Rawlston Morton

September 30, 2020

Core 3 Ch 10

1. Blazor is a framework for building interactive client-side web UI with .NET
2. The AddServerSideBlazor method creates the services that Blazor uses, and the MapBlazorHub method registers the Blazor middleware components. The final addition is to finesse the routing system to ensure that Blazor works seamlessly with the rest of the application.
3. Blazor Server apps are .NET Core projects. The project file for the Blazor Server app is about as simple as it can get: <Project Sdk="Microsoft.NET.Sdk.Web">

<PropertyGroup>

<TargetFramework>netcoreapp3.0</TargetFramework>

</PropertyGroup>

</Project>

1. The component will use the browser’s current URL to locate a Razor Component that can be displayed to the user. If no matching component can be found, then an error message is displayed.
2. <Link> and <NavLink> components, which allow you to navigate to different routes defined in the application. These navigation components can be thought of as being like anchor links on the page that allow you to navigate to other pages in the site. In a traditional website, when you navigate through the application using anchor links, it results in a page refresh, and all the components in the page are re-rendered. Navigation links created with <Link> and <NavLink> do not result in a page refresh; only those certain sections of the page that are defined using the <Route> and that match the URL path are updated.
3. Razor Components, as the name suggests, rely on the Razor approach to annotated HTML elements. The view part of the component is supported by the statements in the @code section. The @code section in this component defines four properties that are decorated with the Parameter attribute, which means the values will be provided at runtime.
4. You can use a razor component that displays a table without knowing which category of order it is dealing with. This is used to avoid duplicating code.
5. So, CRUD stands for (Create, Read, Update, Delete). In SQL these would be: Insert, Select, Update and Delete.
6. Blazor uses the same approach to validation as the rest of ASP.NET Core but, as you will see, applies it a different way to deal with the more interactive nature of Razor Components.
7. The component uses the Inject attribute to declare that it requires an implementation of the IStoreRepository interface, which is one of the ways that Blazor provides access to the application’s services.
8. NavigationManager class is used to programmatically navigate between components without triggering a new HTTP request. The Editor component uses NavigationManager, which is obtained as a service, to return to the Products component after the database has been updated.