1 - Na Pasta Controllers adcione a classe BasketController

2 - Dentro da Classe BasketController

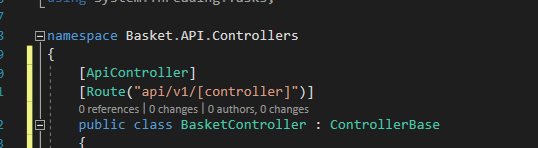
\* Adcione a notação antes do nome da classe "public class BasketController"

[ApiController]

[Route("api/v1/[controller]")]

public class BasketController .....

3 - Faça um extend \* BasketController : ControllerBase.



4 - Começe a implementar os mesmos contratos criados na interface de repositorios, faça a dependencia dela dentro da controller e a implemente.

4.1- Faça uma injeção de dependência para Interface IBasketRepository

\* private readonly IBasketRepository \_repository;

4.2- Faça o construtor com da injeção desta forma:

\* public BasketController (IBasketRepository repository)

{

\_repository = repository ?? throw new ArgumentNullException(nameof(repository));

}

4.3- Feito a injeção de dependência comece a implementar os métodos.

\*[HttpGet("{userName}", Name = "GetBasket")]

[ProducesResponseType(typeof(ShoppingCart),(int)HttpStatusCode.OK)]

public async Task<ActionResult<ShoppingCart>> GetBasket(string userName)

{

var basket = await \_repository.GetBasket(userName);

return Ok(basket ?? new ShoppingCart(userName));

}

\*[HttpPost]

[ProducesResponseType(typeof(ShoppingCart),(int)HttpStatusCode.OK)]

public async Task<ActionResult<ShoppingCart>> UpdateBasket([Frombody] ShopingCart basket)

{

return Ok(await \_repository.UpdateBasket(basket));

}

\*[HttpDelete("{userName}", Name = "DeleteBasket")]

[ProducesResponseType(typeof(void), (int)HttpStatusCode.OK)]

public async Task<IActionResult> DeleteBasket(string userName)

{

await \_repository.DeleteBasket(userName);

return Ok();

}

5 - Feitas as devidas implementações abra a Startup.cs e adcione a Interface IBasketRepository como Service Scope no metodo ConfigureServices..

5.1 - Abaixo do AddStackExchange... digite o codigo para adicionar o serviço que implementamos no Repository e depois no Controller.

\* services.AddScoped<IBasketRepository, BasketRepository> ();