

Links com Tag Helpers

Transcrição

Vamos mexer na interface da nossa aplicação, que atualmente conta com três páginas: o carrossel, o carrinho e o cadastro do cliente. Atualmente, não estamos passando o título da página para a renderização do arquivo `.cshtml`. Além disso, começaremos a implementar os links utilizando ASP.NET Core MVC, já que não conseguimos navegar de uma página para a outra.

Sendo assim, abriremos nosso projeto com as quatro páginas da aplicação, `Carrossel.cshtml`, `Carrinho.cshtml`, `Cadastro.cshtml` e `Resumo.cshtml`. No carrossel, adicionaremos o título, e para isto incluiremos `@` no início do arquivo para indicar o código C#, seguida das chaves.

Daremos um nome para o título desta página, acessando um cenário de dados chamado `viewData`, um dicionário de dados que é passado e renderizado pela *view*, assim como o modelo, para capturar alguma informação, no caso, o título. Colocaremos o nome `Title`, "título" em inglês, como chave deste dicionário. O ASP.NET Core MVC renderizará este título na aba do navegador, cujo nome será "Carrossel".

```
@{  
    ViewData["Title"] = "Carrossel";  
}
```

Faremos o mesmo para as outras páginas, copiando e colando o trecho acima, e trocando "Carrossel" pelos títulos correspondentes, isto é, em:

- `Carrinho.cshtml` teremos "Carrinho";
- `Cadastro.cshtml` teremos "Cadastro";
- `Resumo.cshtml` teremos "Resumo".

Em seguida, voltaremos a `Carrossel.cshtml`, em que o botão "Adicionar" não realiza nenhuma ação. Teremos que criar o link para a página de carrinho, o que faremos com um elemento HTML, a âncora `a`, com o `href` que seria o link para a página de carrinho. Em vez de utilizarmos um link com `href`, utilizaremos uma técnica chamada **Tag Helper**, usada na inclusão de um atributo especial que se inicia com a palavra "asp".

Ao usarmos `<a asp-controller="pedido">Adicionar`, por exemplo, o ASP.NET Core MVC nos dá uma lista de opções de atributos de *tag helper*, sendo uma delas `asp-controller`, que escolheremos porque queremos navegar para uma *action* de um determinado *controller*, neste caso, `pedido`. Também usaremos `asp-action` para navegarmos à página do carrinho:

```
<div class="text-center">  
    <a asp-controller="pedido"  
        asp-action="carrinho"  
        class="btn btn-success">Adicionar</a>  
</div>
```

Note que, com isso, o Visual Studio mudou a cor da âncora e a deixou em negrito, indicando que o elemento `a` não é mais uma âncora HTML, e sim uma *tag helper*, ou seja, um componente do ASP.NET Core utilizado pelo mecanismo de renderização **Razor** para a criação de HTML de maneira mais natural.

A impressão é que estamos incluindo um código HTML sendo que estamos colocando um componente do servidor do ASP.NET Core. Rodaremos a aplicação novamente para vermos o resultado desta alteração. Desta vez, ao passarmos o mouse sobre o botão, o link exibido na parte inferior da tela é o endereço da página referente ao carrinho.

Adiante, implementaremos os outros links para a nossa aplicação utilizando as *tag helpers*.