

```

//-- *****
//-- Project:      HND Data Project
//-- Module:       VehicleDB
//-- Author:       Paul McKillop
//-- Date created: 19 January 2019
//-- *****

using System;
using System.Collections.Generic;
using RentalWPF.Models;
using Dapper;
using System.Windows;

namespace RentalWPF.Data
{
    public static class VehicleDB
    {
        //-- Method to retrieve all vehicles from the database
        public static List<Vehicle> VehiclesGetAll()
        {
            var vehicles = new List<Vehicle>();

            return vehicles;
        }

        //-- Method to add a new domestic vehicle to the database
        public static int VehicleInsertDomestic(VehicleDomestic myVehicleDomestic)
        {
            var conn = AppUtility.DbConnection("RentalAzure");

            try
            {
                //-- open the connection to the database
                if (conn.State == System.Data.ConnectionState.Closed)
                {
                    conn.Open();
                }

                //-- Set up the parameters
                DynamicParameters param = new DynamicParameters();
                param.Add("@VehicleType_id", myVehicleDomestic.VehicleType_Id);
                param.Add("@Model_id", myVehicleDomestic.Model_Id);
                param.Add("@Registration", myVehicleDomestic.Registration);
                param.Add("@DateOnFleet", myVehicleDomestic.DateOnFleet);
                param.Add("@Gearbox", myVehicleDomestic.Gearbox);
                param.Add("@MaxLadenWeight", 0);

                //-- Execute the stored procedure to store the vehicle
                conn.Execute("spVehicle_Insert", param, commandType: System.Data.
                    CommandType.StoredProcedure);
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
            finally
            {
                conn.Close();
            }

            return 1;
        }

        //-- Method to add a new commercial vehicle to the database
        public static int VehicleInsertCommercial(VehicleCommercial
            myVehicleCommercial)

```

```

    {
        var conn = AppUtility.DbConnection("RentalAzure");

        try
        {
            //-- open the connection to the database
            if (conn.State == System.Data.ConnectionState.Closed)
            {
                conn.Open();
            }

            //-- Set up the parameters
            DynamicParameters param = new DynamicParameters();
            param.Add("@VehicleType_id", myVehicleCommercial.VehicleType_Id);
            param.Add("@Model_id", myVehicleCommercial.Model_Id);
            param.Add("@Registration", myVehicleCommercial.Registration);
            param.Add("@DateOnFleet", myVehicleCommercial.DateOnFleet);
            param.Add("@Gearbox", "NA");
            param.Add("@MaxLadenWeight", myVehicleCommercial.MaxLadenWeight);

            //-- Execute the stored procedure to store the vehicle
            conn.Execute("spVehicle_Insert", param, commandType: System.Data.
                CommandType.StoredProcedure);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
        finally
        {
            conn.Close();
        }
        return 1;
    }
}

```