

Laboratório 01

Tema: Gerador de histórias aleatórias

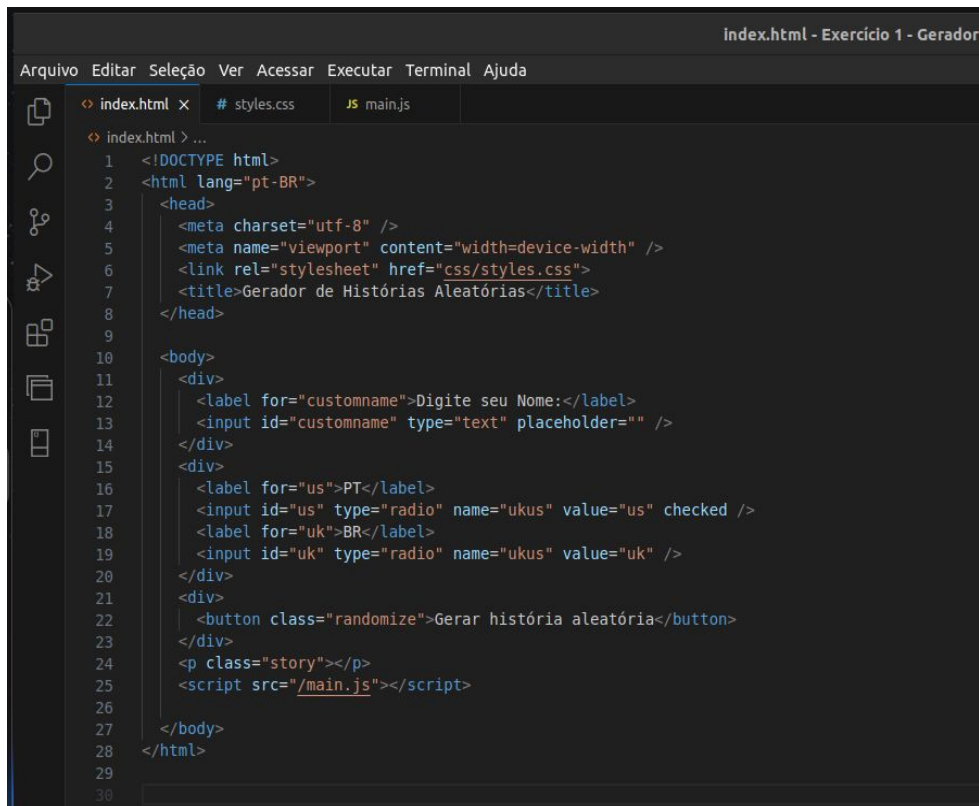
1. Seguir as instruções existentes em:
 - a. https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First_steps/Silly_story_generator
2. No arquivo do texto bruto, substitua o texto por frases de um tema que você gosta (carros, receitas culinárias, moda, séries de streaming, aventuras do Ronaldinho..).
3. Crie a aplicação seguindo o passo a passo sugerido
4. Submeta o link de um arquivo “Google Presentation” com
 - a. prints da demonstração do seu aplicativo e ao final,
 - b. link do seu repositório do GitHub onde os fontes deve estar armazenados.

Descrição

- Para o projeto “Gerador de Histórias aleatórias” foi utilizado o Virtual Studio Code como ambiente de desenvolvimento integrado.
- No VS Code desenvolvi código em HTML, CSS e em JavaScript.
- Baseado com o link e o código oferecido pelo professor e utilizando a criatividade acrescentei mais códigos e adaptei para a versão em PT-BR.
- A seguir está os prints do projeto desenvolvido.

Aluno: Jonas Brito Pacheco.
Web Academy Turma 3.

Apresentação do código - HTML



```
index.html - Exercício 1 - Gerador
Arquivo  Editar  Seleção  Ver  Acessar  Executar  Terminal  Ajuda
index.html x  # styles.css  JS main.js
index.html > ...
1  <!DOCTYPE html>
2  <html lang="pt-BR">
3    <head>
4      <meta charset="utf-8" />
5      <meta name="viewport" content="width=device-width" />
6      <link rel="stylesheet" href="css/styles.css">
7      <title>Gerador de Histórias Aleatórias</title>
8    </head>
9
10   <body>
11     <div>
12       <label for="customname">Digite seu Nome:</label>
13       <input id="customname" type="text" placeholder="" />
14     </div>
15     <div>
16       <label for="us">PT</label>
17       <input id="us" type="radio" name="ukus" value="us" checked />
18       <label for="uk">BR</label>
19       <input id="uk" type="radio" name="ukus" value="uk" />
20     </div>
21     <div>
22       <button class="randomize">Gerar história aleatória</button>
23     </div>
24     <p class="story"></p>
25     <script src="/main.js"></script>
26
27   </body>
28 </html>
29
30
```

Descrição do código

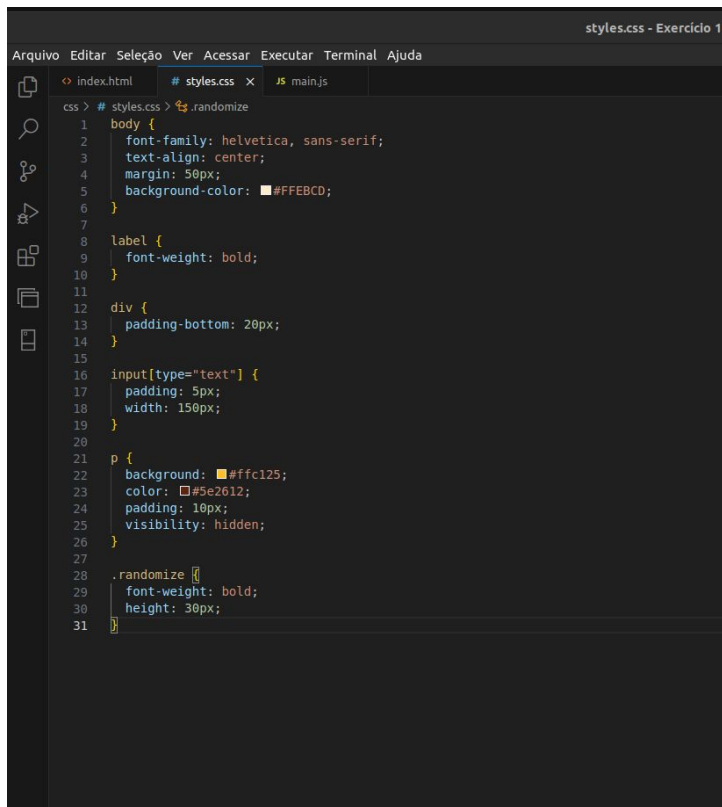
1 - Estrutura:

- `<!DOCTYPE html>`: Declaração do tipo de documento como HTML5.
- `<html lang="pt-BR">`: Elemento HTML raiz com a especificação do idioma como português do Brasil.
- `<head>`: Seção contendo informações sobre o documento, como meta tags e links para folhas de estilo.
 - `<meta charset="utf-8" />`: Define a codificação de caracteres como UTF-8.
 - `<meta name="viewport" content="width=device-width" />`: Configura a visualização em dispositivos móveis.
 - `<link rel="stylesheet" href="css/styles.css">`: Vincula a folha de estilo externa "styles.css".
 - `<title>Gerador de Histórias Aleatórias</title>`: Define o título da página.

2 - Corpo da Página:

- `<body>`: Contém o conteúdo visível da página.
 - `<div>`: Caixa contendo a entrada do usuário para o nome.
 - `<label for="customname">`: Rótulo associado ao campo de texto.
 - `<input id="customname" type="text" placeholder="" />`: Campo de texto para o usuário digitar seu nome.
 - `<div>`: Caixa contendo opções de seleção de idioma.
 - `<label for="us">PT</label>`: Rótulo para a opção de idioma "PT".
 - `<input id="us" type="radio" name="ukus" value="us" checked />`: Botão de rádio para selecionar o idioma "PT".
 - `<label for="uk">BR</label>`: Rótulo para a opção de idioma "BR".
 - `<input id="uk" type="radio" name="ukus" value="uk" />`: Botão de rádio para selecionar o idioma "BR".
 - `<div>`: Caixa contendo o botão para gerar histórias aleatórias.
 - `<button class="randomize">Gerar história aleatória</button>`: Botão com a classe "randomize".
 - `<p class="story"></p>`: Parágrafo vazio com a classe "story" para exibir as histórias geradas.
 - `<script src="/main.js"></script>`: Inclui o script JavaScript "main.js".

Apresentação do código - CSS



```
1 body {
2   font-family: helvetica, sans-serif;
3   text-align: center;
4   margin: 50px;
5   background-color: #FFEBDC;
6 }
7
8 label {
9   font-weight: bold;
10 }
11
12 div {
13   padding-bottom: 20px;
14 }
15
16 input[type="text"] {
17   padding: 5px;
18   width: 150px;
19 }
20
21 p {
22   background: #FFC125;
23   color: #5E2612;
24   padding: 10px;
25   visibility: hidden;
26 }
27
28 .randomize {
29   font-weight: bold;
30   height: 30px;
31 }
```

Descrição do código:

1 - body:

- font-family: Defini a fonte para o corpo da página como "Helvetica" ou, se não disponível, uma fonte genérica "sans-serif".
- text-align: Centraliza o texto no corpo da página.
- margin: Defini margens de 50 pixels em todos os lados do corpo da página.
- background-color: Define a cor de fundo como #FFEBDC.

2 - label:

- font-weight: Define o peso da fonte como negrito para todas as etiquetas na página.

3 - div:

- padding-bottom: Adiciona um espaçamento de 20 pixels na parte inferior de todas as divs na página.

4 - input:

- padding: Adiciona um preenchimento interno de 5 pixels para campos de texto.
- width: Defini a largura dos campos de texto como 150 pixels.

5 - paragraph

- background: Define a cor de fundo do parágrafo como #FFC125
- color: Define a cor do texto como #5E2612
- padding: Adiciona um preenchimento interno de 10 pixels para parágrafos.
- visibility: Define a visibilidade inicial como "hidden" para ocultar os parágrafos.

6 - Randomize:

- .randomize: Estiliza elementos com a classe "randomize".
- font-weight: Define o peso da fonte como negrito para o botão.
- height: Define a altura do botão como 30 pixels.

Apresentação do código - JavaScript

```
main.js - Exercício 1 - Gerador de histórias aleatórias - Visual Studio Code

Arquivo Editar Seleção Ver Acessar Executar Terminal Ajuda
index.html # styles.css JS main.js X

JS main.js > ...
1  /*1. COMPLETE VARIABLE AND FUNCTION DEFINITIONS*/
2
3  const customName = document.getElementById('customname');
4  const randomize = document.querySelector('.randomize');
5  const story = document.querySelector('.story');
6
7  function randomValueFromArray(array){
8      const random = Math.floor(Math.random() * array.length);
9      return array[random];
10 }
11
12 randomize.addEventListener('click', result);
13
14 function result() {
15
16     const originalStory = 'Certo dia eu estava caminhando na floresta, então :inserta: avistou um pássaro azul de :insertx:. Quan
17     const inserta = []
18     const insertb = ['futebol', 'xadrez', 'ping pong']
19     const insertc = ['Zeca pagodinho', 'Arnaldo Antunes', 'Nelson Mandela']
20     const insertd = ['100kg', '30kg', '50kg']
21     const insertx = ['Melô do Jonas', 'Suco de acerola', 'Sorvete de banana']
22     const inserty = ['italiano', 'de sopa quente', 'de salada de frutas']
23     const insertz = ['lançaram um foguete para marte', 'comeram pão com tucumã', 'o pássaro azul se transformou no Darth Vader']
24
25     let storyText = originalStory;
26
27     if(customName.value !== '') {
28         const name = customName.value;
29         storyText = storyText.replace(/:inserta:/g, name);
30     }
31     storyText = storyText.replace(':insertb:', randomValueFromArray(insertb));
32     storyText = storyText.replace(':insertc:', randomValueFromArray(insertc));
33     storyText = storyText.replace(':insertd:', randomValueFromArray(insertd));
34     storyText = storyText.replace(':insertx:', randomValueFromArray(insertx));
35     storyText = storyText.replace(':inserty:', randomValueFromArray(inserty));
36     storyText = storyText.replace(':insertz:', randomValueFromArray(insertz));
37
38     story.textContent = storyText;
39     story.style.visibility = 'visible';
40 }
41
```

Descrição do código:

1 - Realizei referências para os elementos HTML usando `document.getElementById` e `document.querySelector`.

2 - Defini uma função `randomValueFromArray` para obter um valor aleatório de um array.

3 - Adicionei um ouvinte de evento ao botão com a classe `.randomize` para chamar a função `result` quando o botão é clicado.

4 - Dentro da função `result`: Defini uma história original contendo espaços reservados (`:inserta:`, `:insertb:`, etc.). Criei arrays (`inserta`, `insertb`, etc.) com opções de substituição para os espaços reservados. Substitui os espaços reservados na história original pelos valores aleatórios correspondentes.

Atualizei o conteúdo do elemento HTML com a história gerada e tornou o elemento visível.

Link - repositório GitHub

<https://github.com/jonasbrito1/javaScript-web-academy-t3>