using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace \_04\_ChildActionUsing.Models

{

public class Product

{

public int Id { get; set; }

public string Name { get; set; }

public decimal Price { get; set; }

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace \_04\_ChildActionUsing.Models

{

public class ProductCollection

{

public static List<Product> All

{

get

{

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

for (int i = 0; i < 20; i++)

{

products.Add(new Product()

{

Id = i + 1,

Name = "Item Name " + i,

Price = (i + 1) \* 2

});

}

return products;

}

}

}

}

using \_04\_ChildActionUsing.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace \_04\_ChildActionUsing.Controllers

{

public class HomeController : Controller

{

//

// GET: /Home/

public ActionResult Index()

{

return View();

}

public ActionResult ShowTable(int numberOfRows = 5)

{

IEnumerable<Product> products = ProductCollection.All.Take(numberOfRows);

return PartialView("\_Table", products);

}

}

}

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

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

<title>Index</title>

</head>

<body>

<div>

@\*В отличии от частичных представлений дочерние действия позволяют выполнять дополнительные действия в контроллере\*@

@Html.Action("ShowTable", new { numberOfRows = 10 })

</div>

</body>

</html>

@model IEnumerable<\_04\_ChildActionUsing.Models.Product>

<table border="1">

<tr>

<th>

#

</th>

<th>

Name

</th>

<th>

Price

</th>

</tr>

@foreach (var item in Model)

{

<tr>

<td>

@item.Id

</td>

<td>

@item.Name

</td>

<td>

$ @item.Price

</td>

</tr>

}

</table>