



Instituto **Janela**
para o **Mundo**



Let's code

HTML

O QUE É ?

- HTML (Hypertext Markup Language ou Linguagem de Marcação de HiperTexto) é uma linguagem de marcação (não de programação) da web - cada vez que você carrega uma página da web, você está carregando um código HTML. Pense no HTML como o esqueleto de uma página, ele é responsável pelos textos, links, listas e imagens.

INICIANDO...

- HTML é escrito em arquivos com o formato `.html`. Para criar uma página HTML é fácil, entre em seu editor de código (se não tiver uma instalado eu recomendo o Visual Studio Code) e salve o arquivo em branco como `index.html` (você pode nomeá-lo como quiser, mas não esqueça do `.html`).

ESTRUTURA INICIAL

A code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top left. It displays five lines of HTML code with syntax highlighting: line 1 is `<!DOCTYPE html>`, line 2 is `<html>`, line 3 is `<head></head>`, line 4 is `<body></body>`, and line 5 is `</html>`.

```
1 <!DOCTYPE html>
2 <html>
3   <head></head>
4   <body></body>
5 </html>
```

`<!doctype html>` - Essa tag (não tem fechamento) informa ao navegador que o arquivo faz parte de um documento HTML5.

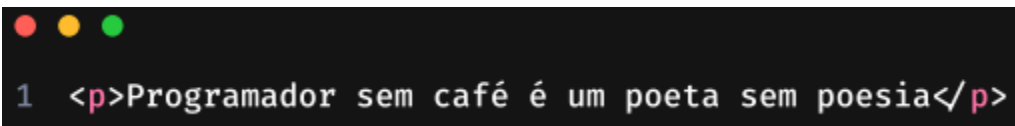
`<html>` - Representa a raiz de um documento HTML.

`<head>` - O head contém informações sobre sua página, mas não o conteúdo que vai aparecer na tela da sua página. Ela conterá coisas como: links para folhas de estilo (CSS), título da página, links de fontes e meta tags e entre outros.

`<body>` - No body contém todo o conteúdo que vai aparecer na tela da sua página. Todo o código que você queira apresentar na página deverá estar dentro dele.

SINTAXE

- Os elementos HTML são escritos usando tags. Todas as tags tem uma chave de abertura e fechamento. A tag de fechamento tem uma barra após o primeiro colchete.
- Exemplo: `<tag>seu nome</tag>`



```
1 <p>Programador sem café é um poeta sem poesia</p>
```

HIERARQUIA

- Os elementos também podem entrar dentro de outros elementos:

```
1 <pai>  
2   <filho>Esta tag está dentro de outra tag</filho>  
3 </pai>
```

INDENTAÇÃO ADEQUADA

- As quebras de linha entre suas tags são super importantes para escrever um bom código HTML. Abaixo temos um bom exemplo disso:

Adequado

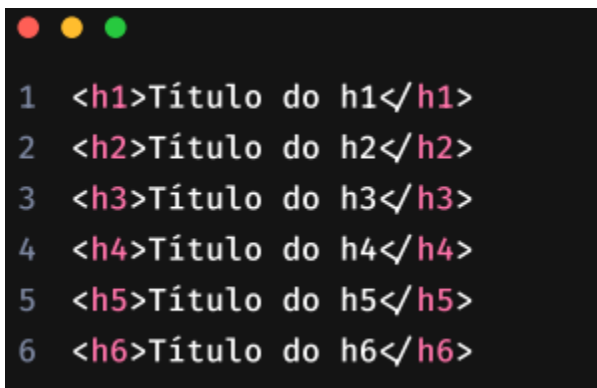
```
1  ...
2  <body>
3    <h1>Aqui temos um título</h1>
4    <p>E aqui um parágrafo</p>
5  </body>
6  ...
```

Não adequado

```
1  <!DOCTYPE html> <html> <head> </head> <body> <h1>
2  Não faça isso</h1> </body> </html>
```

TÍTULOS

- As tags de títulos são representadas em até seis níveis. `<h1>` é o nível mais alto e `<h6>` é o mais baixo. Quanto maior for o nível da tag, maior vai ser o tamanho da fonte apresentada na tela e sua importância no SEO.



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


TEXTO

- As tags de texto, definem diferentes formatações para diversos tipos de texto. Desde estilos de fonte, parágrafos, quebra de linha ou até mesmo spans. Iremos conhecê-las:
 - `<p>` - Sendo a principal tag de texto, é usada para criar um parágrafo.
 - `` - Mesmo tendo a sua funcionalidade parecida com o uso dos parágrafos, os spans geralmente são utilizados para agrupar uma pequena informação.
 - `` - Deixa o texto em negrito.
 - `<i>` - Deixa o texto em itálico.
 - `<hr>` - Cria uma linha horizontal.
 - `` - Adiciona importância (SEO) no texto.

TAGS DE MEDIAS

- As tags de medias ajuda na criação de experiências multimídia em uma página da web. Isso é, adicionar em nossas páginas imagens, vídeos, áudios e entre outros conteúdos de medias.

IMAGEM

```

```

O src vem de source(fonte), ele é atributo da tag , nele vai conter o caminho da imagem que será inserida por você.

O atributo “alt” é recomendado que você sempre utilize para que pessoas com deficiências saibam (através de um leitor) do que se trata a imagem na página.

AUDIO

- A tag <audio> cria um reprodutor de áudio em uma página da web. Ele oferece suporte a controles de mídia, como reprodução, pausa, volume e mudo .

```
<audio controls>  
  <source src="arquivo.mp3" type="audio/mpeg" />  
</audio>
```

Para ativa os controles de áudio e reproduzir, pausar, volume, mudo e download a tag audio precisa do atributo “controls”.

Os formatos de áudio suportados são MP3 , WAV e OGG.

TAG <SOURCE>

- src - URL ou nome do arquivo do áudio a ser reproduzido.
- type - Formato de arquivo do áudio

```
<audio controls>  
  <source src="arquivo.mp3" type="audio/mpeg" />  
</audio>
```

VÍDEO

- A tag `<video>` cria um player de vídeo em uma página da web. Este player oferece suporte à reprodução de vídeo diretamente dentro da página com controles de mídia, como iniciar, parar, volume e outros.

```
<video width="320" controls>  
  <source src="video.mp4" type="video/mp4" />  
</video>
```

Este elemento suporta três formatos diferentes: MP4, WebM e OGG

VÍDEO

- O elemento `<video>` também pode reproduzir arquivos de áudio, mas a tag `<audio>` fornece uma melhor experiência do usuário
- `height` - Define a altura do player de vídeo
- `width` - Define a largura do player de vídeo

I F R A M E

- A tag `<iframe>` renderiza outra página ou recurso HTML em uma página. É comumente usado para incorporar uma página HTML ou um documento PDF.
- A renderização de páginas é um pouco atrasada porque a página é carregada de um servidor remoto.

```
<iframe src="https://stagetree.com.br" title="Desenvolvimento de Software" style="width: 100%; height: 400px"></iframe>
```


I F R A M E

- As páginas com frames não podem ser manipuladas com JavaScript quando a página vem de um domínio diferente, o mesmo com oposto.
- Um bom exemplo da utilização do elemento `<iframe>` é incorporar um vídeo do YouTube em nossa página. Essa tag será responsável por realizar essa ação.
- Para adicionar um vídeo do YouTube em sua página basta clicar em compartilhar (no vídeo) e escolher a opção “incorporar”, logo em seguida colocar todo o código do `<iframe>` no documento HTML.

CANVAS

- A tag `<canvas>` cria um container onde os gráficos podem ser desenhados com JavaScript.
- Canvas é uma boa escolha para desenvolvimento de jogos com uso intensivo de gráficos, então se você deseja criar um jogo em JavaScript a utilização da tag `<canvas>` será essencial.
- O elemento `<canvas>` pode renderizar gráficos 2D e 3D (como linhas, círculos, retângulos, polígonos, imagens de bitmap, texto e muito mais) usando JavaScript.

CANVAS

- Se o navegador não suportar tela, o texto dentro do elemento `<canvas>` será exibido, em nosso caso, será a mensagem “Opa, correu um erro!”

```
<canvas id="jogo" width="400" height="250">  
  Opa, ocorreu um erro!  
</canvas>
```

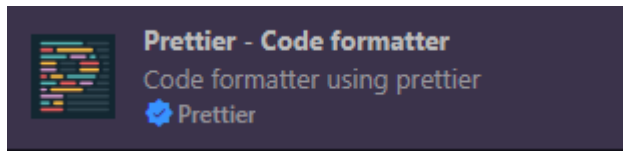
EXERCÍCIO

- Criar um arquivo de texto sobre vocês.
- Salvar em PDF
- Colocar um site qualquer dentro do arquivo HTML utilizando a tag `<iframe>`
- Incluir esse arquivo PDF dentro do HTML
- Incluir um Vídeo dentro de seu HTML utilizando a tag `<video>`
- Incluir um Vídeo(Youtube) dentro de seu HTML utilizando a tag `<iframe>`
- Incluir um Áudio/Música de sua preferência utilizando a tag de `<audio>`

EXERCÍCIO

- Logar na sua conta do Gmail.
- Entrar no Google Drive.
- Criar uma pasta chamada “html”
- Dentro dessa pasta, fazer o upload de um arquivo mp3
- Gerar um link compartilhável desse arquivo para qualquer pessoa.
- Pegar esse link e colocar em uma tag <áudio> dentro de seu arquivo HTML.

IDENTAR CÓDIGO



CTRL + A

ALT + SHIFT + F

CLICAR EM CONFIGURAR

SELECIONAR PRETTIER

DIV

- A tag `<div>` é utilizada para dividir elementos para fins de estilos (usando `class` ou `id`). Ele deve ser utilizado somente quando não tiver outro elemento de semântica.
- Mesmo tendo a sua funcionalidade parecida com o `` que são geralmente utilizados para guardar uma pequena informação. A `<div>` é usada para uma divisão de um conteúdo pois, como o uso dela ajuda a quebrar os elementos em linhas, deixando melhor a visualização.

```
<div>  
  <p>Entrou dentro de uma div</p>  
</div>
```

LINK

- A tag `<a>` (ou elemento âncora) é um Hiperlink, que é usado para vincular uma página a outra. O conteúdo dentro de cada `<a>` precisará indicar o destino do link utilizando o atributo `href`. Se a tag `<a>` não tiver o atributo `href`, ela será apenas um espaço reservado para um hiperlink.
- Por padrão, os links aparecem da seguinte forma em todos os navegadores.
 - Um link não visitado está sublinhado e azul.
 - Um link visitado é sublinhado e roxo.
 - Um link ativo está sublinhado e vermelho;

LINK

```
<a href="http://stagetree.com.br">stagetree</a>
```

TABELAS

- A criação de uma tabela em uma página HTML é feita com a tag `<table>`.
- Entretanto, somente essa tag não é o suficiente para formatar a tabela da maneira correta, pois ela precisa de outras tags para exibir a formatação adequada. Confira quais são eles.

TABELA - TR

- Em todas as tabelas existem linhas. Para isso, utilizamos a tag `<tr>`, ela pode ser inserida em diferentes áreas da tabela, como no cabeçalho, no corpo e no rodapé.

TABELA - TH

- A tag `<th>` é utilizada para inserir o cabeçalho na tabela, ela pode ser inserida em diferentes áreas. Na região do corpo da tabela, ela serve para dar destaque e identificar cada coluna específica.

TABELA - TD

- Além das linhas, toda tabela precisa de uma coluna. Isso significa que é preciso adicionar a tag `<td>` em todas as linhas para criar as colunas desejadas e inserir o conteúdo a ser exibido.

OUTRAS TAGS DE TABELA

- Definindo a legenda de uma tabela: `<caption>`
- Criando um rodapé: `<tfoot>`
- Especifica um grupo de uma ou mais colunas em uma tabela para formatação: `<colgroup>`
- Agrupa o conteúdo do cabeçalho em uma tabela: `<thead>`
- Agrupa o conteúdo do corpo em uma tabela: `<tbody>`
- Agrupa o conteúdo do rodapé em uma tabela: `<tfoot>`

EXEMPLO

```
<table>
  <tr>
    <th>Company</th>
    <th>Contact</th>
    <th>Country</th>
  </tr>
  <tr>
    <td>Alfreds Futterkiste</td>
    <td>Maria Anders</td>
    <td>Germany</td>
  </tr>
  <tr>
    <td>Centro comercial Moctezuma</td>
    <td>Francisco Chang</td>
    <td>Mexico</td>
  </tr>
</table>
```

EXERCÍCIO 1

- Utilizar (ROWSPAN e COLSPAN)

Classe 1			
Classe 2			

LISTAS

- Vamos agora falar um pouco sobre as listas (ordenadas e desordenadas) e como elas funcionam. As listas são muito importantes quando queremos listar itens na página.
- O elemento `` é usado para representar um item que faz parte de uma lista. Este item deve estar contido em um elemento pai: uma lista ordenada (``) ou uma lista desordenada (``).

LISTAS ORDENADAS

- As listas ordenadas (ou numeradas) são usadas para indicar alguma sequência ou numeração.

```
<ol>  
  <li>HTML</li>  
  <li>CSS</li>  
  <li>JavaScript</li>  
  <li>React</li>  
</ol>
```

LISTAS DESORDENADAS

- As listas desordenadas são usadas para listar itens, sem se preocupar com sua sequência. Chamamos apenas de lista de marcadores. Ela segue o mesmo padrão da ordenada apenas mudando de `` para ``. Em seu visual ele mudará de números para pontos.

```
<ul>  
  <li>HTML</li>  
  <li>CSS</li>  
  <li>JavaScript</li>  
  <li>React</li>  
</ul>
```

FORMULÁRIO

- Formulários são um dos principais pontos de interação entre o usuário e sua página. Um formulário HTML é feito de um ou mais widgets. Esses widgets podem ser campos de texto, caixas de seleção, botões, checkboxes, radio buttons e entre outros elementos HTML.
- Para construir o nosso formulário de contato, vamos utilizar os seguintes elementos: `<form>`, `<label>`, `<input>`, `<textarea>` e `<button>`.

FORM

- O elemento `<form>` define um formulário e os seus atributos definem a maneira como esse formulário se comporta.

```
<form action="/" method="post"></form>
```

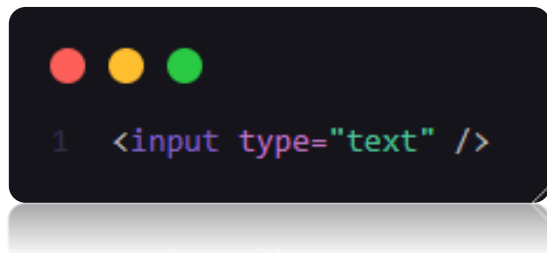
L A B E L

- O elemento `<label>` é a maneira formal de definir um rótulo. Esse elemento é importante para acessibilidade - quando implementados corretamente, os leitores de tela falarão o rótulo de um elemento de formulário juntamente com as instruções relacionadas.

```
<label>Nome:</label>
```

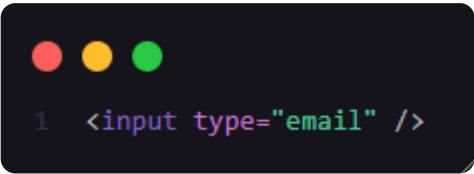
INPUT

- O elemento `<input>` é usado para criar controles interativos que receber dados do usuário. A semântica de um `<input>` varia consideravelmente dependendo do valor de seu atributo `type`.



INPUT

- No input temos o atributo type. Esse atributo define o tipo do nosso input.
Exemplo: o type="email" ele define que o campo aceita só endereço de email.

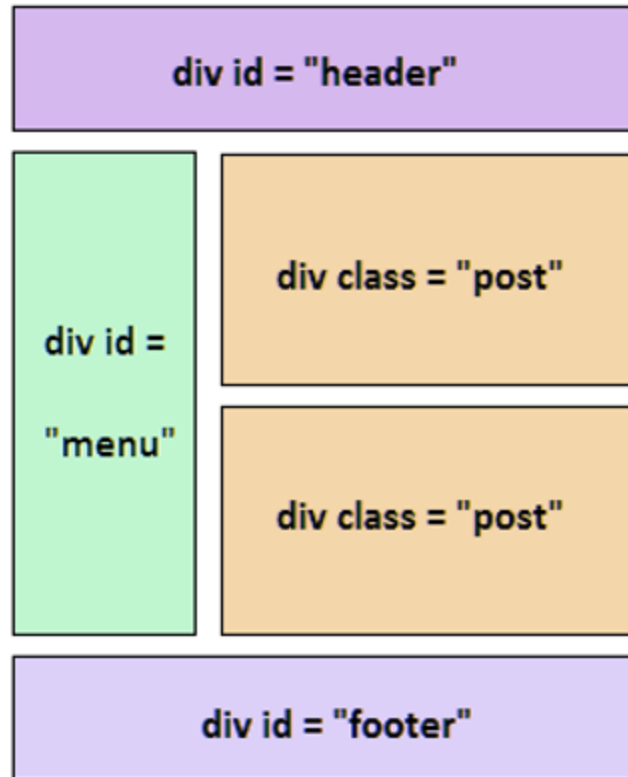


```
1 <input type="email" />
```


INPUT

- O tipo padrão de um input é text, se este atributo não for especificado.
- Um botão sem comportamento padrão: type="button"
- Uma caixa de marcação: type="checkbox"
- Um controle para especificar cores: type="color"
- Um controle para inserir uma data: type="date"
- Um campo para editar um endereço de e-mail: type="email"
- Um controle que permite ao usuário selecionar um arquivo: type="file"
- Um campo de senha cujo valor é ocultado: type="password"
- Um campo de texto com uma só linha para digitar termos de busca: type="search"

EXERCÍCIO 2



- Colocar cor correspondente utilizando o `background-color`.
- Colocar os `id` e `class` correspondente em cada `div`.

EXERCÍCIO 3

- LINK: Colocar um link de um site dentro footer.
- TABELA: dentro de post1 colocar uma tabela.
- FORM: dentro de post2 colocar um formulário.
- H1: colocar na div header um texto utilizando a tag `<h1>`.
- LISTA: na div de menu colocar uma lista não ordenada utilizar a tag ``.
- DICA: Utilizem o que estudados em sala de aula. Lembre-se de usar o atributo `height` e `width`.

EXERCÍCIO 4

- FORM: Colocar um formulário de contato básico no footer.
- VIDEO/IFRAME: dentro de post1 colocar um vídeo.
- AUDIO: dentro de post2 colocar um áudio/música de sua preferência.

EXERCÍCIO 5



- Fazer essa lista de acordo com os conhecimentos passados.

EXERCÍCIO 6



- Fazer essa lista de acordo com os conhecimentos passados.

EXERCÍCIO 7

Receita 2 SALADIM DE FRUTAS

Ingredientes:

3 bananas nanicas maduras
3 maçãs
5 laranjas
10 morangos
1 mamão pequeno

Modo de fazer:

1. Descasque as bananas, as laranjas e o mamão.
2. Separe duas laranjas para espremer e tirar seu suco.
3. Corte todas as frutas em pedaços pequenos.
4. Coloque tudo em uma travessa.
5. Acrescente o suco das duas laranjas e misture tudo.

- Fazer essa lista de acordo com os conhecimentos passados.

EXERCÍCIO 8

NOME DA SUA EMPRESA			
SERVIÇO DE MANUTENÇÃO EM AUTOMÓVEL			OS N°
Cliente		Telefones	
Endereço			
Bairro		Cidade	
CPF/ CPNJ		RG / IE	
Veículo / Ano			
Marca		Placa	
N° Chassis		N° Patrimônio	
Defeito/Reclamação			
SERVIÇOS EXECUTADOS NO LOCAL			
Garantia:		Próxima Revisão:	
Mão de Obra R\$:	Peças R\$:	Total R\$:	
Cliente _____	Técnico _____	DATA: / /	

FIELDSET

- O elemento `<fieldset>` é usado para agrupar elementos em um formulário.

```
<fieldset></fieldset>
```

LEGEND

- O elemento `<legend>` representa um rótulo para o conteúdo do `<fieldset>`.

```
<legend>Contato</legend>
```

EXEMPLO FORMULÁRIO

```
<form action="/" method="post">  
  <label>Nome:</label>  
  <input type="text" />  
  
  <label>Email:</label>  
  <input type="email">  
  
  <input type="submit" value="Enviar" />  
</form>
```

TAGS SEMÂNTICAS

- Na vida sempre temos uma forma correta de fazer as coisas, no HTML não é diferente. As tags semânticas além de deixar o código melhor para o SEO, ele ajuda outros desenvolvedores a entender seu código.
- Informações importantes
 - As tags semânticas não têm nenhum efeito na apresentação na página.
 - As tags semânticas tem significado e deixam seu conteúdo claro.

TAGS SEMÂNTICAS

- Elementos não semânticos:
 - div, span
- Elementos semânticos:
 - Form, table, nav, aside, article, footer, section
- Veja que <div> é amplo, mas a tag <footer> dá um significado (que é o rodapé).

TAGS SEMÂNTICAS

- Portanto em vez de você fazer assim:

```
<div>Sou o rodapé</div>
```

- Faça assim:

```
<footer>Sou o rodapé</footer>
```

SECTION

- O elemento `<section>` é utilizada para marcar as seções de conteúdo de uma página. Com Esse elemento agrupamos de forma lógica nosso conteúdo, separando a informação em áreas diferentes. O principal objetivo é retirar essa responsabilidade das divs.

```
<section></section>
```

ARTICLE

- O elemento `<article>` representa uma composição independente em um documento. Isso é, não precisa do resto do site para contextualizar. Este poderia ser um simples card, um artigo de revista ou jornal, um post de um blog, ou qualquer outra forma de conteúdo independente.
- Para um melhor SEO a tag `<article>` precisa de um título (`<h1>` - `<h6>`) acompanhado

```
<article></article>
```


ASIDE

- O elemento `<aside>` representa uma seção de uma página que consiste de conteúdo que poderia ser considerado separado do conteúdo principal. Essas seções são, muitas vezes, representadas como barras laterais.
- Um bom exemplo de uso são os sites de notícias, a maioria tem uma seção lateral para ver outras notícias que não estão relacionadas com a notícia principal da página.

```
<aside></aside>
```

HEADER

- O elemento `<header>` (cabeçalho) representa um container para conteúdo introdutório (muitas vezes usado na primeira seção da página).

```
<header></header>
```

FOOTER

- O elemento `<footer>` representa um rodapé de um conteúdo. Assim como o `<header>` podemos utilizar ele em vários lugares pela página.

```
<footer></footer>
```

NAV

- O elemento `<nav>` (navegação) representa uma seção de uma página que aponta para outras páginas ou para outras áreas da página, ou seja, uma seção com grupos de links de navegação.

```
<nav></nav>
```

MAIN

- O elemento `<main>` define o conteúdo principal da página. Ele representa o conteúdo mais importante da página, que está diretamente relacionado ao tópico central do documento. É importante utilizar apenas uma vez esse elemento por arquivos HTML criado.

```
<main></main>
```

DETAILS

- O elemento HTML cria um widget de divulgação no qual as informações são visíveis apenas quando o widget é alternado para um estado "aberto". Um resumo ou rótulo deve ser fornecido usando o elemento.

```
<details>  
  <summary>Título</summary>  
  Aqui se escreve o texto  
</details>
```

S U M M A R Y

- O elemento HTML especifica um resumo, legenda ou legenda para a caixa de divulgação de um elemento. Clicar no elemento alterna o estado do elemento pai aberto e fechado.

```
<details>  
  <summary>Título</summary>  
  Aqui se escreve o texto  
</details>
```

FIGCAPTION

- Representa uma legenda ou uma legenda associada com uma figura ou ilustração descrita pelo resto dos dados do elemento <figure> que é seu elemento pai.

```
<figure>
  
  <figcaption>An elephant at sunset</figcaption>
</figure>
```


FIGURE

- É um elemento HTML que representa um escopo em que uma imagem será instanciada nesse local.

```
<figure>
  
  <figcaption>An elephant at sunset</figcaption>
</figure>
```

MARK

- Representa um trecho de destaque em um texto, por exemplo, uma sequência de texto marcado como referência, devido à sua relevância em um contexto particular, pode ser utilizado em uma página que mostra os resultados de uma busca onde todas as instâncias da palavra pesquisadas receberam destaque.

```
<p>O elemento <mark> é usado para <mark>destacar</mark> partes do texto</p>
```

TIME

- Representa o tempo tanto no formato de 24 horas ou como uma data precisa no calendário Gregoriano (com informações opcionais de tempo e fuso horário)

```
<article>  
|   <p>This article was created on <time pubdate>2011-01-28</time>.</p>  
</article>
```

```
<p>The concert took place on <time datetime="2001-05-15 19:00">May 15</time>.</p>
```

DATALIST

- O elemento HTML contém um conjunto de elementos que representam as opções permitidas ou recomendadas disponíveis para escolha em outros controles.

```
<label>Escolha o sabor do sorvete</label>
<input list="ice-cream-flavors">

<datalist id="ice-cream-flavors">
  <option value="Chocolate">
  <option value="Coconut">
  <option value="Mint">
  <option value="Strawberry">
  <option value="Vanilla">
</datalist>
```

DIALOG

- O elemento HTML representa uma caixa de diálogo ou outro componente interativo, como um alerta, inspetor ou subjanela que pode ser ignorado.

```
<dialog open>  
  <p>Ola mundo</p>  
  <button>OK</button>  
</dialog>
```

PROGRESS

- O elemento HTML exibe um indicador que mostra o andamento da conclusão de uma tarefa, geralmente exibido como uma barra de progresso.

```
<progress id="file" max="100" value="70"> 70% </progress>
```

METER

- Representa um valor escalar dentro de um intervalo conhecido ou um valor fracionário

```
<meter id="fuel"  
  min="0" max="100"  
  low="33" high="66" optimum="80"  
  value="50">  
  at 50/100  
</meter>
```

META TAGS

- SEO (Search Engine Optimization) é o conjunto de técnicas responsáveis por fazer com que determinada página de um site apareça nas primeiras páginas do Google. Quanto melhor for o SEO de um site, maiores as chances de que a página apareça logo de cara para quem pesquisa por determinado assunto.
- A tag `<meta>` define metadados (informações) sobre um documento HTML. As tags `<meta>` sempre vão dentro do elemento `<head>` e normalmente são usadas para especificar o conjunto de caracteres, descrição da página, palavras-chave, autor do documento e configurações da janela de visualização.

META TAGS

- Informações importantes:
 - Os metadados não serão exibidos na página, mas podem ser analisados por máquina.
 - Os metadados são usados por navegadores, mecanismos de busca e outros serviços da web.

TITLE

- Title define o título do documento. Ele é mostrado na aba da página do navegador.

```
<title>Nome da página</title>
```

DESCRIPTION

- Description contém a descrição do conteúdo da página, vários navegadores usam esta meta como descrição padrão da página quando é marcada.

```
<meta name="description"  
  content="Descrição sobre a página"  
>
```

CHARSET

- Este atributo define a codificação de caracteres usados na página. O conjunto de caracteres UTF-8 cobre quase todos os caracteres e símbolos do mundo. Sempre tenha ele em suas páginas.

```
<meta charset="UTF-8">
```

KEYWORDS

- Keywords é uma meta de palavras-chaves associadas ao conteúdo da página, contendo strings separadas por vírgulas.

```
<meta name="keywords" content="css, html, js">
```

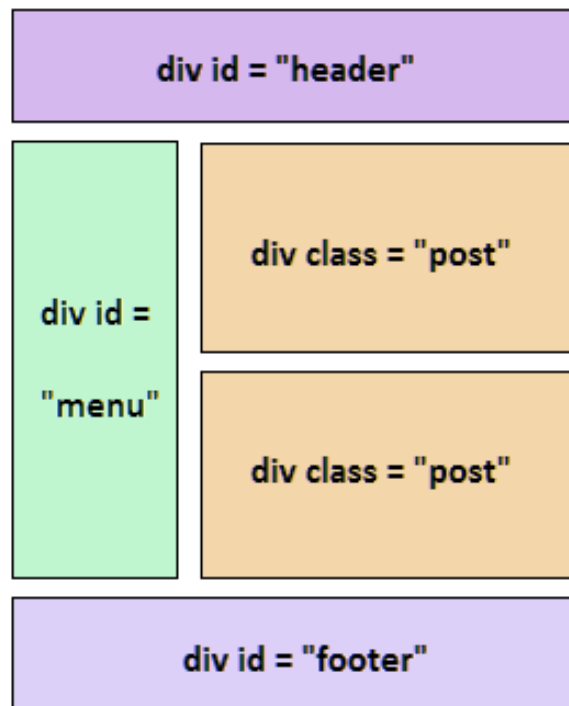
AUTHOR

- Author define o nome do autor do documento.

```
<meta name="author" content="José Victor">
```

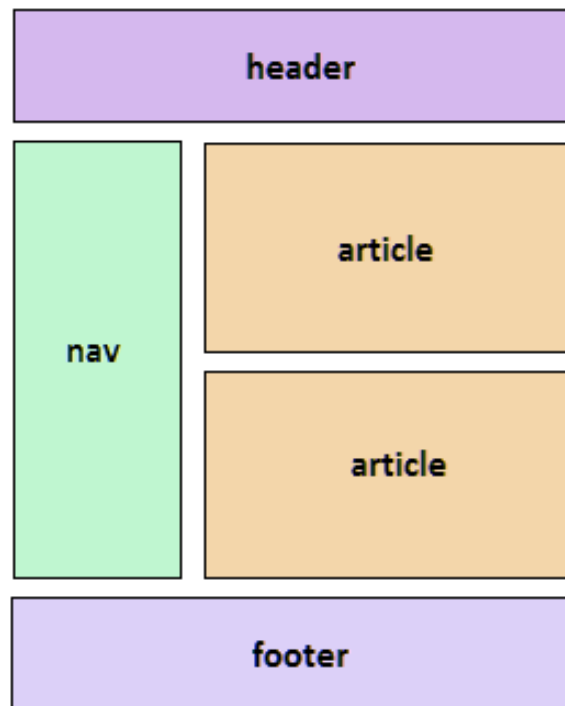
DIFERENÇAS

HTML4

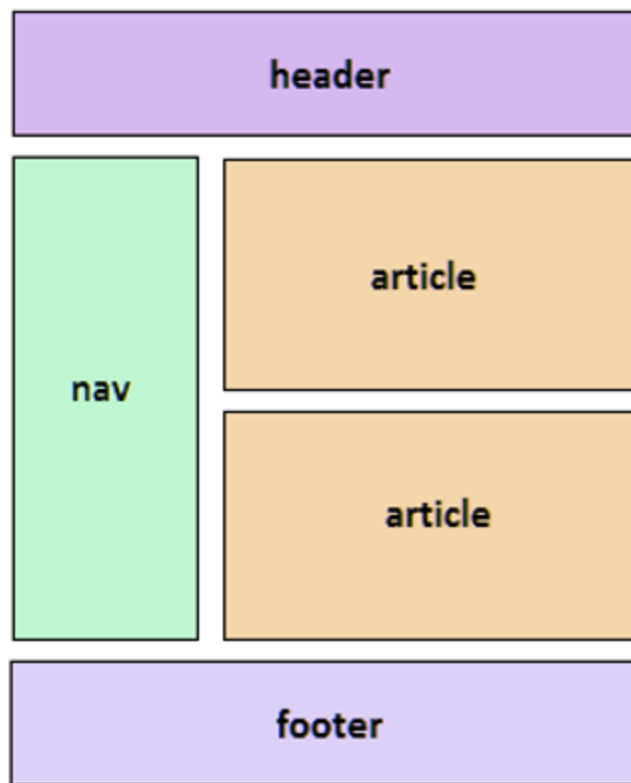


VS

HTML5



EXERCÍCIO



- Colocar cor correspondente utilizando o background-color.
- Colocar os componentes apresentados em uma documento/página html.

EXERCÍCIO

- LINK: Colocar um link de um site dentro footer.
- TABELA: dentro de article 1 colocar uma tabela
- FORM: dentro de article 2 colocar um vídeo um formulário
- DICA: Utilizem o que estudados em sala de aula. Lembre-se de usar o atributo height e width.

EXERCÍCIO 2

- FORM: Colocar um formulário de contato básico no footer.
- VIDEO/IFRAME: dentro de article 1 colocar um vídeo.
- AUDIO: dentro de article 2 colocar um áudio/música de sua preferência.

TRABALHO FINAL

- Defina as marcações dessa pagina de perfil.

