**Source Code Of Practice Project 4 :**

**Github Link:**

<https://github.com/11812142/SimpliLearnProject/tree/master/ASP.%20NET/Simpli_Practice4>

1. **Bal Library**

**Bal.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Bal

{

public class BAL

{

public int student\_id { get; set; }

public string student\_name { get; set; }

public int subject\_marks { get; set; }

}

}

1. **Dal Library**

**Dal.cs**

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Bal;

namespace Dal

{

public class DAL

{

public bool Insert(BAL school)

{

// SqlConnection cn = new SqlConnection(ConfigurationManager.ConnectionStrings["NorthCnString"].ConnectionString);

SqlConnection cn = new SqlConnection("Data Source=DESKTOP-9E6TH3I\\SQLEXPRESS;Initial Catalog=school;Integrated Security=True");

SqlCommand cmdInsert = new SqlCommand("insert into marks(student\_id,student\_name,subject\_marks) values(@student\_id,@student\_name,@subject\_marks)", cn);

cmdInsert.Parameters.AddWithValue("@student\_id", school.student\_id);

cmdInsert.Parameters.AddWithValue("@student\_name", school.student\_name);

cmdInsert.Parameters.AddWithValue("@subject\_marks", school.subject\_marks);

cn.Open();

int i = cmdInsert.ExecuteNonQuery();

bool status = false;

if (i == 1)

{

status = true;

}

cn.Close();//finally

cn.Dispose();//finally

return status;

}

public bool Update(BAL school)

{

// SqlConnection cn = new SqlConnection(ConfigurationManager.ConnectionStrings["NorthCnString"].ConnectionString);

SqlConnection cn = new SqlConnection("Data Source=DESKTOP-9E6TH3I\\SQLEXPRESS;Initial Catalog=school;Integrated Security=True");

SqlCommand cmdUpdate = new SqlCommand("[dbo].[Updatemarks]", cn);

cmdUpdate.CommandType = System.Data.CommandType.StoredProcedure;

cmdUpdate.Parameters.AddWithValue("@p\_studid", school.student\_id);

cmdUpdate.Parameters.AddWithValue("@p\_studname", school.student\_name);

cmdUpdate.Parameters.AddWithValue("@p\_submarks", school.subject\_marks);

cn.Open();

int s = cmdUpdate.ExecuteNonQuery();

bool statusd = false;

if (s == 1)

{

statusd = true;

}

cn.Close();//finally

cn.Dispose();//finally

return statusd;

}

public BAL Find(int id)

{

// SqlConnection cn = new SqlConnection(ConfigurationManager.ConnectionStrings["NorthCnString"].ConnectionString);

SqlConnection cn = new SqlConnection("Data Source=DESKTOP-9E6TH3I\\SQLEXPRESS;Initial Catalog=school;Integrated Security=True");

SqlCommand cmdSelect = new SqlCommand("[dbo].sp\_Findmarks", cn);

cmdSelect.CommandType = System.Data.CommandType.StoredProcedure;

cmdSelect.Parameters.AddWithValue("@p\_studid", id);

SqlParameter p1 = new SqlParameter();

p1.ParameterName = "@p\_mark\_studname";

p1.SqlDbType = System.Data.SqlDbType.NVarChar;

p1.Size = 20;

p1.Direction = System.Data.ParameterDirection.Output;

cmdSelect.Parameters.Add(p1);

SqlParameter p2 = new SqlParameter();

p2.ParameterName = "@p\_marks\_submarks";

p2.SqlDbType = System.Data.SqlDbType.NVarChar;

p2.Size = 20;

p2.Direction = System.Data.ParameterDirection.Output;

cmdSelect.Parameters.Add(p2);

cn.Open();

cmdSelect.ExecuteNonQuery();

BAL found = new BAL();

found.student\_name = p1.Value.ToString();

found.subject\_marks = Convert.ToInt32(p2.Value);

cn.Close();

cn.Dispose();

return found;

}

public List<BAL> list()

{

// SqlConnection cn = new SqlConnection(ConfigurationManager.ConnectionStrings["NorthCnString"].ConnectionString);

// SqlConnection cn = new SqlConnection("Data Source =DESKTOP-LUAVTH3; Initial Catalog = school; Integrated Security = True");

SqlConnection cn = new SqlConnection("Data Source=DESKTOP-9E6TH3I\\SQLEXPRESS;Initial Catalog=school;Integrated Security=True");

SqlCommand cmdlist = new SqlCommand("select \* from marks", cn);

cn.Open();

SqlDataReader dr = cmdlist.ExecuteReader();

List<BAL> emplist = new List<BAL>();

if (dr.HasRows)

{

while (dr.Read())

{

BAL bal = new BAL();

bal.student\_id = Convert.ToInt32(dr["student\_id"]);

bal.student\_name = dr["student\_name"].ToString();

bal.subject\_marks = Convert.ToInt32(dr["subject\_marks"]);

emplist.Add(bal);

}

}

cn.Close();

cn.Dispose();

return emplist;

}

public bool Delete(int id)

{

// SqlConnection cn = new SqlConnection(ConfigurationManager.ConnectionStrings["NorthCnString"].ConnectionString);

SqlConnection cn = new SqlConnection("Data Source=DESKTOP-9E6TH3I\\SQLEXPRESS;Initial Catalog=school;Integrated Security=True");

SqlCommand cmdDelete = new SqlCommand("[dbo].sp\_Deletemarks", cn);

cmdDelete.CommandType = System.Data.CommandType.StoredProcedure;

cmdDelete.Parameters.AddWithValue("@p\_id", id);

cn.Open();

int i = cmdDelete.ExecuteNonQuery();

bool status = false;

if (i == 1)

{

status = true;

}

cn.Close();//finally

cn.Dispose();//finally

return status;

}

}

}

1. **Helper Library**

**Helper.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Bal;

using Dal;

namespace Helper

{

public class Helper1

{

DAL dal = null;

public Helper1()

{

dal = new DAL();

}

public bool Addmarks(BAL employee)

{

return dal.Insert(employee);

}

public bool Editmarks(BAL employee)

{

return dal.Update(employee);

}

public BAL searchmarks(int empid)

{

return dal.Find(empid);

}

public List<BAL> listmarks()

{

return dal.list();

}

public bool removemarks(int employee\_id)

{

return dal.Delete(employee\_id);

}

}

}

1. **In SimpliPractice4**

**ValuesController----- Controller**

using Bal;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web.Http;

using Helper;

using Simpli\_Practice4.Models;

namespace Simpli\_Practice4.Controllers

{

public class ValuesController : ApiController

{

// GET api/values----receive

Helper1 obj = null;

public ValuesController()

{

obj = new Helper1();

}

[HttpGet]

public List<marksmodel> marklist()

{

//return new string[] { "value1", "value2" };

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

empbal = obj.listmarks();

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

foreach (var item in empbal)

{

//Employees emp = new Employees();

emps.Add(new marksmodel

{

student\_id = item.student\_id,

student\_name = item.student\_name,

subject\_marks = item.subject\_marks

});

}

return emps;

}

public marksmodel marksbyid(int id)

{

BAL empbal = new BAL();

empbal = obj.searchmarks(id);

marksmodel emp = new marksmodel();

emp.student\_id = empbal.student\_id;

emp.student\_name = empbal.student\_name;

emp.subject\_marks = empbal.subject\_marks;

return emp;

//return "value";

}

// POST api/<controller> --- Update

public HttpResponseMessage Postmarks([FromBody] marksmodel empdata)

{

BAL empbal = new BAL();

empbal.student\_id = empdata.student\_id;

empbal.student\_name = empdata.student\_name;

empbal.subject\_marks = empdata.subject\_marks;

bool ans = obj.Addmarks(empbal);

if (ans)

{

return Request.CreateResponse(HttpStatusCode.OK);

}

else

{

return Request.CreateResponse(HttpStatusCode.NotAcceptable);

}

}

// PUT api/<controller>/5 -- add

[HttpPut]

public HttpResponseMessage Putmarks([FromBody] marksmodel empdata)

{

BAL empbal = new BAL();

empbal.student\_id = empdata.student\_id;

empbal.student\_name = empdata.student\_name;

empbal.subject\_marks = empdata.subject\_marks;

bool ans = obj.Editmarks(empbal);

if (ans)

{

return Request.CreateResponse(HttpStatusCode.OK);

}

else

{

return Request.CreateResponse(HttpStatusCode.NotAcceptable);

}

}

// DELETE api/<controller>/5

public HttpResponseMessage Deletemarks(int id)

{

bool ans = obj.removemarks(id);

if (ans)

{

return Request.CreateResponse(HttpStatusCode.OK);

}

else

{

return Request.CreateResponse(HttpStatusCode.NotAcceptable);

}

}

}

}

**Marksmodel.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace Simpli\_Practice4.Models

{

public class marksmodel

{

public int student\_id { get; set; }

public string student\_name { get; set; }

public int subject\_marks { get; set; }

}

}

**Used Stored Procedure in this projects ----- Follow Github Link**

**https://github.com/11812142/SimpliLearnProject/tree/master/MS%20SQL**