Instructor example

Web University by Harsha Vardhan

What is Routing?

Routing is functionality that map incoming request to the route handler. The route can have route parameters to receive values from the URL. Using the route, routing can find a route handler based on the URL. All the routes are registered when the application is started. There are two types of routing supported by ASP.NET Core

1. The conventional routing
2. Attribute routing

The Routing uses routes to map incoming requests with the route handler and Generates URL that is used in response. Mostly, the application has a single collection of routes and this collection is used for the process of the request. The RouteAsync method is used to map incoming requests (that match the URL) with available in route collection.

How Routing works in ASP.NET Core?

Routing is used to handle incoming HTTP requests for the app. Routing finds matching executable endpoint for incoming requests. These endpoints are registered when app starts. Matching process use values from incoming request url to process the requests. You can configure the routing in middleware pipeline of configure method in startup class.

```
app.UseRouting(); // It adds route matching to middleware pipeline

// It adds endpoints execution to middleware pipeline
app.UseEndpoints(endpoints =>
{
 endpoints.MapGet("/", async context =>
 {
  await context.Response.WriteAsync("Hello World!");
 });
});
```

When will you prefer attribute routing over conventional routing?

Attribute routing gives you more control over the URIs in your web application, when you want to create each url independently – that means, when all URLs of the application doesn’t follow any common pattern.

What are the important route constraints?

The Route Constraints helps us to filter out or restrict the unwanted values from reaching the controller action or middleware.

What is the purpose of the wwwroot folder?

The wwwroot folder contains static files and compiled assets, such as JavaScript, CSS, and images that your web application needs. wwwroot is the only folder in the entire project that's exposed as-is to the browser.

How do you change the path of wwwroot folder?

We need to set path of the wwwroot folder in the WebRootPath propertyof the WebApplicationOptions class.

```
var builder = WebApplication.CreateBuilder(new WebApplicationOptions() {
 WebRootPath = "foldername"
});
```