

**Documentação do código de  
contador de palavras e  
letras**

**Ândrya Alicya de Lima da silva  
Ayla Christine Barbosa  
Dandara Lima Duarte**

([andryaalicya0801@gmail.com](mailto:andryaalicya0801@gmail.com))  
([kallebstuart@gmail.com](mailto:kallebstuart@gmail.com))  
([christineayla371@gmail.com](mailto:christineayla371@gmail.com))

**RELATÓRIO TÉCNICO DO ICMC**

**3D Colégios  
Novembro/2024**

Os códigos utilizados foram feitos em Html, css e php. O html que estrutura a página, o css é para dar estilo, e o php é o que programa tudo.

Agora vamos analisar o código e definir o que cada parte faz:

1. **\$texto = trim(\$texto);**
  - **O que faz:** A função `trim()` remove espaços em branco (ou outros caracteres de controle) do início e do final de uma string. Ou seja, ela limpa a string `$texto` removendo qualquer espaço extra antes e depois da palavra ou frase.
  - **Exemplo:** Se `$texto = " olá "`, após `trim()`, `$texto` será `"olá"`.
2. **str\_word\_count(\$texto);**
  - **O que faz:** A função `str_word_count()` conta o número de palavras em uma string. Ela retorna o número de palavras que podem ser identificadas pela função (considera palavras separadas por espaços ou outros delimitadores padrão).
  - **Exemplo:** Se `$texto = "Olá mundo"`, o retorno será `2`, pois há duas palavras: "Olá" e "mundo".
3. **str\_replace(' ', ',', \$texto);**

- **O que faz:** A função `str_replace()` substitui todas as ocorrências de um caractere ou substring por outro.
- No caso do código, parece que há um erro de sintaxe. O primeiro parâmetro (‘ ’) está usando aspas curvas ao invés de aspas simples ou duplas padrão. Supondo que a intenção seja substituir os espaços por vírgulas, o código correto seria `str_replace(' ', ',', $texto);`. Isso substituiria todos os espaços no texto por vírgulas.
- **Exemplo:** Se `$texto = "Olá mundo"`, após `str_replace(' ', ',', $texto);`, o resultado seria "Olá,mundo".

#### 4. `strlen($texto);`

- **O que faz:** A função `strlen()` retorna o número de caracteres de uma string, incluindo espaços e pontuação.
- **Exemplo:** Se `$texto = "Olá mundo"`, o retorno de `strlen($texto)` será **10**, porque "Olá mundo" tem 10 caracteres no total (contando o espaço).

#### 5. `return`

- **O que faz:** A palavra-chave `return` é usada para devolver um valor ou sair de uma função em PHP. Quando você usa `return`, o valor especificado será retornado para o código que chamou a função. Se não houver valor, a função simplesmente terminará sua execução.
- **Exemplo:** `return $texto;` retornaria o conteúdo de `$texto` para quem chamou a função.

#### 6. `if ($_SERVER['REQUEST_METHOD'] == 'POST')`

- **O que faz:** Este condicional verifica se o método de requisição HTTP utilizado na página é o **POST**. Ele é comumente utilizado para saber se o formulário foi enviado por meio de uma requisição POST (normalmente usada em formulários).
- **Exemplo:** Se o formulário foi enviado com o método **POST**, o código dentro do `if` será executado.

#### 7. `$_POST['texto'];`

- **O que faz:** `$_POST` é um array superglobal em PHP que contém os dados enviados via requisição HTTP do tipo POST. `$_POST['texto']` acessa o valor do campo `texto` de um formulário enviado por POST.

**Exemplo:** Se o formulário enviado contém um campo chamado `texto` com o valor "Olá mundo", `$_POST['texto']` será igual a "Olá mundo".

Esse é o HTML, e as principais partes e suas funções são:

#### 1. \*\*Estrutura do Documento:\*\*

- `<!DOCTYPE html>`: Declara o tipo de documento como HTML5.
- `<html lang="pt">`: Inicia o documento HTML com o idioma definido como português.

## 2. \*\*Cabeçalho (`<head>`):\*\*

- `<meta charset="UTF-8">`: Define a codificação de caracteres para UTF-8, permitindo a exibição correta de caracteres especiais.
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`: Torna a página responsiva em dispositivos móveis.
- `<title>Contador de Palavras e Letras</title>`: Define o título da página no navegador.
- `<link rel="stylesheet" href="estilo.css">`: Vincula um arquivo CSS externo para estilizar a página.

## 3. \*\*Corpo (`<body>`):\*\*

- \*\*Formulário (`<form>`):\*\* Contém um campo de texto (`<textarea>`) onde o usuário pode inserir ou colar um texto. O formulário envia o conteúdo via `POST` para o servidor (a página recarrega ao ser submetido).
  - \*\*Campo de texto (`<textarea>`):\*\* Usado para o usuário inserir texto. O PHP exibe o texto inserido de volta no campo, caso haja um valor.
  - \*\*Botão (`<button>`):\*\* Submete o formulário para calcular as palavras e letras.

## 4. \*\*Resultados (`<div class="resultados">`):\*\*

- Exibe os resultados do contador de palavras e letras. As contagens são mostradas dentro de elementos `<span>` com IDs `palavras` e `letras`. Os valores são preenchidos com as variáveis PHP `\$palavras` e `\$letras`, que devem ser calculadas no backend.

Este é o CSS, e aqui estão os principais comandos:

## 1. \*\*Estilos gerais:\*\*

- `font-family: Arial, sans-serif;` - Define a fonte do corpo como Arial, com uma fallback para sans-serif.
- `background-color: #a6a7f8;` - Define a cor de fundo do corpo da página.
- `display: flex; justify-content: center; align-items: center; height: 100vh;` - Centraliza o conteúdo vertical e horizontalmente na tela.

2. \*\*Container (caixa central):\*\*

- `background: #7375eb;` - Define a cor de fundo do contêiner.
- `padding: 20px;` - Adiciona espaçamento interno.
- `border-radius: 8px;` - Bordas arredondadas no contêiner.
- `box-shadow: 0 4px 8px rgba(0, 0, 0, 0.5);` - Sombra suave ao redor do contêiner.

3. \*\*Título (`h1`):\*\*

- `font-size: 30px;` - Define o tamanho da fonte do título.
- `color: #000000;` - Define a cor do título como preta.

4. \*\*Área de texto (`textarea`):\*\*

- `width: 95%;` - Largura de 95% do contêiner.
- `height: 150px;` - Define uma altura fixa.
- `border: 5px solid #d0e8f7;` - Borda com uma cor suave e espessura de 5px.
- `resize: none;` - Impede que a área de texto seja redimensionada pelo usuário.

5. \*\*Botão (`button`):\*\*

- `background-color: #0f0c99;` - Cor de fundo azul escuro.
- `color: #fff;` - Cor do texto branca.
- `border-radius: 4px;` - Bordas arredondadas.
- `cursor: pointer;` - Indica que é clicável.

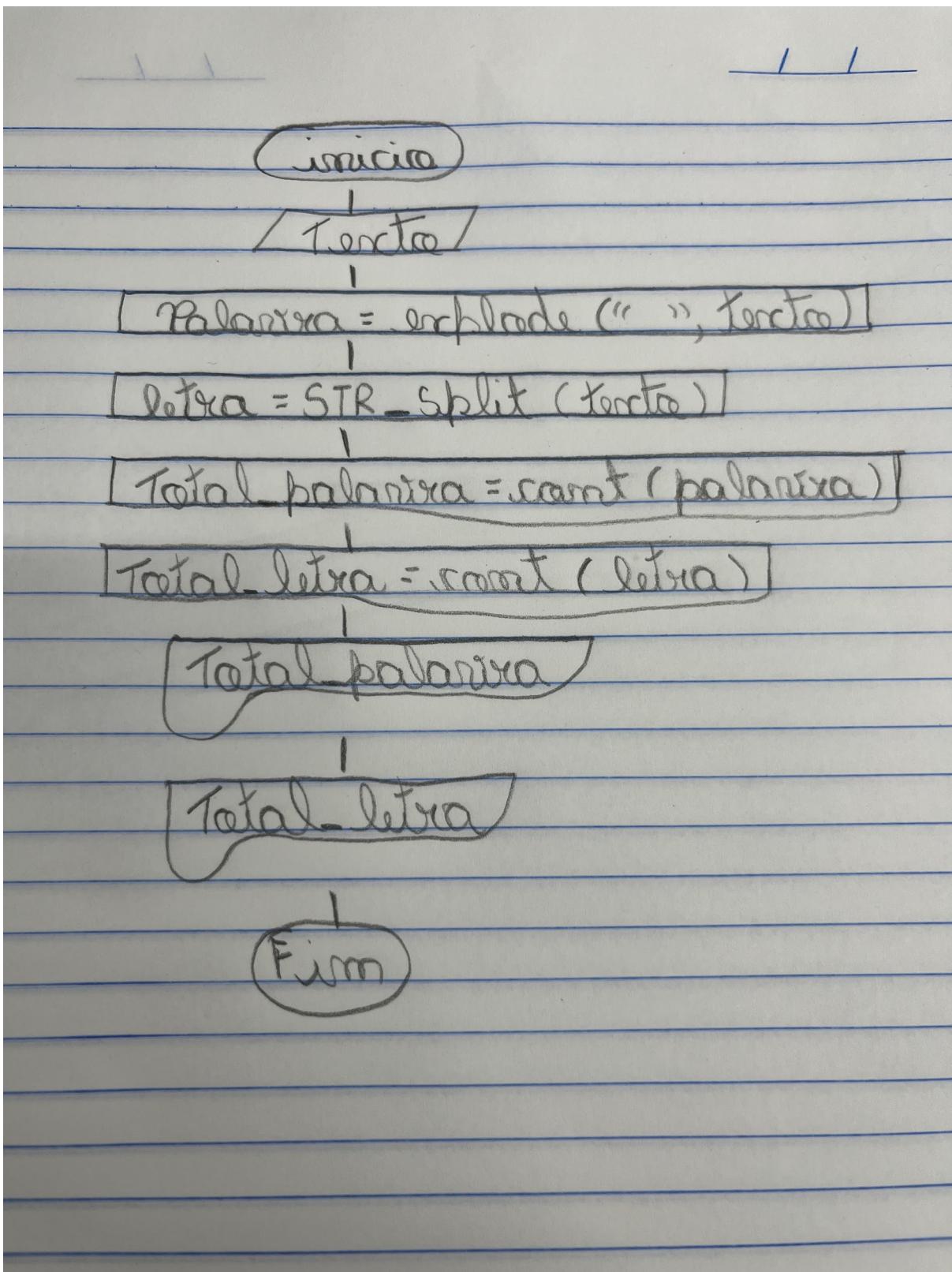
6. \*\*Efeito hover no botão:\*\*

- `background-color: #0e0c70;` - Altera a cor do botão para um tom mais escuro quando o usuário passa o mouse sobre ele.

7. \*\*Resultados (`.resultados`):\*\*

- `margin-top: 20px;` - Adiciona espaço acima dos resultados.
- `font-size: 18px;` - Tamanho de fonte maior para os resultados.
- `.resultados span { font-weight: bold; color: #0a0872; }` - Torna os textos dentro da tag `span` em negrito e de cor azul escura.

## Fluxograma



inicio

caracter: Texto, palavra, letra

escrever: ("texto");  
le (texto);

Palavra - explode (" ", texto);

letra = str.split (texto);

resultado ← total.palavra

escrever: ("O resultado é": resultado);

resultado ← total.letra

escrever: ("O resultado é": resultado);

fim.

Como vai ficar a tela inicial

