

# Divs e Herança em CSS

Usabilidade, desenvolvimento web, mobile e jogos

Prof. Me. Gustavo Torres Custódio  
gustavo.custodio@anhembis.edu.br

# Conteúdo

**Herança**

**Seletores**

**Design Responsivo**

**Exercícios**



Divs e herança em CSS

# Herança

# Herança

- Se o CSS de algumas propriedades do <body> for alterado, isso será refletido nos elementos de sua página.

# Herança

- Se o CSS de algumas propriedades do `<body>` for alterado, isso será refletido nos elementos de sua página.
  - A propriedade `font-family` possui esse comportamento.
    - Se você mudar a fonte do `<body>`, a fonte da página inteira vai mudar.
  - Dizemos que a propriedade `font-family` é **herdada** pelos elementos filhos.

# Herança

- O código HTML:

# Herança

- O código HTML:

```
<html>
  <head>
    <title></title>
    <link> <meta>
  </head>
  <body>
    <h1></h1>
    <h2></h2>
    <p></p>
    <div>
      <p> <em></em> </p>
      <p> <a></a> </p>
    </div>
    <ul>
      <li> <p></p> </li>
      <li></li>
      <li> <em></em> </li>
    </ul>
  </body>
</html>
```

# Herança

- O código HTML:

```
<html>
  <head>
    <title></title>
    <link> <meta>
  </head>
  <body>
    <h1></h1>
    <h2></h2>
    <p></p>
    <div>
      <p> <em></em> </p>
      <p> <a></a> </p>
    </div>
    <ul>
      <li> <p></p> </li>
      <li></li>
      <li> <em></em> </li>
    </ul>
  </body>
</html>
```

- Possui a estrutura de herança:

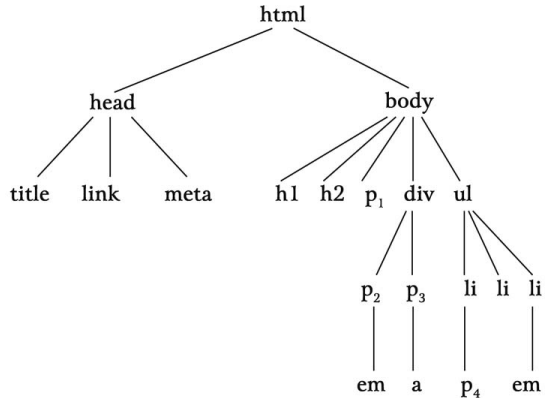


# Herança

- O código HTML:

```
<html>
  <head>
    <title></title>
    <link> <meta>
  </head>
  <body>
    <h1></h1>
    <h2></h2>
    <p></p>
    <div>
      <p> <em></em> </p>
      <p> <a></a> </p>
    </div>
    <ul>
      <li> <p></p> </li>
      <li></li>
      <li> <em></em> </li>
    </ul>
  </body>
</html>
```

- Possui a estrutura de herança:



# Herança

- Outras propriedades **não são herdadas** por elementos filhos.

# Herança

- Outras propriedades **não são herdadas** por elementos filhos.
  - A cor de fundo background-color não é herdada.

# Herança

- Outras propriedades **não são herdadas** por elementos filhos.
  - A cor de fundo background-color não é herdada.
- É possível forçar que algumas propriedades sejam herdadas usando o inherit.

# Herança

- Outras propriedades **não são herdadas** por elementos filhos.
  - A cor de fundo background-color não é herdada.
- É possível forçar que algumas propriedades sejam herdadas usando o inherit.
- No exemplo, o padding do <body> é herdado por uma <div>.

# Herança

## HTML:

```
<div class="page">
  <h1>Usabilidade, desenvolvimento etc...</h1>
  <p>Nessa disciplina, estudamos IHC, desenvolvimento
  de páginas web, desenvolvimento para mobile e jogos.</p>
  <p>Agora estamos em web.</p>
</div>
```

## CSS:

```
body {
  font-family: Arial, Verdana, sans-serif;
  color: #665544;
  padding: 10px;
}

.page {
  border: 1px solid #665544;
  background-color: #efefef;
  padding: inherit;
}
```

# Herança

## HTML:

```
<div class="page">
  <h1>Usabilidade, desenvolvimento etc...</h1>
  <p>Nessa disciplina, estudamos IHC, desenvolvimento
  de páginas web, desenvolvimento para mobile e jogos.</p>
  <p>Agora estamos em web.</p>
</div>
```

## CSS:

```
body {
  font-family: Arial, Verdana, sans-serif;
  color: #665544;
  padding: 10px;
}

.page {
  border: 1px solid #665544;
  background-color: #efefef;
  padding: inherit;
}
```

## Usabilidade, desenvolvimento etc...

Nessa disciplina, estudamos IHC, desenvolvimento de páginas web,  
desenvolvimento para mobile e jogos.

Agora estamos em web.

# Herança

- O elemento `<div>` permite agrupar um conjunto de elementos em um único bloco.



# Herança

- O elemento `<div>` permite agrupar um conjunto de elementos em um único bloco.
  - Você pode por exemplo criar uma `<div>` contendo todos os elementos do cabeçalho do site.
  - O conteúdo de uma `<div>` é iniciado em uma nova linha.

# Herança

- Vantagens de herança:

# Herança

- Vantagens de herança:
  - Redução da quantidade de regras de estilo.
  - Facilitar a reutilização e manutenção do código.
  - Folhas de estilo carregam mais rapidamente.
  - Produtividade.



Divs e herança em CSS

# Seletores

# Seletores

- Anteriormente vimos os 3 tipos básicos de seletores CSS:

# Seletores

- Anteriormente vimos os 3 tipos básicos de seletores CSS:
  - elementos, class e id.

# Seletores

- Anteriormente vimos os 3 tipos básicos de seletores CSS:
  - elementos, class e id.
- É possível gerar seletores mais complexos combinando os existentes.

# Seletores

- Anteriormente vimos os 3 tipos básicos de seletores CSS:
  - elementos, class e id.
- É possível gerar seletores mais complexos combinando os existentes.
- Existem 3 **elementos de combinação** disponíveis:



# Seletores

- Anteriormente vimos os 3 tipos básicos de seletores CSS:
  - elementos, class e id.
- É possível gerar seletores mais complexos combinando os existentes.
- Existem 3 **elementos de combinação** disponíveis:
  - **Seletores descendentes;**
  - **Seletores filhos;**
  - **Seletores irmãos.**

## Seletores Descendentes

- Um seletor descendente combina dois ou mais seletores separados por um **espaço em branco**.

## Seletores Descendentes

- Um seletor descendente combina dois ou mais seletores separados por um **espaço em branco**.
- Ele encontra elementos que sejam descendentes do primeiro elemento declarado no seletor.

## Seletores Descendentes

- Um seletor descendente combina dois ou mais seletores separados por um **espaço em branco**.
- Ele encontra elementos que sejam descendentes do primeiro elemento declarado no seletor.
- Exemplo:
  - `div p { color: "#f00" }`
  - Muda a cor da fonte apenas de parágrafos `<p>` que são filhos de uma `<div>`.

## Seletores Descendentes

- Podemos também usar seletores descendentes usando o **id** ou **classe** dos elementos.

## Seletores Descendentes

- Podemos também usar seletores descendentes usando o **id** ou **classe** dos elementos.

1 <ul id="nav">	1 #nav li { display:inline; }
2     <li><a href="#">Link 1</a></li>	2 #nav a { font-weight:bold; }
3 </ul>	~
4 <ul>	~
5     <li><a href="#">Link 2</a></li>	~
6     <li><a href="#">Link 3</a></li>	~
7 </ul>	~
	~
	~

## Seletores Descendentes

- Apenas o primeiro link será afetado.
  - Ele é o único que possui o id correspondente.

## Seletores Descendentes

- Apenas o primeiro link será afetado.
  - Ele é o único que possui o id correspondente.

> [Link 1](#)

- [Link 2](#)
- [Link 3](#)



## Seletores Filhos

- Um seletor filho seleciona um elemento filho de outra tag.
  - Funciona apenas para filhos imediatos de uma tag.

## Seletores Filhos

- Um seletor filho seleciona um elemento filho de outra tag.
  - Funciona apenas para filhos imediatos de uma tag.
- Exemplo:

1	<div>	1	div > strong { color:#f00; }
2	<strong>Primeiro elemento</strong>	2	
3	<p> <strong>Segundo elemento</strong>	3	
	> </p>	4	
4	</div>	5	
5		6	

## Seletores Filhos

- Um seletor filho seleciona um elemento filho de outra tag.
  - Funciona apenas para filhos imediatos de uma tag.
- Exemplo:

1	<div>		1	div > strong { color:#f00; }
2	<strong>Primeiro elemento</strong>		2	
3	<p> <strong>Segundo elemento</strong>		3	
	> </p>		4	
4	</div>		5	
5			6	

- Apenas os elementos dentro da tag <strong>, filhos de <div> serão afetados.
- ou seja, o <strong> dentro de <p> não será afetado.

## Seletores Irmãos

- Seletores irmãos selecionam elementos HTML que estão “ao lado” de elementos específicos, no mesmo nível de herança.

## Seletores Irmãos

- Seletores irmãos selecionam elementos HTML que estão “ao lado” de elementos específicos, no mesmo nível de herança.
- Há dois tipos de seletores para irmãos:

## Seletores Irmãos

- Seletores irmãos selecionam elementos HTML que estão “ao lado” de elementos específicos, no mesmo nível de herança.
- Há dois tipos de seletores para irmãos:
  - irmãos adjacentes (+);
  - irmãos gerais (~).

## Irmãos Adjacentes

- Um seletor de irmão adjacente seleciona elementos pelo sinal de adição.

## Irmãos Adjacentes

- Um seletor de irmão adjacente seleciona elementos pelo sinal de adição.
  - O seletor tem como alvo um elemento que está imediatamente depois de outro.
  - Os dois elementos devem ter o mesmo pai.



## Irmãos Adjacentes

- Um seletor de irmão adjacente seleciona elementos pelo sinal de adição.
  - O seletor tem como alvo um elemento que está imediatamente depois de outro.
  - Os dois elementos devem ter o mesmo pai.
  - Exemplo:

```
p + p {color:#f00; }
```

```
<div>  
  <p>Primeiro parágrafo</p>  
  <p>Segundo parágrafo</p>  
</div>
```

- Essa regra será aplicada ao segundo parágrafo.

## Irmãos Gerais

- O seletor de irmãos gerais permite selecionar elementos que possuem o mesmo pai.

## Irmãos Gerais

- O seletor de irmãos gerais permite selecionar elementos que possuem o mesmo pai.
  - Elementos combinados pelo til (~).
  - Não existe a necessidade dos elementos serem adjacentes.

# Irmãos Gerais

- Exemplo:

# Irmãos Gerais

- Exemplo:
  - Selecionar apenas os parágrafos irmãos de `<h1>`.

# Irmãos Gerais

- Exemplo:
  - Selecionar apenas os parágrafos irmãos de `<h1>`.

```
<div id="conteudo" role="main">  
  <h1>Meu conteúdo</h1>  
  <p>parágrafo 1</p>  
  <p>parágrafo 2</p>  
  <p>parágrafo 3</p>  
</div>
```

```
#conteudo h1~p {  
  margin-top: 5px;  
  background-color: slategrey;  
  width: 300px;  
  height: 30px;  
  margin-left: 40px;  
}
```

## Aplicação: Menu Horizontal

- Vamos contruir um menu horizontal utilizando CSS.

## Aplicação: Menu Horizontal

- Vamos contruir um menu horizontal utilizando CSS.

```
<div id="menuhorizontal" role="navigation">
  <ul>
    <li> <a href="#">Home</a> </li>
    <li> <a href="#">Produtos</a> </li>
    <li> <a href="#">Links</a> </li>
    <li> <a href="#">Contato</a> </li>
  </ul>
</div>
```

```
#menuhorizontal ul {
  padding: 0px;
  margin: 0px;
  list-style: none;
}
#menuhorizontal ul li {
  display: inline;
}
```



## Aplicação: Menu Horizontal

```
#menuhorizontal ul li a {  
    /*preenchimento superior e inferior 2px*/  
    /*preenchimento esquerdo e direito 10px*/  
    padding: 2px 10px;  
    display: inline-block;  
    color: white;  
    text-decoration: none;  
    border-bottom: 3px solid white;  
}
```

## Aplicação: Menu Horizontal

```
#menuhorizontal ul li a {  
    /*preenchimento superior e inferior 2px*/  
    /*preenchimento esquerdo e direito 10px*/  
    padding: 2px 10px;  
    display: inline-block;  
    color: white;  
    text-decoration: none;  
    border-bottom: 3px solid white;  
}
```



# Pseudoclasses

- Uma pseudo-classe CSS especifica um estado especial do elemento selecionado.

# Pseudoclasses

- Uma pseudo-classe CSS especifica um estado especial do elemento selecionado.
  - Exemplo: hover.
    - o elemento possui um estilo específico para quando o mouse está sobre ele.

## Aplicação: Menu Horizontal

- Vamos adicionar uma pseudo-classe em nosso menu horizontal.

## Aplicação: Menu Horizontal

- Vamos adicionar uma pseudo-classe em nosso menu horizontal.

```
#menuhorizontal ul li a:hover {  
    color: #6D6D6D;  
    border-bottom: 3px solid slateblue;  
}
```



Divs e herança em CSS

# Design Responsivo

# Design Responsivo

- Cada vez mais sites são acessados por meio de celulares e tablets.



## Design Responsivo

- Cada vez mais sites são acessados por meio de celulares e tablets.
- As telas que usuários de dispositivos móveis utilizam possuem tamanho diferentes de *desktops* e *notebooks*.

## Design Responsivo

- Cada vez mais sites são acessados por meio de celulares e tablets.
- As telas que usuários de dispositivos móveis utilizam possuem tamanho diferentes de *desktops* e *notebooks*.
- Precisamos modificar o site para atender esses usuários.

# Design Responsivo

- Uma forma de fazer isso é criando uma versão do site para dispositivos móveis.

# Design Responsivo

- Uma forma de fazer isso é criando uma versão do site para dispositivos móveis.
  - Alguns sites possuem um caminho para versões *mobile*:
    - m.site.com.br

# Design Responsivo

- Uma forma de fazer isso é criando uma versão do site para dispositivos móveis.
  - Alguns sites possuem um caminho para versões *mobile*:
    - m.site.com.br
- Outra forma de fazer isso é no CSS utilizando *Media queries*.

## Media Query

- *Media Query* é uma técnica introduzida no CSS 3.

## Media Query

- *Media Query* é uma técnica introduzida no CSS 3.
- Utiliza o media, onde um bloco CSS só é executado sob condições.

# Media Query

- *Media Query* é uma técnica introduzida no CSS 3.
- Utiliza o media, onde um bloco CSS só é executado sob condições.

```
@media only screen and (max-width: 600px) {  
  body {  
    background-color: lightblue;  
  }  
}
```

- Neste exemplo, se a largura da tela tem 600 pixels ou menos, a cor de fundo dela será azul claro.
  - O trecho `only screen` evita que navegadores que não suportam *media queries* apliquem o CSS.



## Design Responsivo

- Vamos fazer com que o menu horizontal se torne um menu vertical, caso a largura do site seja maior que 700px.

## Design Responsivo

```
@media only screen and (min-width: 700px) {  
    #menuhorizontal ul li {  
        display: block;  
        border: 1px solid white;  
        border-radius: 5px;  
        width: 20%;  
    }  
  
    #menuhorizontal ul li a {  
        border-bottom: none;  
    }  
}
```

## Design Responsivo





Divs e herança em CSS

## Exercícios

## Exercício 1

- Crie uma página de contato contendo e-mail e área de texto para comentários.
- Use o CSS para alterar o estilo da página.
  - Faça com que o botão enviar mude de cor quando o mouse passar por ele.
  - Faça com que o campo de e-mail e a TextArea fiquem uma do lado da outra quando o comprimento da tela for maior que 800px.

# Referências



Duckett, J. (2011).

*HTML & CSS: design and build websites*, volume 15.

Wiley Indianapolis, IN.

Obrigado

[gustavo.custodio@anhembi.br](mailto:gustavo.custodio@anhembi.br)