CREATE PROCEDURE dbo.GetProdDetails

(

@ProductID int,

@ProductName nvarchar(40) output,

@UnitPrice money output

)

AS

SET NOCOUNT ON

Select @ProductName = ProductName,

@UnitPrice = UnitPrice

From Products

Where ProductID = @ProductID

private void btnRetrieveSingleRow\_Click(object sender, System.EventArgs e)

{

Database db = DatabaseFactory.CreateDatabase("Northwind");

DBCommandWrapper dbc =

db.GetStoredProcCommandWrapper("GetProdDetails");

dbc.AddInParameter("@ProductID", DbType.Int32, 1);

dbc.AddOutParameter("@ProductName", DbType.String, 50);

dbc.AddOutParameter("@UnitPrice", DbType.Currency, 8);

db.ExecuteNonQuery(dbc);

lblResult.Text = "Product Name:"+ dbc.GetParameterValue("@ProductName") + " ";

lblResult.Text += "Unit Price:" + dbc.GetParameterValue("@UnitPrice");

}

DbCommand comm1 = db.GetStoreProcCommand("GetProdDetails");

db.ExecuteNonQuery(comm1);

System.Data.IDbConnection dbConnection = new System.Data.OleDb.OleDbConnection(Generales.connecte);

System.Data.IDbCommand dbCommand = new System.Data.OleDb.OleDbCommand();

System.Data.OleDb.OleDbParameter parameter1 = new System.Data.OleDb.OleDbParameter();

System.Data.OleDb.OleDbParameter parameter2 = new System.Data.OleDb.OleDbParameter();

parameter1.OleDbType = System.Data.OleDb.OleDbType.Numeric;

parameter1.ParameterName = "Age";

parameter1.Value = 33;

parameter2.OleDbType = System.Data.OleDb.OleDbType.Char;

parameter2.ParameterName = "Name1";

parameter2.Value = "Alexander";

dbCommand.Connection = dbConnection;

dbCommand.CommandText = "insert into test1 (Age, Name1) values (?,?)";

dbCommand.Parameters.Add(parameter1);

dbCommand.Parameters.Add(parameter2);

dbConnection.Open();

dbCommand.ExecuteNonQuery();

dbConnection.Close();

==================================================

Shared Sub GetSalesByCategory(ByVal connectionString As String, \_

ByVal categoryName As String)

Using connection As New SqlConnection(connectionString)

' Create the command and set its properties.

Dim command As SqlCommand = New SqlCommand()

command.Connection = connection

command.CommandText = "SalesByCategory"

command.CommandType = CommandType.StoredProcedure

' Add the input parameter and set its properties.

Dim parameter As New SqlParameter()

parameter.ParameterName = "@CategoryName"

parameter.SqlDbType = SqlDbType.NVarChar

parameter.Direction = ParameterDirection.Input

parameter.Value = categoryName

' Add the parameter to the Parameters collection.

command.Parameters.Add(parameter)

' Open the connection and execute the reader.

connection.Open()

Dim reader As SqlDataReader = command.ExecuteReader()

If reader.HasRows Then

Do While reader.Read()

Console.WriteLine("{0}: {1:C}", reader(0), reader(1))

Loop

Else

Console.WriteLine("No rows returned.")

End If

End Using

End Sub

Public Sub CreateMyOleDbCommand(connection As OleDbConnection, \_

queryString As String, parameters() As OleDbParameter)

Dim command As New OleDbCommand(queryString, connection)

command.CommandText = "SELECT \* FROM Customers WHERE Country = ? AND City = ?"

command.Parameters.Add(parameters)

Dim j As Integer

For j = 0 To command.Parameters.Count - 1

command.Parameters.Add(parameters(j))

Next j

Dim message As String = ""

Dim i As Integer

For i = 0 To command.Parameters.Count - 1

message += command.Parameters(i).ToString() + ControlChars.Cr

Next i

Console.WriteLine(message)

End Sub

STORE PROCEDURE CON DATASET

Imports System.Data.SqlClient

Public Function LoadDataCapacity(ByVal strRegion As String, ByVal strSubRegion As String) As System.Data.DataTable

Dim oDASQL As clsDAHelperSQL

Dim cnn As SqlConnection

Dim cmd As SqlCommand

oDASQL = New clsDAHelperSQL

cnn = New SqlConnection(Common.ConnectionString)

cmd = New SqlCommand

cmd = New System.Data.SqlClient.SqlCommand("usp\_GetScorecardSiteCapacity", cnn)

cmd.CommandType = CommandType.StoredProcedure

oDASQL.AddCommandParameter(cmd, "@Region", SqlDbType.VarChar, ParameterDirection.Input, , , , strRegion)

oDASQL.AddCommandParameter(cmd, "@SubRegion", SqlDbType.VarChar, ParameterDirection.Input, , , , strSubRegion)

cnn.Open()

Dim datAdapter As New SqlDataAdapter(cmd)

Dim ds As New DataSet

datAdapter.Fill(ds)

cnn.Close()

cnn.Dispose()

Return ds.Tables(0)

End Function

Imports System.Data.SqlClient

Public Function LocationSite(ByVal strLoc As String) As String

Dim oDASQL As clsDAHelperSQL

Dim cnn As SqlConnection

Dim cmd As SqlCommand

Dim strLocation As String

strLocation = ""

Try

oDASQL = New clsDAHelperSQL

cnn = New SqlConnection(Common.ConnectionString)

cmd = New SqlCommand

cnn.Open()

If CType(cnn.State And ConnectionState.Open, Boolean) Then

With cmd

.Connection = cnn

.CommandType = CommandType.StoredProcedure

.CommandText = "usp\_GetSiteLocation"

.Parameters.Clear()

oDASQL.AddCommandParameter(cmd, "@strAbbrev", SqlDbType.VarChar, ParameterDirection.Input, , , , strLoc)

strLocation = cmd.ExecuteScalar()

End With

End If

Catch ex As Exception

Log.AddLog(ex, enmLogType.Error)

strLocation = ""

Finally

oDASQL = Nothing

cnn = Nothing

cmd = Nothing

End Try

Return strLocation

End Function