Business Object Layer (Class Library)

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.ComponentModel.DataAnnotations.Schema;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace BOLayer

{

[Table("Inventory")]

public class Inventory

{

[Key]

[DatabaseGenerated(DatabaseGeneratedOption.Identity)]

public int id { get; set; }

[Required(ErrorMessage = "Item Name is must")]

[MinLength(10, ErrorMessage = "Min characters 10 are needed")]

[MaxLength(20, ErrorMessage = "Max 20 characteres are allowed")]

public string name { get; set; }

[Required(ErrorMessage = "Qty is must")]

[Range(1, 100, ErrorMessage = "Qty Range is 1- 100")]

public int? qty\_stock { get; set; }

[Required(ErrorMessage = "Vendor Name is must")]

public string vendor { get; set; }

[DataType(DataType.Password)]

[Required]

public string password { get; set; }

[DataType(DataType.Password)]

[Required]

[Compare("password", ErrorMessage = "do not match")]

public string comparepassword { get; set; }

//

//[DataType(DataType.EmailAddress)]

[EmailAddress(ErrorMessage = "Not correct Format")]

public string EMail { get; set; }

}

}

**All your classes will be here**

**2nd Project : Data Access Layer** **(Class Library Project)**

**Here Add reference to BOLayer Library , Add DbContext Class & one interface**

Add InventoryDbContext Class

using BOLayer;

using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DALLayer

{

class InventoryDbContext : DbContext

{

public DbSet<Inventory> Inventories { get; set; }

private void FixEfProviderServicesProblem()

{

var instance = System.Data.Entity.SqlServer.SqlProviderServices.Instance;

}

}

}

Dal.cs

**using BOLayer;**

**using System;**

**using System.Collections.Generic;**

**using System.Data.Entity.Validation;**

**using System.Linq;**

**using System.Text;**

**using System.Threading.Tasks;**

**using System.Web.ModelBinding;**

**namespace DALLayer**

**{**

**public interface IInventory**

**{**

**List<Inventory> GetInventories();**

**void InsertInventory(Inventory obj);**

**void DeleteInventory(int id);**

**}**

**}**

In App.config file, add

<entityFramework>

<providers>

<provider invariantName="System.Data.SqlClient" type="System.Data.Entity.SqlServer.SqlProviderServices, EntityFramework.SqlServer"/>

</providers>

</entityFramework>

<connectionStrings>

<add name="InventoryDbContext" connectionString="server=LAPTOP-53S2KQS8\SQLEXPRESS;initial catalog=StockDB;integrated security=true" providerName="System.Data.SqlClient"/>

</connectionStrings>

**3rd Project Business Access Layer** **(Class Library)**

**Here , Add reference to BOLayer & DALLyer**

using BOLayer;

using DALLayer;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace BALLayer

{

public class Class1 : IInventory

{

BOLayer.Inventory inventory =new BOLayer.Inventory();

public void DeleteInventory(int id)

{

DALLayer.InventoryDbContext db = new InventoryDbContext();

// db.Inventories.Remove(id);

}

public List<Inventory> GetInventories()

{

DALLayer.InventoryDbContext db = new InventoryDbContext();

return db.Inventories.ToList();

}

public void InsertInventory(Inventory obj)

{

DALLayer.InventoryDbContext db = new InventoryDbContext();

db.Inventories.Add(obj);

db.SaveChanges();

}

}

}

**4th Project : MVC Client**

**Add reference to BALLayer, BOLayer**

using System;

using System.Collections.Generic;

using System.Data.Entity.Validation;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace MVCClient.Controllers

{

public class InventoryController : Controller

{

// GET: Inventory

public ActionResult Index()

{

BALLayer.Class1 bal = new BALLayer.Class1();

List<BOLayer.Inventory> inventories = new List<BOLayer.Inventory>();

inventories = bal.GetInventories();

return View(inventories);

}

// GET: Inventory/Details/5

public ActionResult Details(int id)

{

return View();

}

// GET: Inventory/Create

public ActionResult Create()

{

BOLayer.Inventory inventory = new BOLayer.Inventory();

return View(inventory);

}

// POST: Inventory/Create

[HttpPost]

[HandleError]

public ActionResult Create(BOLayer.Inventory inventory)

{

try

{

BALLayer.Class1 x = new BALLayer.Class1();

x.AddInventory(inventory);

// TODO: Add insert logic here

return RedirectToAction("Index");

}

catch (DbEntityValidationException dbError)

{

foreach (DbEntityValidationResult entityErr in

dbError.EntityValidationErrors)

{

foreach (DbValidationError error in entityErr.ValidationErrors)

{

string str = "";

str = "Error Property Name " +

error.PropertyName + " Error: " + error.ErrorMessage;

return View("error");

}

}

}

return View();

}

// GET: Inventory/Edit/5

public ActionResult Edit(int id)

{

return View();

}

// POST: Inventory/Edit/5

[HttpPost]

public ActionResult Edit(int id, FormCollection collection)

{

try

{

// TODO: Add update logic here

return RedirectToAction("Index");

}

catch

{

return View();

}

}

// GET: Inventory/Delete/5

public ActionResult Delete(int id)

{

return View();

}

// POST: Inventory/Delete/5

[HttpPost]

public ActionResult Delete(int id, FormCollection collection)

{

try

{

// TODO: Add delete logic here

return RedirectToAction("Index");

}

catch

{

return View();

}

}

}

}

Here in web.config file, copy entries of connectionstring from DALLayer