



**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**
RIO GRANDE DO SUL
Câmpus Feliz

CSS

Prof. Moser Fagundes

Programação III
ADS

Sumário

- Introdução
- Sintaxe
- Seletores e propriedades
- Exercícios

O que é o CSS?

- **CSS = Cascading Style Sheets**
- Serve para descrever **como** os elementos HTML são mostrados na nossa tela.
- Permite centralizar a configuração dos layouts.
- Permite controlar vários layouts de uma única vez.
- CSS externos são salvos em arquivos .css

CSS como solução

- HTML **nunca** teve como objetivo conter tags para formatar conteúdo:
 - HTML serve para **descrever** conteúdo!
- Na versão 3.2 do HTML, foram incluídos elementos como ****, entre outros...
 - Neste momento, começou o pesadelo...
- CSS removeu a formatação de estilo do HTML!

Sintaxe do CSS

Um conjunto de regras do CSS consiste de um **seletor** e um **bloco de declarações**.

- Seletor aponta para um elemento HTML
- Declarações são propriedades no formato **nome:valor;**

Seletor

h1

Bloco de declarações

{color:blue; font-size:12px;}

Seletores do CSS

Seletores podem ser usados para “encontrar” elementos com base no seu:

- *Nome*
- *Identificador*
- *Classe*
- *Atributos*

Seleção por nome do elemento

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
p {  
    text-align: left;  
    background-color: blue;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>
```

```
    <a id="grm" href="http://www.gremio.net">Gremio</a>
```

```
</p>
```

```
</body>
```

```
</html>
```

Seleção por identificador

Para designar uma classe no CSS, usamos o símbolo #

```
<!DOCTYPE html>
<html>
<head>
  <style>
    #grm {
      text-align: left;
      background-color: blue;
    }
  </style>
</head>
<body>
  <p>
    <a id="grm" href="http://www.gremio.net">Gremio</a>
  </p>
</body>
</html>
```


Seleção por classe

Para designar uma classe no CSS, usamos um ponto .

```
<!DOCTYPE html>
<html>
<head>
  <style>
    .times {
      background-color: yellow;
    }
  </style>
</head>
<body>
  <p>
    <a href="http://www.ifrs.edu.br">IFRS</a><br>
    <a class="times" href="http://www.gremio.net">Gremio</a><br>
    <a class="times" href="http://www.internacional.com.br">Inter</a>
  </p>
</body>
</html>
```

Como incluir um CSS?

Podemos incluir um CSS de 3 maneiras:

- *Inline*
- *Interno*
- *Externo*

Inline

Na abordagem inline o CSS vai dentro do elemento (tag) o qual visa formatar. Nesse caso, usamos o atributo **style**.

Exemplo:

```
<h2 style="color:red">Feliz!</h2>
```

Interno

No modo interno, a folha de estilo CSS vai declarada entre as tags `<head>` e `</head>`.

Exemplo:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
  p {
```

```
    text-align: left;
```

```
    background-color: blue;
```

```
  }
```

```
</style>
```

```
</head>
```

```
...
```

Externo

Nesta última maneira, colocamos o código CSS em um arquivo externo e colocamos uma referência para este arquivo entre as tags `<head>` e `</head>`.

Exemplo:

```
<!DOCTYPE html>  
<html>  
<head>
```

```
<link rel="stylesheet" type="text/css" href="meu_estilo.css">
```

```
</head>
```

```
...
```

Externo

Importante:

O arquivo **.css** não pode conter tags HTML!

Exemplo de arquivo **CSS** externo:

```
body{  
    background: black;  
}  
  
.times {  
    background-color: yellow;  
}
```

Ordem de prioridade

A seguinte ordem de prioridade é seguida quando temos estilos declarados em diferentes locais:

- **Inline** (maior prioridade);
- **Externo e interno** (entre eles, levamos em conta a ordem de declaração dentro do cabeçalho; levamos em conta o último estilo lido);
- **Default** (estilo definido pelo navegador).

Agrupamento de seletores

Suponha a seguinte situação em um CSS:

```
h1 {  
  color: blue;  
}
```

```
h2 {  
  color: blue;  
}
```

```
p {  
  color: blue;  
}
```


Agrupamento de seletores

Neste casos, podemos agrupar seletores que possuem as mesmas propriedades e valores.

Apenas temos que separar estes seletores com uma virgula.

```
h1, h2, p {  
    color: blue;  
}
```

Comentários

Os comentários no código CSS começam com `/*` e terminam com `*/`

Exemplo:

```
h1, h2, p {  
    color: blue;  
    /* Aqui vai o seu comentário */  
    text-align: left;  
}  
  
/* Seu comentário pode  
   ter várias linhas  
   como este.. */
```

Background

Para modificar o background de um elemento temos as seguintes propriedades a nossa disposição:

`background-color`

`background-image`

`background-repeat`

Background


A propriedade `background-color` serve para modificar a cor de fundo dos elementos. Por exemplo:

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      background-color: lightblue;
    }
  </style>
</head>
<body>
  <h1>Teste!</h1>
</body>
</html>
```

Background

A propriedade `background-color` serve para modificar a cor de fundo dos elementos. Por exemplo:

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      background-color: lightblue;
    }
  </style>
</head>
<body>
  <h1>Teste!</h1>
</body>
</html>
```



Troque a cor de background do elemento h1

Background


A propriedade `background-image` serve para trocar a imagem de fundo de um elemento.

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      background-image: url(fundo.jpg);
    }
  </style>
</head>
<body>
  <h1>Teste!</h1>
</body>
</html>
```

Background

A propriedade `background-image` serve para trocar a imagem de fundo de um elemento.

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      background-image: url(fundo.jpg);
    }
  </style>
</head>
<body>
  <h1>Teste!</h1>
</body>
</html>
```



Faça *download* de uma imagem de fundo e coloque a mesma como fundo do elemento `<h1>`.

Background

Se não declararmos o modo de repetição, a imagem de fundo se **repete** tanto horizontalmente como verticalmente.

Para mudar esta configuração, usamos a propriedade `background-repeat`, a qual pode assumir os seguintes valores:

- `no-repeat`
- `repeat-x`
- `repeat-y`

Background

Exemplo:

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      background-image: url(fundo.jpg);
      background-repeat: repeat-x;
    }
  </style>
</head>
<body>
  <h1>Teste!</h1>
</body>
</html>
```

Bordas

Podemos especificar o **estilo**, **largura** e **cor** das bordas dos elementos HTML. Para tal, usamos as seguintes propriedades:

`border-style`

`border-width`

`border-color`

Bordas

A propriedade `border-style` pode assumir os seguintes valores:

`dotted`

`solid`

`groove`

`inset`

`none`

`dashed`

`double`

`ridge`

`outset`

`hidden`

Adicione uma borda ao elemento `<h1>` do exemplo anterior.

Bordas

A largura da borda é definida pela propriedade `border-width` e podemos definir larguras diferentes para cada lado do elemento.

Por exemplo:

```
border-width: 2px;
```

ou

```
border-width: 2px 10px 4px 20px;  
/* topo direita embaixo esquerda */
```

Margens

As margens no CSS são usadas para gerar espaços **ao redor** dos componentes HTML.

*O espaço gerado pela margem é **fora** (ao redor) da borda do componente.*

As margens são completamente transparentes e eles não podem ter cor de background.

Margens

As margens são definidas usando as propriedades:

`margin-top`

`margin-right`

`margin-bottom` e

`margin-left`

Alternativamente, podemos usar somente a propriedade `margin` que define o mesmo valor para todas as margens.

Margens

Exemplos:

```
margin-top: 20px;  
margin-right: 30px;  
margin-bottom: 20px;  
margin-left: 30px;
```

ou

```
margin: 50px;
```

ou

```
margin: 20px 30px 20px 30px;  
/* topo direita embaixo esquerda */
```

Margens

Exemplo:

```
<!DOCTYPE html>
<html>
<head>
  <style>
    p.m1 {
      margin: 50px;
      border: solid;
    }
    p.m2 {
      margin: 200px 200px 200px 100px;
      border: solid;
    }
  </style>
</head>

/* Arquivo completo no Moodle */
```


Padding

O **padding** no CSS é usado para gerar espaços **ao redor do conteúdo** dos componentes HTML.

*O espaço gerado pelo padding se localiza **entre o conteúdo e a borda** do componente.*

O padding é afetado pela cor de background do elemento!

Padding

O padding é definido usando as propriedades:

`padding-top`

`padding-right`

`padding-bottom` e

`padding-left`.

Alternativamente, podemos usar somente a propriedade `padding` que define o mesmo valor para todos os lados.

Padding

Exemplos:

```
padding-top: 20px;  
padding-right: 30px;  
padding-bottom: 20px;  
padding-left: 30px;
```

ou

```
padding: 50px;
```

ou

```
padding: 20px 30px 20px 30px;  
/* topo direita embaixo esquerda */
```

Exemplo:

Padding

```
<!DOCTYPE html>
<html>
<head>
  <style>
    p.m1 {
      margin: 50px;
      padding: 100px;
      border: solid;
      background-color: gray;
    }
    p.m2 {
      margin: 200px 100px 200px 100px;
      padding: 30px 50px 30px 50px;
      border: dotted;
      background-color: yellow;
    }
  </style>
</head>

/* Arquivo completo no Moodle */
```

Margem e padding com %

Podemos definir os valores das margens e do padding em função do % das dimensões dos componentes, conforme exemplo a seguir:

```
<style>
  p.m1 {
    margin: 10%;
    padding: 10%;
    border: solid;
    background-color: gray;
  }
  /* Nesse caso, o tamanho da margem e do
  padding mudam se redimensionamos a pagina */
</style>
```

Dimensões: altura e largura

Para definir a **largura** de um elemento, usamos a propriedade `height`. Para definir a sua **altura**, usamos a propriedade `width`.

Se não especificarmos estas propriedades, é o navegador que calcula (nesse caso o valor da propriedade é `auto`).

Texto

Quanto as fontes, podemos mudar os atributos:

`color`

`text-align`

`text-decoration`

`text-transform`

`text-indent`

`letter-spacing`

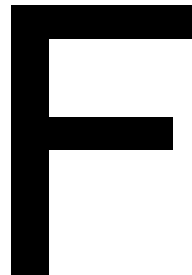
`line-height`

`word-spacing`

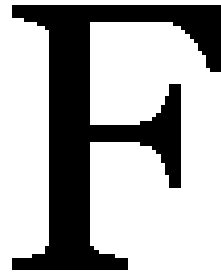
Fontes

As fontes no CSS podem ser usadas para definir propriedades como a família da fonte, tamanho e o estilo de um texto.

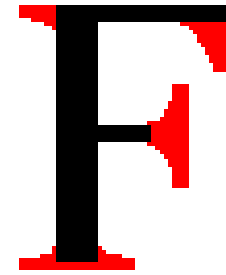
Serif versus Sans Serif



Sans-serif



Serif



Serif
(red serifs)

Fontes

Exemplos fontes segundo a sua família:

Serif

Times New Roman
Georgia

Sans-serif

Arial
Verdana

Monospace

Courier New
Lucida Console

* além das famílias acima, temos **cursive** e **fantasy**

Fontes

A propriedade `font-family` pode conter vários nomes de fontes, de modo que se o navegador não suporta a primeira fonte da lista, ele tenta a segunda e assim sucessivamente.

Caso o nome da fonte tenha mais de uma palavra, então ela deve vir cercada de aspas, como por exemplo: `"Times New Roman"`

Fontes

Outras propriedades das fontes:

`font-size`

`font-style`

`font-weight`

Links

Os links podem ter o seu estilo configurado de acordo com os seus estados, que são:

`a:link /* normal, sem ter sido visitado /*`

`a:visited /* link visitado */`

`a:hover /* quando passamos o mouse sobre */`

`a:active /* link no momento do click */`

Outros Exemplos

- Como formulários são compostos de tags HTML, é possível personalizar os mesmos usando CSS.

Exemplo_Slide_45.html

- Menu com o framework Bootstrap

Exemplos-Menu-Horizontal.rar

Exercícios

Lista 3 no Moodle.

Referências

- Seletores

https://www.w3schools.com/cssref/css_selectors.asp

- Unidades

https://www.w3schools.com/cssref/css_units.asp