

Ajude Nestor a enviar um formulário para sua API

Nestor está enviando os dados de seu formulário através do código abaixo.

Aqui o código para enviar a requisição...

```
public async Task PostLivroAsync(LivroUpload livro)
{
    var resposta = await httpClient.PostAsync($"livros", livro);
    resposta.EnsureSuccessStatusCode();
}
```

...e aqui o código da action:

```
[HttpPost]
public async Task<IActionResult> Incluir([FromQuery] LivroUpload livro)
{
    await _api.PostLivroAsync(livro);
    //continuação do código...
}
```

Mas o código não está nem compilando. O que Nestor deve fazer para fazer com que o livro seja incluído através da API?

Selecione uma alternativa

A Alterar o método de envio da requisição para `GetAsync()` .

B Criar um objeto do tipo `HttpContent` e *embrulhar* o livro nele, passando esse objeto como argumento do método `PostAsync()` , assim:

```
public async Task PostLivroAsync(LivroUpload livro)
{
    var content = new HttpContent(livro);
    var resposta = await httpClient.PostAsync($"livros", content);
    resposta.EnsureSuccessStatusCode();
}
```

C Criar um objeto do tipo `MultipartFormDataContent` e *embrulhar* o livro nele, passando esse objeto como argumento do método `PostAsync()` , conforme código abaixo (nele omiti o método `CreateMultipartFormData()`). Além disso é preciso mudar o *binding* do argumento livro para `[FromForm]` .

```
public async Task PostLivroAsync(LivroUpload livro)
{
    var content = CreateMultipartFormData(livro);
    var resposta = await httpClient.PostAsync($"livros", content);

    resposta.EnsureSuccessStatusCode();
}
```

D

Criar um objeto do tipo `MultipartFormDataContent` e *embrulhar* o livro nele, passando esse objeto como argumento do método `PostAsync()`, assim (aqui omiti o método `CreateMultipartFormData()`):

```
public async Task PostLivroAsync(LivroUpload livro)
{
    var content = CreateMultipartFormData(livro);
    var resposta = await httpClient.PostAsync($"livros", content);
    resposta.EnsureSuccessStatusCode();
}
```