

≡ 05

## Adiantando a paginação

Agora que você viu como aplicar filtros e ordenação em sua API, já consegue iniciar o trabalho de paginação.

Reflita comigo: se eu quero suportar requisições com o seguinte formato...

<http://localhost:6000/api/v2.0/livros?tamanho=50&pagina=3> (<http://localhost:6000/api/v2.0/livros?pagina=3&tamanho=50>)

...quais são os primeiros passos que preciso implementar na API?

Escolha as alternativas corretas.

Selezione 2 alternativas

- A** Criar a action responsável por atender a requisição de paginação.

```
[HttpGet]  
public Task<IActionResult> PaginaDeLivros()  
{  
    //código aqui...  
}
```

- B** Receber um objeto do tipo LivroPaginacao como argumento de entrada da action que retorna a coleção de livros, usando o atributo `[FromQuery]`.

```
public IActionResult ListaDeLivros(  
    [FromQuery] LivroFiltro filtro,  
    [FromQuery] LivroOrdem ordem,  
    [FromQuery] LivroPaginacao paginacao)  
{  
    //código aqui...  
}
```

- C** Uma classe que encapsula os valores de tamanho e página.

```
public class LivroPaginacao  
{  
    public int Tamanho { get; set; }  
    public int Pagina { get; set; }  
}
```