namespace SegmentVariables

{

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using System.Web.Routing;

public class RouteConfig

{

public static void RegisterRoutes(RouteCollection routes)

{

routes.IgnoreRoute("{resource}.axd/{\*pathInfo}");

routes.MapRoute(

name: "Default",

url: "{controller}/{action}/{id}",

defaults: new { controller = "Home", action = "Index", id = "DefaultId" }

);

// !!!!!!

// Некторые имена зарезервированы и не могут использоваться как имена для переменных сегментов.

// К таким именам относятся controller, action и area

}

}

}

namespace SegmentVariables.Controllers

{

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

public class HomeController : Controller

{

//

// GET: /Home/

public ActionResult Index()

{

// RouteData.Values - коллекция значений сегментов, доступных в момент обработки текущего запроса

// ViewBag - динамический словарь, который позволяет передавать данные в представление

ViewBag.Variable = RouteData.Values["id"];

return View();

}

//

// GET: /CustomVariable/

// Отображение переменной id на параметр метода доступа.

// MVC Framework будет пытаться преобразовать значение, полученное из URL, в тип указанный в параметрах.

public ActionResult CustomVariable(int? id)

{

ViewBag.Variable = id;

// Метод действия CustomVariable по умолчанию должен возвращать представление CustomVariable.cshtml из директории Views/Home

// Строковое значение, переданное в качестве параметра метода View, позволяет задать другое представление, которое вернется в ответ на запрос

return View("Index");

}

}

}

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

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

<title>Index</title>

</head>

<body>

<div>

@\* @ViewBag.Variable вывод значение свойства из динамической коллекции\*@

<h2>Variable: @ViewBag.Variable</h2>

<a href="/">Variable = DefaultId (~/)</a> <br />

<a href="/Home/Index/123">Variable = 123 (~/Home/Index/123)</a><br />

<a href="/Home/CustomVariable/456">Variable = 456 (~/Home/CustomVariable/456)</a><br />

</div>

</body>

</html>