```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Data.SqlClient;
using System.Configuration;
using System.Data;

namespace CsharpWebAppDemo.Database.Models
{
    public class Author
    {
        public List<AuthorModel> GetAllAuthors()
        {
            //Connection Open
            //Build and Execute Command (Select AuthorId, AuthorName from ListAuthor)
            //Generate List<AuthorModel>
            //Return List<AuthorModel>

            string CS = ConfigurationManager.ConnectionStrings
              ["SqlDbCS"].ConnectionString;

            SqlConnection conn = new SqlConnection(CS);
            SqlCommand cmd = new SqlCommand();
            SqlDataReader DataReader;

            cmd.CommandText = "SELECT AuthorId, AuthorName FROM ListAuthor";
            cmd.CommandType = System.Data.CommandType.Text;
            cmd.Connection = conn;

            if (conn.State == System.Data.ConnectionState.Closed)
            {
                conn.Open();
            }

            DataReader = cmd.ExecuteReader();

            List<AuthorModel> ListAuthor = new List<AuthorModel>();
            AuthorModel Author;

            while (DataReader.Read())
            {
                Author = new AuthorModel();
                Author.AuthorId = (int)DataReader["AuthorId"];
                Author.AuthorName = DataReader["AuthorName"].ToString();
                ListAuthor.Add(Author);
            }



            conn.Close();
            cmd.Dispose();
```

```
            DataReader.Close();

            return ListAuthor;

        }

        public List<AuthorModel> Authors()
        {
            //Connection Open
            //Build and Execute Command (Select AuthorId, AuthorName from ListAuthor)
            //Generate List<AuthorModel>
            //Return List<AuthorModel>

            string CS = ConfigurationManager.ConnectionStrings
              ["SqlDbCS"].ConnectionString;

            SqlConnection conn = new SqlConnection(CS);
            SqlCommand cmd = new SqlCommand();
            SqlDataReader DataReader;

            cmd.CommandText = "SELECT AuthorId, AuthorName FROM ListAuthor";
            cmd.CommandType = System.Data.CommandType.Text;
            cmd.Connection = conn;

            if (conn.State == System.Data.ConnectionState.Closed)
            {
                conn.Open();
            }

            DataReader = cmd.ExecuteReader();

            List<AuthorModel> ListAuthor = new List<AuthorModel>();
            AuthorModel Author;

            while (DataReader.Read())
            {
                Author = new AuthorModel();
                Author.AuthorId = (int)DataReader["AuthorId"];
                Author.AuthorName = DataReader["AuthorName"].ToString();
                ListAuthor.Add(Author);
            }

            conn.Close();
            cmd.Dispose();
            DataReader.Close();

            return ListAuthor;

        }

        public List<AuthorModel> AuthorsDisconnected()
        {
```

```
            //Connection Open
            //Build and Execute Command (Select AuthorId, AuthorName from ListAuthor)
            //Generate List<AuthorModel>
            //Return List<AuthorModel>

            string CS = ConfigurationManager.ConnectionStrings
              ["SqlDbCS"].ConnectionString;
            string QueryStr = "SELECT * FROM ListAuthor";

            using (SqlConnection DBConnection = new SqlConnection(CS))
            {
                DataTable dt = new DataTable();
                SqlDataAdapter da = new SqlDataAdapter(QueryStr, DBConnection);

                da.Fill(dt);

                List<AuthorModel> ListAuthor = new List<AuthorModel>();
                AuthorModel Author;

                //Complicated Code
                //for (int i=0; i<dt.Rows.Count; i++)
                //{
                //    Author = new AuthorModel();
                //    Author.AuthorId = int.Parse(dt.Rows[i]["AuthorId"].ToString());
                //    Author.AuthorName = dt.Rows[i]["AuthorName"].ToString();
                //    ListAuthor.Add(Author);
                //}

                //Simplified Version
                foreach(DataRow row in dt.Rows)
                {
                    Author = new AuthorModel();
                    Author.AuthorId = int.Parse(row["AuthorId"].ToString());
                    Author.AuthorName = row["AuthorName"].ToString();
                    ListAuthor.Add(Author);
                }

                return ListAuthor;
            }

        }

        public List<AuthorModel> AuthorsDisconnectedDS()
        {
            //Connection Open
            //Build and Execute Command (Select AuthorId, AuthorName from ListAuthor)
            //Generate List<AuthorModel>
            //Return List<AuthorModel>

            string CS = ConfigurationManager.ConnectionStrings
              ["SqlDbCS"].ConnectionString;
            string QueryStr = "SELECT * FROM ListAuthor";
```

```
            using (SqlConnection DBConnection = new SqlConnection(CS))
            {
                DataSet dt = new DataSet();
                SqlDataAdapter da = new SqlDataAdapter(QueryStr, DBConnection);

                da.Fill(dt);

                List<AuthorModel> ListAuthor = new List<AuthorModel>();
                AuthorModel Author;


                //Simplified Version
                foreach (DataRow row in dt.Tables[0].Rows)
                {
                    Author = new AuthorModel();
                    Author.AuthorId = int.Parse(row["AuthorId"].ToString());
                    Author.AuthorName = row["AuthorName"].ToString();
                    ListAuthor.Add(Author);
                }

                return ListAuthor;
            }

        }

        public SqlDataReader GetAllAuthors2()
        {
            //Connection Open
            //Build and Execute Command (Select AuthorId, AuthorName from ListAuthor)
            //Generate List<AuthorModel>
            //Return List<AuthorModel>

            string CS = ConfigurationManager.ConnectionStrings
              ["SqlDbCS"].ConnectionString;

            SqlConnection conn = new SqlConnection(CS);
            SqlCommand cmd = new SqlCommand();
            //SqlDataReader DataReader;

            cmd.CommandText = "SELECT AuthorId, AuthorName FROM ListAuthor";
            cmd.CommandType = System.Data.CommandType.Text;
            cmd.Connection = conn;

            if (conn.State == System.Data.ConnectionState.Closed)
            {
                conn.Open();
            }

            return cmd.ExecuteReader();

        }
```

```
    }
}
```