

Objetivos:

- Desenvolver um servidor utilizando Node.js com o framework Express;
- Criar rotas HTTP;
- Aplicar estruturas de decisão, repetição, conversão de tipos de dados e manipulação de arrays.

Descrição da atividade:

Implemente uma aplicação cliente-servidor para permitir ao usuário calcular o somatório, a média, o fatorial e verificar se um número é primo. A base do código encontra-se disponível no repositório:

<https://github.com/arleysouza/algo-atv3>

- **Nota:** 0,25 pts. na média final
- **Data de entrega:** 19/nov
- **Forma de entrega:** individual e presencial (entregas por meio do Teams não serão aceitas)

Requisitos funcionais:

A aplicação deve disponibilizar as seguintes rotas e comportamentos:

1. Calcular o somatório dos números em um intervalo:

Rota: GET /operacoes/somatorio

Exemplo de Requisição	Resposta Esperada
/operacoes/somatorio	{"erro": "Parâmetro 'inicio' é obrigatório"}
/operacoes/somatorio?inicio=a	{"erro": "Parâmetro 'fim' é obrigatório"}
/operacoes/somatorio?inicio=a&fim=b	{"erro": "Parâmetro 'inicio' precisa ser um número"}
/operacoes/somatorio?inicio=10&fim=b	{"erro": "Parâmetro 'fim' precisa ser um número"}
/operacoes/somatorio?inicio=10&fim=1	{"erro": "O valor de 'inicio' não pode ser maior que 'fim'"}
/operacoes/somatorio?inicio=1&fim=10	{"resultado": 55}

2. Calcular o fatorial de um número

Rota: GET /operacoes/fatorial

Exemplo de Requisição	Resposta Esperada
/operacoes/fatorial	{"erro": "Parâmetro 'numero' é obrigatório"}
/operacoes/fatorial?numero=a	{"erro": "Parâmetro 'numero' precisa ser um número"}
/operacoes/fatorial?numero=-5	{"erro": "O valor de 'numero' não pode ser negativo"}

Exemplo de Requisição	Resposta Esperada
/operacoes/fatorial?numero=5	{"resultado":120}

3. Calcular a média de uma lista de números

Rota: GET /operacoes/media

Exemplo de Requisição	Resposta Esperada
/operacoes/media	{"erro":"Parâmetro 'numeros' é obrigatório"}
/operacoes/media?numeros=a,b,c	{"erro":"Parâmetro 'numeros' deve conter apenas números e vírgulas"}
/operacoes/media?numeros=1;2;3	{"erro":"Parâmetro 'numeros' deve conter apenas números e vírgulas"}
/operacoes/media?numeros=1,x,3	{"erro":"Parâmetro 'numeros' deve conter apenas números e vírgulas"}
/operacoes/media?numeros=4.32	{"resultado":4.32}
/operacoes/media?numeros=1,2.9,3	{"resultado":2.3000000000000003}

4. Verificar se um número é primo

Rota: GET /operacoes/primo

Exemplo de Requisição	Resposta Esperada
/operacoes/primo	{"erro":"Parâmetro 'numero' é obrigatório"}
/operacoes/primo?numero=a	{"erro":"Informe um número inteiro"}
/operacoes/primo?numero=1	{"erro":"Informe um número inteiro igual ou maior que 2"}
/operacoes/primo?numero=9	{"resultado":false}

5. Obter o Máximo Divisor Comum (MDC)

Rota: GET /operacoes/mdc

Exemplo de Requisição	Resposta Esperada
/operacoes/mdc	{"erro":"Parâmetro 'a' é obrigatório"}
/operacoes/mdc?a=x	{"erro":"Parâmetro 'b' é obrigatório"}
/operacoes/mdc?a=10&b=y	{"erro":"Parâmetro 'b' precisa ser um número"}
/operacoes/primo?mdc?a=10&b=15	{"resultado":5}

Requisitos adicionais

A aplicação deverá fornecer um front-end simples para que o usuário possa:

- Informar os valores de entrada;
- Enviar as requisições às rotas da API;
- Visualizar os resultados retornados.

← → ↻ localhost:3003

Testar Operações

Somatório

Início

Fim

Resultado: 45

Cálculo da média

Números (separados por vírgula)

Resultado: 5.066666666666666

Cálculo do fatorial

Número

Resultado: 120

Verificar número primo

Número

Resultado: true