**HTML e CSS na Prática**

Capítulo 2 – Tópico 2: Elaboração de Página Web

**Editores de Texto HTML**

A quantidade de editores de texto disponíveis é muito grande. Os editores são essenciais para o desenvolvimento das páginas web, já que possuem funções que auxiliam o desenvolvedor, como autofechamento, autocomplete, diferenciação de trechos do código por cores, entre outras. *Visual Studio Code, Coda, Textmate, Dreamweaver, Sublime Text, Notepad ++ e muitos outros aplicativos existem para essa função.*

**Container Principal** – Main Container

Existem várias tags que podem servir de container para outros elementos, como <section>, <article> e <main>. Como estamos familiarizados com a tag <div>, usaremos essa tag como container principal do site.

Texto

Descrição gerada automaticamente

**Menu**

Todo site precisa de um menu de navegação. De modo geral, o menu fica localizado no topo do site, próximo ao logo. Para isso, existem tags específicas.

Com a chegada do HTML5, algumas tags foram criadas para dar maior entendimento ao código de uma página HTML. A tag <header> foi uma delas. O objetivo da tag <header> é encabeçar o conteúdo do site. Assim como as divs, a tag <header> é o container para outros elementos.

**Margin e Padding**

Por padrão, alguns elementos HTML já vêm com valores predefinidos em suas propriedades. Por exemplo, os parágrafos <p> possuem margin e padding nativos. Desse modo, a página mantém o mínimo de legibilidade mesmo que o CSS não seja carregado. E o que são margin e padding?

* Margin: A propriedade margin do CSS corresponde à dimensão da margem de cada elemento. Além disso, a margin evita que os elementos fiquem grudados uns nos outros.

Interface gráfica do usuário, Diagrama, Aplicativo

Descrição gerada automaticamente

Observe que a parte em azul claro é a margin do elemento. Essa parte ajuda a separar um elemento do outro.

* Padding: O padding é o espaçamento interno dos elementos.

Diagrama

Descrição gerada automaticamente

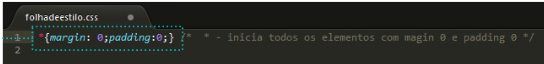
Uma imagem contendo Gráfico de mapa de árvore

Descrição gerada automaticamenteA parte em verde claro é o padding do elemento. Observe que essa parte auxilia a ajustar o espaçamento interno de um elemento.

**Unidade de Medidas** ===================>

**\***

O seletor \* é um seletor universal e representa todos os elementos.



**Definição das Propriedades de uma Class**

Você se lembra da class que colocamos na primeira div que criamos?

Diagrama

Descrição gerada automaticamenteVamos usar essa class para alterar a cor e o estilo da fonte que nossa página terá. Para isso, utilizaremos as propriedades color e font-family. A propriedade color define a cor que o texto terá. De modo geral, o valor utilizado para a definição de cor é o hexadecimal.

Nesse caso, estamos definindo que a fonte (letra) dos elementos que tiverem a class container-principal será da família Verdana e a cor será uma tonalidade de cinza (#777).

Texto

Descrição gerada automaticamente

**Header**

Agora, vamos inserir formatação para o header.

A tag header também pode ser usada como seletor. Desse modo, podemos utilizar o nome da tag para inserir um estilo por meio do CSS. No entanto, quando fazemos isso, todos os elementos criados com essa tag terão o mesmo estilo.

Como iremos ter apenas um header em cada página, isso não será um problema.

Tela de celular com aplicativo aberto

Descrição gerada automaticamente

**Position**

O CSS possui uma propriedade chamada position. A propriedade position trabalha com coordenadas comandadas pelas seguintes propriedades:

Texto

Descrição gerada automaticamentePor definição, os elementos começam todos alinhados à esquerda – left. Caso algum valor seja definido para left, não faz sentindo atribuir valor para right. O mesmo ocorre com relação a top e bottom. Se fizermos isso, ele dará prioridade para o top e o left.

A propriedade position precisa de um ponto inicial para calcular a coordenada e posicionar o elemento na tela. Além disso, a propriedade position possui três valores possíveis.

* *Fixed: O valor fixed fixa o elemento nas coordenadas que atribuirmos. Mesmo com a rolagem da tela, o elemento permanece FIXO na posição definida.*
* *Relative: O ponto inicial para o cálculo de elementos com position definida como relative é o canto superior esquerdo do próprio elemento.*
* *Absolute: Diferente do relative, o valor absolute tem como ponto inicial o elemento em que ele está inserido – seu pai.*

No exemplo que acabamos de ver, definimos a position como absolute. Isso significa que a div logo terá como coordenada inicial o top e left do container do qual faz parte – o header.

Como o header tem o tamanho da janela e está no topo da página, a div logo se posicionará no canto superior da tela. Também definimos a largura da div logo em 300px e a altura como auto. Dessa forma, a div irá se adequar de modo proporcional ao valor da largura.

**Formatação da Imagem**

Agora, precisamos formatar a imagem que inserimos. Para isso, vamos usar uma técnica conhecida como aninhamento.

Como já temos um seletor class para nossa div, não precisamos criar outro para a imagem. Iremos aninhar outro seletor. Nesse caso, pode ser o próprio elemento img.

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Perceba que não criamos outra class para <img>, apenas utilizamos a técnica de aninhamento para fazer a configuração. Observe também que o width foi definido como 90%. Isso significa 90% de 300px, que corresponde à largura da div logo, e um margin-left de 10%. Dessa forma, a imagem do logo terá 280px de largura e 20px de margin-left.

**Aparência do Menu**

Para finalizar a formatação do nosso topo, precisamos definir a aparência do menu (<nav>).

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

1. Usaremos, novamente, uma tag como seletor. Nesse caso, será a tag <nav> juntamente com a técnica do aninhamento para definiras propriedades dos itens da lista <ul>/<li> e dos links <a>.

2. O float corresponde à flutuação de um elemento. Essa propriedade trabalha muito bem com o position relative e é indispensável para o desenvolvimento sem tabelas - tableless. No caso do nav, usamos o float left nas linhas para forçar os itens <li> de nossa lista <ul> a flutuar para esquerda. Com isso, as opções do nav permanecem em uma mesma linha.

3. A propriedade color não tem segredos.

4. Já conhecemos a propriedade margin. Nesse caso, vamos utilizar essa propriedade para separar um item do outro.

5. Também aumentamos a letra do menu por meio da propriedade font-size, com a unidade de medida em. Desse modo, a font (letra) irá se adequar ao tamanho da tela.

6. Outro ponto importante é que utilizamos o :hover para atribuir uma cor diferente quando o usuário passar o mouse sobre o menu.

**Section**

Agora que já definimos o estilo para o topo da página, iremos adicionar a imagem principal da home.Em vez de uma div como container para a imagem principal do site, usaremos a tag <section>

*O elemento HTML section <section> representa uma seção genérica de um documento ou um agrupamento temático de conteúdo.*

**Pop-ups**

Para deixar a página mais elegante e com mais interação, utilizaremos pop-ups para mostrar os textos de conteúdo da home.

O pop-up é uma janela que abre no navegador da internet quando acessamos uma página na web ou um link de redirecionamento. A nova janela pode apresentar informações de destaque do site. Na maioria dos casos, as pop-ups apresentam publicidades e anúncios.

O pop-up que iremos inserir não vai abrir em uma nova janela do navegador. Vamos utilizar um conceito conhecido como Janela Modal. A janela modal é uma janela secundária ou janela filha da tela principal. A função dessa janela secundária é fornecer informações importantes para o usuário.

Para isso, criaremos três divs que vão servir de container para os textos da página.

* *<div id=”openmodal” class="modalDialog"> </div>*
* *<div id=”openmodal1” class="modalDialog"> </div>*
* *<div id=”openmodal2” class="modalDialog"> </div>*

Observe que criamos três divs com IDs distintos, mas com classes iguais. Após criar as três divs e atribuir-lhes ID e class, chegou a hora de adicionar os elementos de cada pop-up. Para o box com o texto, vamos criar uma div. Dentro dessa div, vamos inserir as tags:

* **<a>** com a função de fechar o pop-up
* **<h1>** para o título do conteúdo
* **<p>** para definir um novo parágrafo

Também vamos colocar dois atributos – o href e class –, com os seguintes valores:

*o <a href="#close" class="close"> X </a>*

Texto

Descrição gerada automaticamente

Observe que adicionamos a letra X como texto do link. A letra X funcionará como um botão de fechar. Em breve, explicaremos como se dará o processo de fechamento.

*Motor de Renderização*

Um motor de renderização é um software que transforma linguagem de marcação em informação de formatação. Com isso, o conteúdo formatado fica visível em monitores e demais aparelhos que possuem navegadores. O Safari e o Google Chrome utilizam o motor **webkit**. Já o navegador Mozila Firefox utiliza o prefixo **moz** para referenciar seu motor de renderização. Também temos os prefixos O, que remete ao navegador Opera, e MS, que remete ao Internet Explorer.

Ícone

Descrição gerada automaticamente

Transition

Neste caso, utilizaremos a propriedade transition para realizar a mudança de valor da propriedade opacity. Esse valor foi definido juntamente com as demais propriedades da classe .modalDialog e seu valor inicial é 0, ou seja, o elemento estará totalmente transparente.

Com a propriedade transition, estamos informando ao navegador que o motor de renderização deve realizar a troca de forma suave, quando o elemento com a class modalDialog sofrer uma mudança na propriedade opacity.

A troca deve ser feita em 400 ms (milissegundos), utilizando o efeito ease-in para determinar o ritmo em que a transição será feita.

Texto

Descrição gerada automaticamente

Existem alguns padrões predefinidos para a utilização do *transition*. Por exemplo:

* Linear
* *Ease*
* *Ease-in*
* *Ease-out*
* *Ease-in-out*.

Também podemos personalizar o efeito de transição. No entanto, é necessário ter um conhecimento intermediário das propriedades do CSS.

Pointer-Events

A propriedade pointer-events com o valor none informa ao navegador que os elementos que possuem a class modalDialog não devem ser alvo de eventos realizados pelo mouse. Por exemplo: cliques e hovers.

Desse modo, as divs com essa classe modalDialog tem o tamanho total da janela e estão acima de todos os outros elementos, por conta do z-index com valor 99999. Com isso, qualquer interação com os elementos abaixo dela seria impossível.

Para que isso não ocorra, definimos o valor none para a propriedade pointer-events. O valor indica ao navegador que o elemento não deve receber eventos realizados pelo mouse.

Display

a propriedade display permite que você indique ao navegador como os elementos devem ser renderizados pelo motor.

Existem cerca de 20 valores possíveis para a propriedade display. Iremos utilizar apenas os valores block e none.

* *Block: O valor block indica ao navegador que o elemento deve ser renderizado em bloco, juntamente com seus filhos ou elementos contidos nele.*
* *None: O valor none permite desativar a exibição do elemento. Desse modo, o documento é renderizado como se o elemento não existisse.*

Âncoras

Quando trabalhamos com âncoras em um site, utilizamos a tag <a>.

Dito de outro modo, podemos utilizar um ID como valor do href de uma tag <a>. Desse modo, criamos uma ligação ou âncora da tag <a> com o elemento que possui o ID informado.

Ao clicar no link, o usuário é direcionado para a posição em que o elemento está. Com isso, o usuário visualiza o conteúdo que deseja de forma rápida.

Podemos relacionar esse recurso a um objeto, de forma a manipular seu comportamento com o CSS. Para tanto, precisamos da pseudoclasse :target.

Pseudoclasse: A pseudoclasse é uma palavra adicionada a seletores. Tal palavra indica o estado especial de um elemento ativo. Neste caso, quando clicamos no link (tag <a>) é ativada a ligação (âncora), e a tela é direcionada até o local em que elemento que possui o mesmo ID do atributo href da tag <a> está. Com isso, esse elemento pode ser manipulado pelo seletor :target.

Texto

Descrição gerada automaticamente1. Estamos informando ao navegador que a div ficará visível (display: block)

2. A opacidade mudará para 1 (opacity : 1) quando clicarmos no link ou na âncora. Isso vai ocorrer porque o clique ativará o transition colocado anteriormente na class modalDialog. Desse modo, o pop-up vai aparecer suavemente

3. A propriedade pointer-events foi modificada para auto, possibilitando a interação com as divs.

Elementos do pop-up

Texto

Descrição gerada automaticamenteNem todos os navegadores são compatíveis com os elementos do HTML 5. As propriedades transition e border-radius podem apresentar um comportamento não satisfatório em navegadores mais antigos.

1. *Line-height:* A propriedade line-height define a altura que cada linha do texto terá. No caso do botão close, iremos ajustar a altura que a letra X terá dentro do elemento.

2. *Text-align*: A propriedade text-align permite alinhar o texto. Os valores dessa propriedade são:

* Left (padrão) – atribui alinhamento à esquerda.
* Center – atribui alinhamento centralizado.
* Right – atribui alinhamento à direita.
* Justify – atribui alinhamento justificado.
* Initial – atribui os valores padrão de formatação de texto que o navegador usa.
* Inherit – herda as propriedades de alinhamento de texto do pai (elemento em que o texto está contido).

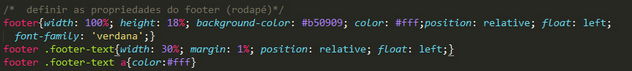
3. *Text-decoration :* A propriedade text-decoration permite inserir formatação ao texto por meio do CSS. Por exemplo, podemos inserir underline. Para dar destaque aos links, toda tag <a> possui cor diferente e underline (\_ sublinhado) quando é adicionada. Para eliminar esse comportamento, usamos a propriedade text-decoration com valor none.

4. *border-radius*: Assim como na formatação da div, também indicamos ao navegador como ele deve renderizar a borda do elemento.

**Footer**

A largura do footer (width) será de 100%. Desse modo, o footer terá o tamanho do container em que está inserido. Como a div ‘container-principal’ tem o tamanho da janela, o footer também terá a largura da janela.

Definição de Estilo de Texto



Interface gráfica do usuário

Descrição gerada automaticamenteInterface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente1. 2.

**Criação das Demais Páginas**

Texto

Descrição gerada automaticamenteNo segundo link do menu, criamos uma âncora para o topo. Como já estamos na página clientes, o redirecionamento para essa página é desnecessário. Clique na imagem para ampliá-la.

*Será que teremos que formatar o menu novamente?*

Não. Utilizamos o seletor <header> para definir o estilo do menu. Com isso, sempre que essa tag for inserida, o estilo definido no CSS será herdado. Como a folha de estilo é a mesma para todas as páginas, a aparência será idêntica. O logo também adotará o mesmo comportamento nas páginas em que for inserido, já que utiliza a class logo para definir suas propriedades.

Adição de Conteúdo – Logo, Resumo e Link

Cada div será composta de logo, um breve resumo e o link para a página do cliente

Adição de Conteúdo – Estilo

Com exceção das propriedades border e font-size, as demais propriedades já foram detalhadas anteriormente. Dessa forma, direcionaremos nossa atenção para as propriedades border e font-size.