**Phase-End Project**

**EMS Full Stack App-Requirement**

**Source code:**

**BLClass1.cs:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace BLL

{

public class BLClass1

{

public int EmpCode { get; set; }

public string EmpName { get; set; }

public string Email { get; set; }

public int DeptCode{ get; set; }

public DateTime DOB { get; set; }

}

}

**DAL:**

**Class1.cs:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.ComponentModel.DataAnnotations;

using System.Data.Entity;

using System.Linq;

using System.Runtime.Remoting.Contexts;

using System.Text;

using System.Threading.Tasks;

namespace DAL

{

public class EmpProfile

{

[Key]

[Required]

public int EmpCode { get; set; }

[MaxLength(20, ErrorMessage = "Not allowed above 20 chars")]

[MinLength(2, ErrorMessage = "Not allowed below 2 chars")]

public string EmpName { get; set; }

public string Email{ get; set; }

public int DeptCode{ get; set; }

public DateTime DateOfBirth { get; set; }

public virtual ICollection<DeptMaster> I { get; set; }

// [DataType(DataType.DateTime,ErrorMessage ="not valid date")]

}

public class DeptMaster

{

[Key]

[Required]

public int DeptCode { get; set; }

[MaxLength(20, ErrorMessage = "Not allowed above 20 chars")]

[MinLength(2, ErrorMessage = "Not allowed below 2 chars")]

public string DeptName { get; set; }

public virtual ICollection<EmpProfile> e { get; set; }

// [DataType(DataType.DateTime,ErrorMessage ="not valid date")]

}

public class EmpDBInitializer : DropCreateDatabaseIfModelChanges<MyContext1>

{

protected override void Seed(MyContext1 context)

{

IList<DeptMaster> defaultStandards = new List<DeptMaster>();

defaultStandards.Add(new DeptMaster() { DeptCode=1,DeptName="IT" });

context.DeptTable.AddRange(defaultStandards);

base.Seed(context);

}

}

public class MyContext1 : DbContext

{

public MyContext1() : base("MyContext1")

{

//createdatabase if not exists

//drop create always

//drop create if model changes

Database.SetInitializer<MyContext1>(new DropCreateDatabaseIfModelChanges<MyContext1>());

Database.SetInitializer(new EmpDBInitializer());

}

public virtual DbSet<EmpProfile> EmpTable { get; set; }

public virtual DbSet<DeptMaster> DeptTable { get; set; }

}

}

**EmpOperation.cs:**

using BLL;

using System;

using System.Collections;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.Linq;

using System.Runtime.Remoting.Contexts;

using System.Text;

using System.Threading.Tasks;

namespace DAL

{

public class EmpOperation

{

public bool DeleteEmployeeDetails(int id)

{

try

{

MyContext1 context = new MyContext1();

List<EmpProfile> s = context.EmpTable.ToList();

EmpProfile r = s.Find(pr => pr.EmpCode == id);

context.EmpTable.Remove(r);

context.SaveChanges();

return true;

}

catch (Exception ex)

{

return false;

}

}

public BLClass1 search(int id)

{

MyContext1 context = new MyContext1();

List<EmpProfile> customers = context.EmpTable.ToList();

EmpProfile obj = customers.Find(cust => cust.EmpCode == id);

// List<BLClass1> cblist = new List<BLClass1>();

BLClass1 b = new BLClass1();

b.EmpName = obj.EmpName;

b.EmpCode = obj.EmpCode;

b.Email=obj.Email;

b.DeptCode=obj.DeptCode;

b.DOB = obj.DateOfBirth;

return b;

//context.SaveChanges();

}

public bool Insert(BLClass1 bal)

{

try

{

MyContext1 context = new MyContext1();

EmpProfile b = new EmpProfile();

b.EmpCode = bal.EmpCode;

b.EmpName = bal.EmpName;

b.Email = bal.Email;

b.DeptCode = bal.DeptCode;

b.DateOfBirth = bal.DOB;

context.EmpTable.Add(b);

context.SaveChanges();

return true;

}

catch(Exception ex)

{

return false;

}

}

public bool UpdateEmployeeDetails(BLClass1 bal)

{

try

{

MyContext1 context = new MyContext1();

List<EmpProfile> customers = context.EmpTable.ToList();

EmpProfile obj = customers.Find(cust => cust.EmpCode == bal.EmpCode);

obj.EmpName = bal.EmpName;

obj.Email = bal.Email;

obj.DeptCode = bal.DeptCode;

obj.DateOfBirth = bal.DOB;

// context.Updatebookdetails();

context.SaveChanges();

return true;

}

catch(Exception ex)

{

return false;

}

}

public List<BLClass1> GetAllEmployeeDetails()

{

MyContext1 context = new MyContext1();

List<EmpProfile> clist = context.EmpTable.ToList();

List<BLClass1> cblist = new List<BLClass1>();

foreach (var item in clist)

{

cblist.Add(new BLClass1 { EmpCode=item.EmpCode,EmpName=item.EmpName,Email=item.Email,DeptCode=item.DeptCode,DOB=item.DateOfBirth });

}

return cblist;

}

}

}

**EmpsController:**

using BLL;

using DAL;

using Project.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Security.Cryptography;

using System.Web.Http;

namespace Project.Controllers

{

public class EmpsController : ApiController

{

// GET api/<controller>

EmpOperation obj = null;

public EmpsController()

{

obj = new EmpOperation();

}

// [Route("GetAllMarks")]

[HttpGet]

public List<Emp> GetEmployeeList()

{

//sub\_mark --model

//BLclass1

List<BLClass1> empbal = new List<BLClass1>();

empbal = obj.GetAllEmployeeDetails();

List<Emp> emps = new List<Emp>();

foreach (var item in empbal)

{

//Employees emp = new Employees();

emps.Add(new Emp { EmpCode = item.EmpCode, EmpName = item.EmpName, Email = item.Email, DeptCode = item.DeptCode, DOB = item.DOB });

}

return emps;

}

// GET api/<controller>/5

// [Route("FindById/{id:int:min(1)}")]

[Route("FindById/{id:int?}")]

public Emp GetEmployeeByID(int id)

{

BLClass1 empbal = new BLClass1();

empbal = obj.search(id);

Emp emp = new Emp();

//emp.Id = empbal.Id;

emp.EmpCode= id;

emp.EmpName = empbal.EmpName;

emp.Email = empbal.Email;

emp.DeptCode = empbal.DeptCode;

emp.DOB = empbal.DOB;

return emp;

}

// POST api/<controller>

public HttpResponseMessage PostEmpDetails([FromBody] Emp empdata)

{

BLClass1 empbal = new BLClass1();

empbal.EmpCode = empdata.EmpCode;

empbal.EmpName = empdata.EmpName;

empbal.Email = empdata.Email;

empbal.DeptCode = empdata.DeptCode;

empbal.DOB = empdata.DOB;

bool ans = obj.Insert(empbal);

if (ans)

{

return Request.CreateResponse(HttpStatusCode.OK);

}

else

{

return Request.CreateResponse(HttpStatusCode.NotAcceptable);

}

}

// PUT api/<controller>/5

public HttpResponseMessage PutEmpDetails([FromBody] Emp empdata)

{

BLClass1 empbal = new BLClass1();

empbal.EmpCode= empdata.EmpCode;

empbal.EmpName = empdata.EmpName;

empbal.Email = empdata.Email;

empbal.DeptCode= empdata.DeptCode;

empbal.DOB=empdata.DOB;

bool ans = obj.UpdateEmployeeDetails(empbal);

if (ans)

{

return Request.CreateResponse(HttpStatusCode.OK);

}

else

{

return Request.CreateResponse(HttpStatusCode.NotAcceptable);

}

}

// DELETE api/<controller>/5

public HttpResponseMessage DeleteEmpDetails(int id)

{

bool ans = obj.DeleteEmployeeDetails(id);

if (ans)

{

return Request.CreateResponse(HttpStatusCode.OK);

}

else

{

return Request.CreateResponse(HttpStatusCode.NotAcceptable);

}

}

}

}

**Emp.cs:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace Project.Models

{

public class Emp

{

public int EmpCode { get; set; }

public string EmpName { get; set; }

public string Email { get; set; }

public int DeptCode { get; set; }

public DateTime DOB { get; set; }

}

}