<!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>

@RenderBody()

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

@RenderSection("scripts", required: false)

</body>

</html>

@{

Layout = "~/Views/Shared/\_Layout.cshtml";

}

namespace MvcRazor.Models

{

public class Product

{

public int ProductId { get; set; }

public string Name { get; set; }

public string Description { get; set; }

public decimal Price { get; set; }

public string Category { get; set; }

}

}

using MvcRazor.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace MvcRazor.Controllers

{

public class ProductController : Controller

{

//

// GET: /Product/

public ActionResult Index()

{

List<Product> products = new List<Product>();

products.Add(new Product()

{

ProductId = 1,

Name = "Шариковая ручка",

Description = "Синяя шариковая ручка с колпачком и прозрачным корпусом.",

Price = 3m,

Category = "Канцтовары"

});

products.Add(new Product()

{

ProductId = 2,

Name = "Бумага A4",

Description = "Стандартная бумага для цветной и чёрно-белой печати.",

Price = 15m,

Category = "Канцтовары"

});

products.Add(new Product()

{

ProductId = 2,

Name = "Мобильный телефон",

Description = "Мобильный телефон с фотокамерой.",

Price = 250m,

Category = "Техника"

});

// Возвращаем представление из директории Views/Products/Index.cshtml

// Параметр передающийся в метод View() является моделью, которая будет доступна только на чтение в представлении Index

return View(products);

}

}

}

@\*Все операторы Razor начинаются с символа '@'

Данное представление строго типизированное. Строка №5 указывает на тип объекта модели.

Модель, переданная в контроллере в качестве параметра метода View, может быть использована по ссылке @Model\*@

@model IEnumerable<MvcRazor.Models.Product>

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

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

<title>Index</title>

</head>

<body>

@foreach (var item in Model)

{

@\*Чтение свойства элемента модели\*@

<h3>Имя продукта: @item.Name</h3>

if (item.Category == "Канцтовары")

{

@\*Razor понимает, что оператор в теле if начинается с HTML дескриптора,

и поэтому относится к нему как к разметке, которую нужно генерировать\*@

<p>На канцтовары действует <b>скидка 10%</b></p>

}

if (item.Price < 5)

{

@\* Префикс строки @: указывает что строку нужно обработать так, как если она будет HTML разметкой\*@

@:стоимость доставки 10 грн.

}

<hr />

}

</body>

</html>