

Ensinando

HTML



CSS



PROFESSORES VICTOR FERNANDES & SABRINA ANDRADE



HTML

O que é?

HTML não é uma linguagem de programação; é uma linguagem de marcação, usada para definir a estrutura do seu conteúdo. HTML consiste de uma série de elementos, que você usa para delimitar ou agrupar diferentes partes do conteúdo para que ele apareça ou atue de determinada maneira. As tags anexas podem transformar uma palavra ou imagem num hiperlink, pode colocar palavras em itálico, pode aumentar ou diminuir a fonte e assim por diante.

Estrutura de páginas



`<html>`

`<head></head>`

`<body></body>`

`<footer></footer>`

`</html>`



Estrutura

- <!DOCTYPE html> Define o tipo de documento.
- <html> Define o documento html.
- <head> Define as configurações da página.
- <link> Adicionar folhas de estilo e scripts.
- <title> Título do documento.
- <body> Define o corpo do documento.
- <!-- --> Comentário dentro de um html.

Tags básicas

- <header>** Define conteúdos de um cabeçalho.
- <nav>** Define área com links para navegação.
- <main>** Container com conteúdo principal.
- <main>** Define uma seção.
- <section>** Define um artigo.
- <article>** Área para conteúdos à parte.
- <aside>** Para definir o rodapé de uma seção ou página.
- <footer>**

Exemplos de Tags



<h1>



<h2>



<h3>



<h4>



<h5>



<h6>

Exemplos de Tags

<title>

Define o título da página que é exibido na barra do navegador.

Insere uma imagem na página, especificando o caminho para a imagem e atributos como altura e largura.

<div>

Cria uma divisão ou contêiner genérico para agrupar e estilizar outros elementos.

<form>

Cria um formulário para coletar dados dos usuários, como campos de entrada de texto, botões e menus suspensos.

Cabeçalhos

Os elementos de cabeçalhos permitem especificar que certas partes do seu conteúdo são títulos ou subtítulos. Da mesma forma que um livro tem o título principal e os capítulos possuem títulos e subtítulos, um documento HTML também tem. HTML contém 6 níveis de título, <h1> - <h6>, embora você normalmente só use de 3 a 4:

```
<!-- 4 níveis de título -->
<h1>Meu título principal</h1>
<h2>Meu título de alto nível</h2>
<h3>Meu subtítulo</h3>
<h4>Meu segundo subtítulo</h4>
```

Formulários

<form> Define um formulário.

<label> Define a legenda ou descrição do input.

<input> Especifica um campo de entrada onde o usuário pode inserir dados.

<textarea> Campo de entrada de texto de várias linhas.

<button> Botão clicável.

<select> Campo com itens selecionáveis.

<option> Define as opções para o select.

Textos

<h1> Define um título com mais importância, podendo ter até o h6 sendo título menos importante.

<p> Define um parágrafo.

**** Define um texto importante / negrito.

**** Texto enfatizado.

<sub> Texto subscrito.

<sup> Texto sobrescrito.

Tags genéricas

<div> Para definir um conteúdo.

**** Define um conteúdo inline.

**** Texto em negrito.

<i> Define um texto em itálico.

Listas

**** Para listas não ordenadas.

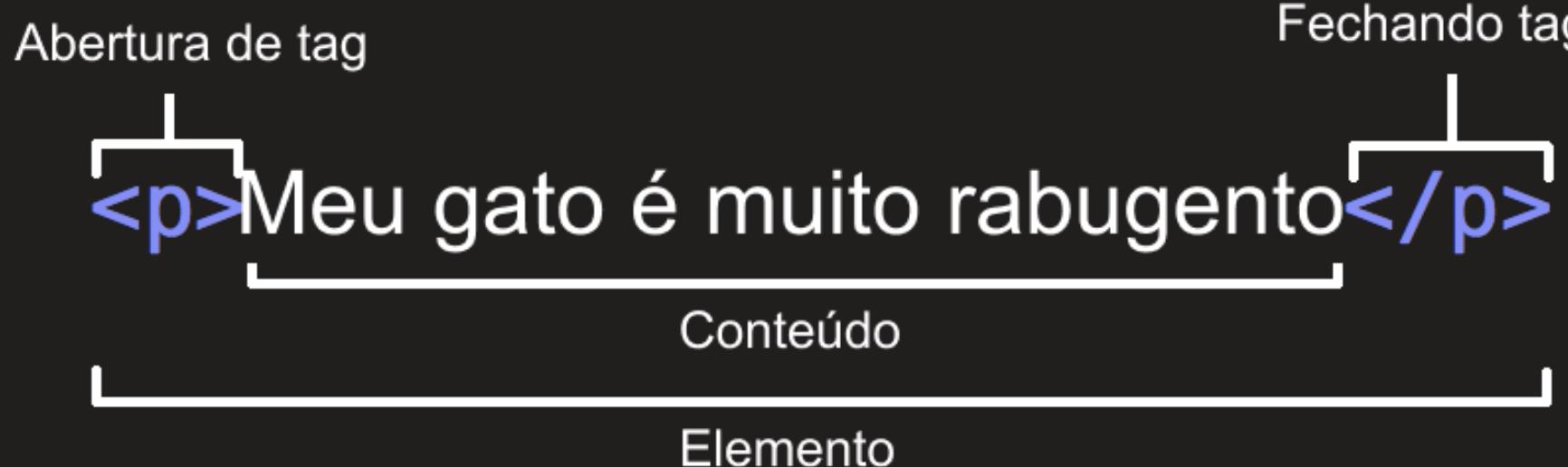
**** Define um lista ordenada.

**** Para colocar um item da lista.

Marcações em HTML

Um elemento HTML é separado de outro texto em um documento por "tags", que consistem no nome do elemento entre "<" e ">". O nome de um elemento dentro de uma tag é insensível a maiúsculas e minúsculas. Isto é, pode ser escrito em maiúsculas, minúsculas ou um mistura. Por exemplo, a tag `<title>` pode ser escrita como `<Title>`, `<TITLE>` ou de qualquer outra forma.

A marcação HTML inclui "elementos" especiais, como `<head>`, `<title>`, `<body>`, `<header>`, `<footer>`, `<article>`, `<section>`, `<p>`, `<div>`, ``, ``, `<aside>`, `<audio>`, `<canvas>`, `<datalist>`, `<details>`, `<embed>`, `<nav>`, `<output>`, `<progress>`, `<video>`, ``, ``, `` e muitos outros.



Alguns exemplos de marcações

<html> - Esse elemento envolve todo o conteúdo da página e às vezes é conhecido como o elemento raiz.

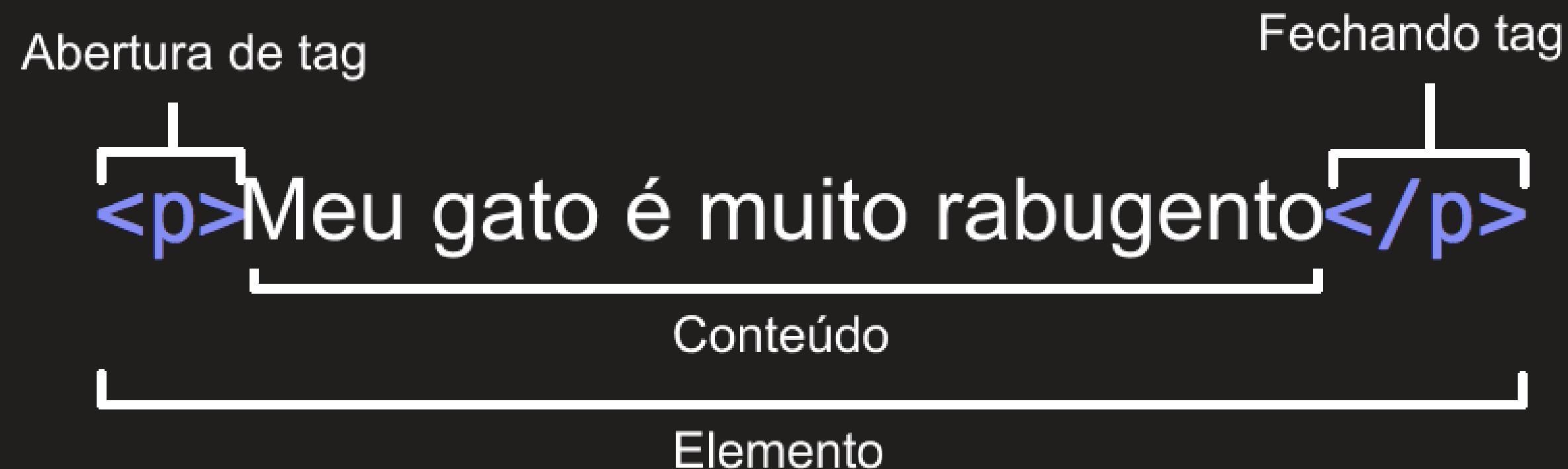
<head> - Esse elemento age como um recipiente de tudo o que você deseja incluir em uma página, ele também pode ser usado para colocar o estilo Css na página.

<body> - o elemento **<body>**. Contém todo o conteúdo que você quer mostrar ao público que visita sua página, seja texto, imagens, vídeos, jogos, faixas de áudio reproduzíveis ou qualquer outra coisa.

**** - isso incorpora uma imagem na nossa página na posição que aparece. Isso é feito pelo atributo src (source), que contém o caminho para nosso arquivo de imagem.

Elementos em HTML

Vamos explorar esse parágrafo mais profundamente.



As principais partes de um elemento são:

1. A tag de abertura: *Consiste no nome do elemento (no caso, p), envolvido em parênteses angulares de abertura e fechamento.* Isso demonstra onde o elemento começa, ou onde seu efeito se inicia – nesse caso, onde é o começo do parágrafo.
2. A tag de fechamento: *Isso é a mesma coisa que a tag de abertura, exceto que inclui uma barra antes do nome do elemento.* Isso demonstra onde o elemento acaba – nesse caso, onde é o fim do parágrafo. Esquecer de incluir uma tag de fechamento é um dos erros mais comuns de iniciantes e pode levar a resultados estranhos.
3. O conteúdo: *Esse é o conteúdo do elemento, que nesse caso é apenas texto.*
4. O elemento: *A tag de abertura, a de fechamento, e o conteúdo formam o elemento.*

Elementos também podem ter atributos, que parecem assim:

Atributo

`<p class="editor-note">Meu gato é muito rabugento</p>`

Atributos contém informação extra sobre o elemento que você não quer que apareça no conteúdo real. Aqui, *class* é o nome do atributo e editor-note é o valor do atributo. O atributo class permite que você forneça ao elemento um identificador que possa ser usado posteriormente para aplicar ao elemento informações de estilo e outras coisas.

Aninhando Elementos

Você pode colocar elementos dentro de outros elementos também – isso é chamado de aninhamento. Se quiséssemos afirmar que nosso gato é *muito* mal-humorado, poderíamos envolver a palavra "muito" em um elemento , o que significa que a palavra deve ser fortemente enfatizada:

<p>Meu gatinho é muito mal humorado.</p> 

Você precisa, no entanto, certificar-se de que seus elementos estejam adequadamente aninhados. No exemplo acima, abrimos primeiro o elemento `<p>`, depois o elemento ``; portanto, temos que fechar primeiro o elemento ``, depois o elemento `<p>`. O código abaixo está **incorrecto**:

```
<p>Meu gatinho é <strong>muito mal humorado.</p><strong> X
```

Os elementos precisam ser abertos e fechados corretamente para que eles estejam claramente visíveis dentro ou fora um do outro. Se eles se sobrepuarem conforme mostrado acima, seu navegador tentará adivinhar o que você estava tentando dizer, o que pode levar a resultados inesperados. *Então não faça isso!*

Elementos Vazios

Alguns elementos não possuem conteúdo e são chamados de *elementos vazios*. Considere o elemento `` que temos na nossa página HTML:

```

```

Ele contém dois atributos, mas não há tag `` de fechamento, e não há conteúdo interno. *Isso acontece porque um elemento de imagem não envolve conteúdo para ter efeito em si mesmo.* Sua proposta é incorporar uma imagem na página HTML no lugar que o código aparece.



CSS

CSS (Cascading Style Sheets ou Folhas de Estilo em Cascata) é uma linguagem de **estilo** usada para descrever a apresentação de um documento escrito em HTML. O CSS descreve como elementos são mostrados na tela, no papel, na fala ou em outras mídias. A versão atual e mais utilizada é a CSS3.



- Introdução ao CSS

Esse módulo inicia com o funcionamento básico do CSS, incluindo seletores e propriedades, escrever regras CSS, aplicar CSS ao HTML, como especificar comprimento, cor, e outras unidades em CSS, cascata e herança, o básico do modelo de caixas, e depuração de CSS.

1. Estilização do texto:

Esse módulo discute os fundamentos da estilização de textos, incluindo a definição de fontes, negritos, itálicos, linhas e espaçamentos de letras, sombras e outras propriedades de textos.



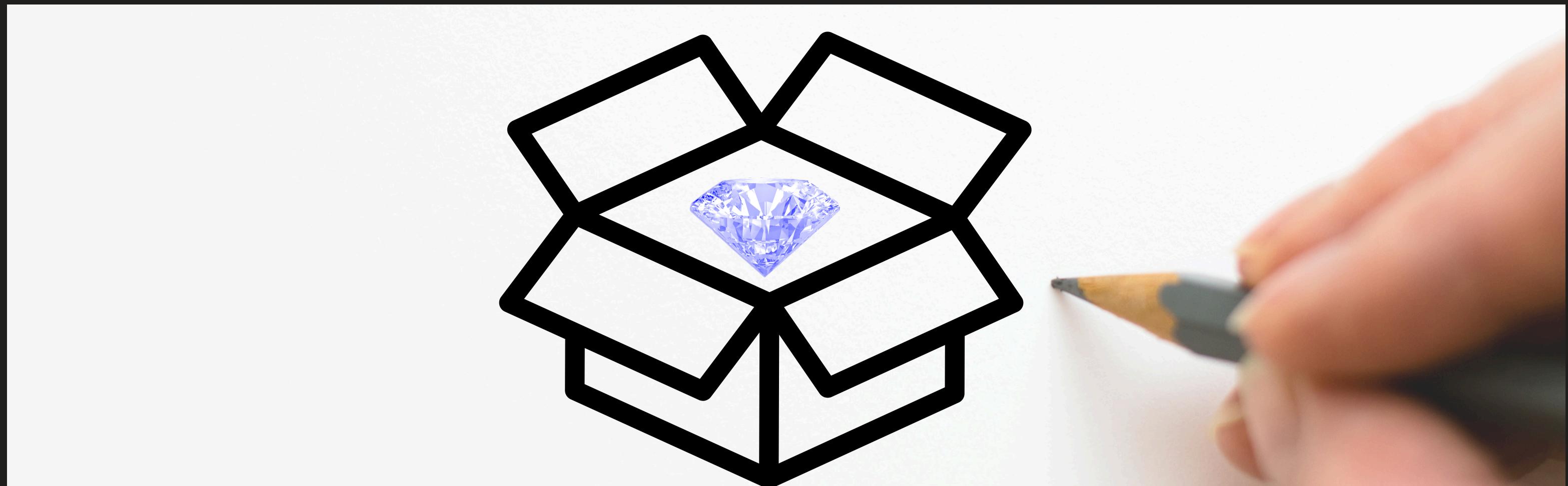
```
h1 {  
    color: red;  
    font-size: 100px;  
    text-align: center;  
}
```

A screenshot of a code editor showing a CSS rule for an h1 selector. The rule sets the color to red, the font size to 100px, and the text alignment to center. The code is written in a dark-themed editor with syntax highlighting.

Foi usado a estilização “color” para colocar a cor vermelha; “Font-size” para aumentar o tamanho da fonte; e “text-align” para alinhar ao meio da página;

2. Estilização de caixas:

Esse módulo foca na estilização de caixas, um dos passos fundamentais para o layout de uma página web. Nesse módulo nós recapitulamos o modelo de caixas, então abordamos o controle do layout das caixas definindo margens, bordas e preenchimento, cores de fundo customizadas, imagens e outras características extravagantes como sombras e filtros em caixas.

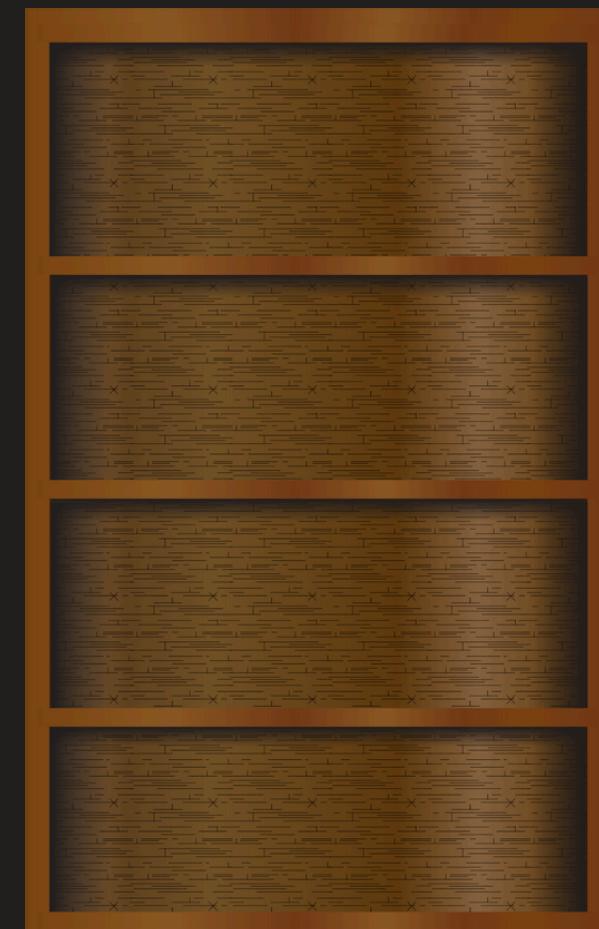


3. Layout CSS:

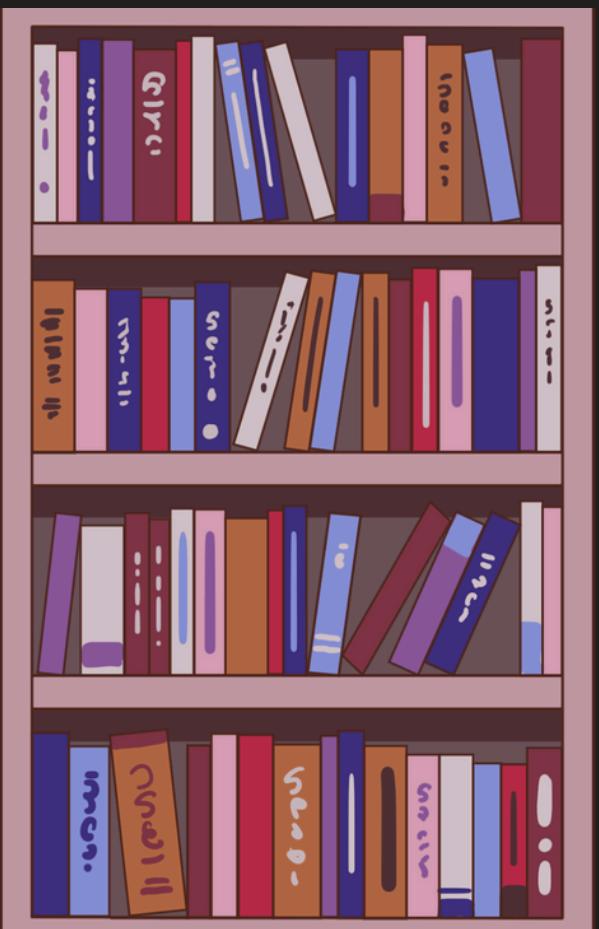
Neste ponto nós já vimos os fundamentos do CSS, como estilizar texto e estilizar e manipular caixas que contém seu conteúdo. Agora veremos como posicionar suas caixas no lugar certo em relação à janela de exibição e um ao outro. Nós cobrimos os pré-requisitos necessários e agora você pode se aprofundar na estilização do CSS com diferentes configurações de exibição, métodos tradicionais de layout envolvendo float e posicionamento, e modernas ferramentas para layout como flexbox.



Conteúdo



Flexbox



Layout

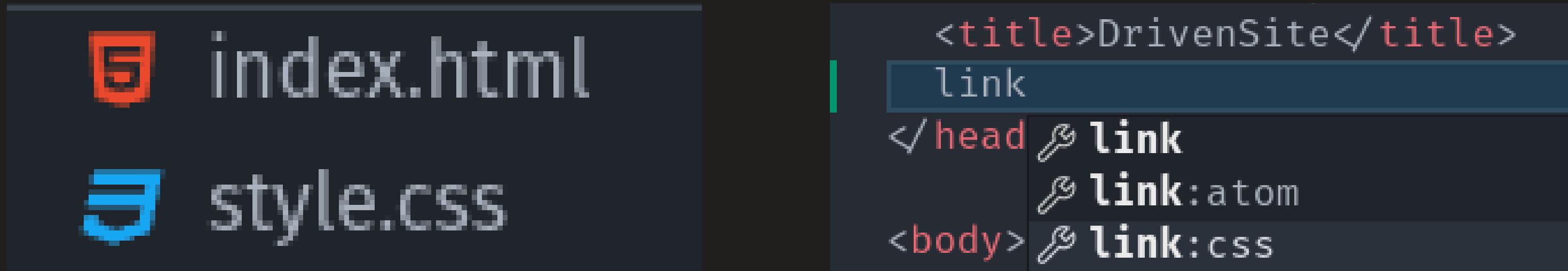
- Então, o que realmente é CSS?

Assim como o HTML, o CSS não é realmente uma linguagem de programação. Também não é uma linguagem de marcação – é uma linguagem de estilização. Isso significa que o CSS permite aplicar estilos seletivamente a elementos em documentos HTML. Por exemplo, para selecionar todos os elementos parágrafo de uma página HTML e tornar o texto dentro deles vermelho, você escreveria este CSS.



- Adicionando CSS para o nosso documento

A primeira coisa que precisamos fazer é falar para o HTML que temos algumas regras CSS que desejamos usar. Há três formas diferentes de aplicar CSS a um documento HTML que você normalmente vai encontrar, contudo, por enquanto, iremos olhar para o mais habitual e útil - vincular o CSS a partir do cabeçalho do seu documento.



Crie um arquivo na mesma pasta em que se encontra o seu documento HTML e salve-o como `style.css`. A extensão `.css` nos informa que se trata de um arquivo CSS.

```
<link rel="stylesheet" href="style.css" />
```



Textos

color: Define a cor do texto dentro de um objeto.

font-family: Define a família das fontes de um objeto.

font-size: Define o tamanho da fonte de um objeto.

font-weight: Define se um texto deve estar, por exemplo, em negrito.

font-style: Define se um texto deve estar, por exemplo, em itálico.



Caixas

border: Dá as definições da borda de um objeto.

border-radius: Serve para definir o quanto arredondada é a borda de um objeto.

display: Serve para modificar a maneira como um objeto será alocado na tela.



Layout

width: Define a largura de um objeto. Muito utilizado em divs e em imagens.

height: Anda junto de width e define a altura de um objeto.

margin: Define o espaço que deve ter entre um objeto e outro. Pode ser utilizado sozinho ou separadamente, como, por exemplo, margin-left, margin-right, margin-top e margin-bottom.

padding: É quase o mesmo que margin, mas invertido: define a distância que um objeto deve ter para com a borda de si mesmo.

background-color: Define a cor do fundo de um objeto.

Importando fonte do Google Fonts

```
@import url('https://fonts.googleapis.com/css2?family=Krona+One&family=Montserrat:wght@400;600&display=swap');  
/* Importa fontes do Google Fonts. "Krona One" e "Montserrat" são as famílias de fontes usadas no site. */
```

Estilizando o corpo da página

```
body{  
    height: 100vh;  
    box-sizing: border-box;  
    background-color: #000000;  
    color: #F6F6F6;  
  
    /* Define o estilo do corpo da página:  
     * - A altura é 100% da altura da janela de visualização (viewport).  
     * - 'box-sizing: border-box' garante que o padding e a borda sejam incluídos no cálculo do tamanho total dos elementos.  
     * - Cor de fundo preta (#000000).  
     * - Cor do texto branco (#F6F6F6)...*/  
}
```

Classe para trocar a cor do título

```
.titulo-destaque{  
    color: #2204FD;  
}  
/* Define uma cor azul (#2204FD) para os elementos que possuem a classe 'titulo-destaque'. */
```

Classe para estilizar e alinhar o principal

```
.apresentacao{  
    margin: 10% 15%;  
    display: flex;  
    align-items: center;  
    justify-content: space-between;  
}  
  
/* Estiliza a seção de apresentação:  
   - Margens definidas para afastar o conteúdo da borda (10% no topo e 15% nas laterais).  
   - Usa display:flexível para organizar os elementos horizontalmente.  
   - 'align-items: center' alinha os itens ao centro verticalmente.  
   - 'justify-content: space-between' distribui os itens horizontalmente com o máximo de espaço entre eles.. */
```

Classe para estilizar o conteúdo

```
.apresentacao__conteudo{  
    width: 615px;  
    display: flex;  
    flex-direction: column;  
    gap: 40px;  
}  
  
/* Estiliza o conteúdo da apresentação:  
   - Define uma largura fixa de 615px.  
   - Usa display:flexível para alinhar elementos em uma coluna.  
   - 'gap: 40px' define um espaço de 40px entre os elementos filhos.. */
```

Classe para colocar fonte e o tamanho do título

```
.apresentacao__conteudo__titulo{  
    font-size: 36px;  
    font-family: 'Krona One', sans-serif;  
}  
  
/* Estiliza o título da apresentação:  
   - Tamanho da fonte definido como 36px.  
   - Fonte 'Krona One' é aplicada ao título.. */
```

Classe para colocar fonte e o tamanho do texto

```
.apresentacao__conteudo__texto{  
    font-size: 24px;  
    font-family: 'Montserrat', sans-serif;  
}  
  
/* Estiliza o parágrafo de texto:  
   - Tamanho da fonte definido como 24px.  
   - Fonte 'Montserrat' é aplicada ao texto.. */
```

Colocando a largura da imagem

```
img {  
    width: 50%;  
}  
/* Define a largura da imagem como 45% da largura do container pai. */
```