

LMS T02 - AP1 - 2024.2 - Campus de Quixadá

Prof. Jefferson de Carvalho Silva

Questão 01 (2,5pts) - Você foi contratado para desenvolver a página principal de um blog de notícias. A página deve ser estruturada de forma semântica usando HTML5 e deve conter as seguintes seções:

1. **Cabeçalho (Header):**

- Deve conter o título do blog e um menu de navegação com pelo menos 3 links (ex.: "Início", "Sobre", "Contato").

2. **Conteúdo Principal (Main):**

- Inclua pelo menos duas seções de notícias. Cada seção deve conter:
 - Um título da seção (ex.: "Notícias Recentes").
 - Dois artigos, cada um com um título, um breve parágrafo de texto e uma imagem ilustrativa (pode ser um link de imagem genérica, como <https://via.placeholder.com/150>).

3. **Rodapé (Footer):**

- Inclua informações de direitos autorais (ex.: "© 2024 Blog de Notícias") e links para redes sociais. O símbolo de copyright é definido, em HTML, como `©`

Estrutura esperada: Utilize as tags semânticas corretamente, como `header`, `nav`, `main`, `section`, `article`, e `footer`. Certifique-se de que o código HTML está bem indentado e validado.

Observação: Não é necessário adicionar CSS ou JavaScript; apenas a estrutura HTML semântica é exigida.

Questão 02 (2,5pts) - Você está desenvolvendo um site e precisa posicionar alguns elementos específicos usando CSS. Crie um código CSS, para o HTML abaixo, que atenda às seguintes necessidades de posicionamento:

1. **Cabeçalho Fixo:**

- Um elemento com a classe `.header` que permanece fixo no topo da página, mesmo quando o usuário rola a tela para baixo.

2. **Seção com Imagem Absoluta:**

- Dentro de uma `div` com a classe `.container`, posicione uma imagem (``) com a classe `.image-absolute` de forma absoluta, alinhada ao canto inferior direito da `div`.

3. **Botão com Posição Relativa:**

- Um botão com a classe `.button-relative` que se afasta 20 pixels da sua posição normal à esquerda.

4. **Texto com Posição Sticky:**

- Um texto com a classe `.text-sticky` que permanece fixo no topo da tela enquanto o usuário rola a seção em que ele está contido.

```

<body>
  <div class="header">Cabeçalho Fixo</div>

  <div class="container">
    <p>Conteúdo principal aqui.</p>
    
  </div>

  <button class="button-relative">Botão Relativo</button>

  <div class="sticky-section">
    <p class="text-sticky">Texto Sticky</p>
    <p>Conteúdo adicional que faz a página rolar.</p>
  </div>
</body>

```

Questão 03 (2,5pts) - Você foi encarregado de criar um efeito visual para um botão em um site. Quando o usuário passar o mouse sobre o botão, ele deve aumentar de tamanho e mudar de cor suavemente.

Tarefa:

Crie um código CSS que estilize um botão com as seguintes características:

1. O botão deve ter uma classe `.btn-animado`.
2. O botão deve ter bordas arredondadas na cor de sua preferência.
3. Quando o mouse passar sobre o botão, ele deve:
 - Aumentar em 20% de tamanho usando `scale()`.
 - Mudar de cor para um tom de verde claro.
4. O efeito de transformação e mudança de cor deve ocorrer suavemente em 0,5 segundos usando `transition`.
5. Exemplo de botão:

```

<button class="btn-animado">Clique Aqui</button>

```

Questão 04 (2,5pts) - Desenvolva um código CSS que organize um menu de navegação na horizontal usando Flexbox. O menu deve conter links organizados dentro de uma lista ``. Além disso, ao passar o mouse sobre os links, eles devem mudar de cor e exibir um efeito visual.

Requisitos:

1. Utilize uma lista `` com a classe `nav-menu` para os itens do menu.
2. Os itens da lista (``) devem conter links (`<a>`) e ser organizados na horizontal usando Flexbox.
3. Os links devem ter um espaçamento uniforme e um estilo visual básico (por exemplo, cor de fundo e padding).
4. Ao passar o mouse sobre os links (`hover`), eles devem mudar de cor e ter uma linha sublinhada animada.
5. Use o HTML abaixo:

```

<nav>
  <ul class="nav-menu">
    <li><a href="#">Início</a></li>
    <li><a href="#">Sobre</a></li>
  </ul>
</nav>

```

```
<li><a href="#">Serviços</a></li>
<li><a href="#">Contato</a></li>
</ul>
</nav>
```

COMO RESPONDER ESSA PROVA:

- Crie uma pasta chamada “LMS_T02_AP1_<sua matricula>”. Por exemplo, se minha matrícula for 123456, o nome da pasta será:
 - LMS_T02_AP1_123456
- Para cada questão, crie um arquivo HTML como o nome como no exemplo abaixo e os coloque dentro da pasta criada no item anterior:
 - questao_01.html
 - questao_02.html
 - questao_03.html
 - questao_04.html
- Caso precise baixar alguma imagem, coloque-a também dentro da pasta raiz, LMS_T01_AP1_<sua matricula>.
- NÃO crie arquivos CSS em separado. Faça tudo que se pede dentro da tag <style>, para cada questão.
- Compacte sua pasta LMS_T02_AP1_<sua matricula> no formato ZIP (evite o formato RAR ou TAR.GZ).
- SUBA o arquivo ZIP no SIGAA, na atividade correspondente PROVA_01.
- OPCIONAL: caso o aluno queira subir o código no GITHUB, não precisa mais enviar um ZIP para o SIGAA. Basta apenas enviar um .TXT com o link do GITHUB.

ATENÇÃO:

- O não obedecimento da organização e nomes dos arquivos acarretará na perda de 1,0 ponto, para cada ocorrência.
- Não serão aceitos arquivos enviados para o meu e-mail.
- O aluno deve pesquisar, apenas, por esses dois sites:
 - w3schools
 - https://github.com/jeffersoncarvalho/ufc_2024_2/tree/main/LMS
 - github pessoal
- Caso o aluno abra qualquer outro site, sua prova será anulada imediatamente com nota ZERO.
- Não será permitido o uso de celular.
- A prova tem duas horas de duração.
- Cada item não implementado em uma questão, acarretará na perda de 0,5 pontos da mesma.