HTML e CSS

Front End Web

Msc. Lucas G. F. Alves

e-mail: lucas.g.f.alves@gmail.com





Planejamento de Aula

Revisão HTML e CSS

Fontes, Alinhamentos e Bordas

Exercícios

Espaçamento, Margem, Seletores e Float

Exercícios





Revisão





Imagens

>

A tag define uma imagem em uma página HTML e necessita de dois atributos preenchidos: src e alt.

O primeiro aponta para o local da imagem e o segundo, um texto alternativo para a imagem caso essa não possa ser carregada ou visualizada.

O HTML 5 introduziu duas novas tags específicas para imagem: <figure> e <figcaption>. A tag <figure> define uma imagem com a conhecida tag . Além disso, permite adicionar uma legenda para a imagem por meio da tag <figcaption>.







Estrutura de Arquivos



Existem algumas recomendações quanto à organização dos arquivos de um site:

Todos os arquivos de um site estarem dentro de uma só pasta.

Ex: livro, tem uma "capa", uma página inicial que possa indicar para o visitante quais são as outras páginas que fazem parte desse projeto e como ele pode acessá-las, como se fosse o índice do site.

Índice também chamado index.html, será a página inicial a menos que alguma configuração determine outra página para esse fim.





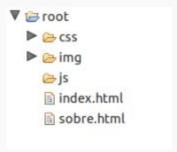


Estrutura de Arquivos



Dentro da pasta do site, no mesmo nível que o index.html, criar mais algumas pastas para manter separados os arquivos de:

- As imagens;
- As folhas de estilo CSS;
- Os scripts;



Assim teremos a pasta do projeto, dentro desta pasta teremos as páginas html junto com as pastas para cada um dos arquivos citados acima.







Links



Quando precisamos indicar que um trecho de texto se refere a um outro conteúdo, seja ele no mesmo documento ou em outro endereço, utilizamos a tag de âncora <a>.

As âncoras tem dois usos. Um deles é a definição de links:

Visite o site do Senac .

A âncora está apenas a palavra "Senac" e quando clicarmos com o cursor do mouse na palavra "Senac", o navegador redirecionará o usuário para o site do Senac, indicado no atributo href.







Links



O outro uso para âncora é a demarcação de destinos para links dentro do próprio documento, o que chamamos de bookmark.

Bookmark é o elemento que tem o atributo id.

```
Mais informações <a href="#info"> aqui </a> . 
Conteúdo da página...
<h2 id="info"> Mais informações sobre o assunto: </h2>
Informações...
```

Ao clicar sobre a palavra "aqui", demarcada com um link, o usuário será levado onde o atributo id foi referenciado.







Links



É possível levar o usuário a um bookmark presente em outra página.

Isso fará com que o usuário que clicar no link seja levado outra página indicada no endereço, especificamente no ponto onde o bookmark "contato" está.







Listas HTML



Não são raros os casos em que queremos exibir uma listagem em nossas páginas. O HTML tem algumas tags definidas para que possamos fazer isso de maneira correta. A lista mais comum é a lista não-ordenada.

```
ul>
    Primeiro item da lista
    <|i>
        Segundo item da lista:
        ul>
             Primeiro item da lista aninhada
             Segundo item da lista aninhada
        Terceiro item da lista
```







Listas HTML



Note que, para cada item da lista não-ordenada, utilizamos uma marcação de item de lista <Ii>. No exemplo acima, utilizamos uma estrutura composta na qual o segundo item da lista contém uma nova lista. A mesma tag de item de lista <Ii> é utilizada quando marcamos uma lista ordenada.

```
    Primeiro item da lista
    Segundo item da lista
    Terceiro item da lista
    Quarto item da lista
    Quinto item da lista
```







Listas HTML



As listas ordenadas também podem ter sua estrutura composta por outras listas ordenadas como no exemplo que temos para as listas não-ordenadas. Também é possível ter listas ordenadas aninhadas em um item de uma lista não-ordenada e vice-versa.

Existe um terceiro tipo de lista que devemos utilizar para demarcar um glossário, quando listamos termos e seus significados. Essa lista é a lista de definição.

```
<dl>
    <dd>HTML</dt>
    <dd>HTML é a linguagem de marcação de textos utilizada para exibir textos como páginas na Internet. </dd>
    <dd>
        <dd>
```

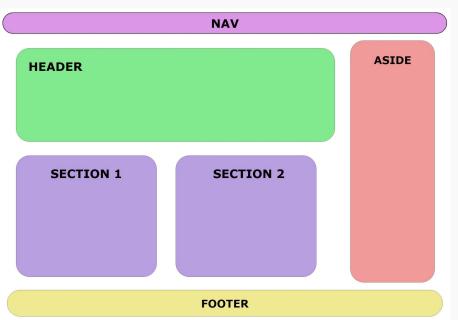


Tags Semânticas



Tags semânticas são tags que possuem um significado, que dão sentido a informação de texto ao navegador e buscadores.

```
São elas:
<body>
    NAV
    HEADER
    MAIN
         SECTION
    FOOTER
    ASIDE
</body>
```







Elementos Estruturais



Vimos muitas tags para casos específícos:

- Títulos com h1;
- Parágrafos com p;
- Imagens com img;
- Links com a etc;
- Listas;

Mas não existe uma tag diferente para cada coisa do universo.

O conjunto de tags do HTML é bem vasto mas é também limitado.

Então como colocar uma tag sem definição?







Elementos Estruturais



Vimos muitas tags para casos específícos:

- Títulos com h1;
- Parágrafos com p;
- Imagens com img;
- Links com a etc;
- Listas;

Mas não existe uma tag diferente para cada coisa do universo.

O conjunto de tags do HTML é bem vasto mas é também limitado.

Então como colocar uma tag sem definição?

Podemos usar as tags <div> e que funcionam como "coringas".

Servem para agrupar conteúdos, tanto um bloco da página quanto um pedaço de texto.









Introdução

Quando escrevemos o HTML, marcamos o conteúdo da página com tags para cada conteúdo.

Quando abrimos a página no navegador mostra as informações com estilos diferentes.

Um h1, por exemplo, fica em **negrito** numa fonte maior.

Parágrafos de texto são espaçados entre si, e assim por diante.

O navegador tem um estilo padrão para as tags que usamos.

Se quisesse um título em vermelho, era só fazer:









Várias outras tags de estilo existiam.

Tags HTML para estilo são má prática hoje em dia e não devem ser usadas.

Assim surgiu o CSS, que é uma outra linguagem, separada do HTML, com objetivo único de cuidar da estilização da página.

Características do CSS:

Bem mais robusto que o HTML para estilização.

Separa as regras de estilo do conteúdo HTML.









Sintaxe do CSS

A sintaxe do CSS tem estrutura simples:

É uma declaração de propriedades e valores separados por um sinal de dois pontos ":", e cada propriedade é separada por um sinal de ponto e vírgula ";" da seguinte maneira:

background-color: yellow; color: blue;

O elemento que receber essas propriedades será exibido com o texto na cor azul e com o fundo amarelo.

Essas propriedades podem ser declaradas de três maneiras diferentes.









Atributo style

A primeira delas é como um atributo style no próprio elemento:

```
O conteúdo desta tag será exibido em azul com fundo amarelo no navegador!
```

Mas isso é uma má prática, então.

Justamente por isso não se recomenda esse tipo de uso na prática, mas sim os que veremos a seguir.









A tag style

Outra maneira de se utilizar o CSS é declarando suas propriedades dentro de uma tag <style>.

Assim estamos declarando o estilo de um elemento em outro lugar do nosso documento.

Precisamos indicar a qual elemento nos referimos.

Fazemos isso utilizando um seletor CSS.

É basicamente setar a tags que receberão o estilo que queremos.









Exemplo: seletor pega todas as tags p e altera sua cor e background:

```
<!DOCTYPE html>
<html>
       <meta charset="utf-8">
       <title>Senac</title>
           background-color: _yellow;
           color: □blue;
   </head>
       O conteudo desta tag será exibido em azul com fundo amarelo! 
       <strong> Também </strong> será exibido em azul com fundo amarelo! 
   </body>
</html>
```









Arquivo externo

A terceira maneira de declarar os estilos é com um arquivo externo.

Este arquivo tem formato com a extensão .css.

Para o HTML usar o CSS precisamos indicar em nosso documento HTML uma ligação entre ele e o CSS.

Com isso nosso HTML fica mais limpo

Utilização e reaproveitamento do mesmo estilos para diversos documentos.

A indicação de uso de CSS(folha de estilos) externa deve ser feita dentro da tag <head> do nosso documento HTML:









Exemplo de HTML com CSS externo:

```
<!DOCTYPE html>
<html>
   <head>
       <meta charset="utf-8">
       <title>Sobre a Mirror Fashion</title>
       <link rel="stylesheet" href="estilos.css">
   </head>
   <body>
       O conte<mark>ú</mark>do desta tag será exibido em azul com fundo amarelo!
       <strong>Tambem</strong> sera exibido em azul com fundo amarelo! 
   </body>
</html>
```









E dentro do arquivo estilos.css colocamos apenas o conteúdo do CSS:

```
p {
    color: □blue;
    background-color: □yellow;
}
```







Imagem de Fundo



A propriedade background-image indica um arquivo de imagem para ser exibido ao fundo do elemento. Por exemplo:

```
h1 {
background-image: url(sobre-background.jpg);
}
```

O navegador vai requisitar um arquivo sobre-background.jpg, que deve estar na mesma pasta do arquivo CSS onde consta essa declaração.







Cores na Web



Propriedades como background-color, color, border-color, entre outras aceitam uma cor como valor. Existem várias maneiras de definir cores quando utilizamos o CSS.

A primeira e mais simples, é usando o nome da cor:

```
h1{ color: red; }
h2{ background: yellow; }
```

É difícil acertar a exata variação de cor que queremos no design. O mais comum é definir a cor com base em sua composição RGB.







Cores na Web



RGB é um sistema de cor bastante comum.

Permitindo especificar até 16 milhões de cores como uma combinação de três cores base:

Vermelho (Red), Verde (Green), Azul (Blue).

Também podemos definir a intensidade de cada um desses três canais básicos, numa escala de 0 a 255.

No CSS, podemos escrever as cores tendo como base sua composição RGB: Exemplo:

h3 { color: rgb(255, 200, 0); }







Cores na Web



Essa sintaxe funciona mas não é a mais comum na prática, por questões de compatibilidade. O mais comum é a notação hexadecimal como #F2EDED. Essa sintaxe tem suporte universal nos navegadores e é mais curta de escrever, apesar de ser mais enigmática.

Exemplo:

h3 { background: #F2EDED;}

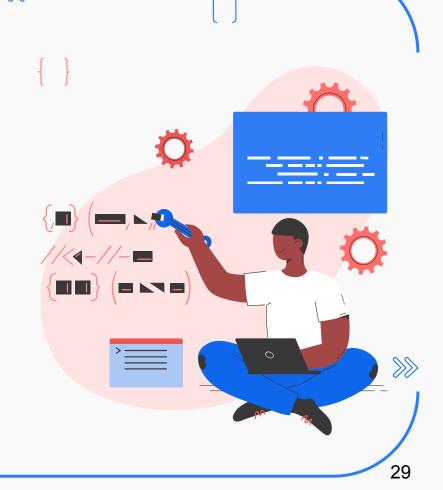
Na notação hexadecimal (que começa com #), temos 6 caracteres. Os primeiros 2 indicam o canal Red, os dois seguintes, o Green, e os dois últimos, Blue.

Seu editor de imagens deve ser capaz de fornecer ambos os valores para você sem problemas.





Propriedades tipográficas e fontes





Propriedades Tipográficas e fontes



As fontes podem ser alteradas com o uso da propriedade **font-family**.

A propriedade **font-family** pode receber seu valor com aspas (nome do arquivo de fonte) ou sem aspas (família da fonte).

Por padrão, os navegadores exibem texto no tipo "serif".

As fontes mais conhecidas são "Times" e "Times New Roman". Elas são chamadas de fontes serifadas pelos pequenos ornamentos em suas terminações.

Site com lista de fontes: https://wavian.com/font-list.html







Propriedades Tipográficas e fontes



Font-family

Pode-se alterar a família de fontes como "sans-serif" (sem serifas), que contém, por exemplo, as fontes "Arial" e "Helvetica".

Pode-se também declarar uma família de fontes "monospace" como, por exemplo, a fonte "Courier".

Exemplo:

```
h1{font-family: serif; }
h2{font-family: sans-serif;}
p{font-family: monospace; }
```







Propriedades Tipográficas e fontes



Pode-se testar se uma fonte existe no computador.

As fontes mais comuns são consideradas "seguras" por serem bem populares.

Exemplo: body { font-family: "Arial", "Helvetica", sans-serif; }

- 1) O navegador verificará se a fonte "Arial" está disponível e a utilizará.
- 2) Caso a fonte "Arial" não esteja disponível, o navegador verificará a próxima fonte a "Helvetica".
- 3) Caso o navegador não encontre também, ele solicita qualquer fonte da família "sans-serif".

Outra tag que serve para manipular a fonte, é a font-style, que define o estilo da fonte que pode ser: normal (normal na vertical), italic (inclinada) e oblique (oblíqua)





Alinhamento e decoração de texto





Alinhamento e decoração de texto



Text-align

Para alinhamento de texto é utilizada a tag text-align.

```
p { text-align: right; }
```

O exemplo determina que os parágrafos tenham o texto alinhado para a direita.

Pode-ser também determinar que seja alinhado ao centro ao definirmos o valor center, ou então definir que o texto vai ocupar toda a largura do elemento aumentando o espaçamento entre as palavras com o valor justify.







Alinhamento e decoração de texto



Por padrão o texto é alinhado à esquerda, com o valor left.

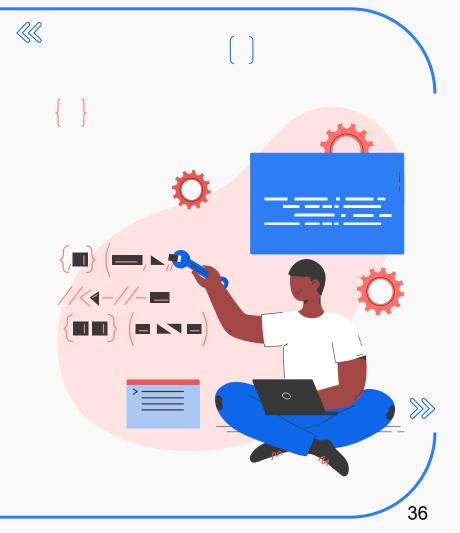
É possível configurar também uma série de espaçamentos de texto com o CSS:

```
p {
          line-height: 3px; /* tamanho da altura de cada linha */
          letter-spacing: 3px; /* tamanho do espaço entre cada letra */
          word-spacing: 5px; /* tamanho do espaço entre cada palavra */
          text-indent: 30px; /* tamanho da margem da primeira linha do texto */
}
```





Bordas





Bordas



Border

É disponível uma série de opções para definição de bordas.

Podemos, para cada borda de um elemento, determinar sua cor, seu estilo de exibição e sua largura.

```
Por exemplo:
```

```
body {
     border-color: red;
     border-style: solid;
     border-width: 1px;
}
```





Exercícios





Exercícios



- 1) Alterar a cor e a fonte do título principal e do corpo na página contato.html, com as propriedades color e font-family.
- 2) O título principal já deve ter um fundo estampado com a imagem, aproveite e coloque uma borda sutil nos subtítulos, para ajudar a separar o conteúdo. Exemplo: h1 { background-image: url(../img/sobre-background.jpg); } h2 { border-bottom: 2px solid #333333; }
- 3) Acerte também a renderização das figuras. Coloque um fundo cinza, uma borda sutil, deixe a legenda em itálico com font-style e alinhe a imagem e a legenda no centro com text-align. Exemplo: figure { background-color: #F2EDED; border: 1px solid #ccc; text-align: center; } figcaption { font-style: italic; }









>>

Padding

A propriedade padding é utilizada para definir uma margem interna (distância entre o limite do elemento, sua borda, e seu respectivo conteúdo) e tem as subpropriedades listadas a seguir:

- padding-top
- padding-right
- padding-bottom
- padding-left



Essas propriedades aplicam uma distância entre o limite do elemento e seu conteúdo acima, à direita, baixo e à esquerda respectivamente. Sentido horário.







Padding

Pode-se definir todos os valores de uma única vez.

Se passado somente um valor para a propriedade padding, esse mesmo valor é aplicado em todas as direções.

```
p { padding: 10px; }
```

Se passados dois valores, o primeiro será aplicado acima e abaixo (mesmo valor para padding-top e padding-bottom) e o segundo será aplicado à direita e à esquerda (padding-right e padding-left).

```
p { padding: 10px 15px; }
```







>>

Padding

Pode ser passado três valores, o primeiro será aplicado acima (padding-top), o segundo será aplicado à direita e à esquerda (padding-right e padding-left) e o terceiro valor será aplicado abaixo do elemento (padding-bottom)

```
p { padding: 10px 20px 15px; }
```

Se forem quatro valores, serão aplicados respectivamente a padding-top, padding-right, padding-bottom e padding-left.

p { padding: 10px 20px 15px 5px; }





Margem





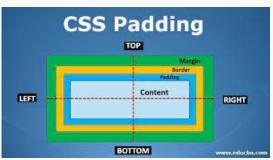
Margem



Margin

A propriedade margin é parecida padding, porém ela adiciona espaço após o limite do elemento, ou seja, é um espaçamento além do elemento em si.

- margin-top
- margin-right
- margin-bottom
- margin-left



Para permitir que o navegador defina qual será a dimensão da propriedade padding ou margin conforme o espaço disponível na tela: definimos o valor auto.

p { margin: 0 auto; }







Dimensões

>>

Um elemento pode ser dimensionado tanto pelas tags de posicionamento, mas também pelas propriedades de HEIGHT será definido a altura e WIDTH a largura do objeto, por exemplo:

height e width — definem dimensões de um elemento (auto, inherit, comprimento em pixels ou %)

max-height e max-width — definem dimensões máximas de um elemento (auto, inherit, comprimento em pixels ou %)

min-height e min-width — definem dimensões mínimas de um elemento (auto, inherit, comprimento em pixels ou %)







Ocultando / exibindo



Comandos:

Retira a elemento da visualização

display:none;

Apenas esconde o elemento, porém o espaço que ele ocupa fica visível.

– visibility: hidden;

Define elemento como visível num bloco (padrão)

– display:block;

Define elemento como visível numa linha (conteúdo disposto em linha)

– display:inline;

Ocultar e re-exibir elementos é muito útil para layouts com abas ou trocas alternadas de elementos de conteúdo.





Seletores de ID





Seletores de ID



É possível aplicar propriedades definindo um valor de seu atributo id. Para isso, o seletor deve iniciar com o caractere "#" seguido do valor correspondente.

```
#cabecalho { color: white; text-align: center; }
```

O seletor acima fará com que todos os elementos dentro de cabecalho tenha seu texto renderizado na cor branca e centralizado.

Como o atributo id deve ter valor único no documento, o seletor deve aplicar suas propriedades declaradas somente àquele único elemento e, por cascata, a todos os seus elementos filhos.





Seletor Hierárquico





Seletores Hierárquico



Pode-se utilizar um seletor hierárquico que permite aplicar estilos aos elementos filhos de um elemento pai:

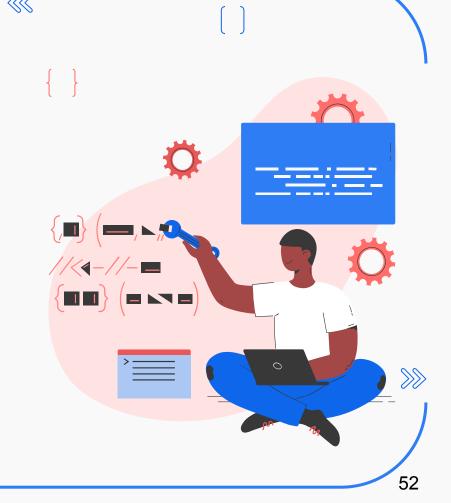
```
#rodape img {
    margin-right: 35px;
    vertical-align: middle;
    width: 94px;
}
```

No exemplo, o estilo será aplicado apenas nos elementos img filhos do elemento com id=rodape.





Fluxo de documento: Float





Fluxo de documento: Float



Float

A propriedade float permite que tiremos um certo elemento do fluxo vertical do documento o que faz com que o conteúdo abaixo dele flua ao seu redor.

Aplicando float a uma imagem fará com que o conteúdo do parágrafo flua ao redor da nossa imagem.

Houve uma perturbação do fluxo HTML, e a imagem parece existir fora do fluxo.







Fluxo de documento: Float



Float

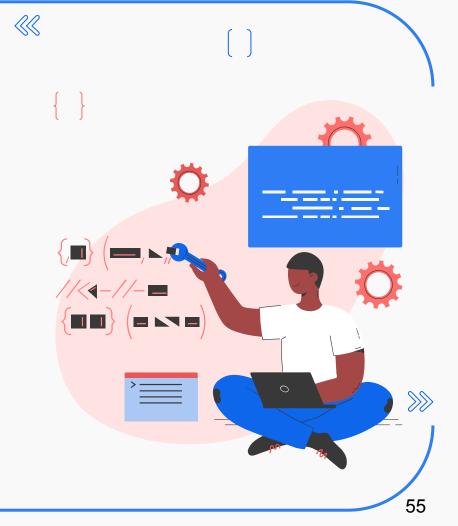
Propriedades

```
    clear — desvincula um elemento do efeito de floating (left, right, both, none, inherit)
    float — especifica uma flutuabilidade para um elemento (left, right, none, inherit)
```





Exercícios





Exercícios



- 1) Aplicar bordas as imagens e textos colocados na página index.html.
- 2) Na página sobre.hmtl, criar um subtítulo com tag h2 chamada Diferenciais, e através de listas, coloquem seus diferenciais como desenvolvedores.
- 3) Aplicar espaçamentos e margem em sua página sobre.html. Exemplo: h1 { padding: 10px; } h2 { margin-top: 30px; } p { padding: 0 45px; text-indent: 15px; } figure { padding: 15px; margin: 30px; } figcaption { margin-top: 10px; }.
- 4) Aplicar seletor id para a figure na página sobre.html. Exemplo:#imagem_autor { margin-left: auto; margin-right: auto; width: 550px; }.
- 5) Crie um rodapé para a página utilizando uma <div> que deve ser inserida como último elemento dentro da tag body
- 6) Colocar uma imagem no rodape e aplicar seletores id e hierarquico para o rodape.







Exercícios CSS



Recriar essa página, e colocar link para a página em conteudo.html.

Fontes:

'Sedgwick Ave', cursive;

'Caveat', cursive:

The Scream
The Scream (Norwegian: Skrik) is the popular name given to each of four versions of a composition, created as both paintings and pastels, by Norwegian Expressionist artist Edvard Munch between 1893 and 1910. The German title Munch gave these works is Der Schrei der Natur (The Scream of Nature). The works show a figure with an agonized expression against a landscape with a tumultuous orange sky. Arthur Lubow has described The Scream as "an icon of modern art, a Mona Lisa for our time." Edvard Munch created the four versions in various

media. The National Gallery in Oslo, Norway, holds one of two painted versions (1893, shown here). The Munch Museum holds the other painted version (1910, see gallery, below) and a pastel version from 1893.



Artist Edward Munch

Year 1893

Type Oil, tempera, pastel and crayon on cardboard

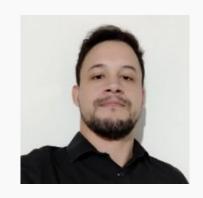
Dimensions 91 cm x 73.5 cm (36 in x 28.9 in)

Location National Gallery, Oslo, Norway

Copied from The Scream Wikipedia



Professor



Lucas G. F. Alves





Obrigado!

E-mail :lucas.g.f.alves@gmail.com





