

<head>

</head>

<body>

# Programação em Ambiente Web I

## *Aula 2*

### *Introdução à CSS*

</body>

<footer>

Instituto de Computação  
UFMT 2023

Programação em Ambiente Web I Prof. Jivago  
Medeiros

>

1  
</footer>

<head>

# CSS

</head>

<body>

- **CSS** (*Cascading Style Sheets*) – Linguagem de folhas de estilo
- **Objetivo:** Separar a estrutura / conteúdo do estilo / formatação em documentos (exemplo, HTML). Definindo e alterando diferentes “folhas de estilo” sempre que necessário.

</body>

<footer>

>

<head>

# Especificação

</head>

<body>

- **Cascading Style Sheets (CSS).** Editor: ETEMAD, E. J.  
Disponível em: <http://www.w3.org/TR/CSS/>

</body>

<footer>

>

<head>

## Outros Materiais

</head>

<body>

- **Tutorias / cursos:**

- W3Schools, **CSS3 Introduction**. Disponível em: [http://www.w3schools.com/css/css3\\_intro.asp](http://www.w3schools.com/css/css3_intro.asp)
- W3C Brasil, **Curso de CSS3**. Disponível em: <http://www.w3c.br/Cursos/CursoCSS3>

- **Outros:**

- CSS-TRICKS: <https://css-tricks.com/>

</body>

<footer>

>

<head>

# CSS

</head>

<body>

- **História:** A primeira versão (CSS1) foi especificada poucos anos depois da primeira especificação publicada do HTML1 (1993 HTML e 1996 CSS). Obtendo notoriedade em nível global apenas no início da primeira década do ano 2000.

</body>

<footer>

>

<head>

# Módulos CSS

</head>

<body>

- A partir do CSS3, a especificação passou a ser dividida em diferentes módulos que podem ser atualizados de forma independente um dos outros, dando origem a diferentes *levels* (níveis).
- Alguns módulos:
  - *Selectors, Color, Background, Box, Media Queries, Namespaces* etc.

</body>

<footer>

>

<head>

# O que é possível formatar CSS ?

</head>

<body>

- Tipo da fonte, cor da fonte, estilo da fonte, bordas (tipo, cor, tamanho), fundo (cor, imagem, comportamento). Posição / posicionamento / tamanho / comportamento / visibilidade / fluides / sombra dos elementos. Elementos / itens / campos dos formulários. Etc, etc, etc...
- ...basicamente, “**tudo**” em um documento HTML

</body>

<footer>

>

<head>

# Sintaxe

</head>

<body>

```
seletor, seletores {  
  propriedade: valor1;  
  outra-propriedade: valor1 valor2 valor3;  
}
```

</body>

<footer>

>



<head>

# Exemplos

</head>

<body>

```
p {  
    color: #060;  
    text-align: center;  
}
```

</body>

<footer>

>

<head>

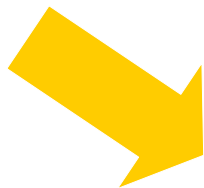
# Seletores

</head>

<body>

- **Seletores:** Definem (selecionam) quais os conjuntos de elementos receberão as propriedades.

Aplica o estilo / formatação (**color** e **text-align**) a **todos** os **elementos** do tipo **<p>**



```
p {  
    color: #060;  
    text-align: center;  
}
```

</body>

<footer>

>

<head>

# Seletores básicos

</head>

<body>

## Nome da *tag* / *elemento*

Aplica o estilo a todos os elementos daquela *tag*

## *#id* do Elemento

Aplica o estilo somente ao elemento que possui aquele ***#id***

## .*nome* da Classe

Aplica a todas os elementos da classe (atributo ***class***)

</body>

<footer>

>

<head>

# Exemplos

</head>

<body>

```
h1 {  
    background-color: #000;  
    color: #FFF;  
    font-size: 32px;  
    text-decoration: underline;  
}  
  
#conteudo {  
    background-color: #CCC;  
    border-bottom: 4px double #000;  
}  
  
.paragrafo-maior {  
    font-size: 18px;  
    font-weight: bold;  
}
```

</body>

<footer>

>

<head>

## Sobre cores em CSS...

</head>

<body>

- Quando trabalhamos com **cores em CSS**, seja cor da fonte (color), cor de fundo (background-color), cor da borda (border-color) entre outras, há **três formas de defini-las**:
  - Pelo nome (em inglês) da cor, ex:  
**black**, **blue**, **brown**, **gold**, gray, **green**, **olive**, **pink**,  
**silver**, **snow**, **violet**, **white**, **yellow**, entre muitas outras:

[https://www.w3schools.com/colors/colors\\_names.asp](https://www.w3schools.com/colors/colors_names.asp)

</body>

<footer>

>

<head>

## Sobre cores em CSS...

</head>

<body>

- Definindo a quantidade de cada cor **R G B**:
  - Podemos definir a "quantidade" de cada cor tanto em **HEXA** decimal quanto em **Decimal**, em ambos, o valor de cada cor vai de 0 à 255 (ou FF em hexadecimal)
  - Para "montarmos" um cor em hexadecimal utilizamos cerquilha (hashtag) "#", ex:
  - background-color: **#1F822D**;

Quantidade de vermelho: **1F**

Quantidade de verde: **82**

Quantidade de azul: **2D**

**Importante:** Podemos definir cores em hexadecimal com apenas três dígitos, exemplo: #333. Nesse caso, o 3 seria duplicado em cada cor, tendo o mesmo valor que #333333. Cores com um ou dois dígitos não são válidas, com quatro e cinco segue a mesma lógica que para três

<footer>

</footer>

<head>

## Sobre cores em CSS...

</head>

<body>

- Para definirmos uma cor em decimal devemos utilizar a função **rgb()**, exemplo:
  - background-color: **rgb(31,130,45)**;  
Quantidade de vermelho: **31**  
Quantidade de verde: **130**  
Quantidade de azul: **45**
- Há uma variação da função **rgb()**, a **rgba()**, onde, além de definirmos a quantidade de vermelho, verde e azul, definimos o "alpha" (a opacidade / transparência da cor) utilizando um número que vai de 0 a 1, exemplo:
  - background-color: **rgba(31,130,45,0.65)**;

</body>

<footer>

>

<head>

## Sobre cores em CSS...

</head>

<body>

- Por fim, além de definirmos cores pelo **nome**, pela **quantidade de RGB** (seja em decimal ou hexadecimal) também podemos definir cores pela posição da Matiz (**Hue**), intensidade de Saturação (**Saturation**) e Luminosidade (**Lightness**) utilizando a função **hsl()**, que é usada de maneira semelhante a **rgb()** e também possui a variação **hsla()**.

</body>

<footer>

>

</footer>



<head>

## Sobre cores em CSS...

</head>

<body>

- **Matiz (Hue):** posição, em graus (0 a 360) na roda de cores. 0 (ou 360) é vermelho, 120 é verde e 240 azul.
- **Saturação:** porcentagem (0 a 100) da intensidade da cor. O valor 100 é a cor “inteira”.
- **Luminosidade:** porcentagem (0 a 100) de preto/branco na cor, 0 é preto 100 é branco.
  - Exemplo:
    - `hsl(128,76,51);`

</body>

<footer>

>

</footer>

&lt;head&gt;

# Sobre medidas em CSS...

&lt;/head&gt;

&lt;body&gt;

## Unidades Absolutas

Nome	Unidade	Descrição
Centímetros	cm	Define o tamanho/largura/altura de um objeto, elemento, font, etc em centímetros.
Milímetros	mm	Define o tamanho/largura/altura de um objeto, elemento, font, etc em milímetros.
Polegadas	in	Define o tamanho/largura/altura de um objeto, elemento, font, etc em polegadas.
Pontos	pt	Define o tamanho/largura/altura de um objeto, elemento, font, etc em pontos (Unidade muito utilizada na impressão de textos e materiais, 72pt = 1in).

&lt;/body&gt;

&lt;footer&gt;

&gt;

&lt;/footer&gt;

&lt;head&gt;

# Sobre medidas em CSS...

&lt;/head&gt;

&lt;body&gt;

## Unidades Relativas

Nome	Unidade	Descrição
Pixels*	px	Número de pixels relativo a resolução do dispositivo de exibição.
Porcentagem	%	O tamanho/largura/altura de um objeto ou elemento é definido em porcentagem, exemplo: 10%, 25%, etc.
	em	Tamanho em relação a fonte atual. Ex: um elemento herda uma fonte com tamanho 20px e possui <b>border-width: 2em</b> , significa que a borda terá o tamanho de 40px (duas vezes 20px).

\*Pixels muitas vezes é colocado como uma unidade de medida relativa, pois, o tamanho dos elementos definidos utilizando pixels acabam sendo **proporcionais** a resolução da tela: se a resolução for maior, o objeto aparenta ser menor.

&lt;footer&gt;

&gt;

&lt;/footer&gt;

&lt;head&gt;

# Sobre medidas em CSS...

&lt;/head&gt;

&lt;body&gt;

## Unidades Relativas

Nome	Unidade	Descrição
Viewport Height	vh	Unidade relativa à altura da <b>viewport</b> do documento (vai de 1 a 100, 1vh = 1% da altura da viewport)
Viewport Width	vw	Unidade relativa à largura da <b>viewport</b> do documento (vai de 1 a 100, 1vw = 1% da largura da viewport)

**Importante:** Revisaremos as unidades relativas quando estivermos falando de layouts responsivos. Por hora, utilizaremos, principalmente, pixels, por ser uma unidade de medida mais simples e consequentemente didática.

## Mais informações sobre unidades de medida em CSS:

<http://www.w3.org/Style/Examples/007/units.html#units>

&lt;/body&gt;

&lt;footer&gt;

&gt;

<head>

# Combinar seletores

</head>

<body>

É possível especificar o seletor de **.classe** e **#ID** para uma *tag* específica, ex:

```
div#conteudo {  
    background-color: #CCC;  
    border-bottom: 4px double #000;  
}
```

```
p.paragrafo-maior {  
    font-size: 18px;  
    font-weight: bold;  
}
```

</body>

<footer>

>

<head>

# Seletores para elementos aninhados

</head>

<body> Se adicionarmos um **espaço** entre os seletores, a semântica "muda completamente":

```
div#conteudo {  
    ...  
}
```



Seleciona elemento **div** com o **id** **conteudo**:

```
<div id="conteudo">
```

```
div #conteudo {  
    ...  
}
```



Seleciona **qualquer elemento** com **id** **conteudo** que tenha um elemento **div** como **pai** (direto ou indireto):

```
<div>  
    <p id="conteudo"> </body>
```

<footer>

>

<head>

# Seletores para elementos aninhados

</head>

<body>

É possível aplicar estilo a elementos dentro de outros elementos, exemplo:

```
div#conteudo p.paragrafo-maior strong {  
    color: #F00;  
    font-weight: bold;  
}
```

Nesse caso, todos os p.paragrafo-maior que estiverem dentro do div#conteudo (direta ou indiretamente) receberão as propriedades.

</body>

<footer>

>

<head>

# Seletores para elementos aninhados

</head>

<body>

Se quisermos aplicar o estilo apenas aos elementos que sejam filhos imediatos utilizamos > (maior que) :

```
div#conteudo > p.paragrafo-maior {  
    color: #F00;  
    font-weight: bold;  
}
```

**Lista rápida com alguns seletores:**

[http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp)

</body>

<footer>

>

</footer>



<head>

## Onde definir os estilos?

</head>

<body>

- Basicamente, os estilos CSS podem ser definidos em três "locais" diferentes:
  - **Diretamente no corpo do documento HTML**, entre as tags **<style></style>**
  - Em um **arquivo externo** (geralmente .css), sendo o arquivo incluído no documento HTML por meio da tag **<link>**
  - E **"inline"**: diretamente no **elemento/tag HTML** por meio do atributo **style**

</body>

<footer>

>

</footer>

&lt;head&gt;

# Arquivo Externo

&lt;/head&gt;

&lt;body&gt;

Seja o arquivo ***estilo.css*** com o seguinte conteúdo:

## estilo.css

```
h1 {  
    background-color: #000;  
    color: #FFF;  
    font-size: 32px;  
    text-decoration: underline;  
}  
  
#conteudo {  
    background-color: #CCC;  
    border-bottom: 4px double #000;  
}  
  
.paragrafo-maior {  
    font-size: 18px;  
    font-weight: bold;  
}
```

&lt;/body&gt;

&lt;footer&gt;

&gt;

&lt;/footer&gt;

<head>

# Arquivo Externo

</head>

<body>

Incluimos a folha de estilos definida em um arquivo **externo** da seguinte forma:

```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="estilo.css" />
  </head>
  <body>
    ...
  </body>
</html>
```



Usamos o atributo **href** para definirmos o local/caminho e o nome do arquivo .css que queremos incluir em nosso documento HTML

</body>

<footer>

>

<head>

## Estilo inline (em linha)

</head>

<body>

Podemos aplicar estilo diretamente aos elementos utilizando o atributo **style**, é o que chamamos de atribuição **inline**. Exemplo:

```
<strong style="color: #F00; font-weight: normal;">  
    Lorem ipsum  
</strong>
```

</body>

<footer>

>

<head>

# Precedência de Seletores

</head>

<body>

No caso de **vários seletores** “apontarem” para o **mesmo elemento**, o que acontece?

</body>

<footer>

>

<head>

## Exemplo

</head>

<body>

**Qual será a cor de fundo do parágrafo?**

```
p {  
  background-color: #CCC;  
}
```

```
p.paragrafo {  
  background-color: #0CC;  
}
```

```
p#um {  
  background-color: #922;  
}
```

```
<p id="um" class="paragrafo">  
  Donec pellentesque vulputate eros  
</p>
```

</body>

<footer>

<head>

E se...

</head>

<body>

```
p {  
  background-color: #CCC;  
}
```

```
p.paragrafo {  
  background-color: #0CC;  
}
```

**E agora?**

```
p#um {  
  background-color: #922;  
}
```

```
<p id="um" class="paragrafo" style="background-color: #BB0">  
  Donec pellentesque vulputate eros  
</p>
```

</body>

<footer>

>

<head>

# Precedência de Seletores

</head>

<body>

- Para saber o resultado, analisa-se qual seletor tem maior precedência. Do maior para o menor, temos:

#id

.classe

elemento

Ainda, o estilo definido *inline (style)* tem maior prioridade que os seletores.

</body>

<footer>

>



<head>

# Propriedades de Texto

</head>

<body>

- **font-size**: tamanho da fonte do texto, ex: 18px
- **font-weight**: o "peso" da fonte (negrito) ex: bold  
<https://developer.mozilla.org/en-US/docs/Web/CSS/font-weight>
- **font-style**: estilos para o texto, ex: itálico  
<https://developer.mozilla.org/en-US/docs/Web/CSS/font-style>
- **text-decoration**: define algumas formatações para o texto, como *underline* e *line-through*.  
<https://developer.mozilla.org/pt-BR/docs/Web/CSS/text-decoration>
- **text-align**: alinhamento do texto, ex: esquerdo, direito, justificado, centro.  
<https://developer.mozilla.org/en-US/docs/Web/CSS/text-align>

</body>

<footer>

>

<head>

## Propriedades importantes

</head>

<body>

- Estudem as propriedades:
  - **Padding** – "margem interna" do elemento
  - **Margin** – "margem externa" do elemento
  - **Display** – Define o comportamento em relação a exibição do elemento na página

</body>

<footer>

>

</footer>

<head>

## Um pouco sobre **border**

</head>

<body>

- **border** é uma propriedade muitas vezes definida de forma composta, por exemplo:

1) **border**: *1px solid #FF0000*;

- Porém, é possível definir a borda de um elemento utilizando três propriedades:

2) **border-width**: *1px*;  
**border-style**: *solid*;  
**border-color**: *#FF0000*;

- Os trechos **1** e **2** são semanticamente equivalentes (produzem o mesmo efeito/resultado)

</body>

<footer>

>

<head>

## Um pouco sobre **border**

</head>

- Os dois exemplos do slide anterior aplicam as propriedades referentes a borda para todos os 4 lados de um elemento: superior, direito, inferior, esquerdo. Podemos definir, um estilo personalizado para cada um dos lados de um elemento utilizando `border-top`, `border-right`, `border-bottom` e `border-left`, exemplo:

```
border-top: 4px solid #FF0000;  
border-right: 2px dotted #0000FF;  
border-bottom: 4px dashed #b6bbff;  
border-left: none;
```

- Assim como fizemos na definição de todas as bordas de uma vez utilizando a propriedade **border**, também é possível definir tamanho, estilo e cor para a borda de cada lado, utilizando, por exemplo, **border-top-width**, **border-top-style** e **border-top-color**.

<footer>

>

</footer>

<head>

# Um pouco sobre background

</head>

<body>

- A propriedade background é utilizada para definir estilos referente ao fundo (background) dos elementos. Podendo essas propriedades serem definidas de forma composta, como por exemplo:

*background: #d1d4ff url('imagem.png') no-repeat scroll left center;*

- Que, individualmente equivale as propriedades:
  - background-color
  - background-image
  - background-repeat
  - background-attachment
  - background-position

</body>

<footer>

>

</footer>

<head>

## Exercício

</head>

<body>

- Aplique estilo a todos elementos da primeira página do exercício da primeira aula:
  - Deve haver **pelo menos um** seletor de elemento, um de classe e um id.
  - Aplicar **propriedades de borda e fundo** a cada um dos 4 **"blocos"** para **diferenciá-los** / destacá-los
  - Aplicar estilos de formatação de texto
  - Incluir via comentário em HTML nome completo e RGA do(s) aluno(s) no início de cada documento HTML

</body>

<footer>

>

</footer>