

Objetivo: crie uma javascript no html para fazer conta de todos os tipos

Adding Task: Crie um arquivo HTML com uma tag <script> para inserir o código JavaScript

Adding Task: Implemente as funções de soma, subtração, multiplicação e divisão em JavaScript

Adding Task: Crie elementos HTML para receber os valores de entrada e exibir o resultado das operações

Adding Task: Faça a chamada das funções de acordo com a ação desejada pelos usuários

Starting task: Crie um arquivo HTML com uma tag <script> para inserir o código JavaScript

Writing code...

```
Executando: "Crie um arquivo HTML com uma tag <script> para inserir o código JavaScript": <!DOCTYPE html>
<html>
<head>
<title>JavaScript Calculator</title>
</head>
<body>
<script src="script.js"></script>
</body>
</html>
```

Adding Task: Crie um arquivo HTML com uma tag <script> para inserir o código JavaScript

Adding Task: Crie funções JavaScript para realizar as operações de potenciação e radiciação.

Starting task: Implemente as funções de soma, subtração, multiplicação e divisão em JavaScript

Writing code...

```
Executando: "Implemente as funções de soma, subtração, multiplicação e divisão em JavaScript": # JavaScript
Arithmetic Functions
```

```
## Introduction
```

In this code, we will implement JavaScript functions for addition, subtraction, multiplication, and division.

```
## Addition
```

To implement the addition function, we can use the `+` operator in JavaScript. The function will take two numbers as parameters and return their sum.

```
```javascript
function addition(a, b) {
return a + b;
```

```
}
...
```

### ## Subtraction

For subtraction, we can use the `-` operator in JavaScript. The function will take two numbers as parameters and return their difference.

```
```javascript  
function subtraction(a, b) {  
  return a - b;  
}  
...
```

Multiplication

To implement multiplication, we can use the `*` operator in JavaScript. The function will take two numbers as parameters and return their product.

```
```javascript  
function multiplication(a, b) {
 return a * b;
}
...
```

### ## Division

For division, we can use the `/` operator in JavaScript. The function will take two numbers as parameters and return their quotient.

```
```javascript  
function division(a, b) {  
  return a / b;  
}  
...
```

Testing

We can test the above functions by calling them with different inputs and checking the returned results. For example:

```
```javascript  
console.log(addition(2, 3)); // Output: 5
console.log(subtraction(5, 2)); // Output: 3
console.log(multiplication(4, 2)); // Output: 8
console.log(division(10, 2)); // Output: 5
...
```

These calls should output the expected results.

---

Adding Task: Implemente as funções de soma, subtração, multiplicação e divisão em JavaScript

---

Adding Task: Crie uma função JavaScript para realizar a operação de potenciação.

---

Starting task: Crie elementos HTML para receber os valores de entrada e exibir o resultado das operações

---

Error attempting to connect to the server.

---

Starting task: Crie elementos HTML para receber os valores de entrada e exibir o resultado das operações

---

Error attempting to connect to the server.

---

Starting task: Crie elementos HTML para receber os valores de entrada e exibir o resultado das operações

---

Writing code...

---

Executando: "Crie elementos HTML para receber os valores de entrada e exibir o resultado das operações": ## HTML Elements

Para começar, vamos criar os elementos HTML necessários para receber os valores de entrada e exibir o resultado das operações.

```
```html
<!DOCTYPE html>
<html>
<head>
  <title>Calculadora</title>
</head>
<body>
<h1>Calculadora</h1>

<label for="num1">Número 1:</label>
<input type="number" id="num1">

  <label for="num2">Número 2:</label>
  <input type="number" id="num2">

<button onclick="soma()">Somar</button>
  <button onclick="subtracao()">Subtrair</button>
<button onclick="multiplicacao()">Multiplicar</button>
<button onclick="divisao()">Dividir</button>

<h2>Resultado:</h2>
<div id="resultado"></div>

<script src="script.js"></script>
</body>
</html>
```
```

Nesse código, criamos uma página HTML com dois campos de entrada para os números, quatro botões para as operações matemáticas (soma, subtração, multiplicação e divisão) e uma área para exibir o resultado. Os botões

chamam funções JavaScript quando são clicados.

Agora vamos para a próxima seção, onde escreveremos o código JavaScript para realizar as operações matemáticas.

---

Adding Task: Crie elementos HTML para receber os valores de entrada e exibir o resultado das operações

---

Adding Task: Crie funções JavaScript para realizar as operações de multiplicação e divisão.

---

Starting task: Faça a chamada das funções de acordo com a ação desejada pelos usuários

---

Error attempting to connect to the server.

---

Starting task: Faça a chamada das funções de acordo com a ação desejada pelos usuários

---

Writing code...

---

Executando: "Faça a chamada das funções de acordo com a ação desejada pelos usuários": ``javascript

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Calculadora</title>
```

```
<script>
```

```
 // Função para realizar a soma de dois números
```

```
function soma(a, b) {
```

```
 return a + b;
```

```
}
```

```
 // Função para realizar a subtração de dois números
```

```
function subtracao(a, b) {
```

```
return a - b;
```

```
}
```

```
 // Função para realizar a multiplicação de dois números
```

```
function multiplicacao(a, b) {
```

```
return a * b;
```

```
}
```

```
 // Função para realizar a divisão de dois números
```

```
function divisao(a, b) {
```

```
 return a / b;
```

```
}
```

```
 // Função para realizar o cálculo da média de um array de números
```

```
function calcularMedia(numeros) {
```

```
var soma = 0;
```

```
for (var i = 0; i < numeros.length; i++) {
```

```
 soma += numeros[i];
```

```
}
return soma / numeros.length;
}

// Função para calcular o fatorial de um número
function calcularFatorial(numero) {
 if (numero === 0 || numero === 1) {
 return 1;
 } else {
 return numero * calcularFatorial(numero - 1);
 }
}

// Função para calcular o quadrado de um número
function calcularQuadrado(numero) {
 return numero ** 2;
}

// Função para calcular a raiz quadrada de um número
function calcularRaizQuadrada(numero) {
return Math.sqrt(numero);
}

// Função para exibir o resultado no HTML
function exibirResultado(resultado) {
 var resultadoElement = document.getElementById("resultado");
 resultadoElement.innerHTML = "Resultado: " + resultado;
}

// Função para realizar a chamada das funções de acordo com a
```

---

Adding Task: Faça a chamada das funções de acordo com a ação desejada pelos usuários

---

Starting task: Crie funções JavaScript para realizar as operações de potenciação e radiciação.

---

Writing code...

---

Executando: "Crie funções JavaScript para realizar as operações de potenciação e radiciação.": Loading...

---

An unknown error occurred. Please try again later.

---

```
Executando: "Crie funções JavaScript para realizar as operações de potenciação e radiciação.": ``javascript
// Function to calculate power
function calculatePower(base, exponent) {
// Calculate the power using the Math.pow() method
return Math.pow(base, exponent);
}
```

```
// Function to calculate square root
function calculateSquareRoot(number) {
 // Calculate the square root using the Math.sqrt() method
 return Math.sqrt(number);
}
...
```

Explicação:

- A função `calculatePower` recebe uma base e um expoente como argumentos e retorna o valor da potência. Ela utiliza o método `Math.pow()` para calcular a potência.
- A função `calculateSquareRoot` recebe um número como argumento e retorna a raiz quadrada desse número. Ela utiliza o método `Math.sqrt()` para calcular a raiz quadrada.

---

Adding Task: Crie funções JavaScript para realizar as operações de potenciação e radiciação.

---

Starting task: Crie uma função JavaScript para realizar a operação de potenciação.

---

Error attempting to connect to the server.

---

Starting task: Crie uma função JavaScript para realizar a operação de potenciação.

---

Error attempting to connect to the server.

---

Starting task: Crie uma função JavaScript para realizar a operação de potenciação.

---

Writing code...

---

Executando: "Crie uma função JavaScript para realizar a operação de potenciação.": ``javascript

```
// Function to calculate the power of a number
function power(base, exponent) {
 // If exponent is 0, return 1
 if (exponent === 0) {
 return 1;
 }

 // If exponent is positive, multiply base by itself recursively
 if (exponent > 0) {
 return base * power(base, exponent - 1);
 }

 // If exponent is negative, calculate the power of (1/base) recursively
 if (exponent < 0) {
 return 1 / (base * power(base, -exponent - 1));
 }
}
...
```

Português:

```
```javascript
// Função para calcular a potência de um número
function potencia(base, expoente) {
  // Se o expoente for 0, retorne 1
  if (expoente === 0) {
    return 1;
  }

  // Se o expoente for positivo, multiplique a base por ela mesma recursivamente
  if (expoente > 0) {
    return base * potencia(base, expoente - 1);
  }

  // Se o expoente for negativo, calcule a potência de (1/base) recursivamente
  if (expoente < 0) {
    return 1 / (base * potencia(base, -expoente - 1));
  }
}
```
```

This code defines a JavaScript function to calculate the power of a number. The function takes two parameters: `base` and `exponent`.

If the exponent is 0, the function returns 1 since any number raised to the power of 0 is 1.

If the exponent is positive, the function recursively multiplies the base by itself `exponent` number of times.

If the exponent is negative, the function calculates the power of (1/base) recursively by negating

---

Adding Task: Crie uma função JavaScript para realizar a operação de potenciação.

---

Adding Task: Crie funções JavaScript para realizar as operações de adição e subtração.

---

Starting task: Crie funções JavaScript para realizar as operações de multiplicação e divisão.

---

Writing code...

---

```
Executando: "Crie funções JavaScript para realizar as operações de multiplicação e divisão.": ```javascript
// Crie uma função JavaScript para realizar a operação de multiplicação
function multiplicacao(a, b) {
 return a * b;
}

// Crie uma função JavaScript para realizar a operação de divisão
function divisao(a, b) {
 return a / b;
}
```

...

Explicação:

- A função `multiplicacao` recebe dois parâmetros `a` e `b` e retorna o resultado da multiplicação entre eles.
- A função `divisao` também recebe dois parâmetros `a` e `b` e retorna o resultado da divisão entre eles.

Essas funções podem ser utilizadas para realizar operações de multiplicação e divisão em JavaScript.

---

Adding Task: Crie funções JavaScript para realizar as operações de multiplicação e divisão.

---

Starting task: Crie funções JavaScript para realizar as operações de adição e subtração.

---

Writing code...

---

Executando: "Crie funções JavaScript para realizar as operações de adição e subtração.": ````javascript

```
// Function to perform addition
```

```
function addition(num1, num2) {
```

```
 return num1 + num2;
```

```
}
```

```
// Function to perform subtraction
```

```
function subtraction(num1, num2) {
```

```
 return num1 - num2;
```

```
}
```

...

Essas funções JavaScript são responsáveis por realizar as operações de adição e subtração, respectivamente. A função `addition` recebe dois números como parâmetros e retorna a soma deles. A função `subtraction` também recebe dois números como parâmetros e retorna a diferença entre eles.

---

Adding Task: Crie funções JavaScript para realizar as operações de adição e subtração.

---

Todas as tarefas foram concluídas. Desligando...

---