**NewsInfo.cs**

[Key]

public int id { get; set; }

public string title { get; set; }

public string description { get; set; }

**NewsInfoContext**

public class NewsInfoContext:DbContext

{

public NewsInfoContext(DbContextOptions<NewsInfoContext> option):base(option)

{

}

public DbSet<NewsInfo> NewsInfos { get; set; }

}

**NewsInfoVM**

public class NewsInfoVM

{

[Key]

public int id { get; set; }

public string title { get; set; }

public string description { get; set; }

}

**Appsettings.json**

{

"ConnectionStrings": {

"DefaultConnection": "Server=(localdb)\\MSSQLLocalDB;Database=nothinganything;Trusted\_Connection=True;MultipleActiveResultSets=true"

}

}

**Startup**

public class Startup

{

//

public IConfiguration Configuration { get;}

public Startup(IConfiguration Con)

{

Configuration = Con;

}

//

public void ConfigureServices(IServiceCollection services)

{

//

services.AddMvc();

services.AddDbContext<NewsInfoContext>(options => options.UseSqlServer(Configuration.GetConnectionString("DefaultConnection")));

//

}

public void Configure(IApplicationBuilder app, IHostingEnvironment env)

{

if (env.IsDevelopment())

{

app.UseDeveloperExceptionPage();

}

//

app.UseMvc(routes =>

{

routes.MapRoute("default", "{controller=Home}/{action=index}/{id?}");

});

//

app.Run(async (context) =>

{

await context.Response.WriteAsync("Hello World!");

});

}

}

**Migration Commands:**

Add-migration firstmigration

Update-database

**MyServices.cs(Interface)**

interface MyServices

{

IActionResult Get();

IActionResult Create(NewsInfoVM vm);

IActionResult Edit(int id, NewsInfo nw);

IActionResult Delete(int id);

}

**HomeController**

public class HomeController : Controller, MyServices

{

public IActionResult Index()

{

return View();

}

NewsInfoContext myCtx;

public HomeController(NewsInfoContext ctx)

{

myCtx = ctx;

}

// Get

public IActionResult Get()

{

return Json(myCtx.NewsInfos.ToList());

}

// Create

[HttpPost]

public IActionResult Create([FromBody]NewsInfoVM vm)

{

NewsInfo nw = new NewsInfo();

nw.title = vm.title;

nw.description = vm.description;

myCtx.NewsInfos.Add(nw);

myCtx.SaveChanges();

return Json(true);

}

// Edit

[HttpPost]

public IActionResult Edit(int id, [FromBody]NewsInfo nw)

{

var a = (from n in myCtx.NewsInfos where id == n.id select n).First();

a.title = nw.title;

a.description = nw.description;

myCtx.SaveChanges();

return Json(true);

}

// Delete

public IActionResult Delete(int id)

{

myCtx.NewsInfos.Remove(myCtx.NewsInfos.Find(id));

myCtx.SaveChanges();

return Json(true);

}

}

**Index.cshtml**

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width" />

<title>Index</title>

<link href="~/Style.css" rel="stylesheet" />

<link href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet" />

<script src="http://code.jquery.com/jquery-3.3.1.min.js"></script>

<script src="https://knockoutjs.com/downloads/knockout-3.4.2.js"></script>

<script type="text/javascript">

var model = {

id: ko.observable(""),

title: ko.observable(""),

description: ko.observable(""),

records: ko.observableArray([])

}

function blank() {

$('#t1').val("");

$('#t2').val("");

$('#t3').val("");

}

function Ajax(method, url) {

var productData = {

"id": $("#t1").val(),

"title": $("#t2").val(),

"description": $("#t3").val(),

};

return new Promise(function (resolve, reject) {

$.ajax({

type: method,

url: url,

data: JSON.stringify(productData),

dataType: "json",

contentType: "application/json",

success: function (data) {

res = data;

resolve(res);

blank();

},

error: function (xhr) {

alert(xhr.responseText)

}

});

});

}

function refresh() {

Ajax("get", "/Home/Get").then(function (a) {

model.records(a);

})

}

refresh();

function work(m, u, a) {

Ajax(m, u).then(function (a) {

refresh();

})

}

function edit(a, b, c) {

$("#t1").val(a);

$("#t2").val(b);

$("#t3").val(c);

}

$(document).ready(function () {

ko.applyBindings();

})

</script>

</head>

<body>

<div class="container">

<h4 style="font-weight:bold">CRUD</h4>

<hr style="border:1px solid #2980b9" />

<div class="form-group">

<label for="id" class="col-md-2">ID</label>

<div class="col-md-10">

<input type="text" id="t1" data-bind="value: model.id" class="form-control" /><br />

</div>

</div>

<div class="form-group">

<label for="title" class="col-md-2">Title</label>

<div class="col-md-10">

<input type="text" id="t2" data-bind="value: model.title" class="form-control" /><br />

</div>

</div>

<div class="form-group">

<label for="description" class="col-md-2">Description</label>

<div class="col-md-10">

<input type="text" id="t3" data-bind="value: model.description" class="form-control" />

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10" style="margin: 20px 0">

<input type="button" class="btn btn-success" value="Add" data-bind="click:function(){work('POST','/Home/Create', alert('Successfully Inserted'))}" />

<input type="button" class="btn btn-info" value="Update" data-bind="click:function(){work('POST','/Home/Edit/'+$('#t1').val(), alert('Successfully Updated'))}" />

<input type="button" class="btn btn-danger" value="Delete" data-bind="click:function(){work('get','/Home/Delete/'+$('#t1').val(), alert('Successfully Deleted'))}" /><br />

</div>

</div>

</div>

<table class="table table-bordered table-striped" id="news" style="width: 80%; margin: 0 auto">

<thead>

<tr>

<th>Title</th>

<th>Description</th>

<th>Action</th>

</tr>

</thead>

<tbody data-bind="foreach:model.records()">

<tr>

<td data-bind="text:title"></td>

<td data-bind="text:description"></td>

<td><a href="#" class="btn btn-primary btn-xs" data-bind="click: function(){edit(id, title, description)}">Edit</a></td>

</tr>

</tbody>

</table>

</body>

</html>