

Documentação do Projeto: Calculadora

Este documento detalha o desenvolvimento de uma calculadora web interativa, estilizada com um tema "AI Core", utilizando **HTML**, **CSS** e **JavaScript**. O projeto visa demonstrar a integração dessas três tecnologias para criar uma interface de usuário funcional e visualmente atraente.

1. Visão Geral do Projeto

A Calculadora é uma aplicação web de front-end que simula as operações básicas de uma calculadora tradicional (adição, subtração, multiplicação e divisão). Sua principal característica é o design moderno e futurista, complementado por animações sutis que melhoram a experiência do usuário. O display exibe a operação em andamento, proporcionando clareza durante o cálculo.

2. Estrutura de Arquivos

O projeto é organizado em três arquivos principais, cada um com uma responsabilidade específica:

- **index.html**: Define a estrutura e o conteúdo da página web, incluindo os elementos da calculadora como o display e os botões.
- **style.css**: Responsável por toda a estilização visual da calculadora, implementando o tema "All Core" com cores escuras, brilhos neon e animações.
- **script.js**: Contém a lógica JavaScript que controla o comportamento da calculadora, como o manuseio de cliques nos botões, atualização do display, realização de cálculos e tratamento de erros.

3. Detalhes da Implementação

- **Meta Tags**: Essenciais para garantir a codificação de caracteres e a responsividade em diferentes dispositivos.
- **Fontes**: A fonte Montserrat é importada via Google Fonts para um visual moderno.
- **Estrutura da Calculadora**: Um contêiner principal (`.calculator`) abriga o display (`.display`) e os botões (`.buttons`).

- **Botões:** Cada botão é um elemento `<button>` com classes específicas (`.number`, `.operator`, `.clear`, `.equals`) para estilização e `ids` para fácil manipulação via JavaScript.
- **Link JS:** O script JavaScript é importado no final do `<body>` para garantir que todos os elementos HTML estejam carregados antes que o JavaScript tente interagir com eles.

3.2. CSS (*style.css*)

- **Paleta de Cores:**
 - Dominada por tons de **azul escuro, cinzas azulados e azul elétrico (#00bfff)**, que é a cor de destaque principal.
 - Vermelho escuro (#8b0000) para o botão "C" e verde escuro (#006400) para o "=" dão contraste, com brilhos em tons de laranja/vermelho e verde limão.
 - Isso cria uma sensação de profundidade e tecnologia, como uma interface de console de IA.
- **Fonte (font-family: 'Roboto Mono', monospace;):**
 - **Roboto Mono** é uma fonte monoespaçada, frequentemente associada a código, terminais e interfaces de dados, reforçando o tema de IA.
 - **Importante:** Assim como antes, você precisa adicionar esta linha no seu `index.html` dentro da tag `<head>` para importar a fonte
- **Calculadora (.calculator):**
 - **Fundo:** Um azul escuro suave (#1a2a3a) para o corpo da calculadora.
 - **Borda:** Uma **borda sólida e robusta** (2px solid #00bfff) no tom de azul elétrico, que é uma característica visual forte de interfaces de IA.
 - **Sombras:** Múltiplas box-shadow com brilho azul elétrico, e um inset shadow para dar a impressão de que a calculadora é uma peça de hardware.
 - **Tamanho e Padding:** Ligeiramente maior e com mais preenchimento para uma sensação mais "sólida".
 - **Display (.display):**
 - **Fundo:** Preto ainda mais profundo (#0a111a).
 - **Cor do Texto:** **Azul elétrico vibrante (#00bfff)**, o ponto focal do tema.
 - **Sombras e Brilho:** inset box-shadow e text-shadow são mais intensos para dar uma sensação de luz forte vindo da tela, como um monitor de dados.
 - **Borda:** Uma borda fina para emoldurar o display.
 - **Botões (.button):**
 - **Fundo:** Azul acinzentado escuro (#2c4155).

- **Borda Individual:** Cada botão agora tem sua própria border: `1px solid #00bfff`; para um visual mais segmentado e "componentizado", como módulos de um sistema.
- **Transições:** Mantidas suaves para uma boa experiência do usuário.
- **Hover/Active:**
 - **:hover:** O botão "levanta" mais (`translateY(-4px)`), fica mais claro e a borda se ilumina.
 - **:active:** O botão tem um `scale(0.98)` sutil (quase imperceptível, mas dá feedback) e o texto do botão muda para a cor de destaque (azul elétrico, laranja, verde) para um feedback de clique mais claro.
- **Cores Específicas dos Botões (`.clear`, `.operator`, `.equals`, `.number`):**
- Mantêm suas cores distintas, mas agora têm **bordas e brilhos que seguem suas cores de destaque** (vermelho/laranja para "C", laranja para operadores, verde para "=", e azul elétrico para números). O texto também muda de cor para a cor de destaque no `:active` para um feedback instantâneo.

JavaScript

O JavaScript gerencia a lógica da calculadora, a interação do usuário e a atualização dinâmica do display.

- **Gerenciamento de Estado:** Variáveis como `currentInput`, `operator`, `previousInput`, `awaitingNextInput` e `displayExpression` controlam o estado da calculadora e a lógica de exibição.
- **updateDisplay():** Função central para atualizar o texto do display, que agora exibe a `displayExpression` (operação completa) ou o `currentInput` (resultado/número atual).
- **Manipulação de Eventos:** Cada botão tem um `event listener` de clique que detecta o tipo de botão (número, operador, limpar, igual) e executa a lógica correspondente.
- **Lógica de Dígitos e Ponto Decimal:** Trata a entrada de números, impedindo múltiplos pontos decimais e substituindo o '0' inicial.
- **Lógica de Operadores:** Gerencia o encadeamento de operações (ex: `5 + 3 + 2`) e atualiza o `previousInput` e a `displayExpression`.
- **Lógica do Sinal de Igual (=):** Aciona a função `calculate()`, exibe o resultado no display e redefine o estado para um novo cálculo.

- **Função `calculate()`:** Executa a operação matemática com base no operador e lida com casos especiais, como divisão por zero. O resultado é armazenado em `currentInput`.
- **`DOMContentLoaded`:** Garante que o script só seja executado após o carregamento completo do DOM, evitando erros de elementos não encontrados.

4. Como Usar

1. Salve os três arquivos (`index.html`, `style.css`, `script.js`) em uma mesma pasta.
2. Abra o arquivo `index.html` em seu navegador web.
3. A calculadora "AL CORE" aparecerá com uma animação de entrada.
4. Interaja com os botões para realizar operações matemáticas.
5. Observe o display que mostra a operação completa enquanto você digita e o resultado final.

5. Contribuições e Melhorias Futuras

Este projeto serve como uma base robusta para uma calculadora web. Possíveis melhorias futuras incluem:

- **Suporte a Teclado:** Permitir que o usuário digite números e operações usando o teclado físico.
- **Histórico de Cálculos:** Adicionar uma funcionalidade para exibir cálculos anteriores.
- **Operações Avançadas:** Implementar funções como porcentagem, raiz quadrada, potência, etc.
- **Temas Personalizáveis:** Permitir que o usuário alterne entre diferentes temas visuais.
- **Responsividade Aprimorada:** Otimizar o layout para telas de dispositivos móveis.
- **Tratamento de Erros Melhorado:** Mensagens de erro mais detalhadas para entradas inválidas.

