Split razor component

There are 2 approaches, to split component HTML and C# code into their own separate files.

1. Partial files approach
2. Base class approach

Single file or Mixed file approach

Both the HTML markup and C# code are in a single file.

// Counter.razor

@page "/counter"

<h1>Counter</h1>

<p>Current count: @currentCount</p>

<button class="btn btn-primary" @onclick="IncrementCount">Click me</button>

@code {

private int currentCount = 0;

private void IncrementCount()

{

currentCount++;

}

}

Copy

Partial files approach

The HTML markup remains in Counter.razor file.

// Counter.razor

@page "/counter"

<h1>Counter</h1>

<p>Current count: @currentCount</p>

<button class="btn btn-primary" @onclick="IncrementCount">Click me</button>

Copy

Remember, when the component is compiled a class with the same name as the component file is generated. Create another class file with name Counter.razor.cs and include the following code in it. Notice the class in this file is implemented as a partial class.

// Counter.razor.cs

public partial class Counter

{

private int currentCount = 0;

private void IncrementCount()

{

currentCount++;

}

}

Copy

You can toggle file nesting, using the File nesting button on the Solution Explorer.

Base class approach

Just like the partial class approach, even with the base class approach, the HTML markup remains in Counter.razor file.

// Counter.razor

@page "/counter"

<h1>Counter</h1>

<p>Current count: @currentCount</p>

<button class="btn btn-primary" @onclick="IncrementCount">Click me</button>

Copy

Move the C# code to a separate class. I named it CounterBase. You can name the class anything you want, but it is a common convention to have the same name as the component but suffixed with the word Base.

In this example, the component name is Counter. So the class that contains the c# code is named CounterBase. The class has to inherit from the built-in ComponentBase class. This class is in Microsoft.AspNetCore.Components namespace.

The access modifier must be at least protected if you wish to access the class member from the HTML.

// CounterBase.cs

public class CounterBase : ComponentBase

{

protected int currentCount = 0;

protected void IncrementCount()

{

currentCount++;

}

}

Copy

Finally, in Counter.razor file do not forget to include the following inherits directive.

@inherits CounterBase

Copy

Slides









