

Weekly project 1
Dharunya S

Create an MVC Web app with the ability to add a food (or) cloth product and view in the table

[Program.cs](#)

```
using Microsoft.EntityFrameworkCore;
using Week1Pro.Context;

var builder = WebApplication.CreateBuilder(args);

// Add services to the container.
builder.Services.AddControllersWithViews();
const string connectionString = "Data Source= PTSQLTESTDB01;"
    + "Initial Catalog=Demo_Dharunya;" +
    "Integrated Security=true;" +
    "TrustServerCertificate=True;";
builder.Services.AddDbContext<AppDbContext>(x =>
x.UseSqlServer(connectionString));
var app = builder.Build();

// Configure the HTTP request pipeline.
if (!app.Environment.IsDevelopment())
{
    app.UseExceptionHandler("/Home/Error");
}
app.UseStaticFiles();

app.UseRouting();

app.UseAuthorization();

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");

app.Run();
```

[AppDbContext.cs](#)

```
using Week1Pro.Models;
using Microsoft.EntityFrameworkCore;

namespace Week1Pro.Context
{
    public class AppDbContext : DbContext
    {
        public AppDbContext(DbContextOptions<AppDbContext> options) : base(options) {
        }

        public DbSet<Product> ProductDet { get; set; }

    }
}
```

[HomeController.cs](#)

```
using Microsoft.AspNetCore.Mvc;
using Microsoft.Identity.Client.Extensions.Msal;
using System;
using System.Diagnostics;
using Week1Pro.Context;
using Week1Pro.Models;
using static System.Formats.Asn1.AsnWriter;

namespace Week1Pro.Controllers
{
    public class HomeController : Controller
    {
        private readonly ILogger<HomeController> _logger;

        private AppDbContext AppDbContext { get; set; }

        public HomeController(ILogger<HomeController> logger, AppDbContext
_appDbContext)
        {
            _logger = logger;
            AppDbContext = _appDbContext;
        }

        public IActionResult Index()
        {
            return View();
        }
    }
}
```

```

public IActionResult Privacy()
{
    return View();
}
public IActionResult ProductDet()
{
    return View();
}
public IActionResult AddProductdet(Product product)
{
    AppDbContext.ProductDet.Add(product);
    AppDbContext.SaveChanges();
    return RedirectToAction("AddProduct");
}
public IActionResult AddProduct() {

    var allproduct = AppDbContext.ProductDet.ToList();
    var tot_price = allproduct.Sum(x => x.Price);
    ViewBag.tot_price = tot_price;

    return View(allproduct);
}
public IActionResult DeleteAction(int id)
{
    var products = AppDbContext.ProductDet.FirstOrDefault(x => x.Id == id);
    if (products != null)
    {
        AppDbContext.ProductDet.Remove(products);
        AppDbContext.SaveChanges();
    }
    return RedirectToAction("AddProduct");
}

public IActionResult Edit(int id)
{
    var product = AppDbContext.ProductDet.FirstOrDefault(x => x.Id == id);
    return View(product);
}

[HttpPost]
public IActionResult Edit(Product product)

```

```

    {
        AppDbContext.ProductDet.Update(product);
        AppDbContext.SaveChanges();
        return RedirectToAction("AddProduct");
    }

    [ResponseCache(Duration = 0, Location = ResponseCacheLocation.None, NoStore =
true)]
    public IActionResult Error()
    {
        return View(new ErrorViewModel { RequestId = Activity.Current?.Id ??
HttpContext.TraceIdentifier });
    }
}

```

[Product.cs](#)

```
namespace Week1Pro.Models
```

```

{
    public class Product
    {
        public int Id { get; set; }
        public string ProName { get; set; }
        public int Price { get; set; }
        public int quantity { get; set; }
    }
}

```

[ProductDet.cs](#)

```

@{
    ViewData["Title"] = "Our Products";
}
@model Week1Pro.Models.Product

```

```

<div class="text-center">
    <h1 class="display-4">Create Edit Products</h1>
</div>
<form asp-action="AddProductdet">

    <label class="mt-4">Product Name</label>
    <input class="form-control" type="text" placeholder="" asp-for="ProName" />
    <label class="mt-4">Price</label>
    <input class="form-control" type="number" placeholder="" asp-for="Price" />
    <label class="mt-4">Quantity</label>

```

```
<input class="form-control" type="number" placeholder="" asp-for="quantity" />
<button class="btn btn-primary mt-4" type="submit">ok</button>
```

```
</form>
```

AddProduct.cs

```
@{
    ViewData["Title"] = "Our Product";
}
@model List<Product>
<div class="text-center">
    <h1 class="display-4">Our Products</h1>
    <a asp-controller="Home" asp-action="ProductDet" class="btn
btn-primary">AddProduct</a>
<br />
    Total price : @ViewBag.tot_price

    <table class="table">
        <thead>
            <tr>
                <th>Product Name</th>
                <th>Price</th>
                <th>Quantity</th>
                <th>Action</th>

            </tr>
        </thead>
        <tbody>
            @foreach (var product in Model)
            {
                <tr>
                    <td>@product.ProName</td>
                    <td>@product.Price</td>
                    <td>@product.quantity</td>
                    <td>
                        <a asp-action="DeleteAction" asp-route-id="@product.Id">Delete</a>
                        <a asp-action="Edit" asp-route-id="@product.Id">Edit</a>
                    </td>
                </tr>
            }
        </tbody>
    </table>
</div>
```

Output:

Our Products

AddProduct

Total price : 10

Product Name	Price	Quantity	Action
slice	10	2	Delete Edit

Create Edit Products

Product Name

Price

Quantity

ok

message

	Id	ProName	Price	quantity
1	1	slice	10	2

