value  End Set  End Property  Public Property TechID() As Nullable(Of Integer)  Get  If m\_TechID.HasValue Then  Return CInt(m\_TechID)  Else  Return Nothing  End If  End Get  Set(ByVal value As Nullable(Of Integer))  m\_TechID = value  End Set  End Property  Public Property DateOpened() As Date  Get  Return m\_DateOpened  End Get  Set(ByVal value As Date)  m\_DateOpened = value  End Set  End Property  Public Property DateClosed() As Nullable(Of Date)  Get  If m\_TechID.HasValue Then  Return m\_DateClosed  Else  Return Nothing  End If  End Get  Set(ByVal value As Nullable(Of Date))  m\_DateClosed = value  End Set  End Property  Public Property Title() As String  Get  Return m\_Title  End Get  Set(ByVal value As String)  m\_Title = value  End Set  End Property  Public Property Description() As String  Get  Return m\_Description  End Get  Set(ByVal value As String)  m\_Description = value  End Set  End Property  Public ReadOnly Property CustomerName() As String  Get  Return CustomerDB.GetCustomerName(m\_CustomerID)  End Get  End Property  Public ReadOnly Property TechnicianName() As String  Get  If m\_TechID.HasValue Then  Return TechnicianDB.GetTechnicianName(m\_TechID.Value)  Else  Return String.Empty  End If  End Get  End Property  Public ReadOnly Property ProductName() As String  Get  Return ProductDB.GetProductName(m\_ProductCode)  End Get  End Property  End Class |
|  | 'Class IncidentDB  Imports System.Data.OleDb  Public Class IncidentDB  Public Shared Function GetOpenIncidents() As List(Of Incident)  Dim IncidentList As New List(Of Incident)  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String = "SELECT CustomerID, ProductCode, TechID, DateOpened, Title FROM Incidents WHERE DateClosed IS NULL"  Dim selectCommand As New OleDbCommand(selectStatement, connection)  Try  connection.Open()  Dim reader As OleDbDataReader = selectCommand.ExecuteReader()  Dim incident As Incident  Do While reader.Read  incident = New Incident  With incident  .CustomerID = CInt(reader("CustomerID"))  .ProductCode = CStr(reader("ProductCode"))  'NUllable TechID  If IsDBNull(reader("TechID")) Then  .TechID = Nothing  Else  .TechID = CInt(reader("TechID"))  End If  .DateOpened = CDate(reader("DateOpened"))  .Title = CStr(reader("Title"))  IncidentList.Add(incident)  End With  Loop  reader.Close()  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return IncidentList  End Function  Public Shared Sub AddIncident(ByVal incident As Incident)  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim insertStatement As String  With incident  insertStatement = String.Format \_  ("INSERT Incidents " &  "(CustomerID, ProductCode, DateOpened, Title, Description) " &  "VALUES (?CustomerID, ?ProdductCode, ?DateOpen, ?Title, ?Description)")  End With  Dim insertCommand As New OleDbCommand(insertStatement, connection)  Try  connection.Open()  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  End Sub  Public Shared Function GetIncident(ByVal incidentID As Integer) As Incident  Dim incident As New Incident  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String =  "SELECT IncidentID, CustomerID, ProductCode, TechID, DateOpened, DateClosed, Title, Description " & \_  "FROM Incidents " & \_  "WHERE IncidentID = ?IncidentID"  Dim selectCommand As New OleDbCommand(selectStatement, connection)  selectCommand.Parameters.AddWithValue("?IncidentID", incidentID)  Try  connection.Open()  Dim reader As OleDbDataReader = selectCommand.ExecuteReader(CommandBehavior.SingleRow)  If reader.Read Then  With incident  .IncidentID = CInt(reader("IncidentID"))  .CustomerID = CInt(reader("CustomerID"))  .ProductCode = reader("ProductCode").ToString  If IsDBNull(reader.GetOrdinal("TechID")) Then  .TechID = Nothing  Else  .TechID = CInt(reader("TechID"))  End If  .DateOpened = CDate(reader("DateOpened"))  If IsDBNull(reader("DateClosed")) Then  .DateClosed = Nothing  Else  .DateClosed = CDate(reader("DateClosed"))  End If  .Title = reader("Title").ToString  incident.Description = reader("Description").ToString  End With  Else  incident = Nothing  End If  reader.Close()  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return incident  End Function  Public Shared Function UpdateIncident(ByVal incident As Incident, ByVal description As String) As Boolean  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim updatestatement As String =  "UPDATE Incidents " &  "SET Description = ?NewDescription " &  "WHERE IncidentID = ?IncidnetID " &  " AND Description = ?Description " &  " AND DateClosed IS NULL"  Dim updateCommand As New OleDbCommand(updatestatement, connection)  updateCommand.Parameters.AddWithValue("?NewDescription", description)  updateCommand.Parameters.AddWithValue("?IncidnetID", incident.IncidentID)  updateCommand.Parameters.AddWithValue("?Description", incident.Description)  Try  connection.Open()  Dim count As Integer = updateCommand.ExecuteNonQuery  If count > 0 Then  Return True  Else  Return False  End If  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  End Function  Public Shared Function CloseIncident(ByVal incident As Incident) As Boolean  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim updateStatement As String = \_  "UPDATE Incidents " &  "SET DateClosed = ?DateClosed " & \_  "WHERE IncidentID = ?IncidentID " & \_  " AND Description =?Descroption" & \_  " AND DateClosed IS NULL"  Dim updateCommand As New OleDbCommand(updateStatement, connection)  updateCommand.Parameters.AddWithValue("?DateClosed", DateTime.Today)  updateCommand.Parameters.AddWithValue("?IncidnetID", incident.IncidentID)  updateCommand.Parameters.AddWithValue("?Description", incident.Description)  Try  connection.Open()  Dim count As Integer = updateCommand.ExecuteNonQuery  If count > 0 Then  Return True  Else  Return False  End If  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  End Function  Public Shared Function GetOpenTechnicianIncidents(ByVal techID As Integer) As List(Of Incident)  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String = \_  "SELECT CustomerID, ProductCode, TechID, DateOpened, Title " & \_  "FROM Incidents " & \_  "WHERE TechID = DateClosed IS NULL"  Dim selectCommand As New OleDbCommand(selectStatement, connection)  selectCommand.Parameters.AddWithValue("?TechID", techID)  Dim openIncidents As New List(Of Incident)  Try  connection.Open()  Dim reader As OleDbDataReader = selectCommand.ExecuteReader()  Dim incident As Incident  Do While reader.Read  incident = New Incident  With incident  .CustomerID = CInt(reader("CustomerID"))  .ProductCode = CStr(reader("ProductCode"))  .DateOpened = CDate(reader("DateOpened"))  .Title = CStr(reader("Description"))  End With  openIncidents.Add(incident)  Loop  reader.Close()  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return openIncidents  End Function  End Class |
|  | 'Class ProductDB  Imports System.Data.OleDb  Public Class ProductDB  Public Shared Function GetProductList() As List(Of Product)  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String =  "SELECT ProductCode, Name " &  "FROM Products " &  "ORDER BY Name"  Dim selectCommand As New OleDbCommand(selectStatement, connection)  Dim products As New List(Of Product)  Try  connection.Open()  Dim reader As OleDbDataReader = selectCommand.ExecuteReader()  Dim product As Product  Do While reader.Read  product = New Product  With product  .ProductCode = CStr(reader("ProductCode"))  .Name = CStr(reader("Name"))  End With  products.Add(product)  Loop  reader.Close()  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return products  End Function  Public Shared Function GetProductName(ByVal productCode As String) As String  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String = String.Format(  "Select Name FROM Products WHERE ProductCode = '{0}'",  productCode)  Dim selectCommand As New OleDbCommand(selectStatement, connection)  Dim productName As String = String.Empty  Try  connection.Open()  productName = CStr(selectCommand.ExecuteScalar())  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return productName  End Function  End Class |
|  | 'Class Technician  Public Class Technician  Private m\_TechID As Integer  Private m\_Name As String  Private m\_Email As String  Private m\_Phone As String  Public Property TechID() As Integer  Get  Return m\_TechID  End Get  Set(ByVal value As Integer)  m\_TechID = value  End Set  End Property  Public Property Name() As String  Get  Return m\_Name  End Get  Set(ByVal value As String)  m\_Name = value  End Set  End Property  Public Property Email() As String  Get  Return m\_Email  End Get  Set(ByVal value As String)  m\_Email = value  End Set  End Property  Public Property Phone() As String  Get  Return m\_Phone  End Get  Set(ByVal value As String)  m\_Phone = value  End Set  End Property  End Class |
|  | 'Class TechnicianDB  Option Strict On  Imports System.Data.OleDb  Public Class TechnicianDB  Public Shared Function GetTechnicianName(ByVal techID As Integer) As String  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String = String.Format(  "SELECT Name FROM Technicians WHERE techID = {0}",  techID)  Dim selectCommand As New OleDbCommand(selectStatement, connection)  Dim technicianName As String = String.Empty  Try  connection.Open()  technicianName = CStr(selectCommand.ExecuteScalar())  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return technicianName  End Function  Public Shared Function GetTechnicianList() As List(Of Technician)  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String =  "SELECT TechID, Name " &  "FROM Technicians " &  "ORDER BY Name"  Dim selectCommand As New OleDbCommand(selectStatement, connection)  Dim technicians As New List(Of Technician)  Try  connection.Open()  Dim reader As OleDbDataReader = selectCommand.ExecuteReader()  Dim technician As Technician  Do While reader.Read  technician = New Technician  With technician  .TechID = CInt(reader("TechID"))  .Name = CStr(reader("Name"))  End With  technicians.Add(technician)  Loop  reader.Close()  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return technicians  End Function  Public Shared Function GetTechnician(ByVal techID As Integer) As Technician  Dim technician As New Technician  Dim connection As OleDbConnection = TechSupportDB.GetConnection  Dim selectStatement As String =  "SELECT TechID, Name, Email, Phone " &  "FROM Technicians " &  "WHERE TechID = ?TechID"  Dim selectCommand As New OleDbCommand(selectStatement, connection)  selectCommand.Parameters.AddWithValue("?TechID", techID)  Try  connection.Open()  Dim reader As OleDbDataReader = selectCommand.ExecuteReader(CommandBehavior.SingleRow)  If reader.Read Then  With technician  .TechID = CInt(reader("TechID"))  .Name = reader("Name").ToString  .Email = reader("Email").ToString  .Phone = reader("Phone").ToString  End With  Else  technician = Nothing  End If  reader.Close()  Catch ex As OleDbException  Throw ex  Finally  connection.Close()  End Try  Return technician  End Function  End Class |
|  | 'Class TechSupportDB  Imports System.Data.OleDb  Public Class TechSupportDB  Public Shared Function GetConnection() As OleDbConnection  Dim connectionString As String = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\bob\TechSupport.mdb"  Dim connection As New OleDbConnection(connectionString)  Return connection  End Function  End Class |