

Introdução ao CSS

Disciplina: Desenvolvimento Web

Prof. Dr. Rafael Will M. de Araujo



Sumário

- Apresentar as folhas de estilo - CSS
- Como funciona e como usar
- Sintaxe básica
- Como colorir, transformar, modificar os elementos HTML

Conteúdo

1 Introdução

2 Sintaxe e seletores

3 Mais estilos

4 Exercício

Cascading Style Sheets: o que são?

- CSS significa *Cascading Style Sheets* (em português: Folhas de Estilo em Cascata)
- Definem como os elementos HTML são exibidos;
- Estilos são normalmente armazenados em Folhas de Estilo (arquivos com extensão css);
- Folhas de estilo externo auxiliam na economia de tempo e trabalho do desenvolvedor;
- Múltiplos estilos podem ser definidos em cascata dentro de um único documento;

Vantagens do CSS

- Facilidade de manutenção;
- Novas possibilidades de apresentação visual;
- Criação de sites *tableless* (sem auxílio de tabelas para organizar os itens na tela);
- Diminuição do tempo de *download*;
- Facilidade de expansão de funcionalidades;

Como o CSS é utilizado?

- Como as folhas de estilos podem ser especificadas:
 - ▷ dentro de um elemento HTML (*inline*)
 - ▷ dentro do elemento `<head> </head>` de uma página HTML (*interno*);
 - ▷ em um arquivo CSS *externo*;
- É possível especificar mais de um estilo

Folha de Estilo *inline* (dentro do elemento)

- Usado para aplicar uma regra apenas para uma *tag* (um elemento) de uma página específica;
- Definido na propriedade **style**;

Exemplo 1 de CSS *inline* (dentro do elemento)

```
1 <body style="background-color: yellow">  
2   <h1> Olá Mundo!</h1>  
3 </body>
```

Exemplo 2 de CSS *inline* (dentro do elemento)

```
1 <body style="background-color: yellow">  
2   <h1 style="color: red"> Olá Mundo!</h1>  
3 </body>
```

- Estilos *inline* possuem prioridade sobre as outras maneiras de especificação de estilos.

Folha de Estilo interna

- Usado geralmente quando queremos definir um estilo que será utilizado em apenas uma página;
- Definido na seção `<head>` `</head>`, através da tag `<style>` `</style>`;

Definindo uma folha de estilo interna

```
1 <html>
2   <head>
3     <style type="text/css">
4       body {background-color: yellow}
5     </style>
6   </head>
7   <body>
8     <h1> Olá Mundo!</h1>
9   </body>
10 </html>
```


Folha de Estilo Externa

- Ideal quando desejamos aplicar o mesmo estilo em várias páginas;
- Cada página deve ter um vínculo (referência) com a folha de estilo;
 - ▷ Construído através da tag `<link>`
- É capaz de alterar todo o visual de uma página mudando-se apenas a referência para um arquivo CSS;

Arquivo *meu_estilo.css*

```
1 body {background-color: yellow}
2 h1 {color: red}
```

Página HTML

```
1 <html>
2   <head>
3     <link rel="stylesheet" type="text/css" href="meu_estilo.css">
4   </head>
5   <body>
6     <h1> Olá Mundo!</h1>
7   </body>
8 </html>
```

A tag link

- Note os atributos do marcador `<link>`:
 - ▷ **rel**: especifica o relacionamento entre o documento HTML e o documento ligado. Para folhas de estilo, o valor do atributo é `"stylesheet"`;
 - ▷ **type**: especifica o tipo do documento. Para CSS, o tipo é `text/css`;
 - ▷ **href**: especifica a localização do arquivo CSS. Pode ser um recurso local ou um recurso externo;
- No marcador `link` apenas o atributo **rel** é obrigatório. Mas para inserir uma folha de estilo, precisamos colocar a referência (através do atributo **href**).

Conteúdo

1 Introdução

2 Sintaxe e seletores

3 Mais estilos

4 Exercício

Sintaxe CSS

- A sintaxe CSS é definida em 3 partes: um **seletor**, uma **propriedade** e um **valor**.

Sintaxe básica do CSS

```
1  seletor {  
2      propriedade: valor;  
3  }
```

- Um seletor simples pode ser um marcador HTML, uma **classe** (atributo *class*) ou um identificador (atributo *id*).

Exemplo de seletor simples para a *tag* body

```
1  body {  
2      background: yellow;  
3  }
```

Classe de Estilo

- Classes de estilo permitem definir diferentes estilos para o mesmo tipo de elemento HTML;
- No arquivo CSS, o nome de uma classe é identificada com um ponto (.)
- **Exemplo:** Dois tipos de parágrafos em uma página Web.

Arquivo CSS

```
1  p.right {  
2      text-align: right;  
3  }  
4  p.center {  
5      text-align: center  
6  }
```

Página HTML

```
1  <p class="right">Alinhado à direita</p>  
2  <p class="center">Centralizado </p>  
3  <p class="right">Outro texto diferente alinhado à direita</p>
```

- O atributo **class** é usado para definir qual classe (categoria) de estilo será utilizada no marcador. Podemos usar a mesma classe em vários elementos diferentes (assim, eles vão pertencer à mesma classe).

Classe de Estilo

- É possível omitir o nome do elemento no seletor:

Exemplo de classe de estilo sem *tag* específica

```
1 .center {text-align: center}
```

- Neste caso todo marcador com a classe center terá o estilo aplicado:

Página HTML

```
1 <h1 class="center">Cabeçalho centralizado</h1>  
2 <p class="center">Parágrafo também centralizado.</p>
```

Outras observações sobre o uso de classes

- O nome da classe e o valor da propriedade **class** não podem iniciar com números;
- Não utilizar espaços para um nome de classe;
- Um marcador HTML pode receber mais de uma classe. Para isso, separa-se cada uma das classes por espaços, dentro do valor do atributo **class**;

Seletor por *id*

- É possível definir estilo para elementos HTML usando o seletor *id*. Ele é definido pelo caractere cerquilha (#).

Exemplo 1 de seletor por *id*

```
1 #titulo-principal { color: green }
```

Página HTML

```
1 <h1 id="titulo-principal">Esse é o título principal</h1>
2 <p>Texto do título principal</p>
3 <h1>Outro título</h1>
4 <p>Texto do segundo título</p>
```


Seletor por *id*

- A regra do estilo abaixo definirá que um elemento p com um id="para1" terá cor vermelha, e outro com id="para2" terá cor azul:

Exemplo 2 de seletor por *id*

```
1 p#para1 { color: red }  
2 p#para2 { color: blue }
```

Página HTML

```
1 <p id="para1">Texto em vermelho</p>  
2 <p id="para2">Texto em azul</p>
```

Agrupamento de seletores

- Também é possível agrupar seletores.
 - ▷ Para agrupar seletores basta separá-los por vírgulas;
 - ▷ O estilo dentro das chaves é aplicado a todos os seletores declarados;

Exemplo de agrupamento de seletores

```
1 h1, h2, h3, h4, h5, h6 {  
2   color: green;  
3 }
```

Página HTML

```
1 <h1>Exemplo</h1>  
2 <h2>Exemplo</h2>  
3 <h3>Exemplo</h3>  
4 <h4>Exemplo</h4>  
5 <h5>Exemplo</h5>  
6 <h6>Exemplo</h6>
```

Selecionando elementos dentro de outros elementos

- É possível selecionar elementos que estão dentro de outro elemento (divs, sections, articles, navs, header, footer, main, etc).

Exemplo

```
1 nav a{
2     text-decoration: none;
3     color: red;
4 }
```

Página HTML

```
1 <nav>
2     Texto dentro da nav<br>
3     <a href="http://www.google.com">Link 1 dentro da nav</a><br>
4     <a href="http://www.bing.com">Link 2 dentro da nav</a><br>
5 </nav>
6 <a href="http://www.google.com">Link 1 fora da nav</a><br>
7 <a href="http://www.bing.com">Link 2 fora da nav</a>
```

- Note que somente os links dentro da tag nav ficarão na cor vermelha.
- Mais detalhes sobre seletores: www.w3schools.com/cssref/css_selectors.asp

Adicionando comentários

- Comentários podem ser adicionados ao arquivo CSS usando os marcadores `/* */`.

Exemplo

```
1 nav a{ /* deixa os links dentro do nav na cor vermelha */
2     text-decoration: none;
3     color: red;
4 }
5
6 a { /* demais links serão verdes */
7     color: green;
8 }
```

Página HTML

```
1 <nav>
2     Texto dentro da nav<br>
3     <a href="http://www.google.com">Link 1 dentro da nav</a><br>
4     <a href="http://www.bing.com">Link 2 dentro da nav</a><br>
5 </nav>
6 <a href="http://www.google.com">Link 1 fora da nav</a><br>
7 <a href="http://www.bing.com">Link 2 fora da nav</a>
```

Valores com múltiplas palavras

- Se o valor é composto de múltiplas palavras, deve ser colocado aspas entre o valor:

Exemplo

```
1 p {  
2   font-family: "sans serif";  
3 }
```

- Se desejar especificar mais de uma propriedade, estas devem ser **separadas por ponto e vírgula**. Para tornar seu estilo mais legível, **descreva cada propriedade em cada linha**:

Exemplo

```
1 p {  
2   text-align: center;  
3   color: black;  
4   font-family: arial;  
5 }
```

Pseudo-classes e pseudo-elementos

- As **pseudo-classes** e **pseudo-elementos** são usados para adicionar efeitos especiais em alguns seletores.
 - ▷ Em resumo, eles permitem especificar um estilo específico para um estado especial do elemento selecionado.

Exemplo

```
1 selector:pseudo-class {  
2     property: value;  
3 }  
  
4 selector.class:pseudo-class {  
5     property: value  
6 }
```

- Alguns exemplos de pseudo-classes:

- ▷ :link
- ▷ :visited
- ▷ :hover
- ▷ :active
- ▷ :focus
- ▷ :first-letter
- ▷ :first-line
- ▷ :first-child
- ▷ :before
- ▷ :after

Pseudo-classes e pseudo-elementos

- Exemplo 1:

Exemplo 1 de pseudo-classes sobre *hyperlinks*

```
1 a:link{ color: red}
2 a:visited{ color: green}
3 a:hover{ color: magenta}
4 a:active{ color: blue}
```

Página HTML

```
1 <a href="#">Exemplo de link</a>
```

- Exemplo 2:

Exemplo 2 de pseudo-classes

```
1 input:focus {background-color: red}
2 p:first-letter{color: green}
3 p:first-line{color: blue}
4 p:first-child{background-color: yellow}
```

Página HTML

```
1 <input type="text">
2 <div>
3   <p>
4     Primeira letra verde
5     Fundo amarelo
6   </p>
7   <p>
8     Primeira linha azul<br>
9     Segunda linha normal
10  </p>
11 </div>
```

Conteúdo

1 Introdução

2 Sintaxe e seletores

3 Mais estilos

4 Exercício

Cores

- Existem 16.777.216 cores disponíveis no CSS.
- Algumas das cores mais comuns possuem nomes pré-definido:
 - ▷ aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, yellow, ...
- Também podem ser definidas por valores RGB:
 - ▷ A cor **red** é o mesmo que `rgb(255,0,0)`, o mesmo que `rgb(100%,0%,0%)`.
- Ou por um código hexadecimal:
 - ▷ A cor **red** é o mesmo que `#ff0000` e o mesmo que `#f00`.
- As cores podem ser aplicadas a um texto usando as propriedades **color** e **background-color**.

Exemplo

```
1 h1 {
2   color: #ffc;
3   background-color: #009;
4 }
5
6 p {
7   color: rgb(255, 0, 0);
8   background-color: rgb(0%, 100%, 100%);
9 }
```

Página HTML

```
1 <h1>Título</h1>
2 <p>Parágrafo</p>
```

Plano de fundo

- Também é possível definir imagens para fundos de elementos HTML:

Exemplo de estilo que define um plano de fundo

```
1 body{
2   background-image: url('imagem_fundo.png');
3 }
```

- Para que imagem não se repita, incluímos a propriedade **background-repeat** com o valor *no-repeat*:

Exemplo de estilo que define um plano de fundo sem repetição

```
1 body{
2   background-image: url('imagem_fundo.png');
3   background-repeat: no-repeat;
4 }
```

Unidades de tamanho

- Em CSS existem várias formas de especificar tamanhos:
 - ▷ **px**: é a unidade para pixel (ponto na tela);
 - ▷ **em**: é um elemento com tamanho x vezes da fonte atual do elemento pai. Se a fonte do elemento pai possui tamanho 12px, 2em terá 24px;
 - ▷ **rem**: funciona da mesma maneira que a anterior, mas a sua referência é o elemento raiz do documento;
 - ◇ Essa referência pode ser definida pelo usuário nas configurações do navegador.
 - ▷ **%**: é a unidade para porcentagem;
 - ▷ Demais unidades: https://www.w3schools.com/cssref/css_units.asp
- Se nenhuma unidade for informada, o padrão é *px*

Exemplo de estilos com unidades de tamanho

```
1 p.porc { font-size: 70%; }
2 p.px { font-size: 28px; }
3 p.pt { font-size: 16pt; }
4 p.em { font-size: 5em; }
```

Página HTML

```
1 <p>Texto</p>
2 <p class="porc">Texto</p>
3 <p class="px">Texto</p>
4 <p class="pt">Texto</p>
5 <p class="em">Texto</p>
```

Estilos de texto

- Os atributos que permitem mudar a forma como o texto é representado são: **font-family, font-size, font-weight, font-style, text-decoration, text-transform;**
- **font-family** indica a fonte a ser usada;
- Algumas observações importantes:
 - ▷ A fonte usada precisa estar presente (instalada) no computador do usuário. Por isso não vale a pena usar fontes “obscuras”;
 - ▷ Podemos especificar várias fontes separadas por vírgulas. O navegador usa a primeira que o utilizador tenha;
 - ▷ Se o nome da fonte tiver mais de uma palavra é necessário usar aspas;
 - ▷ O último valor desta propriedade deverá ser uma classe de fonte mais genérica como: **serif , sans serif, cursive, fantasy, monospace.**

Estilos de texto

- **font-size**: indica o tamanho da fonte. As unidades utilizadas são as mesmas mostradas anteriormente;
- **font-weight**: indica se a fonte é negrito ou não. Valores possíveis: *normal*, *bold*, *bolder*, *lighter*, valores de 100 a 900 (aumentando de 100 em 100).
- **font-style**: indica se o texto deve ser escrito em itálico ou não. Valores: *normal*, *italic*, *oblique*.
- **text-decoration**: indica se o texto deve ser sublinhado ou não. Valores: *none*, *underline*, *overline*, *line-through*.
- **text-transform**: muda a capitalização das letras. Valores: *none*, *capitalize*, *uppercase*, *lowercase*.

Exemplo

```
1 .titulo{
2   font-family: Verdana, Arial, Helvetica, "sans serif";
3   font-size: 16px;
4   font-weight: bold;
5   font-style: normal;
6   color: rgb(245, 100, 50);
7   text-decoration: underline;
8   text-transform: capitalize;
9 }
```

Página HTML

```
1 <p class="titulo">Texto de exemplo</p>
```

Alinhamento do texto

- A propriedade **text-align** permite alinhar o texto dentro de um elemento (atenção: não alinha o elemento em si). Os valores possíveis são: *center*, *left*, *right*, *justify*;
- A propriedade **text-indent** permite indentar a primeira linha de um parágrafo. Os valores desta propriedade podem ser um comprimento ou uma percentagem.

Exemplo

```
1 .direita{
2   text-align: right;
3 }
4
5 .esquerda{
6   text-align: left;
7 }
8
9 .centro{
10  text-align: center;
11 }
12
13 .paragrafo-indentado{
14   text-indent: 30px;
15 }
```

Página HTML

```
1 <p class="direita">Direita</p>
2 <p class="esquerda">Esquerda</p>
3 <p class="centro">Centro</p>
4 <p class="paragrafo-indentado">Lorem ipsum dolor sit amet, consectetur adipiscing elit,
5 sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
```

Espaçamento do texto

- Existem ainda propriedades que permitem mudar o espaçamento entre as várias componentes do texto (letras, palavras, linhas, parágrafos, etc);
- Os atributos **letter-spacing** e **word-spacing** permitem mudar o espaçamento entre letras ou entre palavras. Os valores permitidos são um comprimento ou normal;
- O atributo **line-height** altera a altura de uma linha sem alterar o tamanho da fonte. Os valores permitidos são um comprimento, uma percentagem ou normal.

Exemplo

```
1 .letras{
2   letter-spacing: 10px;
3   background-color: red;
4 }
5
6 .palavras{
7   word-spacing: 10px;
8   background-color: red;
9 }
10
11 .linha{
12   line-height: 50px;
13   background-color: red;
14 }
```

Página HTML

```
1 <p class="letras">Texto</p>
2 <p class="palavras">Texto texto texto</p>
3 <p class="linha">Texto</p>
```

Bordas

- A propriedade **border** permite a aplicação de borda ao *model box*;
- O valor da propriedade é dividido em três partes, definido nessa ordem:
 - ▷ **width**: largura da borda
 - ▷ **style**: tipo da borda
 - ▷ **color**: cor da borda

Exemplo

```
1 seletor{  
2     border: width style color;  
3 }
```


Bordas

- Também é possível se referir à borda diretamente ao lado do **model box** e a propriedade que deve ser alterada:
 - ▷ *border-top-width, border-right-width, border-bottom-width, border-left-width;*
 - ▷ *border-top-style, border-right-style, border-bottom-style, border-left-style;*
 - ▷ *border-top-color, border-right-color, border-bottom-color, border-left-color;*
- **border-width:** recebe valores numéricos
- **border-style:** pode receber os seguintes valores: *solid, dotted, dashed, double, groove, ridge, inset, outset;*
- **border-color:** recebe cores;

Bordas

Exemplo

```
1 .bordas {  
2   border-top-width: 2px;  
3   border-top-color: black;  
4   border-top-style: dashed;  
  
5   border-right-width: 2px;  
6   border-right-color: orange;  
7   border-right-style: solid;  
  
8   border-bottom-width: 2px;  
9   border-bottom-color: yellow;  
10  border-bottom-style: double;  
  
11  border-left-width: 2px;  
12  border-left-color: red;  
13  border-left-style: dotted;  
14 }
```

Página HTML

```
1 <div class="bordas">  
2   div comum com bordas estilizadas  
3 </div>
```

Bordas arredondadas

- Agora é possível definir bordas arredondadas para os elementos, utilizando a propriedade **border-radius** (raio da borda);
- Da mesma forma que a border, a **border-radius** pode ser definida genericamente ou por orientação:
 - ▷ border-top-left-radius
 - ▷ border-top-right-radius
 - ▷ border-bottom-right-radius
 - ▷ border-bottom-left-radius
- Também é possível definir em uma mesma linha, com uma combinação de 4, 3, 2 ou 1 valor:
 - ▷ *border-radius:* esquerda-cima direita-cima direita-baixo esquerda-baixo
 - ▷ *border-radius:* esquerda-cima direita-cima+esquerda-baixo direita-baixo
 - ▷ *border-radius:* esquerda-cima+direita-baixo direita-cima+esquerda-baixo
 - ▷ *border-radius:* esquerda-cima+direita-baixo+direita-cima+esquerda-baixo

Bordas arredondadas

Exemplo

```
1  p {
2    background: lime;
3    padding: 10px;
4    width: 100px;
5    height: 75px;
6  }
7
8  p.quatro-valores {
9    border-radius: 15px 50px 30px 5px;
10 }
11
12 p.tres-valores {
13   border-radius: 15px 50px 30px;
14 }
15
16 p.dois-valores {
17   border-radius: 55px 30px;
18 }
19
20 p.um-valor {
21   border-radius: 50px;
22 }
```

Página HTML

```
1  <p class="quatro-valores">Quatro valores</p>
2  <p class="tres-valores">Três valores</p>
3  <p class="dois-valores">Dois valores</p>
4  <p class="um-valor">Um valor</p>
```

Conteúdo

1 Introdução

2 Sintaxe e seletores

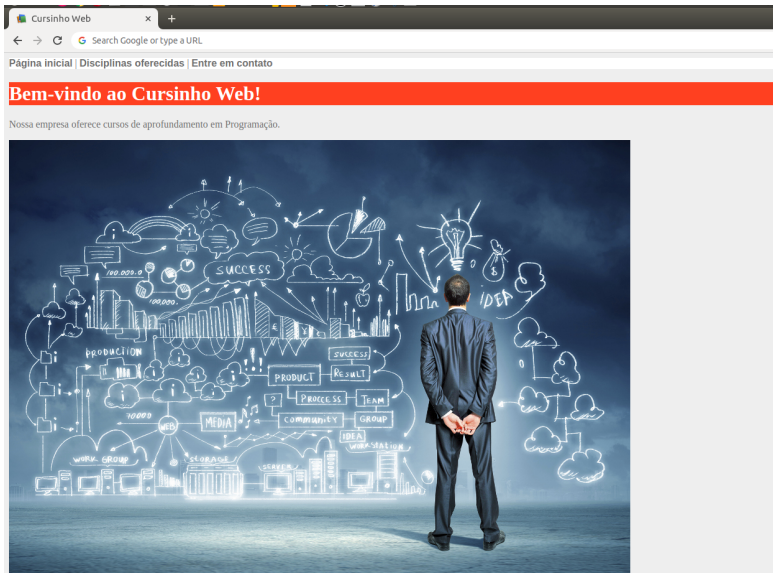
3 Mais estilos

4 Exercício

Exercício

- Construa o arquivo “estilos.css” e adicione-o às páginas “index.html” e “disciplinas.html” através da tag `<link>`.
- Esse arquivo CSS deve conter as seguintes regras (utilize os seletores adequados para implementá-las:
 - ▷ o body deve ter cor de fundo #eeeeee e cor do texto #6f6f6f;
 - ▷ o header e o footer devem usar o branco como cor de fundo;
 - ▷ todo h1 deve ter cor de fundo #ff4020 e cor do texto branco;
 - ▷ todo link dentro de um nav deve ter:
 - ◇ tamanho do texto 1rem
 - ◇ tipos de fonte: Arial, Helvetica, sans serif
 - ◇ negrito
 - ◇ sem sublinhado
 - ▷ além disso, todo link dentro de um nav deve ter os seguintes efeitos:
 - ◇ cor padrão e visitado: #6f6f6f;
 - ◇ ao colocar o mouse em cima dos links, devem mudar para a cor #ff4020 e ficar sublinhados;
 - ◇ enquanto o mouse estiver pressionado sobre o link, ele deve ter cor azul e ficar sublinhado;
 - ▷ os elementos h2 da página “disciplinas.html” (os títulos das disciplinas) devem pertencer à mesma classe, chamada “titulo-article” (modifique o HTML para que isso ocorra). Para esta classe, defina uma regra com o seguinte:
 - ◇ cor do texto *royalblue*
 - ▷ o link do e-mail no rodapé deve ter um id único, chamado “link-email” (modifique o HTML para que isso ocorra), com os seguintes efeitos:
 - ◇ a cor padrão do link, e também se ele já for visitado ou o mouse estiver pressionado sobre ele será azul e ele não deve ser sublinhado;
 - ◇ quando o mouse estiver em cima deste link, sua cor deve mudar para #ff4020 e ele deve ficar sublinhado;

Exercício: *index.html*



Cursinho Web - Todos os direitos reservados

Contato: email@provedor.com

Exercício: *disciplinas.html*

Cursinho Web - Disciplinas

← → ↻ Search Google or type a URL

[Página inicial](#) | [Disciplinas oferecidas](#) | [Entre em contato](#)

Disciplinas oferecidas:

Desenvolvimento Web

Conhecer os conceitos de desenvolvimento de sistemas dinâmicos para a Internet, aplicando as mais recentes tecnologias e métodos para esta área.

Habilidades

- HTML
- CSS
- JavaScript
- Python e Flask

HTML


CSS

JS

Programação Orientada a Objetos

Desenvolver lógica de raciocínio para organizar os elementos necessários para o desenvolvimento de programas orientado a objetos.

- Dicionários
- Tratamento de exceções
- Testes unitários
- Construção de classes, atributos e métodos
- Herança e polimorfismo

python

Cursinho Web - Todos os direitos reservados

Contato: email@provedor.com