

Mentes lógicas:

Primeiros passos na programação descomplicada



Thaís Pisani

1

INTRODUÇÃO À LÓGICA DE PROGRAMAÇÃO

INTRODUÇÃO À LÓGICA DE PROGRAMAÇÃO

A lógica de programação é a base fundamental para qualquer linguagem de programação. Compreender os conceitos básicos ajuda a resolver problemas e criar soluções eficazes. Neste eBook, vamos explorar a lógica de programação usando JavaScript, uma linguagem amplamente utilizada para desenvolvimento web.

O que é lógica da programação?

Lógica de programação é a organização sequencial de instruções para resolver um problema ou executar uma tarefa. É como criar uma receita para um bolo: cada passo deve ser seguido em ordem para obter o resultado desejado.

JS

2

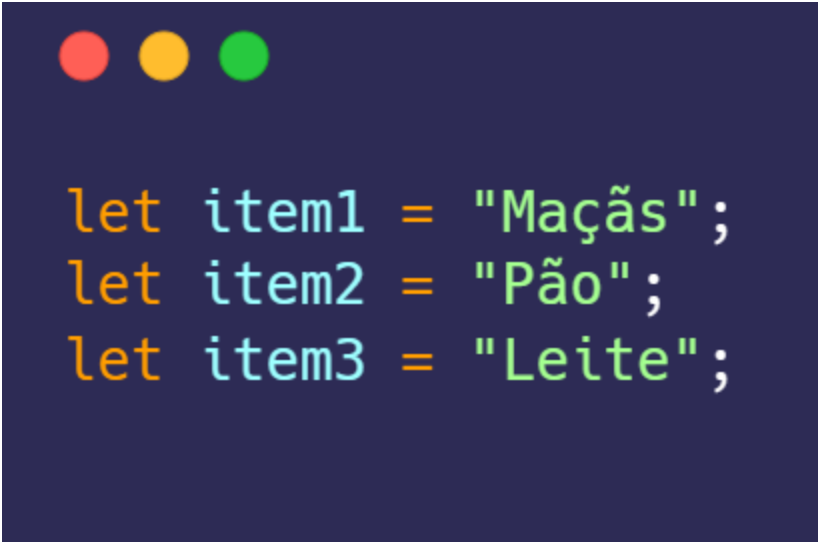
Variáveis: Guardando Informações

VARIÁVEIS:

Guardando Informações

Variáveis são "caixas" onde armazenamos informações que podem ser usadas e manipuladas durante a execução de um programa.

Imagine que você está criando um aplicativo de lista de compras. Você pode usar variáveis para armazenar os itens da lista:



```
let item1 = "Maçãs";  
let item2 = "Pão";  
let item3 = "Leite";
```

3


**Condicionais:
Tomando Decisões**

CONDICIONAIS:

Tomando Decisões

Condicionais permitem que o programa tome decisões com base em certas condições. Usamos "if", "else" e "else if" para criar essas estruturas.

Suponha que você esteja desenvolvendo um sistema de login. Você pode usar condicionais para verificar se o usuário inseriu a senha correta:



```
let senhaCorreta = "12345";
let senhaUsuario = prompt("Digite sua
senha:");

if (senhaUsuario === senhaCorreta) {
  console.log("Acesso permitido");
} else {
  console.log("Acesso negado");
}
```

4

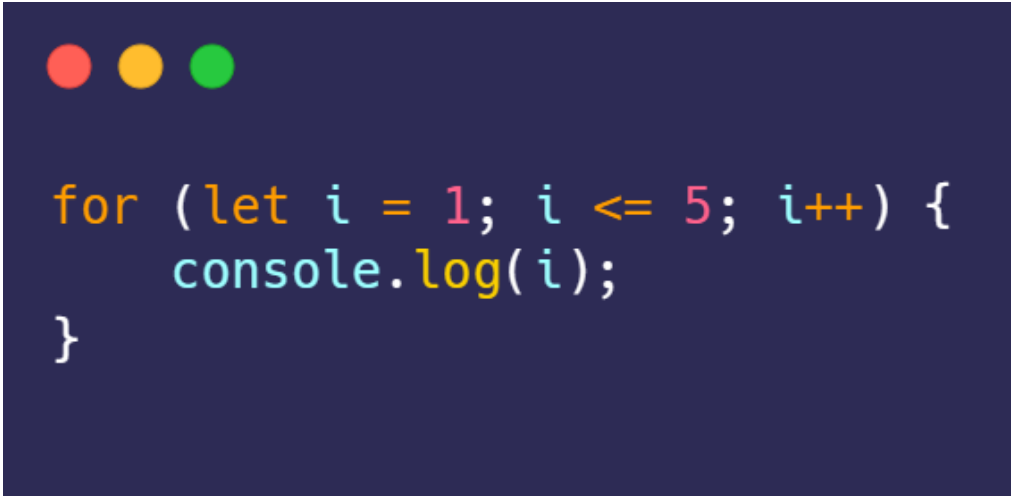
Loops:
Repetindo Tarefas

LOOPS:

Repetindo tarefas

Loops permitem repetir uma série de instruções várias vezes, economizando tempo e esforço na escrita de código repetitivo.

Vamos criar um loop que imprime os números de 1 a 5:



```
for (let i = 1; i <= 5; i++) {  
  console.log(i);  
}
```

5

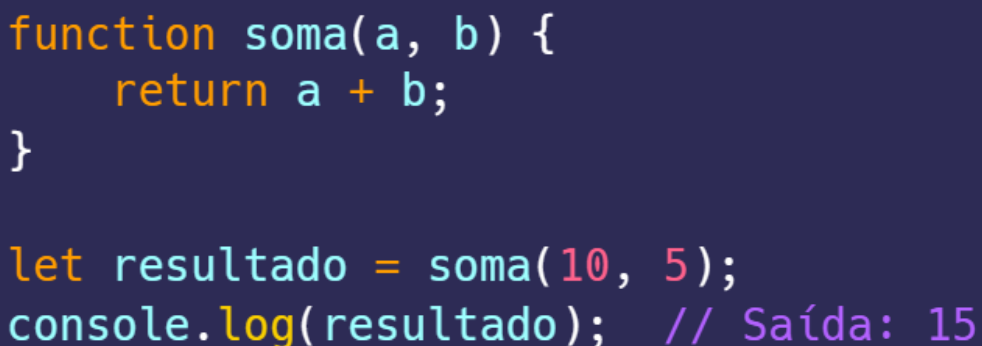
Funções:
Organizando o Código

FUNÇÕES

Organizando o Código

Funções são blocos de código que realizam tarefas específicas e podem ser reutilizados em diferentes partes do programa.

Imagine que você precisa calcular a soma de dois números várias vezes. Em vez de escrever o mesmo código repetidamente, você pode criar uma função:



```
function soma(a, b) {  
    return a + b;  
}  
  
let resultado = soma(10, 5);  
console.log(resultado); // Saída: 15
```

6

Arrays:

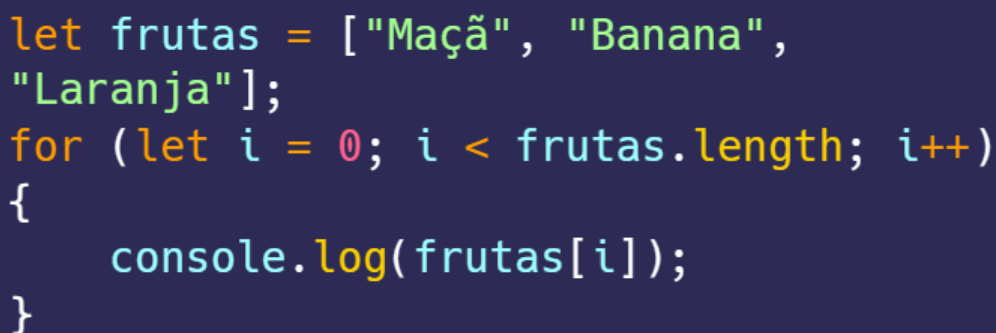
**Armazenando Coleções de
Dados**

ARRAYS

Armazenando Coleções de Dados

Listas são estruturas que permitem armazenar múltiplos valores em uma única variável.

Vamos criar uma lista de frutas e imprimir cada uma delas:



```
let frutas = ["Maçã", "Banana",  
"Laranja"];  
for (let i = 0; i < frutas.length; i++)  
{  
    console.log(frutas[i]);  
}
```

7

**Aplicação Prática:
Calculadora Simples**

APLICAÇÃO PRÁTICA:

Calculadora Simples

Vamos usar o que aprendemos para criar uma calculadora simples:

```
function soma(a, b) {
  return a + b;
}

function subtrai(a, b) {
  return a - b;
}

function multiplica(a, b) {
  return a * b;
}

function divide(a, b) {
  if (b !== 0) {
    return a / b;
  } else {
    return "Divisão por zero não é permitida";
  }
}

while (true) {
  let operacao = prompt("Escolha a operação: soma, subtrai, multiplica, divide ou sair:");

  if (operacao === "sair") {
    break;
  }

  let num1 = parseFloat(prompt("Digite o primeiro número:"));
  let num2 = parseFloat(prompt("Digite o segundo número:"));

  switch (operacao) {
    case "soma":
      console.log("Resultado:", soma(num1, num2));
      break;
    case "subtrai":
      console.log("Resultado:", subtrai(num1, num2));
      break;
    case "multiplica":
      console.log("Resultado:", multiplica(num1, num2));
      break;
    case "divide":
      console.log("Resultado:", divide(num1, num2));
      break;
    default:
      console.log("Operação inválida");
  }
}
```

APLICAÇÃO PRÁTICA:

Calculadora Simples

Essa calculadora permite que você some, subtraia, multiplique e divida números, mostrando como funções e condicionais podem ser usadas juntas para criar um aplicativo útil.

Conclusão

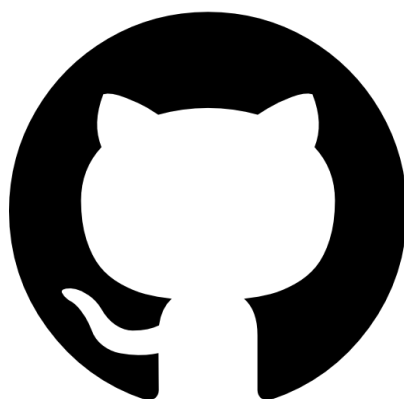
A lógica de programação é a chave para resolver problemas de maneira eficiente. Compreendendo e praticando esses conceitos básicos usando JavaScript, você estará bem encaminhado para se tornar um programador competente. Continue praticando e explorando novas possibilidades!

OBRIGADA POR LER ATÉ AQUI!

Calculadora Simples

Esse Ebook foi gerado por IA, e diagramado por mim.

Esse conteúdo foi gerado com fins didáticos de construção, não foi realizado uma validação detalhada do conteúdo e pode conter erros gerados por uma IA.



<https://github.com/thaispisanis>