Project Development Phase Sprint-04

|  |  |
| --- | --- |
| Date | 16 November 2022 |
| Team ID | PNT2022TMID45954 |
| Project Name | Machine Learning-Based Predictive Analytics for Aircraft Engine |

# AIRCRAFT ENGINE ANALAYSIS:

Imports Microsoft.VisualBasic Imports System

Imports System.Data Imports System.Data.Common Imports System.Data.OleDb

Imports System.Configuration

Imports System.Collections.Specialized Imports SampleApps.ValGnES.GlobalVariables Imports System.Web.HttpContext

Imports System.Drawing Imports System.IO

Imports System.Data.OracleClient Imports System.Web

Imports System.Web.UI.WebControls Imports System.Net.Sockets Imports System.Net

Imports System.Configuration.ConfigurationManager

Namespace GL

Public Class DAL

Public Enum Transaction None

Start Finish

End Enum

Public Enum QueryType SelectQuery InsertQuery UpdateQuery DeleteQuery Adapter

UpdateDeleteAdapter spSelect spInsert spUpdate spDelete

End Enum

Public Enum ReturnType None

Scalar Field Collection Sortedlist ListItem Dataset

Datarow RefCursor spCollection spSortedlist spDataset spOutPut spListItem

End Enum

#Region "Get Connection String From Session"

Public Shared Function ConString() As String Dim ConnectionString As String = ""

'ConnectionString = ConfigurationSettings.AppSettings("ConnStr") ' ConnectionString =

System.Configuration.ConfigurationManager.AppSettings("tamConnectionString1") ConnectionString =

System.Configuration.ConfigurationManager.ConnectionStrings("CString").ConnectionStrin g

'ConnectionString = "Provider=SQLOLEDB;Data Source=(local);User ID=sa;pwd=sa;Initial Catalog=GLPGDMDB"

'ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\tamil\tam.mdb"

'ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=E:\tam.mdb"

'ConnectionString = "Provider=Microsoft.Jet.OLEDB.6.0;Data Source=E:\tam.mdb"

'ConnectionString = "Provider=SQLOLEDB;Data Source=(local);User ID=sa;pwd=sa;Initial Catalog=pubs"

Return ConnectionString End Function

#End Region

#Region "Get Connection Instance"

Protected Shared Function CreateOleDbConnection(ByVal provFlag As Integer) As ConObject

Dim conObj As ConObject = New ConObject Dim Cmd As OleDbCommand = New OleDbCommand

Dim Conn As OleDbConnection = New OleDbConnection Try

Dim ConnString As String = ConString() Dim ErrMsg As String = ""

Conn = Nothing

If provFlag = 0 Then

Conn = New OleDbConnection(ConnString)

Else

Dim provInd As Integer = ConnString.ToLower().IndexOf("provider=") If provInd <> -1 Then

ConnString = "Provider=OraOLEDB.Oracle.1" & ConnString.Substring(ConnString.ToLower().IndexOf(";", provInd))

End If

Conn = New OleDbConnection(ConnString) End If

If Conn Is Nothing Then

Throw New Exception("Error in connection") End If

Conn.Open() Cmd.Connection = Conn

Return New ConObject(Cmd, Conn, ErrMsg) Catch ex As Exception

Return New ConObject(Cmd, Conn, ex.Message) End

Try

End Function #End Region

#Region "ExecuteQuery"

Public Overloads Shared Sub ExecuteQuery(ByVal QueryStr() As String, Optional ByVal TransStart As Transaction = Transaction.None, Optional ByVal TransFinish As Transaction = Transaction.None, Optional ByVal conFlag As Integer = 0, Optional ByVal oracleFlag As Integer = 0)

Dim conObj As ConObject = GetConnection(conFlag, oracleFlag) Dim ErrMsg As String = conObj.ErrMsg

If ErrMsg <> "" Then

Throw New Exception(ErrMsg) : Exit Sub

ElseIf conObj.Conn Is Nothing And conObj.ConnO Is Nothing Then

Throw New Exception("Oracle Connection can not be Established.Please Check Network/ Oracle Connectivity") : Exit Sub

End If

Dim QryStr As String Try

If TransStart = Transaction.Start Then

If Not conObj.Cmd Is Nothing Then conObj.Cmd.Transaction

=

conObj.Cmd.Connection.BeginTransaction Else

conObj.CmdO.Transaction = conObj.CmdO.Connection.BeginTransaction

End If End If

If QueryStr.Length > 0 Then For Each QryStr In QueryStr

If Not QryStr = "" Then

If Not conObj.Cmd Is Nothing Then conObj.Cmd.CommandText

= QryStr conObj.Cmd.ExecuteNonQuery() Else conObj.CmdO.CommandText = QryStr conObj.CmdO.ExecuteNonQuery() End If

End If Next

End If

If TransFinish = Transaction.Finish Then

If Not conObj.Cmd Is Nothing Then conObj.Cmd.Transaction.Commit() Else conObj.CmdO.Transaction.Commit()

End If End If

Catch en As OleDbException

HttpContext.Current.Session("ErrorMsg") = en.ErrorCode g\_Save\_Failed

= ""

If HttpContext.Current.Session("ErrorMsg") = -2147217900 Then g\_Save\_Failed = "Attempt to Save Record Failed due to One or more

errors occurred during " + \_

"processing of command. ORA-00933 SQL command not properly ended." Else g\_Save\_Failed

= "Attempt to Save Record Failed due to "

g\_Save\_Failed = g\_Save\_Failed & HttpContext.Current.Session("ErrorMsg")

End If

If TransStart Then

If Not conObj.Cmd Is Nothing Then conObj.Cmd.Transaction.Rollback()

ElseIf Not conObj.CmdO Is Nothing Then conObj.CmdO.Transaction.Rollback()

End If End If

Throw New Exception(en.ToString)

Finally

DestroyConObj(conObj) End Try

End Sub #End Region

#Region "DestroyConObj"

Protected Shared Sub DestroyConObj(ByVal argConObj As ConObject) If Not argConObj Is Nothing Then

If Not argConObj.Cmd Is Nothing Then Try argConObj.Cmd.Dispose() Catch ex As Exception

End Try End If

If Not argConObj.Conn Is Nothing Then Try

If Not argConObj.Conn.State = ConnectionState.Closed Then argConObj.Conn.Close() End If

argConObj.Conn.Dispose() Catch ex As Exception

End Try End If

End If End Sub

#End Region

Public Shared Function GetConnection(Optional ByVal oracleFlag As Integer = 0, Optional ByVal oracleClientFlag As Integer = 0) As ConObject

Dim conObj As ConObject = New ConObject() Dim ErrMsg As String = ""

Try

If oracleClientFlag = 0 Then conObj = CreateOleDbConnection(oracleFlag) Else

' conObj = CreateOracleConnection() End

If

Catch en As Exception

If en.Message.Contains("ORA-12560") = True Or en.Message.Contains("ORA-12571") = True Then

ErrMsg = "Oracle is not Connected.Either Network is disconnected or Oracle Listener Problem!(" & en.Message & ")"

ElseIf en.Message.Contains("The ConnectionString property has not been initialized") = True Or en.Message.ToUpper.Contains("TIMEOUT") = True Then

Try

For I As Integer = 0 To 1 Try

If oracleClientFlag = 0 Then conObj = CreateOleDbConnection(oracleFlag)

Else

' conObj = CreateOracleConnection()

End If

Catch ex1 As OleDbException

If ex1.Message.Contains("ORA-12560") = True Or ex1.Message.Contains("ORA-12571") = True Then

ErrMsg = "Oracle is not Connected.Either Network is disconnected or Oracle Listener Problem!(" & ex1.Message & ")"

ElseIf ex1.Message.ToUpper.Contains("TIMEOUT") = True

Then

Next

ErrMsg = "Oracle Connection Timeout" Else

ErrMsg = ex1.Message End If

Catch ex1 As Exception ErrMsg

= ex1.Message End Try

Catch ex As OleDbException

If ex.Message.Contains("ORA-12560") = True Or ex.Message.Contains("ORA-12571") = True Then

ErrMsg = "Oracle is not Connected.Either Network is disconnected or Oracle Listener Problem!(" & en.Message & ")"

ElseIf ex.Message.ToUpper.Contains("TIMEOUT") = True Then ErrMsg = "Oracle Connection Timeout"

Else

ErrMsg = ex.Message

Else

End If

Catch ex As Exception ErrMsg

= ex.Message End Try

ErrMsg = en.Message

End If End

Try

Return conObj End Function

#Region "ExecuteQueryForReturnType"

Public Overloads Shared Function ExecuteQuery(ByVal QueryStr As String, ByVal QryType As QueryType, Optional ByVal RetType As ReturnType = ReturnType.None, Optional

ByVal TransStart As Transaction = Transaction.None, Optional ByVal TransFinish As Transaction = Transaction.None, Optional ByVal TName As String = Nothing, Optional ByVal key As Boolean = False, Optional ByVal ds As DataSet = Nothing, Optional ByVal conFlag As Integer = 0, Optional ByVal oracleFlag As Integer = 0) As Object

Dim Sortedorder As Boolean = key Dim conObj As ConObject = Nothing Dim ErrMsg As String

Try conObj = GetConnection(conFlag, oracleFlag) ErrMsg = conObj.ErrMsg

If ErrMsg <> "" Then

Throw New Exception(ErrMsg) : Exit Function

ElseIf conObj.Conn Is Nothing And conObj.ConnO Is Nothing Then Throw New Exception("Oracle Connection can not be

Established.Please Check Network/ Oracle Connectivity") : Exit Function End If

If TransStart = Transaction.Start Then

If Not conObj.Cmd Is Nothing Then conObj.Cmd.Transaction = conObj.Conn.BeginTransaction

ElseIf Not conObj.CmdO Is Nothing Then conObj.CmdO.Transaction = conObj.ConnO.BeginTransaction End If

End If

Select Case QryType

Case QueryType.SelectQuery If QueryStr = "" Then

Exit Select

Else

If Not conObj.Cmd Is Nothing Then conObj.Cmd.CommandText

= QueryStr

Return (ExecuteSelect(RetType, TName, conObj.Cmd))

ElseIf Not conObj.CmdO Is Nothing Then conObj.CmdO.CommandText = QueryStr

Return (ExecuteSelect(RetType, TName, conObj.CmdO)) End If

End If

Case QueryType.InsertQuery, QueryType.UpdateQuery, QueryType.DeleteQuery

If Not conObj.Cmd Is Nothing Then conObj.Cmd.CommandText =

QueryStr

conObj.Cmd.ExecuteNonQuery()

ElseIf Not conObj.CmdO Is Nothing Then conObj.CmdO.CommandText = QueryStr conObj.CmdO.ExecuteNonQuery()

End If

Case QueryType.Adapter

If Not conObj.Cmd Is Nothing Then

Dim adapter As New OleDbDataAdapter(QueryStr, conObj.Conn) adapter.Update(ds, ds.Tables(0).TableName)

adapter.Dispose()

ElseIf Not conObj.CmdO Is Nothing Then ds = New DataSet Dim adapter As New OracleDataAdapter(QueryStr, conObj.ConnO)

adapter.Update(ds, ds.Tables(0).TableName) adapter.Dispose()

End If

Case QueryType.UpdateDeleteAdapter If QueryStr = "" Then

If Not conObj.Cmd Is Nothing Then ds

= New DataSet

Dim adapter As New OleDbDataAdapter("", conObj.Conn) adapter.Update(ds) adapter.Dispose()

ElseIf Not conObj.CmdO Is Nothing Then ds

= New DataSet

Dim adapter As New OracleDataAdapter("", conObj.ConnO) adapter.Update(ds)

adapter.Dispose() End

If End If

End Select

If TransFinish = Transaction.Finish Then conObj.Cmd.Transaction.Commit()

End If

Catch en As OleDbException

If TransStart Then

If Not conObj.Cmd Is Nothing Then conObj.Cmd.Transaction.Rollback()

ElseIf Not conObj.CmdO Is Nothing Then conObj.CmdO.Transaction.Rollback() End If End If

Throw en Finally

DestroyConObj(conObj) End Try

Return Nothing End Function

#End Region

Public Overloads Shared Function GetRecordsetFromSQL(ByRef sSQL As String) As Data.DataTable

'Public Function GetRecordsetFromSQL(ByRef sSQL As String) As Data.DataTable

Dim oDT As New Data.DataTable Dim oCmd As OleDbCommand

Dim oDA As OleDbDataAdapter Dim DBConn As OleDbConnection

' DBConn = New OleDbConnection("Provider=SQLOLEDB;Data Source=(local);User ID=sa;pwd=sa;Initial Catalog=pubs")

DBConn = New OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\tamil\tam.mdb")

oDA = New OleDbDataAdapter(sSQL, DBConn) 'Dim conObj As ConObject = Nothing

'conObj = GetConnection(0, 0) 'If conObj Is Nothing Then

' 'As got the error in HRESULT - COM exception while executing

ERR.RAISE

' 'In order to achieve the same functionality in ported as like

baseline...instroduced VB6.0 interop to raise the error message to COM ' 'ManagedCOMError.GetManagedErrorInfo(vbObjectError +

ppErrorCodes.ppDALErrNoOpenConnection, "ProPrimaDAL.DataAccess.GetRecordsetFromSQL", "No Connection Open")

' 'Err.Raise(vbObjectError + ppErrorCodes.ppDALErrNoOpenConnection,

"ProPrimaDAL.DataAccess.GetRecordsetFromSQL", "No Connection Open")

' 'PGC Migration ID:002 to fix warning Null Reference Exception ' GetRecordsetFromSQL = Nothing

'Else

' conObj.Cmd.Transaction = conObj.Conn.BeginTransaction ' oCmd = New OleDbCommand(sSQL, conObj.Conn)

' ' Dim adapter As New OleDbDataAdapter(QueryStr, conObj.Conn) ' oCmd.CommandType = Data.CommandType.Text

' oDA = New OleDbDataAdapter(oCmd)

oDA.Fill(oDT) ' oCmd.Dispose() oCmd =

Nothing

oDA = Nothing GetRecordsetFromSQL = oDT ' End If

End Function

Public Overloads Shared Function UpdateData(ByVal dt As DataTable, ByVal TableName As String) As Boolean

Dim oCmd As OleDbCommand = Nothing

Dim oDataAdap As OleDbDataAdapter = Nothing

Dim oCmdBuilder As OleDbCommandBuilder = Nothing 'Dim oDA As OleDbDataAdapter

Dim DBConn As OleDbConnection

DBConn = New OleDbConnection("Provider=SQLOLEDB;Data Source=(local);User ID=sa;pwd=sa;Initial Catalog=pubs")

'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

' If moConnection Is Nothing Then

'As got the error in HRESULT - COM exception while executing ERR.RAISE 'In order to achieve the same functionality in ported as like

baseline...instroduced VB6.0 interop to raise the error message to COM 'ManagedCOMError.GetManagedErrorInfo(vbObjectError +

ppErrorCodes.ppDALErrNoOpenConnection, "ProPrimaDAL.DataAccess.UpdateData", "No Connection Open")

' Err.Raise(vbObjectError + ppErrorCodes.ppDALErrNoOpenConnection, "ProPrimaDAL.DataAccess.UpdateData", "No Connection Open")

' Else

' oCmd = New OleDbCommand("Select \* from " & TableName, DBConn, trn) DBConn.Open()

oCmd = New OleDbCommand("Select \* from " & TableName, DBConn, DBConn.BeginTransaction) oCmd.Connection = DBConn oDataAdap =

New OleDbDataAdapter(oCmd) oCmdBuilder = New OleDbCommandBuilder(oDataAdap) oCmdBuilder.QuotePrefix = "[" oCmdBuilder.QuoteSuffix = "]" oDataAdap.TableMappings.Add("Table", TableName) dt.TableName = TableName oDataAdap.Update(dt) oCmd.Transaction.Commit() oCmdBuilder.Dispose() oCmdBuilder = Nothing oDataAdap.Dispose() oDataAdap = Nothing oCmd.Dispose() oCmd = Nothing Return True

Exit Function

End Function

# WEB PAGE OUTPUT

