



HTML para Concursos Públicos

Aula 06 – CSS – Parte 1

A 3D sphere with various web technologies and file formats written on it. The word 'css' is prominently displayed in the center in a large, orange, 3D font. Other words like 'html', 'php', 'javascript', 'css', 'www', 'pdf', 'ajax', 'xhtml', 'seo', 'cms', 'sql', 'web', 'jpg', 'png', 'svg', 'xml', 'json', 'yaml', 'json', 'xml', 'yaml', 'json', 'xml', 'yaml' are scattered across the sphere's surface in a smaller, grey, sans-serif font. The sphere is set against a dark, gradient background.

Iremos estudar nesta aula, como é possível usar a CSS para dividirmos as responsabilidades no desenvolvimento web: enquanto alguns profissionais trabalham o conteúdo de um *site*, outros (com um perfil mais artístico) podem se dedicar apenas à apresentação deste conteúdo.

➤ Atributos de Estilo

Nos primórdios da HTML, podíamos incluir atributos de estilo em *tags* HTML. Porém, isto é uma prática **NÃO** recomendada pela W3C.



Isto quer dizer que os estilos (por exemplo: fonte, cor, posicionamento etc) que regem a APRESENTAÇÃO do conteúdo de um arquivo HTML, podem estar incluídas nas **próprias** *tags* HTML.

Isto resultou num grande problema, pois toda vez que uma regra de apresentação tivesse que ser modificada, precisaríamos alterar vários arquivos HTML, procurar as *tags* cujos atributos de estilo tivessem que ser alterados.

A CSS veio para resolver este problema, ao separar conteúdo da apresentação.

➤ Tipos de CSS

É possível definir estilos (às vezes chamados “regras” de estilo) em três níveis:

- + no próprio elemento HTML (chamado também de *inline*);
- + na seção `<head>` do arquivo HTML (chamada **CSS interna**);
- + ou em arquivos externos (chamada **CSS externa**).

No próprio elemento HTML (**não recomendado**): você pode definir estilos usando o atributo `style` do elemento HTML, este tipo de CSS é também chamado de *inline*. Por exemplo, para definir que um texto de um parágrafo deve ter a cor verde:

```
<p style="color:green">Este parágrafo ficará na cor verde!</p>
```

A propriedade do estilo (`color`) vem em primeiro lugar, depois o símbolo `:` (dois pontos) e por fim o valor da propriedade (`green`).

Este tipo de CSS refere-se APENAS ao elemento onde está inserido. Ou seja, NÃO afetará os outros elementos.

➤ Outros tipos de CSS

Também é possível definir o estilo numa *tag* chamada `<style>` que deve ficar dentro do elemento `<head>` da página HTML (**CSS Interna**). Por exemplo:

```
<html>
  <head>
    <style> p { color : green }
  </style>
</head>
.....
</html>
```



Este estilo, delimitado pelas *tags* `<style>` e `</style>`, definem que todos os parágrafos (*tags* `<p>`) do arquivo HTML ficarão com a cor verde.

Porém, a forma mais recomendada é usar arquivos externos (**CSS Externa**). Nestes arquivos escrevemos várias definições de estilo. Para referenciar tais arquivos, usamos a tag `<link>` da seguinte forma:

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

A tag `<link>` fica dentro do elemento `<head>`.

Os atributos `rel` e `type` definem o *link* como sendo do tipo texto e de estilos.

O atributo `href` define o nome do arquivo onde estão os estilos.

➤ Estrutura de Estilos

Geralmente, um estilo dentro de um arquivo externo ou dentro da *tag* `<style>` assume a seguinte estrutura:

```
seletor {  
    propriedade1 : valor;  
    propriedade2 : valor;  
    ... outras propriedades ...  
}
```

Analizando o nosso primeiro exemplo de estilo...

```
<html>  
  <head>  
    <style> p { color : green }  
  </style>  
  </head>  
  .....  
</html>
```



```
p {  
    color : green;  
}
```

Aqui estamos selecionando o elemento `<p>`, ou seja, o elemento **parágrafo**. Neste exemplo, ele é o **Seletor**.

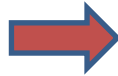
Veremos mais adiante que este **Seletor** pode ser um elemento HTML ou vários diferentes.

A propriedade é `color` e o seu valor é `green`.

➤ Mais estrutura de Estilos

É possível definir o mesmo estilo para vários elementos HTML.

```
h1, h2 {  
    color : gray;  
}
```



Neste exemplo, aplicamos um estilo, que é cor da fonte `gray`, ou seja, cinza, para os elementos `<h1>` e `<h2>`, que são separados por vírgula.

Você pode definir outro estilo de novo para o mesmo elemento.

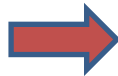
```
h1, h2 {  
    color : gray;  
}  
h1 {  
    background-color: red;  
}
```



Neste exemplo os títulos `<h1>` e `<h2>` terão suas fontes de cor cinza, mas somente o título `<h1>` terá a cor de fundo igual a vermelho (`red`).

Os elementos que estão dentro de um elemento (aninhados) cujo estilo é definido, recebem também estes mesmos valores de estilo.

```
body {  
    color : gray;  
}
```



Neste exemplo todos os elementos que estiverem dentro da *tag* `<body>` herdarão a propriedade `color`, ou seja, suas fontes terão a cor cinza.

➤ Prioridade de Estilos: a “Cascata”

Já que podemos definir estilos em três níveis, se tivermos estilos conflitantes, ou seja, referenciando o mesmo elemento, qual prevalecerá?

Resposta: A regra de estilo mais específica é a que será usada! Mas o que é ser mais específica?

Os arquivos CSS externos são os menos específicos, logo, são os que têm as regras de menor prioridade.



Logo depois, vêm as regras que estão na *tag* `<style>` dentro da seção `<head>` do arquivo HTML.



As regras *inline* são as mais específicas, logo, elas tem maior prioridade!

Daí, finalmente, chegamos a começar a entender o porquê da palavra “**Cascata**”. Mas existe também outra explicação, que veremos a seguir.

➤ Mais sobre Cascata de Estilos

E se houver vários arquivos de estilos externos definidos num mesmo arquivo HTML com alguns estilo coincidentes. Qual estilo ganha?

Resposta: Ganha quem estiver por último! Por exemplo:

```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="estilos1.css">
    <link rel="stylesheet" type="text/css" href="estilos2.css">
  </head>
```

```
p { color : red;
    background-color: blue;
}
```

estilos1.css



```
p { color : white;
}
```

estilos2.css



```
p { color : white;
    background-color: blue;
}
```

A palavra “Cascata” também se aplicará aqui. No exemplo, os dois arquivos CSS se referem a uma propriedade em comum do elemento <p>, que é a propriedade `color`. Neste caso, a tag <p> terá o seu `background-color` igual a `blue` (pois é referenciada apenas em um dos arquivos) e a propriedade `color` igual a `white`, pois vai prevalecer a última regra.

➤ Seletores de Classe e Identificador

Vimos que podemos aplicar estilos em todos os elementos de uma página HTML. Porém, existe um modo de aplicá-los de forma diferenciada. Isso se dá pelos seletores de Classe e Identificador.

Usamos seletores de **Classe** quando queremos aplicar um mesmo estilo em vários elementos diferentes.

Usamos seletores de **Identificador** quando queremos aplicar um estilo a um elemento específico na página HTML.

➤ Seletor de Classe

Um mesmo estilo em vários elementos diferentes:

Uma classe começa com um sinal de ponto (.)

A seguir o nome da classe, que pode conter letras e números.

```
.MeuAzul {  
    color : blue;  
    background-color : black;  
}
```

Estilo

Para aplicar este estilo em elementos HTML, podemos fazer da seguinte forma:

Usamos o atributo **class** para indicar qual classe o elemento deve buscar o estilo.

```
<p class="MeuAzul">Este  
parágrafo em azul e fundo  
preto</p>  
<blockquote class="MeuAzul">  
Este bloco de citação em azul e  
fundo preto.</blockquote>
```

Código HTML

Neste caso, vemos que diferentes elementos (<p> e <blockquote>) podem receber a mesma classe de estilo.

➤ Usos do Seletor de Classe

É possível criar seletores de classe mais restritos, como por exemplo, criar um estilo para apenas um tipo de elemento.

Para isso, o seletor começa com o tipo de elemento `<p>` seguido de ponto (.) e o nome da classe.

```
p.MeuAzul {  
  color : blue;  
  background-color : black;  
}
```

Estilo

Neste exemplo, estamos criando um estilo da classe “MeuAzul” restrito apenas aos elementos `<p>`.

Também é possível que um mesmo elemento possa estar associado a mais de uma classe de estilo. Para isso, colocamos vários estilos separados por espaço no atributo `class`.

```
.MeuAzul {  
  color : blue;  
}  
  
.FundoVerde {  
  background-color : green;  
}
```

Estilo

```
<p class="MeuAzul FundoVerde">Este  
texto ficará na cor azul com um  
fundo na cor verde</p>
```

HTML

Na esquerda temos 2 classes de estilo “MeuAzul” e “FundoVerde”, e no arquivo HTML selecionamos estes 2 estilos para o elemento `<p>`. Para tanto, usamos o atributo `class` e separamos os nomes das classes com espaço.

➤ Seletor de Identificador

Vimos como criar classes que podem ser usados em vários elementos ao longo do nosso código HTML. Mas existem casos em que há elementos específicos e únicos na página HTML, como por exemplo um rodapé ou cabeçalho. Para especificar estilos para estes elementos únicos, usamos o seletor de identificador.

Um seletor de identificador começa com o símbolo #.

```
#rodape {  
  color : blue;  
  font-size : 10px;  
}
```

Estilo

A seguir o nome do identificador, que pode conter letras e números.

Para aplicar este estilo em determinado elemento HTML, podemos fazer da seguinte forma:

Usamos o atributo **id** para indicar elemento vai receber o estilo.

```
<p id="rodape">Este texto vai  
ficar no rodapé de nossa  
página</p>
```

Código HTML

➤ Pseudo-classes

Alguns elementos possuem “estados” que podem ser alterados ao longo da exibição de uma página HTML. Por exemplo, um *link* (elemento `<a>`) pode ter o estado “não visitado”, ou “já visitado”. Para se criar estilos diferentes para cada estado, usamos o conceito de **Pseudo-classes**.

Primeiro definimos o elemento HTML que queremos estilizar.

Depois colocamos o símbolo de dois pontos (:) seguido do “estado”.

```
elemento : estado {  
  
    ... estilos ...  
  
}
```

Estilo de pseudo-classe

➤ Usando as Pseudo-classes

A seguir vemos como usar as pseudo-classes para criar estilos diferentes para cada um dos “estados” de um *link*:

```
a:link {  
  color : green;  
  text-decoration : none;  
}
```

Estilo para quando for um *link* ainda não visitado na página. O estilo será a cor verde e sem sublinhado . (Na próxima aula, veremos mais sobre esta propriedade `text-decoration`).

```
a:visited {  
  color : gray;  
  text-decoration : none;  
}
```

Estilo para quando for um *link* já foi visitado (clicado). O estilo será a cor cinza e sem sublinhado.

```
a:hover {  
  color : red;  
  text-decoration : underline;  
}
```

Estilo para quando o mouse passar pelo *link* mas sem clicá-lo. O estilo será a cor vermelha e com sublinhado.

```
a:active {  
  color : blue;  
  text-decoration : underline;  
}
```

Estilo para quando o *link* for clicado. O estilo será a cor azul e com sublinhado.

Questão 1 (FCC - 2010 - AL-SP - Agente Legislativo de Serviços Técnicos e Administrativos - Processamento de Dados)

Muitos atributos de *tags* nas novas versões da linguagem HTML foram depreciados em favor do uso de

- a) scripts executados do lado do cliente.
- b) folhas de estilo.
- c) scripts executados do lado do servidor.
- d) linguagens de marcação extensíveis.
- e) novas bibliotecas de tags.

Resposta: Letra B. Os atributos de estilo foram depreciados, e atualmente as Folhas de Estilo em Cascata ou *Cascade Style Sheets* (CSS) são recomendadas para definição de estilos.

Questão 2 (FCC - 2009 - TJ-SE - Técnico Judiciário - Programação de Sistemas)

As regras de CSS especificadas dentro da *tag* de HTML que referem-se ao estilo *inline*

- a) podem ser aplicadas por *link* em alguma página HTML.
- b) afetam outros documentos da página HTML.
- c) podem ser aplicadas por *link* no cabeçalho da página HTML.
- d) afetam outras *tags* da página HTML.
- e) afetam somente a *tag* atual da página HTML.

Resposta: Letra E. O tipo *inline* é definido dentro da própria *tag* usando-se o atributo “*style*” e afeta somente esta tag.

Exemplo: `<p style="color:blue">Texto azul</p>`

Questão 3 (CESPE - 2006 - ANATEL - Especialista em Regulação – Informática)

No que se refere às tecnologias usadas na Internet, julgue os seguintes itens.

Um documento HTML é armazenado em um arquivo ASCII e a estrutura do documento é descrita com seqüências de caracteres chamadas *tags*. O estilo de um documento pode ser definido com *tags* HTML ou com a linguagem *Cascading Style Sheets*. O código que define o estilo de um documento HTML tem de ser armazenado no mesmo arquivo no qual está o documento.

() CERTO (X) ERRADO

Resposta: ERRADO. A CSS pode estar definida também em arquivos externos, usando a *tag* `<link>` dentro da *tag* `<head>` do documento HTML.

➤ Resolução de Questões

Questão 4 (CESGRANRIO - 2011 - Transpetro - Analista de Sistemas Júnior)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head><title>Cores</title>
  <style type="text/css">
    /* AQUI */
  </style>
</head>
<body>
  <div id="div1" class="a">DIV 1</div>
  <div id="div2" class="a b">DIV 2</div>
</body>
</html>
```

Qual estilo deve substituir o comentário /* AQUI */ no trecho para que apenas o texto DIV 1 apareça com fundo azul?

- a) div.a {background-color: blue}
- b) div.a {bgcolor: blue}
- c) div.div1 {background-color: blue}
- d) div.div1 {bgcolor: blue}
- e) div#div1 {background-color: blue}

Resposta: Letra E. Não podemos definir um estilo para a classe "a", pois isto afetaria os dois `div`s, pois o segundo `div` pertence também à classe "a". Então, para afetar APENAS o primeiro `div`, usamos o seletor de Identificador (`div#div1`)

➤ Resolução de Questões

Questão 5 (Prova: FCC - 2012 - TRT - 11ª Região (AM) - Analista Judiciário - Tecnologia da Informação)

Considere a página HTML:

```
<html>
  <head>
    <title>Cadastro</title>
    <style type="text/css">
    </style>
  </head>
  <body>
    <p>
      <a href="music.html">Músicas
    </p>
  </body>
</html>
```

A instrução que deve ser utilizada no interior do elemento para retirar o sublinhado do link ao posicionar o ponteiro do mouse sobre ele é

- a) a:hover{text-decoration:underline}
- b) a:hover{text-decoration:none}
- c) a:hover{text-decoration:normal}
- d) a:visited{text-decoration:none}
- e) a:{text-decoration:none}

Resposta: Letra B. O estado “hover” ocorre quando o mouse passa por cima do *link*. E para retirar o sublinhado usamos o estilo “text-decoration:none”.

➤ Resolução de Questões

Questão 6 (FCC - 2010 - TRE-RS - Analista Judiciário - Analista de Sistemas Suporte)

Analise os blocos de código seguintes:

Bloco 1:

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

Bloco 2:

```
<head>
  <style type="text/css">
    body {background-color: red}
    p {margin-left: 20px }
  </style>
</head>
```

Bloco 3:

```
<p style="color: red; margin-left: 20px">
  Este é um parágrafo
</p>
```

Esses blocos representam, respectivamente, as maneiras de inserir folhas de estilo:

- a) interna, externa e inline.
- b) inline, interna e externa.
- c) externa, interna e inline.**
- d) externa e interna, somente.
- e) interna, somente.

Resposta: Letra C.

GABARITO:

- 1 – Letra B
- 2 – Letra E
- 3 – ERRADO
- 4 – Letra E
- 5 – Letra B
- 6 – Letra C

Até a próxima aula!