



HTML para Concursos Públicos

Aula 07 – CSS – Parte 2

A 3D sphere with various web technologies written on it, including 'seo', 'cms', 'sql', 'html', 'php', 'xhtml', 'www', 'ajax', 'pdf', 'jpg', 'web', and 'css'. The 'css' text is highlighted in orange, while the others are in black. A faint 'canva' watermark is visible across the center of the sphere.

Além disso, teremos que também estudar algumas *tags* de estilo que NÃO são mais recomendadas pela W3C, mas que infelizmente AINDA podem ser cobradas por algumas bancas!

➤ Estilos de Fundo (*background*)

Estilos de Fundo dizem respeito à apresentação dos fundos de alguns elementos, e estão associados aos atributos abaixo:

Propriedade **background-color** ou atributo **bgcolor**: define uma **cor** a ser colocada no fundo do elemento HTML. Praticamente TODOS os elementos HTML suportam este atributo. As cores podem ser definidas de três formas:

Por um nome em inglês, por exemplo: `blue`, `black`, `red`, `green` etc.

Por um conjunto de três números de 0 a 255 para especificar tonalidades de vermelho, verde e azul com a função **rgb**. Por exemplo: `rgb(100, 10, 200)`.

Por um conjunto de três números em hexadecimal de 00 até FF para especificar tonalidades de vermelho, verde e azul. Por exemplo: `#640AC8`.

ATENÇÃO: “`bgcolor`” não é mais recomendado. W3C não recomenda mais o seu uso!

➤ Imagens de Fundo (*background*)

Propriedade **background-image** ou atributo **background**: define uma **imagem** a ser colocada no fundo do elemento HTML. No caso da propriedade, usamos a função url para informar qual imagem usar. Por exemplo:

```
<body background="imagens/fundo.jpg">
```

```
<body style="background-image: url('imagens/fundo.jpg') " >
```

ATENÇÃO: "background" não é mais recomendado. W3C não recomenda mais o seu uso!

Propriedade **background-repeat**: determina se a imagem deve ser repetida ou não, e se for repetida, em qual eixo (x : horizontal ou y : vertical). Por exemplo:

```
.imgSemRepet {  
  background-image: url("fundo.jpg");  
  background-repeat: no-repeat; }
```

← Imagem sem repetição

```
.imgVertical {  
  background-image: url("fundo.jpg");  
  background-repeat: repeat-y; }
```

← Imagem repetida verticalmente apenas

```
.imgHorizontal {  
  background-image: url("fundo.jpg");  
  background-repeat: repeat-x; }
```

← Imagem repetida horizontalmente apenas

Se nada for especificado, a imagem será repetida horizontal e verticalmente!

➤ Formatação de Texto

Propriedade **color**: define uma **cor** a ser usada no texto de um elemento HTML. Todo elemento HTML que trata de texto possui este atributo, que segue a regra de valores do atributo `background-color`, ou seja, as três formas: nome, ou função `rgb` ou em hexadecimal. A seguir uns exemplos:

```
<p style="color:green">Texto em verde</p>
```

```
<div style="color:#ffffff">Texto em branco</div>
```

```
<span style="color:rgb(0,255,0)">Texto em verde</span>
```

Repare que usamos três *tags* diferentes: `<p>`, `<div>` e ``, mas poderia ser qualquer uma que envolva um texto, como `<q>`, `<pre>` etc.

Propriedade **text-decoration**: define o estilo do texto, se é sublinhado, traçado etc. A seguir uns exemplos:

```
<p style="text-decoration:underline">Texto</p>
```

→ Sublinhado

```
<p style="text-decoration:line-through">Texto</p>
```

→ Traçado

```
<p style="text-decoration:overline">Texto</p>
```

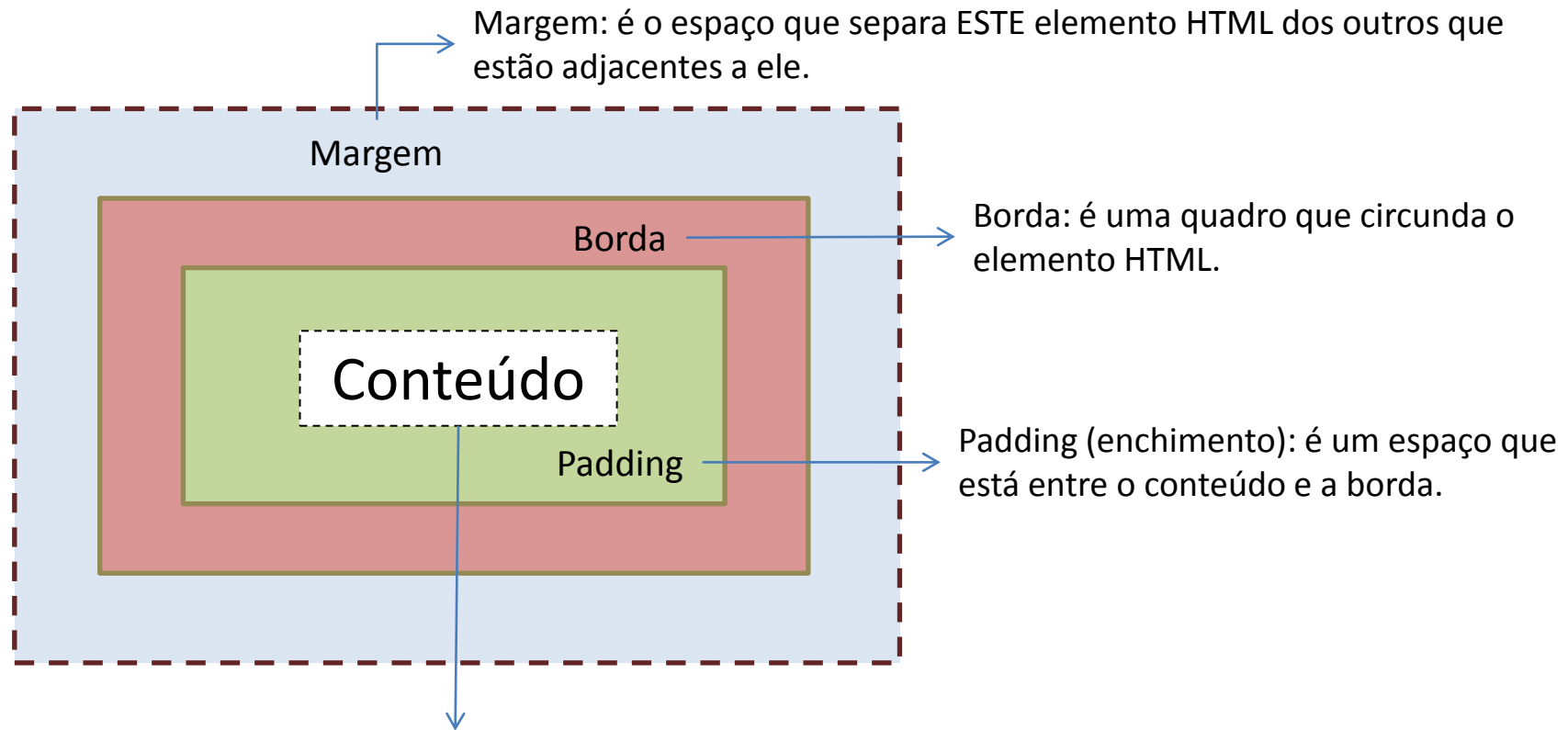
→ Linha superior

```
<p style="text-decoration:blink">Texto</p>
```

→ Piscante

➤ Modelo de Caixa do CSS

CSS usa o conceito de Caixa para modelar vários elementos HTML. Para entender melhor, considere o desenho a seguir:



O conteúdo propriamente dito, pode ser um elemento HTML qualquer, como um parágrafo inteiro, ou uma divisão (<div>) ou um link, entre outros.

Para lidar com as bordas em elementos HTML, precisamos estudar algumas propriedades:

border-style: estilo da borda, pode ser:

dotted: borda pontilhada

dashed: borda tracejada

solid: borda sólida, ou seja, uma linha sem nenhuma falha

double: borda sólida dupla

groove, ridge, inset, outset: bordas com efeito 3D.

border-width: largura da borda. Pode ser definida em pixels, como “3px”, ou com uma palavra, como `thin` (fina), `medium` (médio) ou `thick` (grossa).

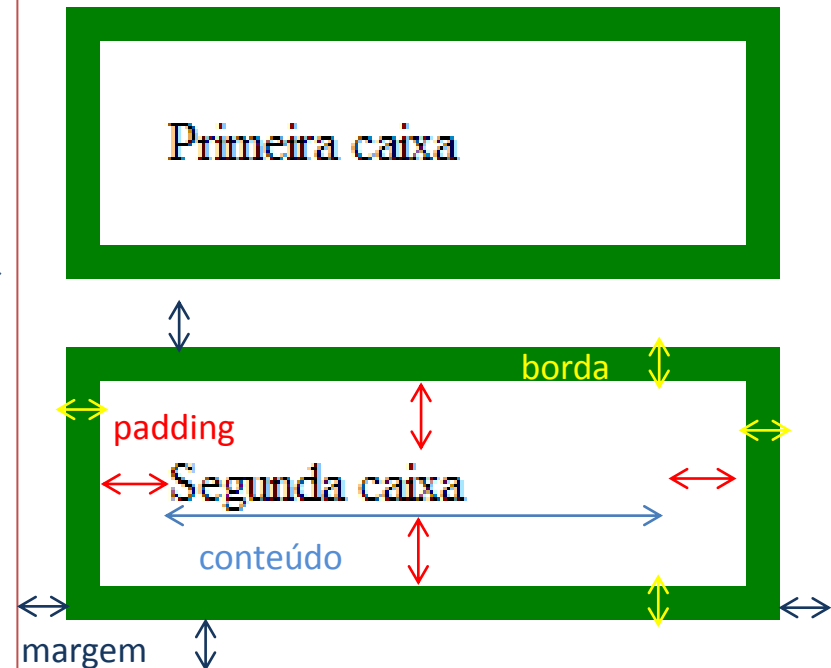
border-color: a cor da borda. Pode ser definida da mesma forma que qualquer propriedade de cor, como a `background-color`.

É possível definir estas três propriedades numa só, respeitando esta ordem, por exemplo:
border: 10px solid green; (define uma borda de largura 10 pixels, do tipo sólida, e de cor verde).

➤ Modelo de Caixa em ação

Vejam agora um exemplo do modelo de Caixa:

```
<html>
  <head>
    <style>
      .caixa {
        width:150px;
        padding:20px;
        border:10px solid green;
        margin:20px;
      }
    </style>
  </head>
  <body>
    <p class="caixa">Primeira caixa</p>
    <p class="caixa">Segunda caixa</p>
  </body>
</html>
```



Para calcular o comprimento total do elemento teríamos que somar:

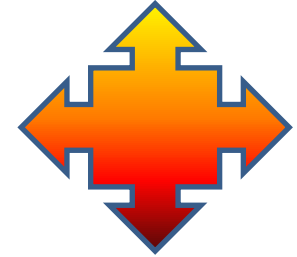
margem esquerda + borda + *padding* esquerdo + conteúdo + *padding* direito + borda + margem direita

No nosso exemplo, teremos: 20 + 10 + 20 + 150 + 20 + 10 + 20 = **250**

E para calcular a altura total do elemento teríamos que somar:

margem inferior + borda + *padding* inferior + conteúdo + *padding* superior + borda + margem superior

➤ Estilos para Posicionamento



É possível determinar o posicionamento de elementos HTML a fim de que possamos ter alguns efeitos de estilo. Para isto existem algumas configurações de posição.

Estático: posiciona os elementos HTML seguindo o fluxo normal do texto. É o **padrão** do HTML!

Fixo (fixed): posiciona os elementos HTML de forma fixa em relação à janela do browser. Mesmo se a janela mudar de tamanho, a posição do elemento não se altera. Seu posicionamento não altera os outros elementos HTML.

Relativo (relative): posiciona os elementos HTML considerando uma relação de distância entre onde o elemento DEVERIA estar. Seu posicionamento afeta e é afetado pelos outros elementos HTML próximos.

Absoluto (absolute): posiciona os elementos HTML sem considerar ou interferir nos outros elementos. Também para o restante dos outros elementos, ele é considerado fora do fluxo. Neste tipo de posicionamento, um elemento HTML pode ficar “por cima” de outro (sobreposição).

➤ Propriedades para posicionar

left: distância em pixels para o início do fluxo ONDE o elemento deveria estar, considerando o eixo horizontal. O eixo horizontal cresce da esquerda para a direita.

top: distância em pixels para o início do fluxo ONDE o elemento deveria estar, considerando o eixo vertical. O eixo vertical cresce de cima para baixo.

z-index: é um índice (um número inteiro) que informa a ordem de sobreposição de elementos HTML. Quanto mais alto, mais acima dos outros elementos ele fica.

Exemplos:

Apesar de terem o mesmo posicionamento (`top` e `left`), os dois elementos `div` não se sobrepõem, pois têm posicionamento relativo, assim o posicionamento de um interfere no outro.

```
<div style="position:relative;top:10px;left:10px;width:150px;
background-color:yellow;">Primeira caixa</div>
<div style="position:relative;top:10px;left:10px;width:150px;
background-color:red;">Segunda caixa</div>
```

Primeira caixa
Segunda caixa

Mas se o posicionamento for absoluto, os elementos não se interferem. Assim, quem for adicionado por último, fica por cima do anterior.

```
<div style="position:absolute;top:10px;left:10px;width:150px;
background-color:yellow;">Primeira caixa</div>
<div style="position:absolute;top:10px;left:10px;width:150px;
background-color:red;">Segunda caixa</div>
```

Segunda caixa

Já vimos que as Tabelas são elementos HTML usados para dispormos conteúdos em formato tabular.

Para adicionar estilos às Tabelas, usamos também o Conceito de Caixa da CSS, pois cada célula da Tabela pode ser considerada uma caixa.

Porém, existem algumas diferenças de nomenclatura entre os Atributos de Tabelas e as Propriedades CSS de Tabelas. A W3C recomenda que usemos APENAS as Propriedades CSS. Abaixo segue uma relação entre elas:

Atributo **align** / Propriedade **text-align**: alinhamento horizontal do texto na célula (<td>). Aceita os seguintes valores: *left*, *right* ou *center*.

Atributo **valign** / Propriedade **vertical-align**: alinhamento vertical do texto na célula (<td>). Aceita os valores: *top*, *bottom* ou *middle*.

Atributo **cellpadding** / Propriedade **padding**: enchimento na célula (<td>), análogo ao conceito de Caixa. Seu valor é em pixels.

Atributo **cellspacing** : espaçamento entre as células (<td>), seu valor é em pixels.

Questão 1 (CESPE - 2012 - TJ-AL - Analista Judiciário - Análise de Sistemas)

Para os desenvolvedores de conteúdo em DHTML é importante entender como funcionam os atributos de estilo de CSS. A respeito desse assunto, assinale a opção correta.

- a) Entre as regras CSS estabelecidas pelo desenvolvedor da página web, as de maior prioridade nas resoluções de conflitos são as regras CSS externa.
- b) O valor padrão do atributo position de CSS para um dado elemento é relative, o qual especifica que o elemento está posicionado de acordo com o fluxo normal de conteúdo do documento.
- c) Elementos de DHTML são estaticamente posicionados, podendo ser posicionados com atributos top e left.
- d) CSS declaradas dentro das tags HTML aplicam regras a todos os elementos da página, pois o conteúdo é misturado com a apresentação.
- e) As regras CSS internas deverão ser declaradas na seção <head> do documento com a tag de estilo <style>.

Resposta: Letra E. Pois: a) as regras CSS mais prioritárias são as inline; b) o valor padrão de posicionamento é static; c) não estudamos ainda, mas DHTML os elementos podem ser dinamicamente posicionados; d) CSS inline só afeta o elemento no qual estiver inserido; e por fim, as CSS internas são definidas realmente dentro da tag <style> que está aninhada na tag <head>.

Questão 2 (CEPERJ - 2012 - PROCON-RJ - Técnico em Informática)

Ao construir um site para a internet, para gerar uma página cujo fundo corresponda à figura procom.jpg, deve ser inserido um parâmetro na tag <body>. Um exemplo correto para essa sintaxe é:

- a) <body screen="procom.jpg"> ... </body>
- b) <body bottom="procom.jpg"> ... </body>
- c) <body bgcolor="procom.jpg"> ... </body>
- d) <body watermark="procom.jpg"> ... </body>
- e) <body background="procom.jpg"> ... </body>

Resposta: Letra E. Podemos usar o atributo `background` para atribuir uma imagem de fundo. Mas a W3C recomenda usarmos de agora em diante a propriedade CSS `background-image`.

Questão 3 (FCC - 2010 - TCE-SP - Agente da Fiscalização Financeira - Informática - Suporte de Web)

O atributo de espaçamento CELLPADDING da linguagem HTML permite

- a) controlar o espaço entre o texto e as bordas da célula.
- b) controlar o espaço entre as células.
- c) mesclar o tamanho das células.
- d) mesclar o conteúdo das células.
- e) controlar o conteúdo das células.

Resposta: Letra A. O atributo `cellpadding` usa o mesmo conceito da Caixa CSS, ou seja, é o “enchimento” entre o conteúdo e a borda da célula. Mas a W3C recomenda usarmos de agora em diante a propriedade CSS `padding`.

Questão 4 (Prova: FCC - 2010 - TCE-SP - Agente da Fiscalização Financeira - Informática - Suporte de Web)

Para determinar uma cor do plano de fundo do browser, por meio da linguagem HTML, deve-se utilizar o atributo

- a) BGCOLOR.
- b) COLOR.
- c) COLORBROWSER.
- d) BODYCOLOR.
- e) BACKGROUND.

Resposta: Letra A. O atributo `b g c o l o r` determina uma cor de fundo que pode ser usada na tag `<body>` e assim, deixar o plano de fundo do *browser* com esta cor. Porém, a W3C recomenda de agora em diante usarmos a propriedade CSS `background-color`.

Questão 5 (FCC - 2009 - TJ-SE - Técnico Judiciário - Programação de Sistemas)

A cor do texto HTML de um único parágrafo da página é definido pelo atributo

- a) Paragraph.
- b) Bgcolor.
- c) Text.
- d) Font.
- e) Color.

Resposta: Letra E. O atributo `color` determina a cor do texto de um elemento HTML.

Questão 6 (ESAF - 2008 - Prefeitura de Natal - RN - Auditor do Tesouro Municipal - Tecnologia da Informação - Prova 2)

Quanto à estrutura, propriedades e sintaxe da linguagem de programação HTML, é correto afirmar que

- a) Os lados da tabela nos quais a borda é apresentada podem ser especificados com o atributo RULES, que pode assumir os valores "box" para borda apresentada em todos os lados e "border" para borda não apresentada.
- b) A apresentação da separação entre as células pode ser especificada com o atributo FRAME, que pode assumir os valores none; groups; rows; cols ou all.
- c) O texto em um documento HTML pode ser destacado usando-se diferentes estilos, físicos ou lógicos, cuja apresentação é independente do navegador utilizado. As tags para se destacar o texto são usadas unicamente no início da ação.
- d) O alinhamento do conteúdo de uma célula pode ser especificado com os atributos ALIGN e VALIGN. O atributo ALIGN pode assumir os valores top, middle, bottom ou baseline e o atributo VALIGN pode assumir os valores left, center, right, justify.
- e) As tabelas de um documento HTML são construídas com tags < TABLE >, < TR >, < TH > e < TD > que identificam, respectivamente, tabelas, linhas, células de cabeçalho e células de dados.

Resposta: Letra E. Pois as outras opções apresentam pequenos erros, sobrando somente a letra E com textos mais corretos.

GABARITO:

- 1 – Letra E
- 2 – Letra E
- 3 – Letra A
- 4 – Letra A
- 5 – Letra E
- 6 – Letra E