



# Exercício

## Aula 16

Prof: Henrique Augusto Maltauro

**Codificar Back-end  
de Aplicações Web**

## Exercício

Criar uma API para simular uma calculadora.

→ Pasta Enums

◆ TipoOperacao

→ Pasta Domain

◆ Parametros

→ Pasta Handlers

◆ CalculadoraHandler

→ Pasta Controllers

◆ CalculadoraController

## Exercício

- enum TipoOperacao

O enum TipoOperacao deve possuir quatro opções para representar cada uma das quatro operações básicas de soma, subtração, multiplicação e divisão.

# Exercício

- classe Parametros

A classe Parametros tem o objetivo de representar os parâmetros do cálculo a ser realizado.

Sendo assim ela deve possuir dois atributos do tipo double para representar os valores a serem calculados, e um atributo do tipo TipoOperacao, para representar qual é a operação a ser realizada no cálculo.

## Exercício

- classe Parametros

Todos os atributos devem ser públicos e marcados com `JsonPropertyName`.

A classe não precisa de método construtor.

## Exercício

- classe `CalculadoraHandler`

A classe `CalculadoraHandler` deverá realizar os processos da nossa calculadora.

Sendo assim, ela deverá possuir um único `método` para realizar o cálculo.

## Exercício

- classe `CalculadoraHandler`

Esse único método deverá retornar um `ActionResult` e receber um parâmetro do tipo `Parametros`, e com essa informação deverá validar qual é o enum `TipoOperacao` que está sendo recebido, e a partir dele realizar o cálculo necessário.

## Exercício

- classe `CalculadoraHandler`

Ainda nesse `método`, ele deve possuir um bloco de `try catch` para validar exceções dos tipos `ArithmeticException`, `DivideByZeroException`, `OverflowException` e `Exception`.



## Exercício

- classe `CalculadoraHandler`

Ainda nesse `método`, se a execução cair em alguns dos blocos `catch`, ele deve retornar um `ActionResult` de código `500` com uma mensagem de erro.

Se tudo ocorrer bem, ele deve retornar um `Ok` com o resultado do cálculo realizado.

## Exercício

- controller `CalculadoraController`

O controller `CalculadoraController` deverá realizar as rotas da nossa `API`, e sendo assim deverá ter um atributo do tipo `CalculadoraHandler`, o qual deve ser inicializado no método construtor.

## Exercício

- controller **CalculadoraController**

O controller deverá possuir uma única rota “**executar**” do tipo GET, a qual deverá receber via JSON um **parâmetro** do tipo **Parametros**, e usa-lo para executar o **método** de cálculo do handler e retornar um **ActionResult** de forma **assíncrona**.