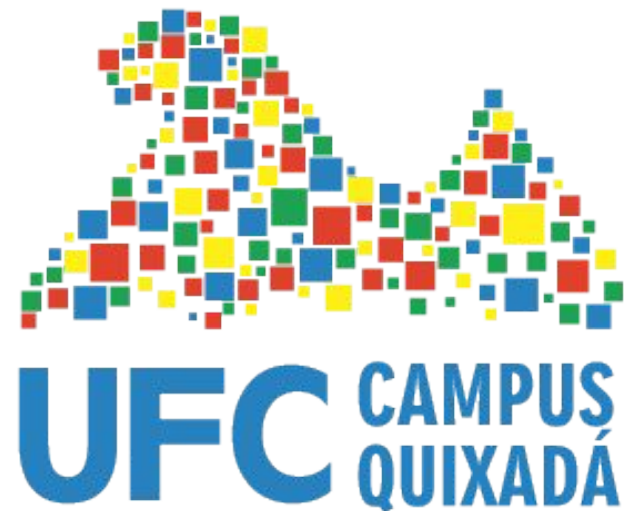


Desenvolvimento de software para Web

HTML5

Prof. Sidartha Carvalho



HTML5: Introdução

- O HTML5 trouxe uma série de mudanças significativas em relação às versões anteriores do HTML, tanto em termos de semântica quanto de funcionalidade.
- Essas mudanças visam melhorar a acessibilidade, a interatividade, a multimídia, e a estrutura do conteúdo nas páginas web.

HTML5: Introdução

- Antes do HTML5, a estrutura de uma página era basicamente construída com `<div>` e ``, o que não fornecia um significado claro sobre o propósito de diferentes partes do conteúdo.
- O HTML5 introduziu várias tags semânticas que oferecem mais significado ao conteúdo da página, facilitando a leitura e a compreensão tanto para os desenvolvedores quanto para ferramentas automatizadas, como motores de busca e leitores de tela.

HTML5: Introdução

- A lista completa do HTML5 inclui as tags mencionadas acima, com destaque para as novas tags semânticas, de mídia, interatividade, e as que aprimoram os formulários e os metadados.
- Essas tags permitem que os desenvolvedores criem sites mais acessíveis, interativos e bem estruturados, além de fornecerem suporte para novas tecnologias como vídeo, áudio, e gráficos dinâmicos.

HTML5: Introdução

- Embora iniciado em 2004, o HTML5 só começou a ser utilizado em 2009.
- Somente em 2014 declarado um padrão finalizado. O seu amplo uso começou logo após a declaração oficial em 2014.
- Hoje, o HTML5 é padrão universal para o desenvolvimento de sites e aplicativos web.
- Amplamente suportado por todos os navegadores modernos e usado em praticamente todos os projetos web, desde blogs simples até aplicativos interativos e sistemas corporativos complexos.

HTML5: Categorias de Tags

- Tags de Estrutura Semântica
- Tags de Mídia
- Tags de Formulários
- Tags de Metadados
- Tags de Scripts e APIs
- Outras Tags

HTML5: Categorias de Tags

- Tags de Estrutura Semântica
- Tags de Mídia
- Tags de Formulários
- Tags de Metadados
- Tags de Scripts e APIs
- Outras Tags

HTML5: Tags Estrutura Semântica

- `<header>`
 - Define a seção de cabeçalho de uma página ou parte de uma página.
- `<footer>`
 - Define a seção de rodapé de uma página ou parte da página.
- `<article>`
 - Representa um conteúdo independente e reutilizável.
- `<section>`
 - Define uma seção genérica de conteúdo.
- `<nav>`
 - Representa um conjunto de links de navegação.
- `<aside>`
 - Define conteúdo tangencial ou relacionado.
- `<main>`
 - Define o conteúdo principal da página.

HTML5: Tags Estrutura Semântica

- `<figure>`
 - Representa conteúdo visual, como uma imagem ou gráfico.
- `<figcaption>`
 - Define uma legenda para o conteúdo dentro de `<figure>`.
- `<mark>`
 - Destaca partes do texto.
- `<progress>`
 - Representa o progresso de uma tarefa.
- `<output>`
 - Representa o resultado de um cálculo ou ação.
- `<summary>`
 - Especifica um resumo para o conteúdo em `<details>`.
- `<details>`
 - Define um conteúdo que pode ser expandido ou recolhido.

HTML5: Tags Estrutura Semântica

- <header>
- Representa a seção introdutória de uma página ou um bloco de conteúdo, geralmente contendo o título, logotipos, menus de navegação e informações de destaque.
- É essencial para estruturar a parte superior da página ou de uma seção, ajudando na navegação e na identificação do conteúdo.

```
<header>
  <h1>Bem-vindo ao meu site</h1>
  <nav>
    <ul>
      <li><a href="#home">Home</a></li>
      <li><a href="#about">Sobre</a></li>
      <li><a href="#contact">Contato</a></li>
    </ul>
  </nav>
</header>
```

HTML5: Tags Estrutura Semântica

- `<nav>`
- Define uma seção de navegação, usada para links que direcionam para outras partes da página ou para outros sites.
- Fundamental para criar menus de navegação, sendo um ponto-chave para a usabilidade e acessibilidade de um site.

```
<nav>
  <ul>
    <li><a href="#section1">Seção 1</a></li>
    <li><a href="#section2">Seção 2</a></li>
    <li><a href="#section3">Seção 3</a></li>
  </ul>
</nav>
```

HTML5: Tags Estrutura Semântica

- `<main>`
- Representa o conteúdo principal da página, sendo a área onde o foco principal do site ou da aplicação é exibido. Deve conter o conteúdo único da página.
- Ajuda navegadores e tecnologias assistivas a identificarem a parte mais importante do conteúdo, aumentando a acessibilidade.

```
<main>
  <article>
    <h2>Notícia Principal</h2>
    <p>Detalhes sobre a notícia...</p>
  </article>
</main>
```

HTML5: Tags Estrutura Semântica

- `<article>`
- Representa um conteúdo independente que pode ser distribuído separadamente, como postagens de blog, artigos de notícias ou itens de um fórum.
- Facilita a organização de blocos de conteúdo independentes e reutilizáveis.

```
<article>  
  <h2>Título do Artigo</h2>  
  <p>Texto do artigo...</p>  
</article>
```

HTML5: Tags Estrutura Semântica

- `<footer>`
- Define a seção final de um documento ou de uma parte da página, geralmente contendo informações de rodapé, como direitos autorais, links de contato e informações adicionais.
- Essencial para estruturar o encerramento do conteúdo, tornando-o visualmente e semanticamente separado do resto da página.

```
<footer>  
  <p>&copy; 2024 Minha Empresa. Todos os direitos reservados.</p>  
  <p><a  
href="mailto:contato@minhaempresa.com">Contato</a></p>  
</footer>
```

HTML5: Tags Estrutura Semântica

- `<aside>`
- Usada para representar conteúdo relacionado ao conteúdo principal de uma página ou seção, mas que não faz parte diretamente do fluxo principal.
- Frequentemente utilizada para barras laterais, citações, notas de rodapé ou informações complementares, como widgets ou anúncios.

```
<main>
  <article>
    <h1>Como economizar energia em casa</h1>
    <p>Economizar energia não é apenas bom para o meio ambiente, mas também reduz
suas despesas.</p>
  </article>

  <aside>
    <h2>Notícias Relacionadas</h2>
    <ul>
      <li><a href="#news1">Dicas para economizar água</a></li>
      <li><a href="#news2">Energia solar: vale a pena?</a></li>
    </ul>
  </aside>
</main>
```

HTML5: Tags Estrutura Semântica

- `<mark>`
- usada para destacar trechos de texto, como se você estivesse marcando com um marcador amarelo em um livro.
- Enfatizar palavras ou frases importantes, especialmente em buscas ou para destacar informações relevantes.

```
<p>Resultados para a busca por "energia":</p>  
<p>A <mark>energia</mark> renovável é fundamental para  
a sustentabilidade do planeta.</p>
```

```
<style>  
  mark {  
    background-color: lightgreen;  
    color: darkgreen;  
    font-weight: bold;  
  }  
</style>
```

```
<p>As palavras <mark>importantes</mark> neste texto estão destacadas com um  
estilo personalizado.</p>
```


HTML5: Tags Estrutura Semântica

- `<progress>`
- Representar o progresso de uma tarefa ou processo em execução, como carregamento de arquivos, progresso de um download ou conclusão de uma atividade.
- É uma ferramenta visual e semântica para mostrar o estado de progresso em uma interface.

```
<label for="file-download">Progresso do download:</label>  
<progress id="file-download" value="70" max="100"></progress>
```

```
<label for="loading">Carregando dados...</label>  
<progress id="loading"></progress>
```

HTML5: Tags Estrutura Semântica

- `<progress>`

```
<style>
  progress {
    width: 100%;
    height: 20px;
  }
  progress::-webkit-progress-bar {
    background-color: #f3f3f3;
  }
  progress::-webkit-progress-value {
    background-color: #4caf50;
  }
</style>
```

```
<label for="task-progress">Progresso da tarefa:</label>
<progress id="task-progress" value="40" max="100"></progress>
```

HTML5: Tags Estrutura Semântica

- `<output>`
- Usada para representar o resultado de um cálculo ou ação realizada em um formulário.
- Útil para exibir resultados dinâmicos com a ajuda de JavaScript ou outros mecanismos.

```
<form oninput="result.value=parseInt(a.value) + parseInt(b.value)">
  <label for="a">Número 1:</label>
  <input type="number" id="a" name="a" value="0">

  <label for="b">Número 2:</label>
  <input type="number" id="b" name="b" value="0">

  <p>Resultado: <output name="result" for="a b">0</output></p>
</form>
```

O atributo `oninput` realiza o cálculo dinâmico da soma dos dois números e exibe o resultado no `<output>`.

O resultado do cálculo é atualizado automaticamente conforme os valores dos campos são alterados.

HTML5: Tags Estrutura Semântica

- `<summary>` e `<details>`
- Usada como um título interativo para a tag `<details>`, que serve para criar uma seção expansível ou colapsável.
- O `<summary>` funciona como o cabeçalho desta seção e é clicável para abrir ou fechar o conteúdo dentro do `<details>`.

```
<details>
  <summary>Capítulo 1: Introdução à Programação</summary>
  <p>Este capítulo cobre os conceitos básicos de lógica e linguagens de programação.</p>
</details>
  <summary>Seção 1.1: O que é um algoritmo?</summary>
  <p>Um algoritmo é uma sequência de passos para resolver um problema.</p>
</details>
  <summary>Seção 1.2: Linguagens de programação populares</summary>
  <p>Python, JavaScript, Java e C++ são exemplos de linguagens amplamente usadas.</p>
</details>
</details>
```

HTML5: Tags Estrutura Semântica

ATIVIDADE

Faça uso de cada uma das 14 tags HTML para definição de estrutura semântica introduzidas no HTML5 descritas nos slides anteriores.

Crie um exemplo para cada tag, pode ser na mesma página HTML.



HTML5: Categorias de Tags

- Tags de Estrutura Semântica
- Tags de Mídia
- Tags de Formulários
- Tags de Metadados
- Tags de Scripts e APIs
- Outras Tags

HTML5: Tags de Mídia

- `<audio>`
 - Define conteúdo de áudio.
- `<video>`
 - Define conteúdo de vídeo.
- `<source>`
 - Define várias fontes de mídia para `<audio>` e `<video>`.
- `<track>`
 - Define legendas ou faixas de texto para vídeos.

HTML5: Tags de Mídia

- <audio> e <source>
 - controls permite exibir controles padrões do navegador
 - Permite informar diferentes formatos de áudio/vídeo para maior compatibilidade

<h1>Exemplo com Áudio e Vídeo Públicos</h1>

<!-- Exemplo de Áudio -->

<h2>Reprodução de Áudio</h2>

<audio controls>

<source src="https://www.soundhelix.com/examples/mp3/SoundHelix-Song-1.mp3" type="audio/mpeg">

Seu navegador não suporta a reprodução de áudio.

</audio>

<audio controls>

<source src="https://www.soundhelix.com/examples/mp3/SoundHelix-Song-1.mp3" type="audio/mpeg">

<source src="https://www.example.com/audio/song.ogg" type="audio/ogg">

Seu navegador não suporta a reprodução de áudio.

</audio>

HTML5: Tags de Mídia

- <video>, <source> e <track>
 - Permite informar diferentes fontes de áudio/vídeo e diferentes legendas

<h2>Reprodução de Vídeo com Legenda</h2>

```
<video controls width="600">  
  <source src="https://www.w3schools.com/html/mov_bbb.mp4" type="video/mp4">  
  <source src="https://www.w3schools.com/html/mov_bbb_backup.mp4" type="video/mp4">  
  <track src="https://www.w3schools.com/html/mov_bbb_subtitles_en.vtt" kind="subtitles" srclang="en"  
label="English">  
  <track src="https://www.w3schools.com/html/mov_bbb_subtitles_fr.vtt" kind="subtitles" srclang="fr"  
label="Français">  
  Seu navegador não suporta a reprodução de vídeo.  
</video>
```

HTML5: Categorias de Tags

- Tags de Estrutura Semântica
- Tags de Mídia
- **Tags de Formulários**
- Tags de Metadados
- Tags de Scripts e APIs
- Outras Tags

HTML5: Tags de Formulário

<input>

- Com novos **tipos de entrada** no HTML5:
 - **color**: Para seleção de cores.
 - **date**: Para seleção de datas.
 - **datetime-local**: Para data e hora locais.
 - **email**: Para entrada de endereços de e-mail (validação integrada).
 - **month**: Para seleção de meses.
 - **number**: Para entrada de números (com controle de incremento/decremento).
 - **range**: Para seleção de valores numéricos dentro de um intervalo.
 - **search**: Campo otimizado para buscas.
 - **tel**: Para números de telefone.
 - **time**: Para horários.
 - **url**: Para endereços de URL (validação integrada).
 - **week**: Para selecionar semanas.

HTML5: Tags de Formulário

- <datalist>
 - Associa uma lista de opções predefinidas a um campo de entrada.
 - Usado com o atributo list no <input>.

```
<input list="browsers" name="browser">
<datalist id="browsers">
  <option value="Chrome">
  <option value="Firefox">
  <option value="Safari">
  <option value="Edge">
</datalist>
```

HTML5: Tags de Formulário

- <meter>
 - Representa medições dentro de um intervalo conhecido.

```
<meter value="0.6" min="0" max="1">60%</meter>
```

HTML5: Tags de Formulário

- Atributo multiple do <select>
 - Permite selecionar mais de uma opção usando CTRL ou Shift
 - Size é a quantidade de elementos exibidos ao usuário

```
<form action="/processar-hobbies" method="post">
  <label for="hobbies">Escolha seus hobbies favoritos:</label>
  <br>
  <select id="hobbies" multiple size="5">
    <option value="musica">Música</option>
    <option value="esportes">Esportes</option>
    <option value="viagem">Viagem</option>
    <option value="leitura">Leitura</option>
    <option value="cozinhar">Cozinhar</option>
    <option value="jogos">Jogos</option>
  </select>

  <br><br>
  <button type="submit">Enviar</button>
</form>
```

HTML5: Tags de Formulário

- Atributo novalidate
 - Não validar todo o <form> mesmo com itens como required
 - Pode ser interessante para realizar validações personalizadas, diferente do que o HTML faz
 - Pode-se usar JavaScript ou validar no Servidor após enviar a requisição

```
<form action="/submit-form" method="post" novalidate>
  <label for="nome">Nome:</label>
  <input type="text" id="nome" name="nome" required>
  <br><br>

  <label for="email">E-mail:</label>
  <input type="email" id="email" name="email" required>
  <br><br>

  <button type="submit">Enviar</button>
</form>
```

HTML5: Categorias de Tags

- Tags de Estrutura Semântica
- Tags de Mídia
- Tags de Formulários
- Tags de Metadados
- Tags de Scripts e APIs
- Outras Tags

HTML5: Tags de Metadados

`<base>` – Define a URL base para todos os links relativos no documento.

`<head>` – Contém metadados sobre o documento, como título, links para folhas de estilo e scripts.

`<link>` – Define relacionamentos entre o documento atual e outros recursos, como links para folhas de estilo ou ícones.

`<meta>` – Define metadados como descrições, palavras-chave ou instruções de configuração para navegadores e motores de busca.

`<style>` – Define regras de estilo internas no documento (CSS).

`<title>` – Define o título da página que aparece na aba do navegador.

`<noscript>` – Define conteúdo a ser exibido se o JavaScript não estiver habilitado no navegador.

HTML5: Tags de Metadados

- `<base>`
 - Definir a URL base para todos os links relativos e recursos em um documento HTML.
 - Ao especificar um caminho relativo em uma tag como `<a>` ou ``, o navegador usará o valor da URL definido no atributo `href` de `<base>` como ponto de partida.

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <base href="https://developer.mozilla.org/" target="_blank">
</head>
<body>
  <a href="en-US/docs/Web/HTML">Documentação de HTML</a>
  <a href="en-US/docs/Web/CSS">Documentação de CSS</a>
  
</body>
```

HTML5: Tags de Metadados

- `<noscript>`
 - Usada para fornecer conteúdo alternativo ou mensagens para navegadores que não suportam JavaScript ou têm o JavaScript desativado.
 - O conteúdo dentro da tag `<noscript>` será exibido apenas nesses casos.

```
<body>
  <noscript>
    <p style="color: red;">Aviso: Este site requer JavaScript para funcionar
    corretamente. Por favor, ative-o no seu navegador.</p>
  </noscript>
</body>
```

HTML5: Categorias de Tags

- Tags de Estrutura Semântica
- Tags de Mídia
- Tags de Formulários
- Tags de Metadados
- Tags de Scripts e APIs
- Outras Tags

HTML5: Tags de Scripts e APIs

- `<script>` – Define um bloco de código JavaScript ou referencia um arquivo externo com código JavaScript.
- `<canvas>` – Permite desenhar gráficos dinâmicos, imagens e outros elementos gráficos via JavaScript.
- `<svg>` – Define gráficos vetoriais escaláveis, usados para criar imagens e animações escaláveis.
- `<iframe>` – Insere um documento HTML dentro de outro, permitindo incorporar conteúdo externo, como vídeos ou outras páginas web.

HTML5: Tags de Scripts e APIs

- `<canvas>`
 - Criar gráficos dinâmicos, animações, desenhos e outros tipos de conteúdo visual programável com JavaScript.
 - A tag sozinha define apenas um espaço no layout; o desenho real é feito usando o contexto de renderização fornecido por JavaScript.

HTML5: Tags de Scripts e APIs

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
content="width=device-width, initial-scale=1.0">
  <title>Exemplo de &lt;canvas></title>
  <style>
    canvas {
      border: 2px solid black;
      display: block;
      margin: 20px auto;
    }
  </style>
</head>
<body>
  <h1>Exemplo de Canvas</h1>

  <!-- Definindo o espaço para o canvas -->
  <canvas id="myCanvas" width="400" height="400">
    Seu navegador não suporta a tag canvas.
  </canvas>
```

```
<script>
  // Obtendo o elemento canvas e seu contexto de desenho
  const canvas = document.getElementById('myCanvas');
  const ctx = canvas.getContext('2d');

  // Definindo as propriedades do desenho
  const centerX = canvas.width / 2;
  const centerY = canvas.height / 2;
  const radius = 100;

  // Desenhando um círculo preenchido
  ctx.beginPath();
  ctx.arc(centerX, centerY, radius, 0, 2 * Math.PI, false);
  ctx.fillStyle = 'skyblue';
  ctx.fill();

  // Desenhando o contorno do círculo
  ctx.lineWidth = 5;
  ctx.strokeStyle = 'darkblue';
  ctx.stroke();

  // Adicionando um texto no canvas
  ctx.font = '20px Arial';
  ctx.fillStyle = 'black';
  ctx.textAlign = 'center';
  ctx.fillText('Olá, Canvas!', centerX, centerY + radius + 30);
</script>
</body>
</html>
```

HTML5: Tags de Scripts e APIs

- `<svg>`
 - Usada para criar gráficos vetoriais escaláveis diretamente no HTML.
 - Suporta formas geométricas, textos, gradientes e animações.
 - Diferente do `<canvas>`, os elementos do `<svg>` são declarativos e fazem parte do DOM, o que permite fácil manipulação via CSS e JavaScript.

HTML5: Tags de Scripts e APIs

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exemplo de SVG</title>
  <style>
    svg {
      display: block;
      margin: 20px auto;
      border: 2px solid black;
    }
  </style>
</head>
<body>
  <h1>Exemplo de Gráfico SVG</h1>

  <!-- Gráfico SVG -->
  <svg width="400" height="400" xmlns="http://www.w3.org/2000/svg">
    <!-- Definindo um gradiente radial -->
    <defs>
      <radialGradient id="grad1" cx="50%" cy="50%" r="50%" fx="50%" fy="50%">
        <stop offset="0%" style="stop-color: skyblue; stop-opacity: 1" />
        <stop offset="100%" style="stop-color: darkblue; stop-opacity: 1" />
      </radialGradient>
    </defs>

    <!-- Um círculo preenchido com o gradiente -->
    <circle cx="200" cy="200" r="100" fill="url(#grad1)" />

    <!-- Adicionando um texto no centro -->
    <text x="200" y="210" font-family="Arial" font-size="20" text-anchor="middle" fill="white">
      SVG Example
    </text>
  </svg>
</body>
</html>
```

HTML5: Categorias de Tags

- Tags de Estrutura Semântica
- Tags de Mídia
- Tags de Formulários
- Tags de Metadados
- Tags de Scripts e APIs
- Outras Tags

HTML5: Outras Tags

- `<wbr>` – Sugere ao navegador onde pode quebrar a linha em um texto longo, caso necessário.
- `<bdi>` – Isola o texto de dentro de um elemento para que a direção do texto não afete o contexto externo (útil para texto bidirecional).
- `<bdo>` – Controla a direção do texto, permitindo a reversão do fluxo do texto em casos especiais.
- `<ruby>` – Define anotações de pronúncia ou tradução para caracteres chineses ou outros caracteres não latinos (usado para caracteres japoneses e chineses).
- `<rt>` – Especifica o texto de anotação para um caractere dentro de `<ruby>`.

HTML5: Outras Tags

- `<wbr>`

```
<div class="container">  
  Esta é uma palavra muito longa:  
  supercalifragilisticexpialidocious<wbr>isawesome!  
</div>
```

HTML5: Outras Tags

<ruby> e <rt>

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exemplo de &lt;ruby> e &lt;rt></title>
  <style>
    ruby {
      font-size: 20px;
      font-family: Arial, sans-serif;
    }
    rt {
      font-size: 14px;
      color: gray;
    }
  </style>
</head>
<body>
  <h1>Exemplo de &lt;ruby> e &lt;rt></h1>
  <p>Aqui está um exemplo de texto com anotações de pronúncia:</p>
  <p>Este é o kanji para "montanha": <ruby>山<rt>やま</rt></ruby> (yama).</p>
  <p>Este é um exemplo com um termo em chinês: <ruby>中国<rt>Zhōngguó</rt></ruby> (China). </p>
  <p>Você também pode usar para explicar acrônimos: <ruby>HTML<rt>HyperText Markup Language</rt></ruby>. </p>
</body>
</html>
```

Dúvidas?



www.shutterstock.com · 744867163

Exercício

Crie um mini-tube (similar ao Youtube) usando as tags de mídia.

Deve conter uma galeria de vídeos que podem ser executados no próprio navegador.

Usar src para vídeos que existem!

Envio pelo SIGAA

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>MiniTube</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      background-color: #f9f9f9;
      color: #333;
    }
    header {
      background-color: #ff0000;
      color: white;
      padding: 10px;
      text-align: center;
    }
    .gallery {
      display: flex;
      flex-wrap: wrap;
      justify-content: center;
      gap: 20px;
      padding: 20px;
    }
    .video-container {
      width: 300px;
      border: 1px solid #ccc;
      border-radius: 5px;
      background-color: white;
      box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);
      overflow: hidden;
      text-align: center;
    }
    video {
      width: 100%;
      height: auto;
    }
    .title {
      padding: 10px;
      font-size: 16px;
      font-weight: bold;
      color: #444;
    }
  </style>
</head>
<body>
  <header>
    <h1>MiniTube</h1>
    <p>Galeria de vídeos para assistir diretamente no navegador</p>
  </header>

  <section class="gallery">
    <!-- Vídeo 1 -->
    <div class="video-container">
      <video controls>
        <source src="https://www.w3schools.com/html/mov_bbb.mp4" type="video/mp4">
        Seu navegador não suporta vídeos.
      </video>
      <div class="title">Big Buck Bunny</div>
    </div>

    <!-- Vídeo 2 -->
    <div class="video-container">
      <video controls>
        <source src="https://www.w3schools.com/html/movie.mp4" type="video/mp4">
        Seu navegador não suporta vídeos.
      </video>
      <div class="title">The Bear</div>
    </div>

    <!-- Vídeo 3 -->
    <div class="video-container">
      <video controls>
        <source src="https://sample-videos.com/video123/mp4/720/big_buck_bunny_720p_1mb.mp4" type="video/mp4">
        Seu navegador não suporta vídeos.
      </video>
      <div class="title">Big Buck Bunny (Curta)</div>
    </div>

    <!-- Vídeo 4 -->
    <div class="video-container">
      <video controls>
        <source src="https://sample-videos.com/video123/mp4/720/sample-5s.mp4" type="video/mp4">
        Seu navegador não suporta vídeos.
      </video>
      <div class="title">Vídeo de 5 segundos</div>
    </div>
  </section>
</body>
</html>
```