**solution**

**CODE:**

**Index.cshtml:**

<div class="jumbotron" style="color:antiquewhite">

<h1 style="color:black" >CAR DEALER</h1>

<p style="color:black" class="lead">WELCOME TO CAR DEALER!</p>

<p><a href="https://localhost:44301/Home/CreateCars" class="btn btn-primary btn-lg"> Go to Cars&raquo;</a></p>

</div>

<div class="row">

<p>Please select the above buttons to add cars to the stock. This is a simple CRUD application which utilizes WEB API concept taught in Cloud Computing Lab.</p>

</div>

**CreateCars.cshtml:**

@model CloudComputingOEL.Models.CarsStock

@{

ViewBag.Title = "CreateCars";

}

<h2>CreateCars</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h4>CarsStock</h4>

<hr />

@Html.ValidationSummary(true, "", new { @class = "text-danger" })

<div class="form-group">

@Html.LabelFor(model => model.CarName, htmlAttributes: new { @class = "control-label col-md-2" })

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

@Html.EditorFor(model => model.CarName, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.CarName, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.CarModel, htmlAttributes: new { @class = "control-label col-md-2" })

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

@Html.EditorFor(model => model.CarModel, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.CarModel, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.CarPrice, htmlAttributes: new { @class = "control-label col-md-2" })

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

@Html.EditorFor(model => model.CarPrice, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.CarPrice, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.CarColor, htmlAttributes: new { @class = "control-label col-md-2" })

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

@Html.EditorFor(model => model.CarColor, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.CarColor, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<input type="submit" value="Create" class="btn btn-default" />

</div>

</div>

</div>

}

<div>

@Html.ActionLink("Back to List", "Index")

</div>

@section Scripts {

@Scripts.Render("~/bundles/jqueryval")

}

**ViewCars.cshtml:**

@model IEnumerable<CloudComputingOEL.Models.CarsStock>

@{

ViewBag.Title = "ViewCars";

}

<h2>ViewCars</h2>

<p>

@Html.ActionLink("Create New", "Create")

</p>

<table class="table">

<tr>

<th>

@Html.DisplayNameFor(model => model.CarName)

</th>

<th>

@Html.DisplayNameFor(model => model.CarModel)

</th>

<th>

@Html.DisplayNameFor(model => model.CarPrice)

</th>

<th>

@Html.DisplayNameFor(model => model.CarColor)

</th>

<th></th>

</tr>

@foreach (var item in Model) {

<tr>

<td>

@Html.DisplayFor(modelItem => item.CarName)

</td>

<td>

@Html.DisplayFor(modelItem => item.CarModel)

</td>

<td>

@Html.DisplayFor(modelItem => item.CarPrice)

</td>

<td>

@Html.DisplayFor(modelItem => item.CarColor)

</td>

<td>

@Html.ActionLink("Edit", "Edit", new { /\* id=item.PrimaryKey \*/ }) |

@Html.ActionLink("Details", "Details", new { /\* id=item.PrimaryKey \*/ }) |

@Html.ActionLink("Delete", "Delete", new { /\* id=item.PrimaryKey \*/ })

</td>

</tr>

}

</table>

**DeleteCars.cshtml:**

@model CloudComputingOEL.Models.CarsStock

@{

ViewBag.Title = "DeleteCars";

}

<h2>DeleteCars</h2>

<h3>Are you sure you want to delete this?</h3>

<div>

<h4>CarsStock</h4>

<hr />

<dl class="dl-horizontal">

<dt>

@Html.DisplayNameFor(model => model.CarName)

</dt>

<dd>

@Html.DisplayFor(model => model.CarName)

</dd>

<dt>

@Html.DisplayNameFor(model => model.CarModel)

</dt>

<dd>

@Html.DisplayFor(model => model.CarModel)

</dd>

<dt>

@Html.DisplayNameFor(model => model.CarPrice)

</dt>

<dd>

@Html.DisplayFor(model => model.CarPrice)

</dd>

<dt>

@Html.DisplayNameFor(model => model.CarColor)

</dt>

<dd>

@Html.DisplayFor(model => model.CarColor)

</dd>

</dl>

@using (Html.BeginForm()) {

@Html.AntiForgeryToken()

<div class="form-actions no-color">

<input type="submit" value="Delete" class="btn btn-default" /> |

@Html.ActionLink("Back to List", "Index")

</div>

}

</div>

**EditCars.cshtml:**

@model CloudComputingOEL.Models.CarsStock

@{

ViewBag.Title = "EditCars";

}

<h2>EditCars</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h4>CarsStock</h4>

<hr />

@Html.ValidationSummary(true, "", new { @class = "text-danger" })

<div class="form-group">

@Html.LabelFor(model => model.CarName, htmlAttributes: new { @class = "control-label col-md-2" })

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

@Html.EditorFor(model => model.CarName, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.CarName, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.CarModel, htmlAttributes: new { @class = "control-label col-md-2" })

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

@Html.EditorFor(model => model.CarModel, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.CarModel, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.CarPrice, htmlAttributes: new { @class = "control-label col-md-2" })

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

@Html.EditorFor(model => model.CarPrice, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.CarPrice, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.CarColor, htmlAttributes: new { @class = "control-label col-md-2" })

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

@Html.EditorFor(model => model.CarColor, new { htmlAttributes = new { @class = "form-control" } })

@Html.ValidationMessageFor(model => model.CarColor, "", new { @class = "text-danger" })

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<input type="submit" value="Save" class="btn btn-default" />

</div>

</div>

</div>

}

<div>

@Html.ActionLink("Back to List", "Index")

</div>

@section Scripts {

@Scripts.Render("~/bundles/jqueryval")

}

**CarDetails.cshtml:**

@model CloudComputingOEL.Models.CarsStock

@{

ViewBag.Title = "CarDetails";

}

<h2>CarDetails</h2>

<div>

<h4>CarsStock</h4>

<hr />

<dl class="dl-horizontal">

<dt>

@Html.DisplayNameFor(model => model.CarName)

</dt>

<dd>

@Html.DisplayFor(model => model.CarName)

</dd>

<dt>

@Html.DisplayNameFor(model => model.CarModel)

</dt>

<dd>

@Html.DisplayFor(model => model.CarModel)

</dd>

<dt>

@Html.DisplayNameFor(model => model.CarPrice)

</dt>

<dd>

@Html.DisplayFor(model => model.CarPrice)

</dd>

<dt>

@Html.DisplayNameFor(model => model.CarColor)

</dt>

<dd>

@Html.DisplayFor(model => model.CarColor)

</dd>

</dl>

</div>

<p>

@Html.ActionLink("Edit", "EditCars", new { /\* id = Model.PrimaryKey \*/ }) |

@Html.ActionLink("Back to List", "Index")

</p>

**CarController.cs (API CONTROLLER):**

using CloudComputingOEL.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web.Http;

namespace CloudComputingOEL.Controllers

{

public class CarController : ApiController

{

// GET: api/Car

[HttpGet]

public IEnumerable<CarsStock> GetAllcarDetails()

{

CarsStock ST = new CarsStock();

List<CarsStock> li = new List<CarsStock>();

ST.CarName = "xx";

ST.CarModel = "xx";

ST.CarPrice = "xx";

ST.CarColor = "red";

li.Add(ST);

return li;

}

// GET: api/Car/5

public IEnumerable<CarsStock> Get(int id)

{

CarsStock ST = new CarsStock();

List<CarsStock> li = new List<CarsStock>();

if (id == 1)

{

ST.CarName = "xx";

ST.CarModel = "xx";

ST.CarPrice = "xx";

ST.CarColor = "red";

li.Add(ST);

}

return li;

}

// POST: api/Car

public void Post([FromBody]string value)

{

}

// PUT: api/Car/5

public void Put(int id, [FromBody]string value)

{

}

// DELETE: api/Car/5

public void Delete(int id)

{

}

}

}

**HomeController.cs:**

using CloudComputingOEL.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace CloudComputingOEL.Controllers

{

public class HomeController : Controller

{

public ActionResult Index()

{

ViewBag.Title = "Car Home Page";

return View();

}

public ActionResult CreateCars()

{

return View("CreateCars");

}

public ActionResult CarDetails()

{

return View("CarDetails");

}

public ActionResult ViewCars()

{

CarsStock obj = new CarsStock

{

CarName = "asd",

CarModel = "gag",

CarPrice = "123",

CarColor = "red"

};

return View("ViewCars");

}

public ActionResult DeleteCars()

{

return View("DeleteCars");

}

public ActionResult EditCars()

{

return View("EditCars");

}

}

}

**CarsStock.cs (MODEL):**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace CloudComputingOEL.Models

{

public class CarsStock

{

public string CarName { get; set; }

public string CarModel { get; set; }

public string CarPrice { get; set; }

public string CarColor { get; set; }

}

}

**SOME CHANGES IN layout.cshtml:**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

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

<title>@ViewBag.Title</title>

@Styles.Render("~/Content/css")

@Scripts.Render("~/bundles/modernizr")

</head>

<body>

<div class="navbar navbar-inverse navbar-fixed-top">

<div class="container">

<div class="navbar-header">

<img src="C:\Users\Spring2020\Pictures\Camera Roll\biryani1.jpg" />

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

@Html.ActionLink("Car Dealer", "Index", "Home", new { area = "" }, new { @class = "navbar-brand" })

</div>

<div class="navbar-collapse collapse">

<ul class="nav navbar-nav">

<li>@Html.ActionLink("Home", "Index", "Home", new { area = "" }, null)</li>

<li>@Html.ActionLink("API", "Index", "Help", new { area = "" }, null)</li>

<li>@Html.ActionLink("Create", "CreateCars", "Home", new { area = "" }, null)</li>

<li>@Html.ActionLink("Details", "CarDetails", "Home", new { area = "" }, null)</li>

<li>@Html.ActionLink("Delete", "DeleteCars", "Home", new { area = "" }, null)</li>

<li>@Html.ActionLink("View", "ViewCars", "Home", new { area = "" }, null)</li>

<li>@Html.ActionLink("Edit", "EditCars", "Home", new { area = "" }, null)</li>

</ul>

</div>

</div>

</div>

<div class="container body-content">

@RenderBody()

<hr />

<footer>

<p>&copy; @DateTime.Now.Year - CarDealership by Diyan</p>

</footer>

</div>

@Scripts.Render("~/bundles/jquery")

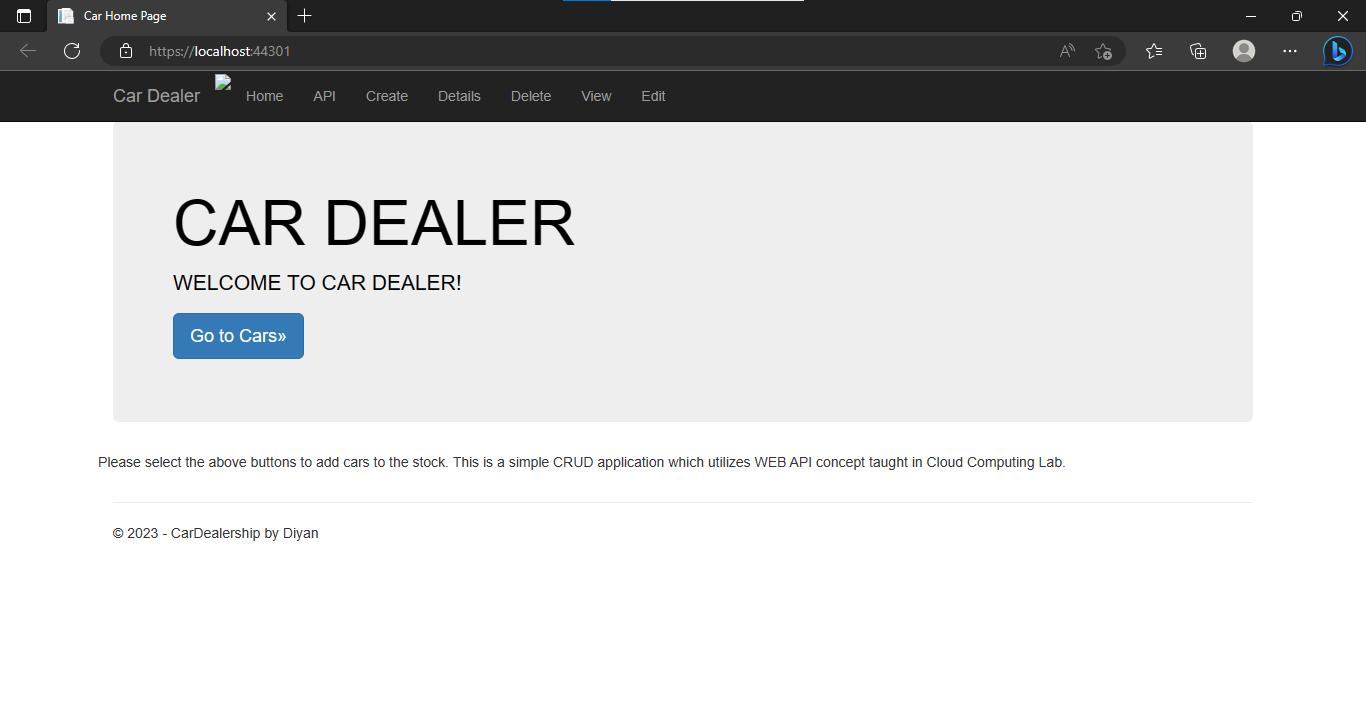
@Scripts.Render("~/bundles/bootstrap")

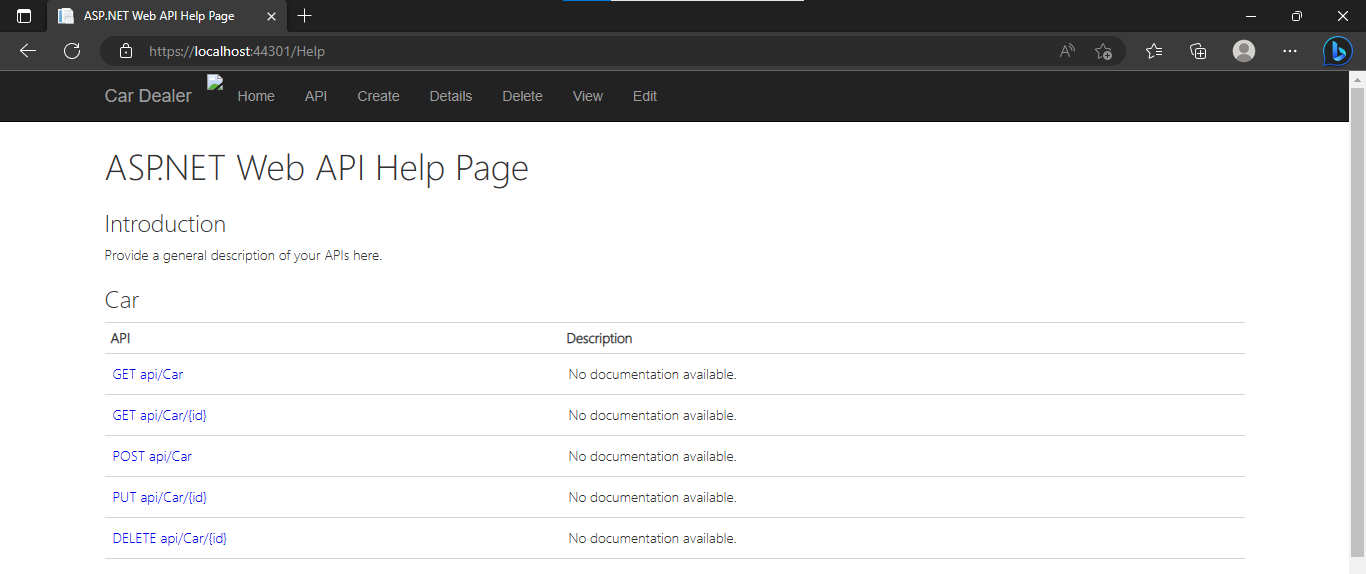
@RenderSection("scripts", required: false)

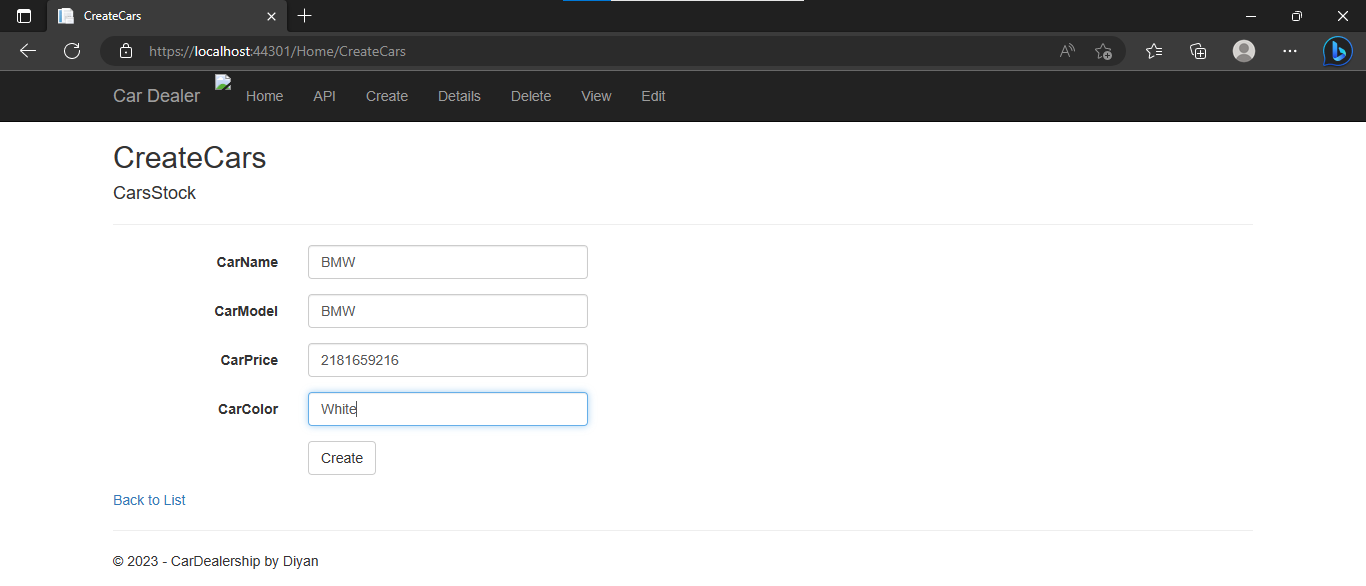
</body>

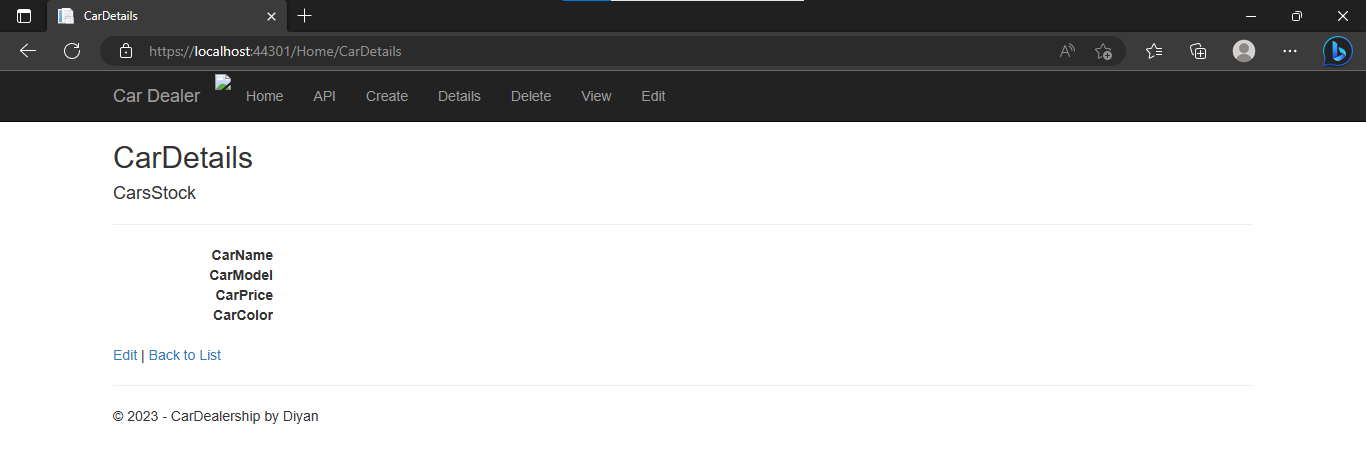
</html>

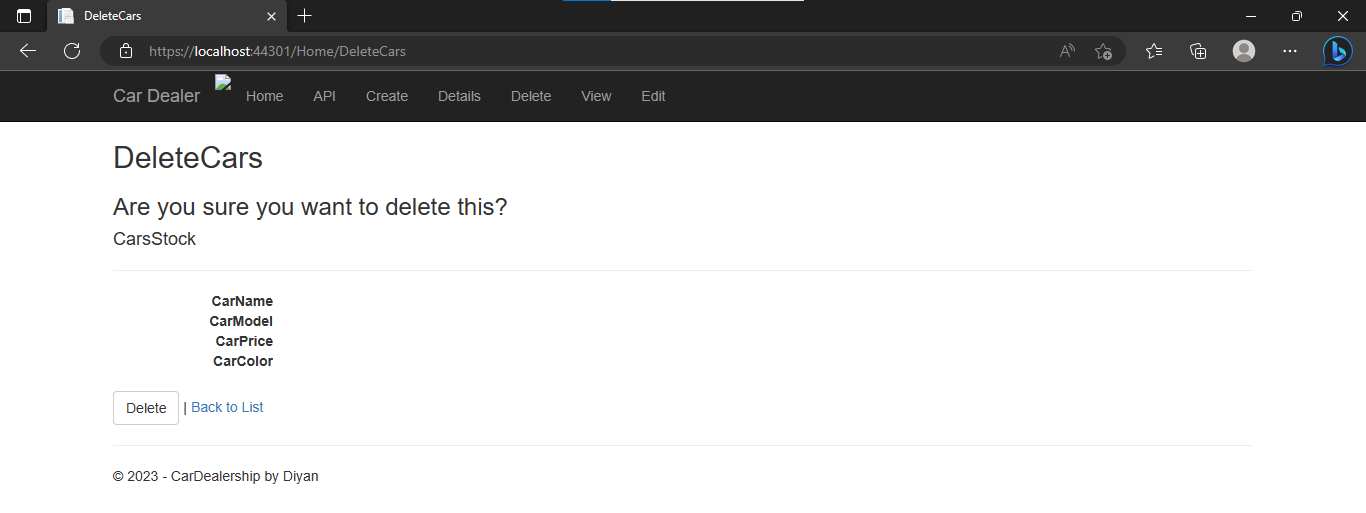
**SCREENSHOTS:**

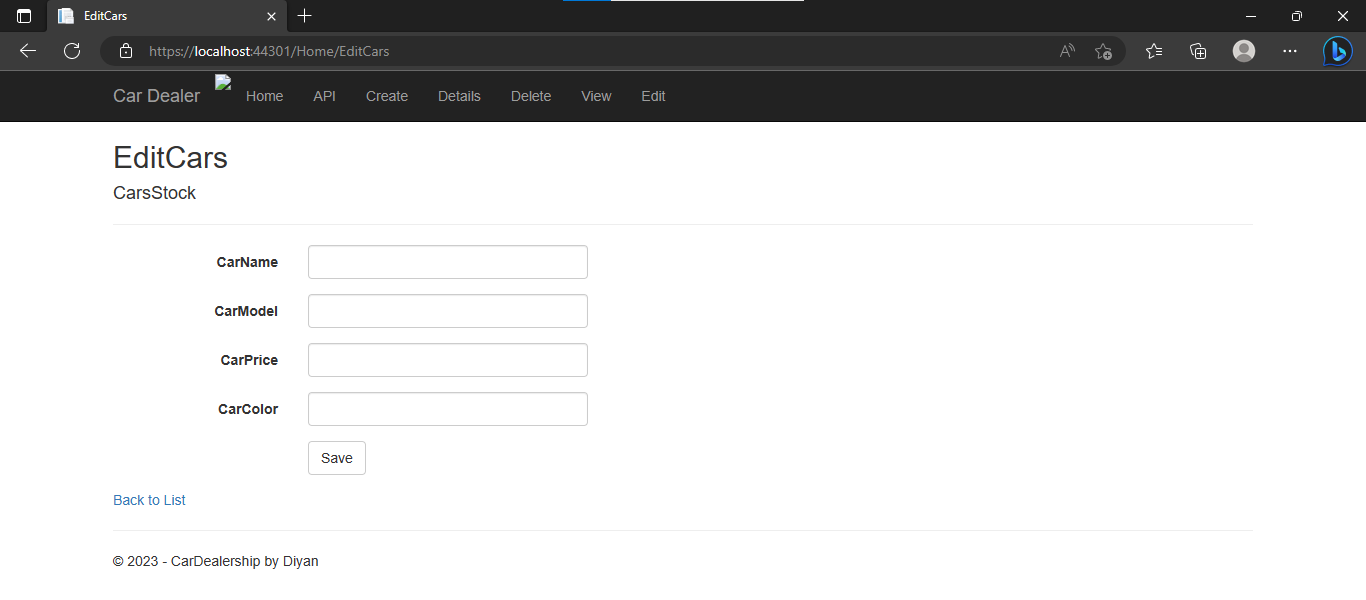
****

****

****

****

****

****