Routing

In an ASP.NET MVC application, the browser's URL will be pointing to a **particular method** within a **particular controller**.

Routing is the process performed by your application that takes a URL and translates it into a request for that **particular method** within the controller.
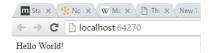
Let's say that we have a simple MVC application with the following:

One controller: **HomeController**
One method: **Index()**
One view: **Index.cshtml**   This view will just display the words "Hello World!".

When we run the application from within Visual Studio, we see that the URL that is used is something like is seen below. (The number will be different.)

Notice that in the url, the controller called HomeController is not specified and the Index() method is not specified.

And yet, the Index method was executed and the view was displayed. That is because Index() is the default method and HomeController is the default controller.

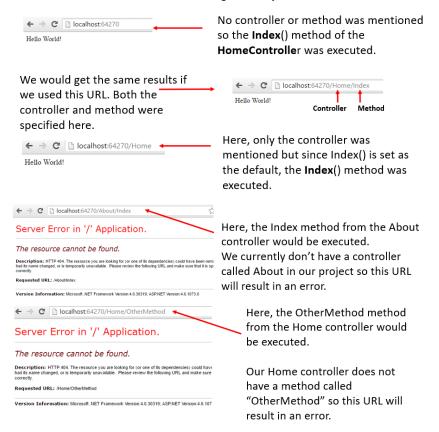## RouteConfig.cs in App_Start sets the default routing behavior.

This means that the application will interpret any URL with a pattern like ****/**** (**any two words separated with a slash**) this way: The first word will tell the application which controller to look in, and the second word with tell the application which action method in that controller should be executed.

Whenever we create a new ASP.NET MVC application, Index() is set as the default method and HomeController is set as the default controller. This is set in RouteConfig.cs as we just saw.

No controller or method was mentioned so the **Index**() method of the **HomeControlle**r was executed.

We would get the same results if we used this URL. Both the controller and method were specified here.

Here, only the controller was mentioned but since Index() is set as the default, the **Index**() method was executed.

Here, the Index method from the About controller would be executed.
We currently don't have a controller called About in our project so this URL will result in an error.

Here, the OtherMethod method from the Home controller would be executed.

Our Home controller does not have a method called "OtherMethod" so this URL will result in an error.

We can change RouteConfig.cs if we would like.

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6  using System.Web.Routing;
7
8  namespace WebApplication1
9  {
10     public class RouteConfig
11     {
12         public static void RegisterRoutes(RouteCollection routes)
13         {
14             routes.IgnoreRoute("{resource}.axd/{*pathInfo}");
15
16             routes.MapRoute(
17                 name: "Default",
18                 url: "{controller}/{action}/{id}",
19                 defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }
20             );
21         }
22     }
23  }
24
```

Here are some examples of other types of routing behaviors we might want to implement.

```
routes.MapRoute(null, "", new { controller = "Product", action = "List", category = (string)null, page = 1 });

routes.MapRoute(null, "Page{page}", new { controller = "Product", action = "List", category = (string)null }, new { page = @"\d+" });

routes.MapRoute(null, "{category}", new { controller = "Product", action = "List", page = 1 });

routes.MapRoute(null, "{category}/Page{page}", new { controller = "Product", action = "List" }, new { page = @"\d+" });

routes.MapRoute(null, "{controller}/{action}");
```

Given the routes in the code above, answer the question below.

For each URL, tell what method would be executed in what controller. And is there any extra information passed?

www.mysite.com   List method in the Product controller. A value of 1 will be passed in for the variable called page and null will be passed in for the variable called category.

www.mysite.com/Store/Create   Create method in the Store controller. No information is passed.

www.mysite.com/Page4   List method in the Product controller. A value of 4 will be passed in for the variable called page and null will be passed in for the variable called category.

www.mysite.com/Sports/Page2   List method in the Product controller. Sports will be passed in for the variable called category and a value of 2 will be passed in for the variable called page.

www.mysite.com/Lumber   List method in the Product controller. Lumber will be passed in for the variable called category and a value of 1 will be passed in for the variable called page.

End-of-Class Questions

Assume RouteConfig.cs includes the following statements:

```
routes.MapRoute(null, "{school}/Page{page}", new { controller = "Home", action = "Index" });
routes.MapRoute(null, "{action}/{controller}");
```

What URL would you use if you wanted to execute the MyMethod method inside the controller called CarsController?

www.mysite.com_____

Assume the browser sends a request from the URL www.mysite.com/CCU/Page1.

What method would be run?

In what controller would that method be?

What value would be sent into the method for the variable called school?

What value would be sent into the method for the variable called page?