

# Programação Web Responsiva

## CSS e Vídeos

Prof. Bruno Azevedo

UNIP – Universidade Paulista



# Propriedade Background-image

- Define uma imagem de fundo para um elemento.
- Pode usar arquivos locais ou URLs externas.
- Exemplo:

```
body {  
    background-image: url("imagem.jpg");  
}
```

- A imagem pode ser ajustada com outras propriedades.

# Propriedade Background-repeat

- Controla a repetição da imagem de fundo.
- Valores:
  - repeat (padrão): repete horizontal e verticalmente.
  - no-repeat: sem repetição.
  - repeat-x / repeat-y: repete apenas em uma direção.
- Exemplo:

```
body {  
  background-image: url("imagem.jpg");  
  background-repeat: no-repeat;  
}
```

# Propriedade Background-size

- Define o tamanho da imagem de fundo.
- Pode usar valores em pixels, porcentagem, ou palavras-chave.
- Valores comuns:
  - cover: cobre todo o elemento mantendo a proporção.
  - contain: ajusta a imagem para caber dentro do elemento.
- Exemplo:

```
body {  
  background-image: url("imagem.jpg");  
  background-size: cover;  
}
```

- Outro exemplo:

```
body {  
  background-image: url("imagem.jpg");  
  background-size: 50% 50%;  
}
```

# Introdução às Paletas de Cores

- No desenvolvimento Web, as cores desempenham um papel importante na criação de layouts visuais atraentes e funcionais.
- Cores podem ser aplicadas a diferentes elementos HTML, como texto, fundo, bordas, botões e links.
- Uma **paleta de cores** é uma seleção de cores que são combinadas de forma harmoniosa para criar um visual coeso.
- A escolha da paleta de cores afeta a usabilidade, acessibilidade e estética de um site.

# Ferramentas Online para Gerar Paletas de Cores

- Colors: <https://colors.co/>
- Paleta de Cores da Nuvemshop: <https://paletadecores.com/>
- Roda de Cores da Adobe Color:  
<https://color.adobe.com/pt/create/color-wheel>
- Colorhunt: <https://colorhunt.co/>
- Colormind: Utiliza inteligência artificial para gerar paletas de cores.  
<http://colormind.io/>

# Representação de Cores em HTML

- Em HTML, as cores podem ser definidas de várias formas. Algumas das mais comuns são:
  - Nomes de cores: São nomes predefinidos como `red`, `green`, `blue`, entre outros.
  - Códigos hexadecimais: Representados por `#RRGGBB`, como `#FF5733` (um tom de laranja).
  - Códigos RGB: Combinam valores de Red, Green e Blue no formato `rgb(R, G, B)`, como `rgb(255, 87, 51)`.
  - Códigos RGBA: Similar ao RGB, mas com um valor adicional para Alpha, que controla a opacidade da cor, variando de 0 (totalmente transparente) a 1 (totalmente opaco). Exemplo: `rgba(255, 87, 51, 0.5)`.
  - Códigos HSL: Usam matiz (H), saturação (S) e luminosidade (L) para definir a cor. Exemplo: `hsl(9, 100%, 60%)`.
- Diferentes métodos de representação de cores oferecem flexibilidade na definição de tonalidades e transparências em HTML.

# Usando Nomes de Cores e Códigos Hexadecimais

- Exemplo de uso com nome de cor:

```
<body style="background-color: lightblue;">  
<h1 style="color: darkred;">Título com cor personalizada</h1>  
<p>Texto com cor padrão.</p>  
</body>
```

- Exemplo de uso com código hexadecimal:

```
<body style="background-color: |#FF5733|;">  
<h1 style="color: #333333;">Título com cor hexadecimal</h1>  
</body>
```



# Usando Códigos RGB e RGBA

- Exemplo de uso com código RGB:

```
<body style="background-color: rgb(255, 87, 51);">  
<h1 style="color: rgb(51, 51, 51);">Texto com cor RGB</h1>  
</body>
```

- A função `rgba()` permite definir cores com transparência (canal alfa, que varia de 0 (totalmente transparente) a 1 (totalmente opaco)).
- Exemplo de uso com código RGBA:

```
<body style="background-color: rgba(255, 87, 51, 0.5);">  
<h1 style="color: rgba(0, 0, 0, 0.8);">Texto com transparência</h1>  
</body>
```

# Usando Códigos HSL e HSLA

- Exemplo de uso com código HSL:

```
<body style="background-color: hsl(20, 100%, 70%);">  
<h1 style="color: hsl(0, 0%, 20%);">Texto com HSL</h1>  
</body>
```

- Exemplo de uso com código HSLA:

```
<body style="background-color: hsla(20, 100%, 70%, 0.5);">  
<h1 style="color: hsla(0, 0%, 20%, 0.8);">Texto com transparência</h1>  
</body>
```

# Como Inserir Vídeos em HTML5

- No HTML5, a tag <video> permite inserir vídeos diretamente na página sem a necessidade de plugins externos.
- a tag <video> define a área onde o vídeo será exibido.
- A tag <source> define diferentes formatos de vídeo para compatibilidade entre navegadores.

▷ Exemplo:

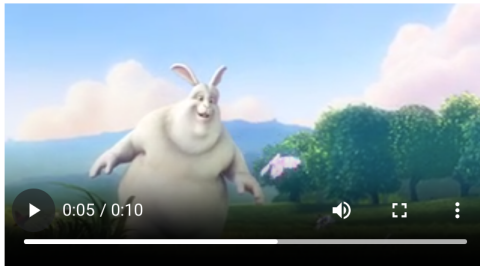
```
<video controls>
```

```
  <source src="videos/video.mp4" type="video/mp4">
```

```
  <source src="videos/video.webm" type="video/webm">
```

Seu navegador não suporta a tag de vídeo.

```
</video>
```



# Atributos Comuns da Tag <video>

- A tag <video> pode ser personalizada com diferentes atributos:
  - controls: Adiciona controles para o usuário (play, pause, volume, etc.).
  - autoplay: Inicia o vídeo automaticamente assim que a página for carregada.
  - loop: Reproduz o vídeo em laço contínuo.
  - muted: Inicia o vídeo com o áudio desativado.
  - width e height: Definem as dimensões do vídeo.
- ▷ Exemplo com mais atributos:

```
<video controls autoplay loop muted width="800">  
  <source src="video.mp4" type="video/mp4">  
  <source src="video.webm" type="video/webm">  
  Seu navegador não suporta a tag de vídeo.  
</video>
```

# Atividade Prática 10

- ❶ Criar uma página HTML chamada `midia.html` contendo:
  - ❶ Um fundo com uma imagem aplicada através do CSS.
  - ❷ Um título `<h1>` com uma cor diferente e sombra no texto.
  - ❸ Um vídeo embutido na página, com opção de reprodução automática.
  - ❹ Um contêiner `<div>` para organizar o conteúdo.
- ❷ **Ampliação:** criar uma página para promover uma ONG da sua região, exibindo imagens e um vídeo institucional.
- ❸ Cada aluno deve produzir um relatório de 1 a 3 páginas contendo:
  - **Resumo teórico:** explicação sobre manipulação de imagens e vídeos no CSS.
  - **Código-fonte comentado:** explicação de cada configuração aplicada.