

```

using System;
using System.Data;
using Oracle.DataAccess.Client;
using CustomersBus;

namespace CustomersOracleData
{
    /// <summary>
    /// CustData Class designed to handle
    /// all database activities for Customers table.
    /// Author: Doug Streitenberger
    /// </summary>
    public class CustData
    {
        /// <summary>
        /// Default Instantiation of Class
        /// </summary>
        ///
        public CustData()
        {

        }

        /// <summary>
        /// Populates a DataTable with all Customer Information
        /// Author: Doug Streitenberger
        /// </summary>
        /// <returns>DataTable of all Customer Information.</returns>
        public static DataTable CustomersEnum()
        {
            using (OracleConnection cn = new OracleConnection(GetConnectionString()))
            {
                using (OracleCommand cmd = new OracleCommand("pkgCustomers.f_CustomersEnum_ref", cn))
                {
                    cmd.CommandType = CommandType.StoredProcedure;
                    OracleParameter prmReturnCursor = cmd.Parameters.Add("c_CustomersEnum_ref"
                        , OracleDbType.RefCursor
                        , ParameterDirection.ReturnValue);

                    OracleDataAdapter da = new OracleDataAdapter(cmd);

                    DataTable dt = new DataTable();
                    da.Fill(dt);
                    return dt;
                }
            }
        }
    }
}

```

```

/// <summary>
/// Populates a DataTable with all Customer Information based on search params entered.
/// Search Criteria are formatted such that if values are entered, the returned datatable
/// will be filtered according to those entries. Otherwise, all records will be returned.
/// Author: Doug Streitenberger
/// </summary>
/// <param name="CustomerId"></param>
/// <param name="CompanyName"></param>
/// <param name="ContactName"></param>
/// <returns>DataTable of all Customer Information.</returns>
public static DataTable CustomerSearchEnum(string CustomerId, string CompanyName, string ContactName)
{
    CustomerId = CustomerId.Trim();
    CompanyName = CompanyName.Trim();
    ContactName = ContactName.Trim();

    using (OracleConnection cn = new OracleConnection(GetConnectionString()))
    {
        using (OracleCommand cmd = new OracleCommand("pkgCustomers.f_CustomersEnumSearch_ref", cn))
        {
            cmd.CommandType = CommandType.StoredProcedure;

            OracleParameter prmReturnCursor = cmd.Parameters.Add("c_CustomersEnum_ref"
                , OracleDbType.RefCursor
                , ParameterDirection.ReturnValue);

            OracleParameter prmCustomerID = cmd.Parameters.Add("i_CustomerID_tx"
                , OracleDbType.Char
                , 7);
            prmCustomerID.Value = FormatSearchCriteria(CustomerId);

            OracleParameter prmCompanyName = cmd.Parameters.Add("i_CompanyName_tx"
                , OracleDbType.Varchar2
                , 52);
            prmCompanyName.Value = FormatSearchCriteria(CompanyName);

            OracleParameter prmContactName = cmd.Parameters.Add("i_ContactName_tx"
                , OracleDbType.Varchar2
                , 52);
            prmContactName.Value = FormatSearchCriteria(ContactName);

            OracleDataAdapter da = new OracleDataAdapter(cmd);

            DataTable dt = new DataTable();
            da.Fill(dt);
            return dt;
        }
    }
}

```

```

/// <summary>
/// If an entry is made to a search criteria, the search criteria
/// will be concatenated as follows: '%SearchCriteria%'
/// (a % on each end of the string).
/// Otherwise an '%' is returned for that search criteria.
/// Author: Doug Streitenberger
/// </summary>
/// <param name="SearchCriteria">Search Criteria as entered by the user.</param>
/// <returns>Formatted Search Criteria String.</returns>
private static string FormatSearchCriteria(string SearchCriteria)
{
    if (SearchCriteria.Length == 0)
    {
        SearchCriteria = "%";
    }
    else
    {
        SearchCriteria = "%" + SearchCriteria + "%";
    }
    return SearchCriteria;
}

/// <summary>
/// Populates DataSet with Customer Information based on param CustomerId.
/// Author: Doug Streitenberger
/// </summary>
/// <param name="CustomerId">Requested CustomerId.</param>
/// <returns>DataSet of Selected Customer Information</returns>
public static DataSet CustomersEnumByIDRpt(string CustomerId)
{
    using (OracleConnection cn = new OracleConnection(GetConnectionString()))
    {
        using (OracleCommand cmd = new OracleCommand("pkgCustomers.f_CustomersEnumByID_ref", cn))
        {
            cmd.CommandType = CommandType.StoredProcedure;
            OracleParameter prmReturnCursor = cmd.Parameters.Add("c_CustomersEnumAllFields_ref"
                , OracleDbType.RefCursor
                , ParameterDirection.ReturnValue);

            OracleParameter prmCustomerID = cmd.Parameters.Add("i_CustomerID_tx"
                , OracleDbType.Char
                , 5);
            prmCustomerID.Value = CustomerId;

            OracleDataAdapter da = new OracleDataAdapter(cmd);
            DataSet ds = new DataSet();

            da.Fill(ds);
            return ds;
        }
    }
}

```

```

    }
}

/// <summary>
/// Populates a DataSet of the Top 5 Customers By Sales
/// based on Start And End Dates entered.
/// Author: Doug Streitenberger
/// </summary>
/// <returns>DataSet of Top 5 Customers</returns>
public static DataSet CustomersEnumTop5(string sStartDate, string sEndDate)
{
    using (OracleConnection cn = new OracleConnection(GetConnectionString()))
    {
        using (OracleCommand cmd = new OracleCommand("pkgCustomers.f_CustomersEnumTop5_ref ", cn))
        {
            cmd.CommandType = CommandType.StoredProcedure;
            OracleParameter prmReturnCursor = cmd.Parameters.Add("c_CustomersTop5_ref"
                , OracleDbType.RefCursor
                , ParameterDirection.ReturnValue);

            OracleParameter prmStartDate = cmd.Parameters.Add("i_Start_Dt"
                , OracleDbType.Date);

            OracleParameter prmEndDate = cmd.Parameters.Add("i_End_Dt"
                , OracleDbType.Date);
            try
            {
                prmStartDate.Value = Convert.ToDateTime(sStartDate);
                prmEndDate.Value = Convert.ToDateTime(sEndDate);
            }
            catch { }
            OracleDataAdapter da = new OracleDataAdapter(cmd);
            DataSet ds = new DataSet();

            da.Fill(ds);
            return ds;
        }
    }
}

/// <summary>
/// On record insert, determines if CustomerID already exists in the Customers table based on param CustomerId.
/// Author: Doug Streitenberger
/// </summary>
/// <param name="CustomerId">Requested CustomerId</param>
/// <returns>Boolean whether CustomerId already exists in the database.</returns>
public static bool UniqueCustomerID(string CustomerId)
{
    bool bUnique = true;

```

```

using (OracleConnection cn = new OracleConnection(GetConnectionString()))
{
    using (OracleCommand cmd = new OracleCommand("pkgCustomers.f_UniqueCustomerID_nr", cn))
    {
        cmd.CommandType = CommandType.StoredProcedure;
        OracleParameter prmReturnCount = cmd.Parameters.Add("o_CountOfRecords_nr"
            , OracleDbType.Int32
            , ParameterDirection.ReturnValue);

        OracleParameter prmCustomerID = cmd.Parameters.Add("i_CustomerID_tx"
            , OracleDbType.Char
            , 5);
        prmCustomerID.Value = CustomerId;

        cn.Open();
        cmd.ExecuteNonQuery();
        Int32 iCountofRecords = (Int32)prmReturnCount.Value;
        if (iCountofRecords > 0)
        {
            bUnique = false;
        }
    }
    return bUnique;
}

}

/// <summary>
/// Inserts Customer Record based on Customer object.
/// Author: Doug Streitenberger
/// </summary>
/// <param name="customer">Customer Object</param>
public static void CustomerAdd(Customer customer)
{
    using (OracleConnection cn = new OracleConnection(GetConnectionString()))
    {
        using (OracleCommand cmd = new OracleCommand("pkgCustomers.p_CustomersInsert", cn))
        {
            cmd.CommandType = CommandType.StoredProcedure;
            OracleParameter prmCustomerID = cmd.Parameters.Add("i_CustomerID_tx"
                , OracleDbType.Char
                , 5);
            prmCustomerID.Value = customer.CustomerId;

            OracleParameter prmCompanyName = cmd.Parameters.Add("i_CompanyName_tx"
                , OracleDbType.Varchar2
                , 40);
            prmCompanyName.Value = customer.CompanyName;

            OracleParameter prmContactName = cmd.Parameters.Add("i_ContactName_tx"
                , OracleDbType.Varchar2

```

```

        , 30);
    prmContactName.Value = customer.ContactName;

    OracleParameter prmPhone = cmd.Parameters.Add("i_Phone_tx"
        , OracleDbType.Varchar2
        , 24);
    prmPhone.Value = customer.PhoneNumber;

    OracleParameter prmFax = cmd.Parameters.Add("i_Fax_tx"
        , OracleDbType.Varchar2
        , 24);
    prmFax.Value = customer.FaxNumber;

    cn.Open();
    cmd.ExecuteNonQuery();
    }
}

/// <summary>
/// Updates Customer Record based on Customer Object.
/// Author: Doug Streitenberger
/// </summary>
/// <param name="oldCustomer">Customer Object Before Update For Data Concurrency</param>
/// <param name="customer">Customer Object of Updated Data</param>
/// <returns>Int32 iCount;
/// Should return value of '0' record updated.
/// If '-1': Data Concurrency Error.
/// </returns>

public static int CustomerUpdate(Customer oldCustomer, Customer customer)
{
    Int32 iCount = 0;
    using (OracleConnection cn = new OracleConnection(GetConnectionString()))
    {
        using (OracleCommand cmd = new OracleCommand("pkgCustomers.f_CustomersUpdate_nr", cn))
        {
            cmd.CommandType = CommandType.StoredProcedure;
            OracleParameter prmReturnCount = cmd.Parameters.Add("l_result"
                , OracleDbType.Int32
                , ParameterDirection.ReturnValue);

            OracleParameter prmCustomerID = cmd.Parameters.Add("i_CustomerID_tx"
                , OracleDbType.Char
                , 5);
            prmCustomerID.Value = customer.CustomerId;

            OracleParameter prmOldCompanyName = cmd.Parameters.Add("i_OldCompanyName_tx"
                , OracleDbType.Varchar2
                , 50);

```

```

prnOldCompanyName.Value = oldCustomer.CompanyName;

OracleParameter prnOldContactName = cmd.Parameters.Add("i_OldContactName_tx"
    , OracleDbType.Varchar2
    , 50);
prnOldContactName.Value = oldCustomer.ContactName;

OracleParameter prnOldPhone = cmd.Parameters.Add("i_OldPhone_tx"
    , OracleDbType.Varchar2
    , 50);
prnOldPhone.Value = oldCustomer.PhoneNumber;

OracleParameter prnOldFax = cmd.Parameters.Add("i_OldFax_tx"
    , OracleDbType.Varchar2
    , 50);
prnOldFax.Value = oldCustomer.FaxNumber;

OracleParameter prnCompanyName = cmd.Parameters.Add("i_CompanyName_tx"
    , OracleDbType.Varchar2
    , 50);
prnCompanyName.Value = customer.CompanyName;

OracleParameter prnContactName = cmd.Parameters.Add("i_ContactName_tx"
    , OracleDbType.Varchar2
    , 50);
prnContactName.Value = customer.ContactName;

OracleParameter prnPhone = cmd.Parameters.Add("i_Phone_tx"
    , OracleDbType.Varchar2
    , 50);
prnPhone.Value = customer.PhoneNumber;

OracleParameter prnFax = cmd.Parameters.Add("i_Fax_tx"
    , OracleDbType.Varchar2
    , 50);
prnFax.Value = customer.FaxNumber;

cn.Open();
cmd.ExecuteNonQuery();
iCount = (Int32)prnReturnCount.Value;
    }
}
return iCount;
}

```

```

/// <summary>
/// Deletes Customer record based upon CustomerId.
/// Author: Doug Streitenberger
/// </summary>
/// <param name="CustomerId">CustomerId of the selected record</param>

```

```

public static void CustomerDelete(string CustomerId)
{
    using (OracleConnection cn = new OracleConnection(GetConnectionString()))
    {
        using (OracleCommand cmd = new OracleCommand("pkgCustomers.p_CustomersDelete", cn))
        {
            cmd.CommandType = CommandType.StoredProcedure;
            OracleParameter prmCustomerID = cmd.Parameters.Add("i_CustomerID_tx"
                , OracleDbType.Char
                , 5);
            prmCustomerID.Value = CustomerId;

            cn.Open();
            cmd.ExecuteNonQuery();
        }
    }
}

private static string GetConnectionString()
{
    // Establish a database connection string for
    // the CustData Class.
    CustomersOracleDBConn objDBConn = new CustomersOracleDBConn();
    string sDbConn = objDBConn.sConn;
    objDBConn = null;
    return sDbConn;
}
}
}

```