
Codificação para Front-End

APRESENTAÇÃO

Hoje iremos iniciar a **codificação para Front-End** e desenvolver o nosso primeiro site do zero, utilizando **HTML5** e **CSS3**.

E para iniciar, precisamos escolher um programa chamado **IDE**, que significa **Integrated Development Environment**, que é só um jeito bonito de dizer **editor de código**, ou traduzindo, **ambiente de desenvolvimento integrado**.

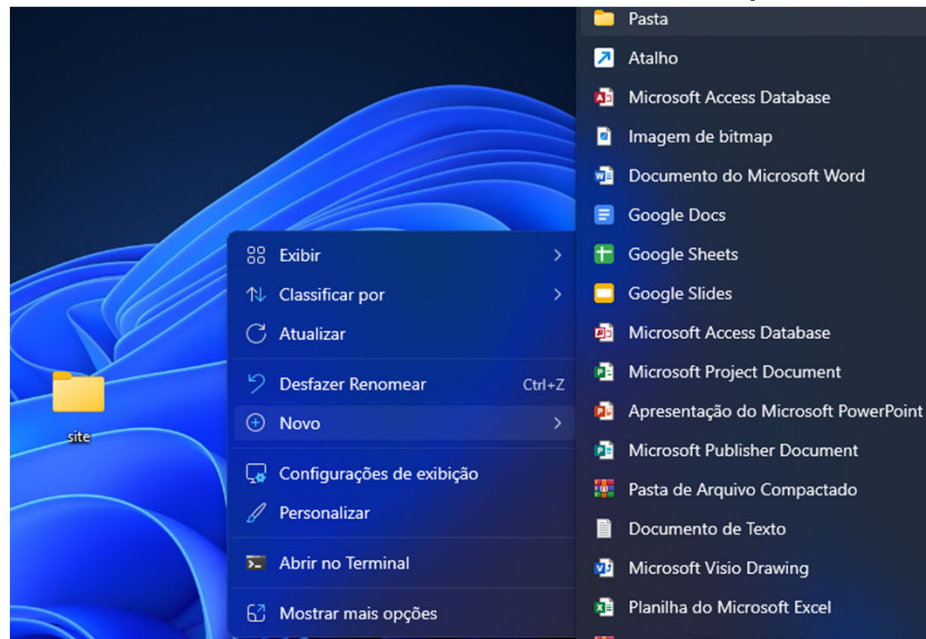
Existem vários editores de código por aí, mas não se preocupe, diferente dos **designers**, que precisam de varias ferramenta paga, como por exemplo o **Photoshop**, nós podemos usar ferramentas gratuitas, que é o que a maior parte dos desenvolvedores usam.

A questão é qual escolher. As mais utilizadas são **Visual Studio Code**, **Sublime Text**, **Eclipse**, **NetBeans**, **Notepad++**, **Dreamweaver** e por aí vai.

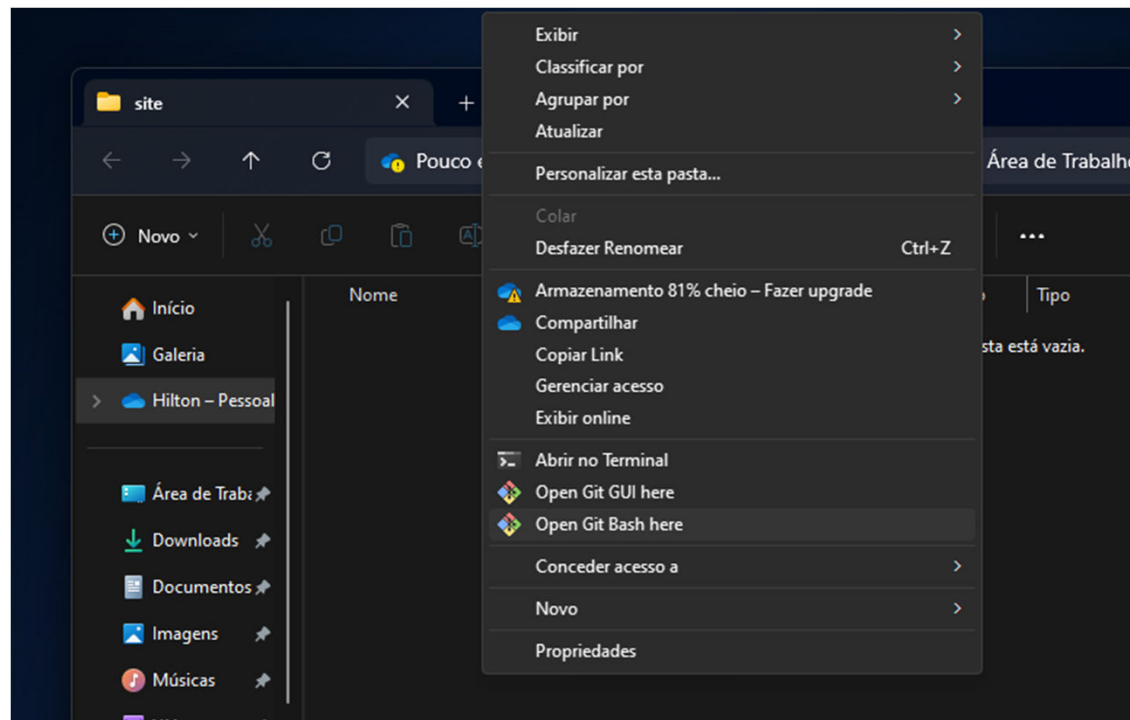
Para desenvolver nossas atividades, iremos utilizar o **Visual Studio Code**, fazemos em paralelo a utilização do **git** e **github**, para **salvar** e **versionar** nossos códigos.

Agora como já instalamos tudo para editarmos nosso código, precisamos criar uma pasta para organizar os arquivos do nosso site.

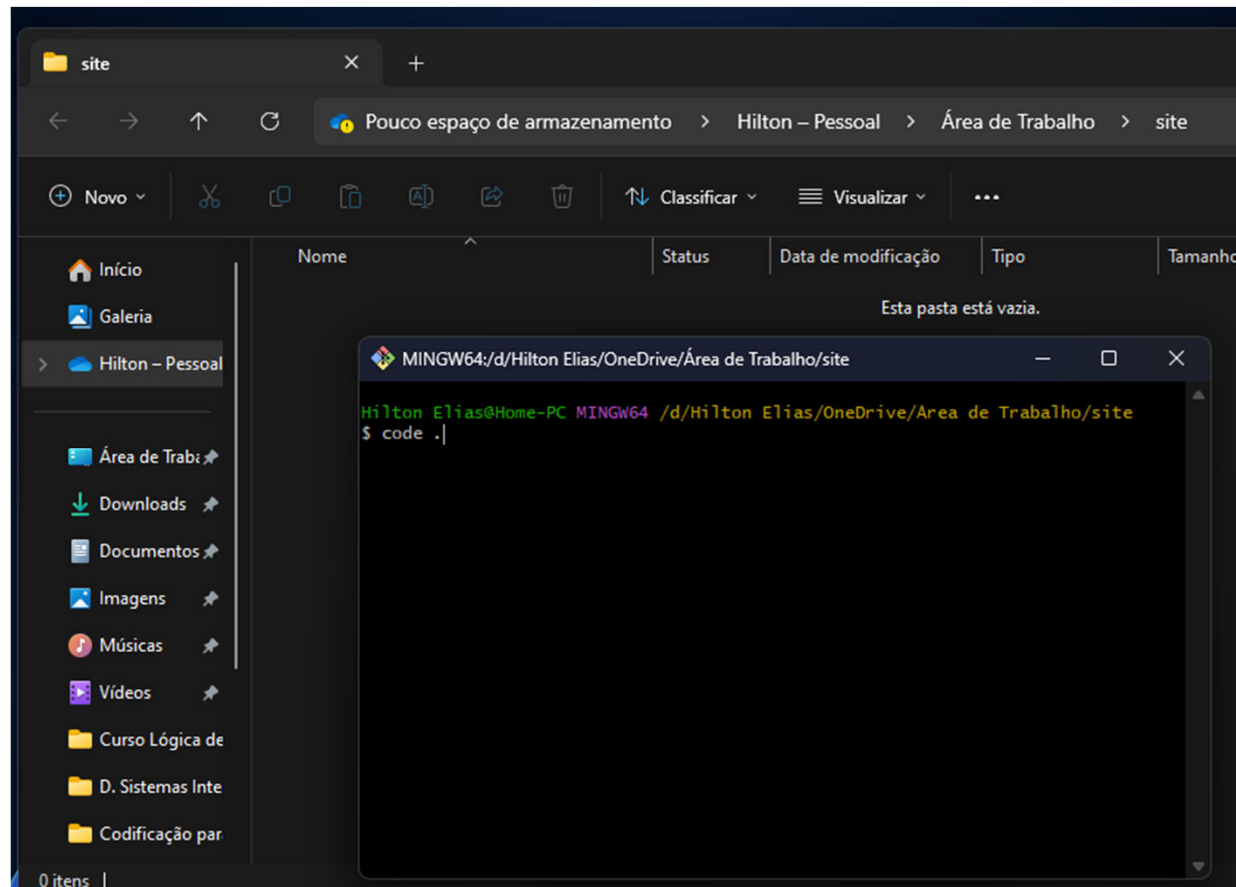
Vamos criar uma pasta chamada site no desktop, pois fica mais fácil de acessar, agora é só clicar aqui com o botão direito do mouse, novo, pasta e dar o nome de site, tudo em minúsculas.



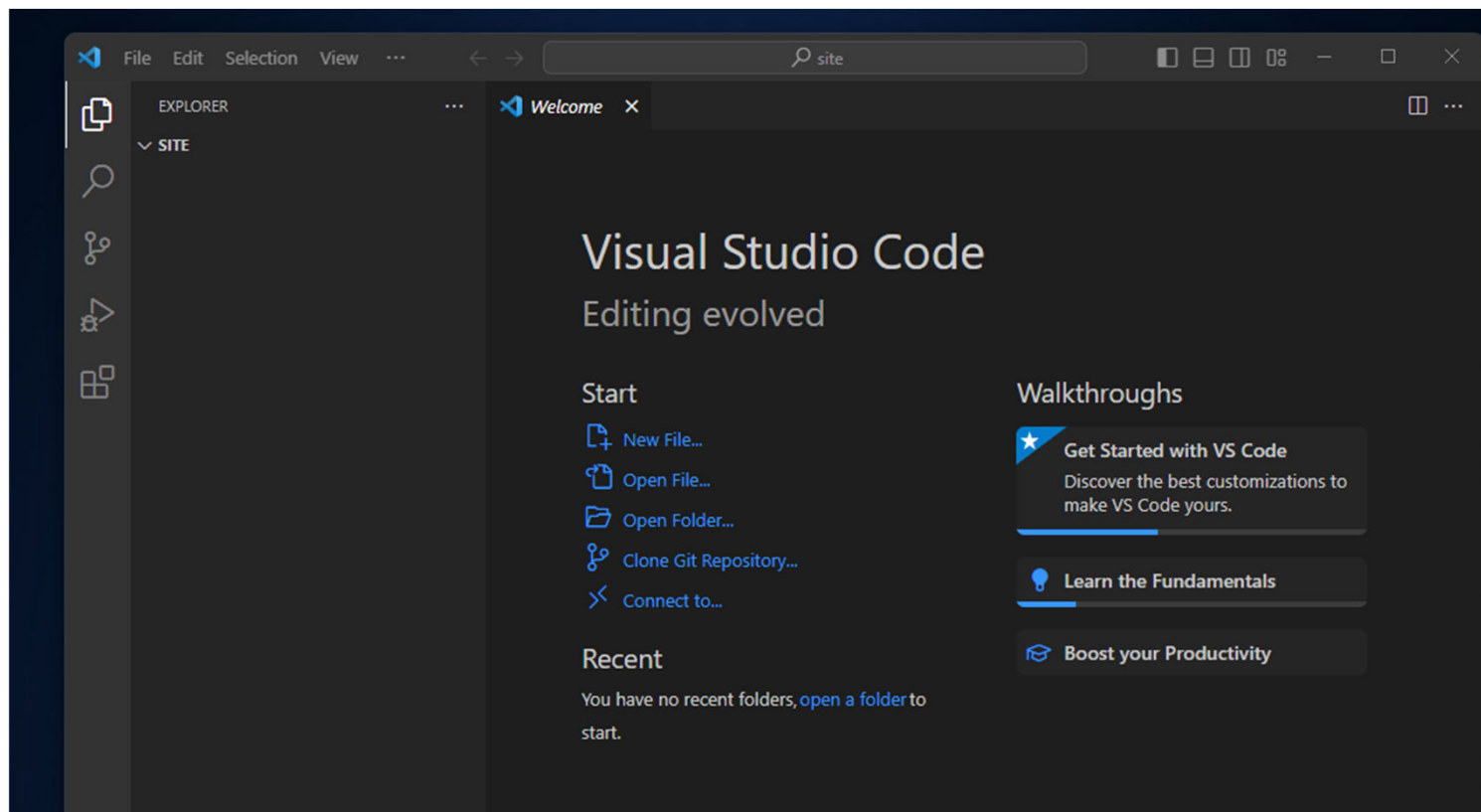
Agora é preciso apontar o programa IDE para aquela pasta que acabamos de criar. Vamos entrar na pasta site, clicar com o direito do mouse, mostrar mais opções, open git bash here.



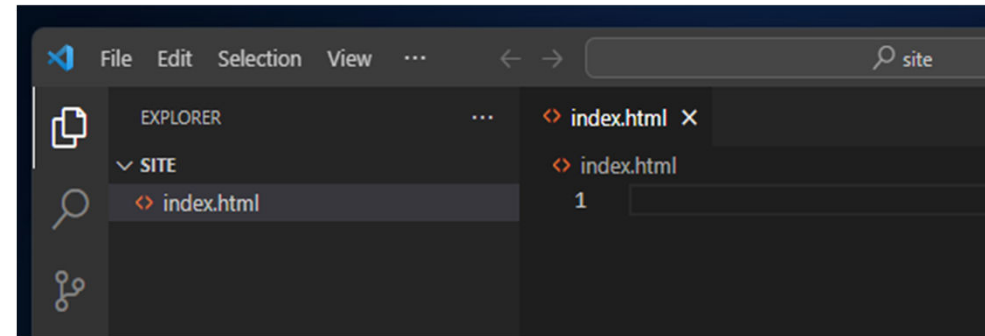
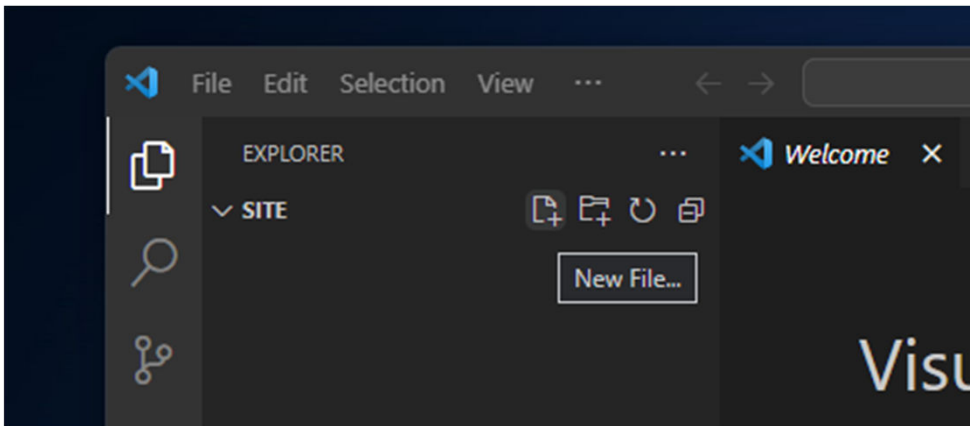
Agora com o terminal do git aberto, iremos executar o comando `code .`, para que seja aberto a pasta `site` no Visual Studio Code.



Agora os arquivos do nosso site irão aparecer do lado esquerdo, mas como vocês podem ver, não tem muita coisa acontecendo aqui, porque a gente não criou nada, simplesmente uma pasta.

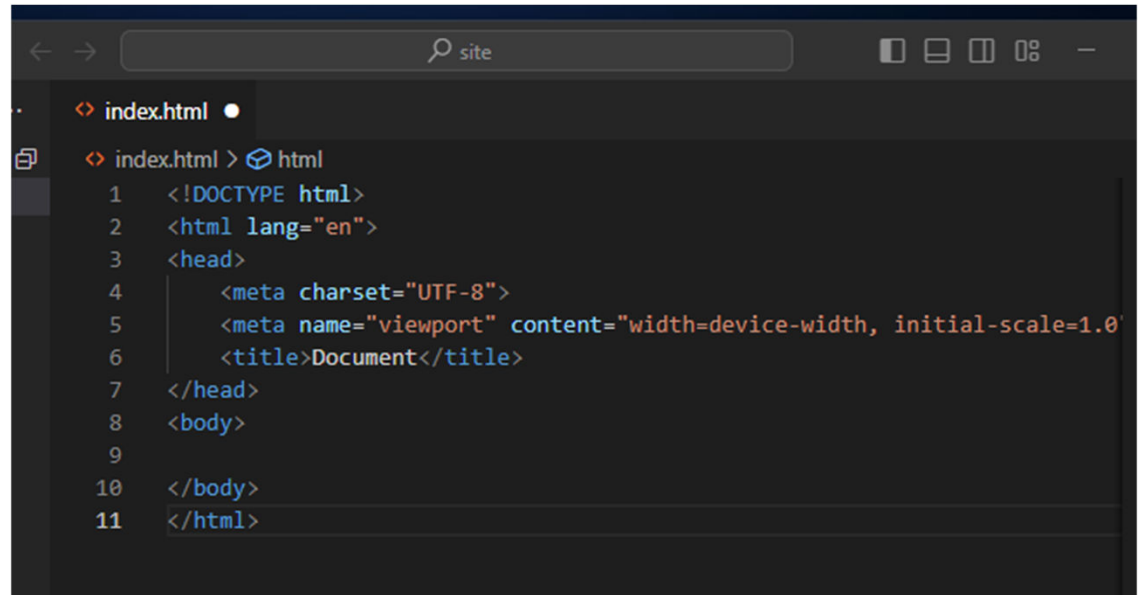
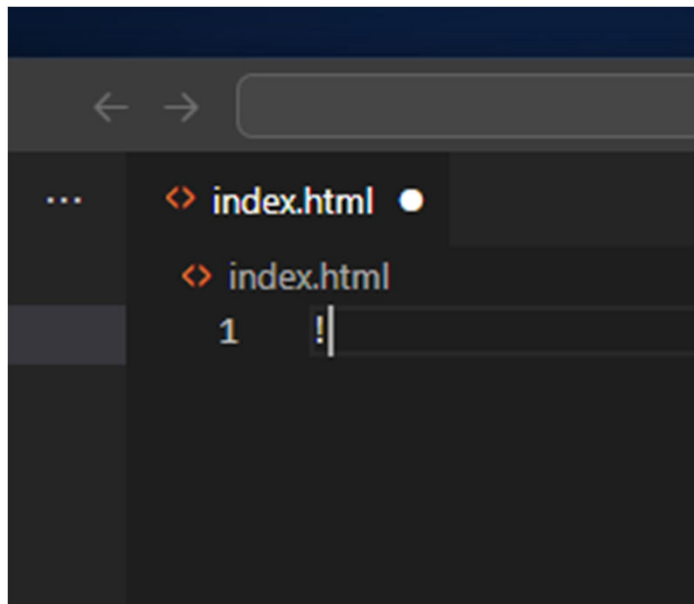


Vamos criar nosso primeiro arquivo, New File. Isso vai criar uma página em branco, vamos escolher um nome para esse nosso arquivo.



Como nosso primeiro arquivo do site, ele tem que se chamar **index** que significa em inglês **índice**. Isso é a nossa página inicial. Agora eu vou acrescentar **.html** para falar que esse nosso arquivo é um arquivo do tipo **HTML**. Quando codificarmos **CSS**, iremos criar arquivos **.css** e assim por diante.

Vamos criar nosso primeiro código em HTML, no Visual Studio Code, na linha 1 se colocarmos uma ! e dar Enter, ele inicia um código HTML simples para nos darmos início ao nosso site.



A primeira linha de um arquivo **HTML** precisa conter: **DocType HTML**. Isso serve para dizer ao navegador que é o que irá ler em nosso site, saber qual versão de HTML que nós estamos usando.

No momento isso significa **HTML5**. E por que a gente tem que falar isso? Antigamente existia varias versões de **HTML** e era preciso escrever um monte de coisa para dizer qual versão estava sendo utilizada.

Na próxima linha, temos um **elemento HTML**, também conhecido como **tag HTML**. Ela é representada por sinal <, o nome da **tag** e um sinal >. A primeira **tag**, se chama **html** mesmo.

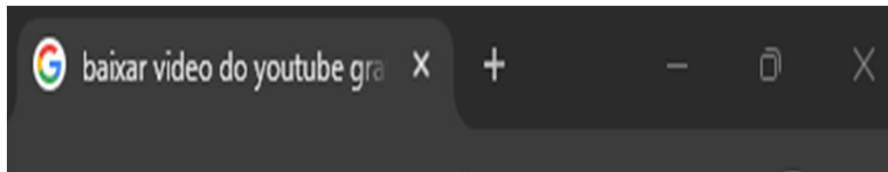
Todo o nosso site vai aqui dentro. Percebam que a **tag** se repetiu lá embaixo, mas com uma **barra** na frente. Isso significa que ela encerra o elemento que foi aberto aqui. Isso possibilita que novos elementos apareçam aqui dentro.

Dentro dessa **tag HTML** só podem ir dois elementos, o <**head**> e o <**body**>. Mas percebam que eles estão dentro, entre a **tag** <**html**> e a **tag que fecha a tag** </**html**>. E veja que cada elemento possui sua abertura e o seu fechamento. Qual única diferença que o fechamento é a mesma tag de novo, só que com uma barra na frente.

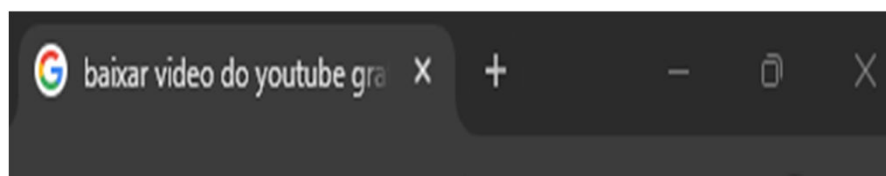
A linguagem **HTML** é feita com essa estrutura de árvore, onde elementos vão dentro de outros elementos e por aí vai.

Agora vamos ver o que cada um desses elementos aqui tem de diferente. O **<head>** é onde a gente coloca as informações sobre o site. Coisas, por exemplo, como o título da página.

Esse título vai aparecer aqui na aba do navegador. Esse título também aparece nos resultados de busca, veja só. Outra coisa que aparece também que a gente coloca no **<head>** é a descrição do site. Essa descrição aparece também aqui embaixo.



Tem também os ícones de favoritos. que são esses ícones que ficam aqui quando a gente favorita no navegador. Fora isso, a gente tem também vínculo com arquivos **CSS**, **arquivos de script**, **fontes e etc**.



Vamos começar do básico. Primeiro, vamos colocar um título na página.

Como que a gente faz isso?

Agora vamos escrever o primeiro **elemento**, que é o **elemento de título**, **<title>**.

Esse elemento especial que vai dentro do **<head>** significa o título da nossa página. Vai aparecendo em resultados de busca ou na aba do navegador, vamos escrever **Meu primeiro site**.

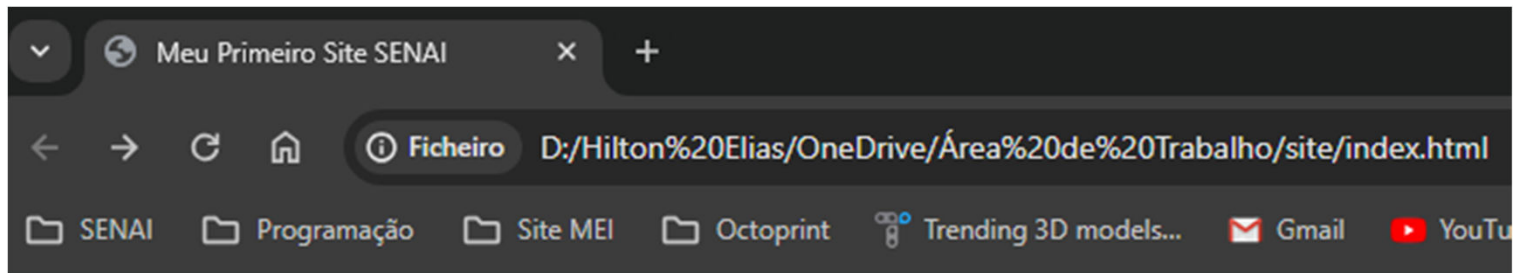
Agora vamos salvar e abrir no navegador e conferir se está tudo correto.

Criaremos um espaço dentro da **tag <body>**, que é onde se coloca todo o conteúdo do site, como **títulos, botões, imagens, etc.** Essa é a grande diferença entre o que vai dentro da **tag <head>**, que são apenas informações do site.

Nesta parte inserimos o primeiro elemento e um dos mais importantes, o elemento **<h1>**. Esse elemento significa título. Então ele é um título que vai aparecer no meio do conteúdo do nosso site e é utilizado o tempo todo, além de ser muito importante para fazer chamadas para os conteúdos.

Esse **h** vem de **heading** em inglês, que significa **chamada ou título**, o **1** serve para indicar qual é a importância desse título. Existem seis títulos, então **h1 ao h6**. Mais na frente veremos melhor. É o **único elemento** que possui essa regra de números.

Aqui no **<h1>** vou colocar o texto do título, iremos colocar uma série de matérias com link, e será a base de nossa primeira página. Vamos ao nosso site e vejam que o título apareceu dentro do navegador.

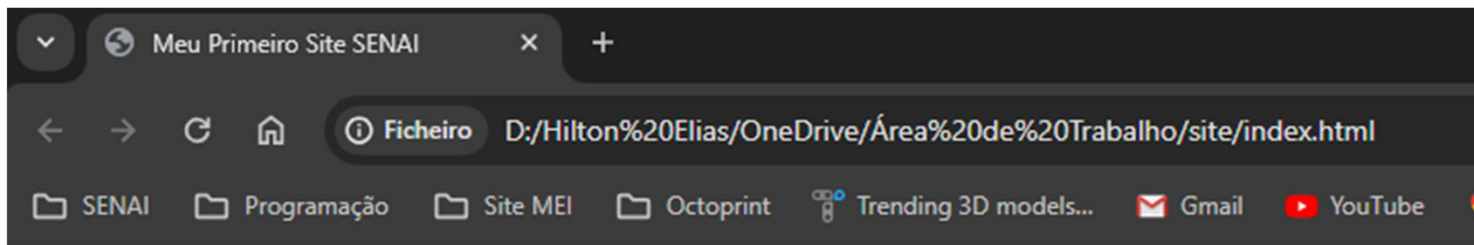


Primeiro dia de Codificação

O **título** da **tag <title>** do **<head>** foi para a aba do navegador, que é bem diferente do **título <h1>** que é onde colocamos o conteúdo do site, dentro da **tag <body>**.

A partir desse ponto, acrescentaremos mais uma linha e criaremos um elemento chamado **<p>**, esse elemento serve para colocar um texto de parágrafo, por isso leva esse nome. Elementos de parágrafo normalmente possuem várias linhas.

Agora vamos acrescentar aqui um texto para chamar atenção para a notícia.
<p>**Notícias da Indústria**</p>.



Primeiro dia de Codificação

Notícias da Indústria

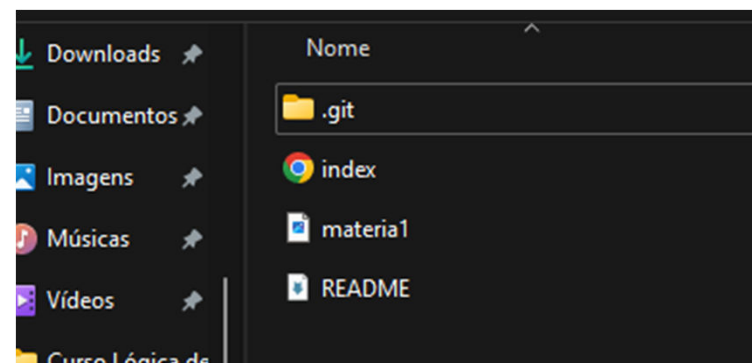
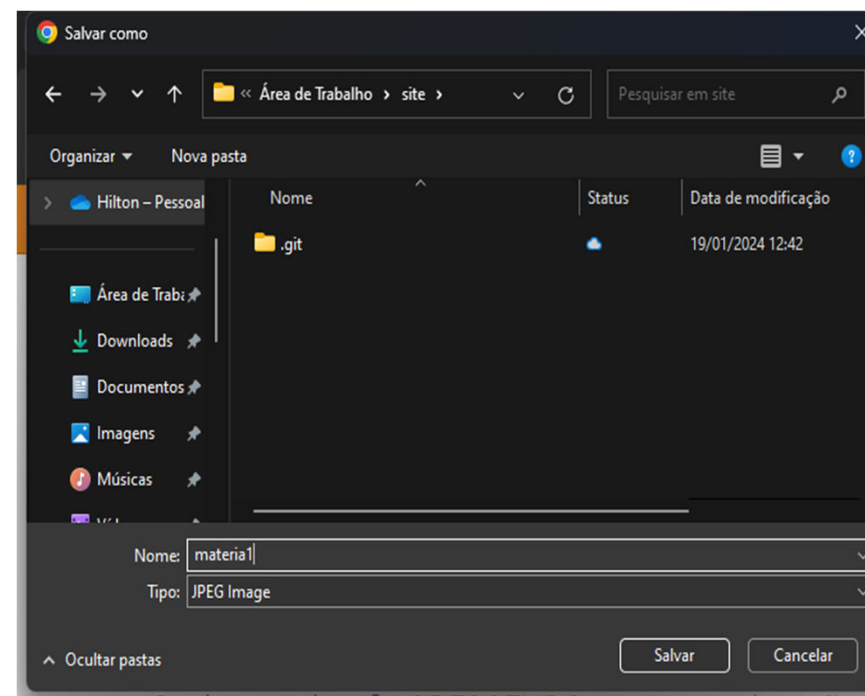
Vamos criar outra linha no código e criar uma matéria.

Vamos fazer uma série de matérias, cada uma, será um conjunto de vários elementos. Toda matéria terá uma imagem, um título e um link para essa matéria.

Vamos começar com imagem, iremos buscar uma no site do SENAI como referência, escolha uma matéria, clique com o botão direito em cima dessa imagem e salvar imagem como, verifique se você está dentro da pasta site, troque o nome dessa imagem colocando em minúsculas e sem caracteres especiais, como materia1.

Outra coisa importante é lembrar qual é o formato dessa imagem, podemos usar quatro tipos de imagem (JPG, PNG, GIF e SVG). Se o formato que estiver aparecendo para você for algo diferente disso, escolha outra imagem. Mais na frente eu vou explicar a diferença entre elas.

Agora na pasta do nosso site temos três arquivos, o arquivo index.html, matéria1.jpg e o README.md.



Antes de colocar a imagem, vamos adicionar o título da matéria. Utilizando o elemento <h1>. Iremos ao site de referencia e copiar o texto do título da matéria e colar ele aqui dentro do <h1>.

```
<> index.html M ●
<> index.html > html > body > h1
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Meu Primeiro Site SENAI</title>
7  </head>
8  <body>
9      <h1>Primeiro dia de Codificação</h1>
10     <p>Notícias da Indústria</p>
11     <h1>Vagas de verão SENAI: cursos com desconto para concluir ainda nas férias</h1>
12 </body>
13 </html>
```

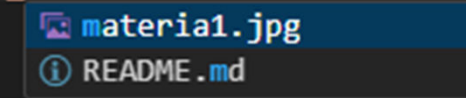
Embaixo do `<h1>` iremos adicionar um elemento novo que se chama ``. E perceba que quando você colocar o sinal de maior, ele não vai fechar assim como o elemento `<meta>`.

O elemento `` é um desses elementos especiais que não tem a tag de fechamento.

Dentro do elemento `` é preciso mostrar qual é a imagem que irá aparecer aqui. Eu posso ter quantas imagens eu quiser, só tenho que apontar qual imagem eu quero.

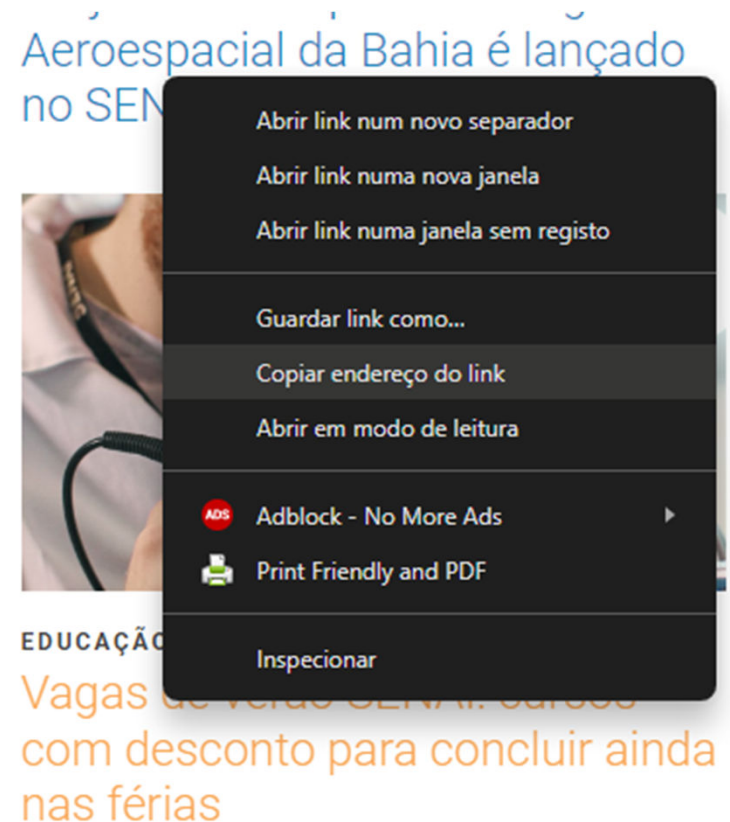
Dentro da tag, vou criar um atributo chamado `src`, assim der um Enter, ele se autocompleta, já vai preencher a formatação básica do que é o atributo e vai procurar dentro da nossa pasta os arquivos possíveis para colocar aqui dentro.

```
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Meu Primeiro Site SENAI</title>
7  </head>
8  <body>
9      <h1>Primeiro dia de Codificação</h1>
10     <p>Notícias da Indústria</p>
11     <h1>Vagas de verão SENAI: cursos com
12     
13
14 </body>
15 </html>
```



Agora iremos inserir o link para essa matéria, que nada mais é do que um botão que onde clica e leva para algum lugar, neste caso, essa matéria no site possui um link.

Se vocês passarem o mouse em cima das coisas, vocês vão ver que ela fica às vezes uma mãozinha, isso indica que tem um link ali dentro. Se eu clicar com o direito, eu consigo copiar o endereço do link.



Antes de colar o link, preciso escrever inserir um novo elemento <a>, que significa âncora e dentro dele colocar o texto que vai ser clicável, um texto chamado Saiba Mais.

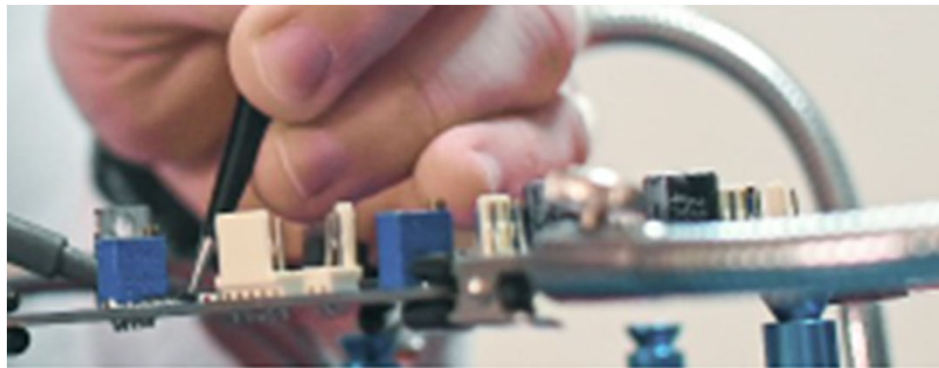
```
8 <body>
9   <h1>Primeiro dia de Codificação</h1>
10  <p>Notícias da Indústria</p>
11  <h1>Vagas de verão SENAI: cursos com desconto
12  
13  <a>Saiba mais!</a>
14 </body>
15 </html>
```

Agora eu preciso apontar pra onde esse link deve me levar. Dentro da abertura do elemento <a>, de um espaço, e iremos acrescentar um aqui atributo chamado href. Nesse atributo é onde devemos colar o link que foi copiado do site de referencia. Percebam que está dentro de aspas aqui em laranja.

Assim todo aquele link enorme apareceu aqui dentro.

```
8 <body>
9   <h1>Primeiro dia de Codificação</h1>
10  <p>Notícias da Indústria</p>
11  <h1>Vagas de verão SENAI: cursos com desconto para concluir ainda nas férias</h1>
12  
13  <a href="https://noticias.portaldaindustria.com.br/noticias/educacao/vagas-de-verao
14 </body>
15 </html>
```

Iremos salvar o arquivo, voltar ao nosso site, atualizar a página e verificar que o link apareceu.



Saiba mais!

O mais importante é verificar se esse texto está com uma cor azul ou roxa, pois assim saberemos que o link foi criado corretamente. Se passarmos o mouse irão ver que vira uma mãozinha e se clicar, ele vai nos levar para o link que eu apontei.

A partir desse ponto iremos criar sessões de conteúdo para organizar a nossa página. Por enquanto não temos muita coisa aqui, mas logo essa página vai ficar bem longa. Por isso iremos subdividir nossa página em partes.

A primeira coisa a fazer é acrescentar comentários, que nada mais são do que textos que acrescentamos em qualquer lugar no meio do código que ficaram invisíveis aos usuários do site. Este elemento `<!--` abre o comentário, e este `-->` fecha. Todo que é escrito aqui não aparece, é como se não existisse, não aparece nada, mas o texto aparece aqui nós. É uma observação em que posso colocar no meio do código que não aparece no meio do site.

```
<!--Matéria 1-->  
<h1>Vagas de verão SENAI: cursos com desconto para concluir a  
  
<a href="https://noticias.portaldaindustria.com.br/noticias/edu
```

Existe outra coisa que podemos fazer pra dividir o nosso conteúdo. Vamos aprender a lidar com sessões de conteúdo.

Vejam essa matéria, percebem que ela é um grupo de elementos <html>. Ela é composta de um título, uma imagem e um link. Só que está solta, esse comentário é só para identificar, mas ainda não há uma separação.

```
<!--Matéria 1-->  
<h1>Vagas de verão SENAI: cursos com desconto para concluir ainda nas férias</h1>  
  
<a href="https://noticias.portaldaindustria.com.br/noticias/educacao/vagas-de-verao
```

Imagina que mais para frente, nós teremos menu de navegação, rodapé, formulário, as coisas não podem ficar soltas dessa forma, senão o código fica bagunçado e sem sentido.

Para isso, existem as sessões e nós vamos aprender a primeira e uma das mais importantes sessões do HTML5, o <article>. Vamos abrir o elemento <article> antes do comentario, percebam que ele já colocou a tag de fechamento do <article>, mas eu vou recortar aqui esse fechamento e vou envolver o conteúdo da minha matéria.

```
<!--Matéria 1-->
<article></article>|
<h1>Vagas de verão SENAI: cursos com desconto para concluir ainda

<a href="https://noticias.portaldaindustria.com.br/noticias/educac
```

Então depois do título, imagem e o link, é onde a minha matéria acaba. Aqui eu vou colar o fechamento do <article>.

```
<!--Matéria 1-->
<article>
<h1>Vagas de verão SENAI: cursos com desconto para concluir ainda nas fés

<a href="https://noticias.portaldaindustria.com.br/noticias/educacao/vaga
|</article>
```

Vou selecionar esse conteúdo (título, imagem e link) e apertar a tecla Tab, assim fica muito mais fácil de entender onde o artigo abre e fecha, e também muito mais fácil de entender o que está dentro e o que está fora dele.

Por isso é muito importante o Tab. Parece besteira, mas vocês conseguem identificar esse grupo muito mais fácil. É como se criasse um mini site aqui dentro. Tudo que essa matéria precisa está dentro desse artigo.

```
<!--Matéria 1-->
<article>
  <h1>Vagas de verão SENAI: cursos com desconto para concluir ainda nas férias</h1>
  
  <a href="https://noticias.portaldaindustria.com.br/noticias/educacao/vagas-de-verao-senai-
</article>
```

E agora vem a parte legal, se quiser criar uma outra matéria que nada mais é do que uma cópia dessa primeira, basta eu selecionar este elemento <article> e todo o seu conteúdo dentro, até onde ela fecha e colar em baixo.

Agora temos dois <articles>, mas vamos trocar as informações dessa segunda matéria. Coloque o comentário igual o da matéria 1, logo em cima do novo <article>.

Vou entrar no site de referencia novamente e procurar uma nova matéria. Vou escolher uma matéria e copiar o título.

Vou selecionar o texto dentro do <h1> da matéria 2, vou deletar e colar. Pronto, agora tem o título da nova matéria.

Agora é trocar imagem, vou lá no site de referencia, vou clicar com direito na imagem, salvar a imagem como, vou me certificar que estou dentro da pasta correta do site e, super importante, trocar o nome da imagem. Sempre troque o nome da imagem por um nome curto, sem espaços e sem acentos. Vou chamar de matéria2.

Agora só falta colocar o link da matéria. Volte no site de referencia, procure onde tem link, clicar com direito, copiar endereço do link. Vá dentro das aspas, tomar cuidado para não apagar todas as aspas, colar o link. Pronto, agora temos as duas matérias.

```
<!--Matéria 2-->
<article>
  <h1>Conheça o desafio CRESCENDO da temporada 2023-2024 da FRC</h1>
  
  <a href="https://noticias.portaldaindustria.com.br/noticias/infogra
</article>
```

Agora vamos fazer uma terceira e quarta matéria. Basta repetir os passos utilizados para fazer essa segunda matéria.

Agora que já temos as quatro matérias, vejam só como o nosso código já é bem longo, apesar de todas essas matérias serem iguais.

Então, sempre vejam se as matérias estão dentro do seu próprio <article>. Cada matéria, cada conjunto de elementos de título, imagem e link. Tem o comentário em cima, que é opcional.

Agora vamos conhecer mais alguns elementos para criar mais estrutura em nosso site. A maioria dos sites que se vê tem algumas divisões bem claras.

A parte de cima do site se chama cabeçalho, ela normalmente é a introdução para todo o site. Normalmente ela contextualiza você, em que site você está, e te apresenta também outro item importante, que é o menu de navegação. Alguns possuem também um sistema de busca interno, outros links e tudo mais.

Outra parte importante é a área de conteúdo, que é a área que já começamos a trabalhar, colocando as matérias.

E bem no final da página, temos o rodapé, que normalmente tem o mapa do site com os links para todas as áreas, um disclaimer (isenção de responsabilidade) de direitos autorais, alguns links importantes, às vezes links para redes sociais e tudo mais, e isso encerra a nossa página. A maior parte dos sites possuem essa estrutura, e o nosso site não vai ser diferente.

No início do código, na tag <body>, vou criar um espaço e vou criar um elemento chamado <header>, esse elemento é o cabeçalho.

```
<body>  
    
  <header></header>  
    
  <h1>Primeiro dia de Codificação</h1>  
  <p>Notícias da Indústria</p>
```

Vamos colocar o conteúdo do cabeçalho, a princípio teremos duas coisas apenas por enquanto. O nome do nosso site, normalmente é o logo, que no caso vai ser só um texto mesmo, e o menu de navegação.

Para o nome do site, eu vou usar um link, quando a gente passa o mouse em cima, ele é um link, e ele sempre leva para a página inicial.

```
<body>  
    
  <header>  
    <a>Alunos do SENAI</a>  
  </header>
```

Agora precisamos colocar o href, que é o endereço para onde esse link vai me levar, que é para a única página que temos na verdade, que é a página inicial, que é index.html.

```
<header>  
  <a href="index.html">Alunos do SENAI</a>  
</header>
```

Agora vamos ver como é que faz um menu de navegação. Para isso a gente precisa primeiro aprender o que é um menu de verdade. Ele nada mais é do que uma lista de links. Para fazer isso no HTML existem elementos específicos para isso. Ainda dentro do <header> vou dar um espaço depois do link. O elemento responsável pela lista no HTML é o elemento . Abre e fecha como todos que já vimos.

Agora eu preciso colocar os itens da lista. Os itens da lista devem entrar dentro de um outro elemento chamado , isso para cada elemento dessa lista. Esse nosso site terá duas páginas no menu apenas, por enquanto.

```
<ul>  
  <li></li>  
</ul>
```

Então se essa lista tiver dois itens, ela vai ter duas , cada será responsável por um item. No primeiro item vou escrever Home, que é a nossa página inicial, e no segundo item vou escrever contato, que vai ser uma página que faremos mais a frente.

```
<ul>
  <li>Home</li>
  <li>Contato</li>
</ul>
```

Agora, nós temos a lista, os itens, mas precisamos acrescentar os links, eles vão dentro das , em volta do texto em que se quer clicar, como quero que o Home seja clicável, vou colocar o link em volta dele, dentro da , vou colocar o endereço do link, que é o atributo href dentro do <a>, e o endereço para home, index.html.

```
<a href="index.html">Alunos do SENAI</a>

<ul>
  <li>
    <a href="index.html">Home</a>
  </li>
  <li>
    <a href="contato.html">Contato</a>
  </li>
</ul>
```

Faremos a mesma coisa com o contato, mesmo que não tenha essa página a contato, mas tenho certeza que ela vai se chamar contato.html. Tudo isso é uma lista de links que normalmente associa-se como um menu de navegação, pra navegar entre as nossas páginas do site.

Agora temos que lembrar que o nosso logo, o nome que escrevi, também deve fazer parte da navegação. Então pra determinar que tudo isso aqui é navegação, vamos criar um outro elemento em volta deles.

Vamos aprender o elemento `<nav>`, vou retirar a tag de fechamento e colocar depois da ``. Selecione o que está dentro e do `<nav>` e dar um Tab para ficar alinhado, assim fica mais fácil de enxergar o que está dentro do elemento `<nav>`, que estão dentro do elemento `<header>`.

```
<header>

  <nav>
    <a href="index.html">Alunos do SENAI</a>

    <ul>
      <li>
        <a href="index.html">Home</a>
      </li>
      <li>
        <a href="contato.html">Contato</a>
      </li>
    </ul>
  </nav>
</header>
```

Vamos revisar, `<header>` significa cabeçalho é separar algo que é uma introdução para o que vem embaixo. `<nav>` é a navegação e serve para dizer que o que está dentro dele é para navegar dentro do site, ou seja, vai ter um monte de links para páginas internas do nosso projeto. O `` significa lista, e ``, item de lista.

Importante perceber que várias coisas que marcamos no HTML não ficam visíveis para o usuário. Por exemplo, ele consegue ver o nosso Logo e a nossa lista de links, mas não consegue ver o que é o `<header>`, ele não consegue ver o que é o `<nav>`, nem o `<article>`. Para ele, essas coisas só ficarão aparentes quando a gente acrescentar algum estilo visual.

Então, para que escrever tudo isso se o usuário nem enxerga isso? Primeiro, que fica muito mais fácil para nós, desenvolvedores, enxergarmos o nosso código e diferenciar o que é cada parte.

Vejam como fica mais fácil quando se subdivide o conteúdo. O `<header>` e o `<nav>` marcam todo o nosso cabeçalho, as nossas matérias dentro de cada elemento `<article>`. Dessa forma vamos organizando muito melhor.

Agora vamos acrescentar no final desse site, antes de fechar a tag `<body>` lá embaixo, um outro elemento que normalmente encerra o conteúdo, o famoso rodapé, chamado `<footer>` no HTML.

Vamos abrir <footer>, vamos deixar bem simples a princípio, vamos apenas criar um parágrafo e escrever “Você chegou ao fim do nosso conteúdo!”.

Agora, nossa primeira página já possui o básico de uma página HTML, possui um cabeçalho, um menu de navegação, um conteúdo e um rodapé.

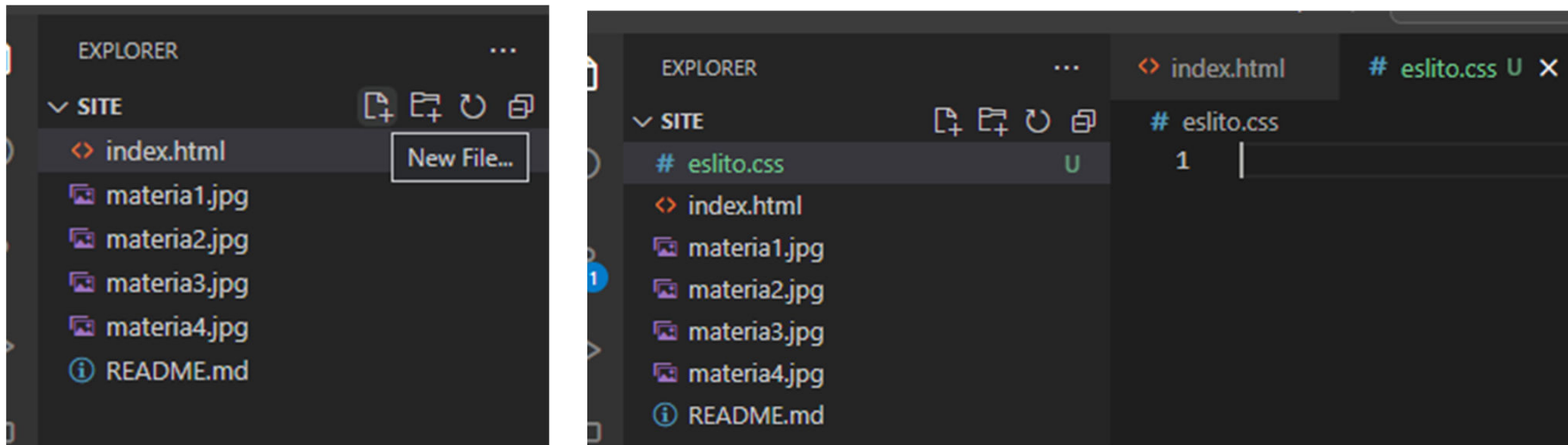
```
<footer>
  <p>Você chegou ao fim do nosso conteúdo!</p>
</footer>

</body>
</html>
```

Agora vamos tratar um pouco da parte visual de nosso site, aprender como modificar a fonte, a cor de fundo, a cor do link, como fazer o menu ficar de lado a lado, etc.

O CSS é quem irá cuidar de tudo isso. A primeira coisa que temos entender é que o CSS é uma linguagem bem diferente do HTML, ele serve para dar uma aparência para modificar todas essas coisas que colocamos na tela com o HTML. Mas sem o HTML, o CSS não tem como agir.

Agora iremos implementar o CSS em nossa página, clicando em New File(arquivo novo) e criar o estilo.css.



Percebam que a extensão agora é .css, e isso vai diferenciar dos arquivos html. Temos que entender que por mais que eu criei esse arquivo, e porque está na mesma pasta que o nosso html, não quer dizer que vai funcionar. Precisamos vincular os dois arquivos um ao outro, isso é feito no html.

Agora estamos com dois arquivos abertos, o estilo.css e o index.html.

Retornaremos ao index.html, para vincular os dois arquivos, precisamos modificar o <head> do nosso site, criaremos uma linha nova aqui e vamos criar um elemento chamado <link>. Não confunda o elemento <link> com os elementos <a> de âncora.

```
<head>
  <title>Meu Primeiro Site SENAI</title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link>
</head>
```

Esses links <a> que estão dentro do body servem para criar botões de navegação. Já o elemento <link> dentro do <head> significa vínculo em inglês. E é pra isso que ele serve, pra trazer outros arquivos aqui pra dentro, não pra clicar e levar pra algum lugar. Bem diferente, o elemento <link>, ele não fecha que nem os outros elementos, como por exemplo o <title> ou o próprio <head>.

Ele funciona que nem o <meta>, ou seja, ele não precisa fechar. Portanto já dá pra imaginar que precisamos colocar atributos dentro, vou dar um espaço antes do sinal de maior, e criar um atributo que a gente até já conhece, são usados nos próprios links do <body>, href.

Isso vai trazer o vínculo com o arquivo de dentro do nosso site.

O arquivo que quero trazer é o estilo.css, apos as aspas, vou dar mais um espaço e vou criar um outro atributo chamado rel (tipo de relação) e dentro das aspas, vou escrever stylesheet (folha de estilos). Esse atributo serve para trazer varios outros tipos de arquivo, não apenas css.

```
<head>
  <title>Meu Primeiro Site SENAI</title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="estilo.css" rel="stylesheet">
</head>
```

Agora temos nosso arquivo HTML devidamente linkado com o arquivo CSS.

Vamos salvar o index.html e voltar para o arquivo estilo CSS, que está vazio, e vamos começar a escrever para saber se deu certo.

Para testar começaremos a primeira modificação em nosso site, escrevendo o seguinte, body, abre chaves, background-color, dois pontos, red, ponto e vírgula. Aqui estou chamando o elemento <body> do nosso HTML, e dentro das chaves vou colocar propriedades CSS, que são características e a primeira característica que vou dar se chama background color. O dois pontos é para falar qual cor que vai ser e essa é a cor red (vermelho), ponto e vírgula para falar que essa linha acabou e fecha as chaves.

Podemos ver que o CSS é bem diferente do HTML.

```
<> index.html M    # eslito.css U ●
# eslito.css > body
1  body {
2    background-color: red;
3  }
```

Resumindo, aqui apontamos qual elemento quero alterar alguma propriedade, estou falando para o <body> ficar com o fundo vermelho.



Agora podemos deixar nosso site muito mais interessante com o poder do CSS. Mais pra frente nós vamos aprender formas mais complexas de nomear cor.

Vamos aprender algumas propriedades básicas de CSS, para dar personalidade a nossa página. Vamos revisar o que temos no nosso HTML, porque para modificar no nosso CSS, temos que entender primeiro o que conseguimos modificar no HTML.

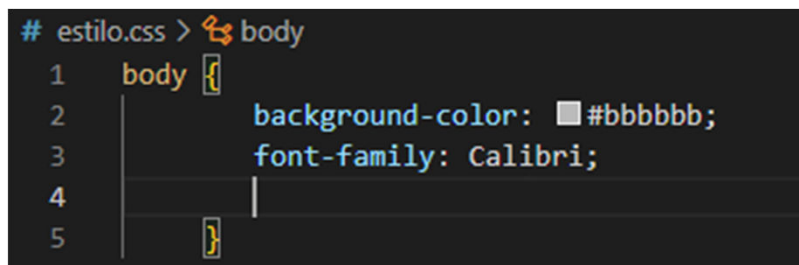
Em nossa index.html, temos o <body>, que podemos chamar pelo CSS, como nós já vimos antes, onde podemos modificar o fundo, por exemplo. Podemos chamar o elemento <header>, o elemento <nav>, <a>, , , <h1>, <p>, <article>, e o <footer>.

A segunda alteração que podemos fazer para melhorar nosso site, é o padrão de fontes, podemos mudar o tamanho dos textos, a família, a cor.

Na última alteração tínhamos colocado o fundo vermelho em nosso site usando a tag <body>, que envolve todo o conteúdo da página. Essa mesma chamada, <body>, também serve para nós alterarmos alguns padrões, por exemplo, como os padrões de fonte.

Vamos acrescentar uma linha aqui depois do ponto e vírgula onde termina a alteração do background (cor de fundo) e vou chamar uma propriedade que vai mudar a família de fontes.



Ela se chama font-family, por enquanto, usaremos apenas fontes instaladas em nosso computador, mas mais para frente vamos aprender a usar outras fontes, poderemos usar a fonte que quisermos. A fonte que vou usar é a Calibri. Basta escrever o nome da fonte.

A screenshot of a code editor with a dark background. The text shows a CSS file named 'estilo.css' with a selector 'body'. The code is as follows:



```
# estilo.css > body
1 body {
2     background-color: #bbbbbb;
3     font-family: Calibri;
4 }
5
```

Vamos aumentar a fonte, a propriedade se chama font-size, vamos colocar 18px (pixels). Alguns textos mudaram de tamanho, como os itens do cabeçalho, parágrafo e links. Mas perceberam que os títulos não mudaram de tamanho. Se colocarmos essas propriedades de fonte no <body>, a maioria dos itens irá absorver essas características ao mesmo tempo, mas outras não.

Isso porque os títulos já vêm com uma propriedade aplicada diretamente pelo navegador, por mais que não tenhamos colocado nada.

```
# estilo.css >  body
1  body {
2      background-color:  #bbbbbb;
3      font-family: Calibri;
4      font-size: 20px;
5  }
```

Agora vamos alterar a cor dos textos padrão. A propriedade que modifica a cor dos textos é a color, não existe text color. Em vez de ficar chamando pelo nome da cor, vocês vão aprender uma coisa chamada hexadecimal. Para isso vou precisar colocar o símbolo # (jogo da velha, ou que muita gente acaba chamando de hashtag). Depois desse símbolo, irão seis dígitos que representam as cores. Vamos apenas escrever 333333, vai nos dar um cinza bem escuro, quase preto.

```
# estilo.css > ...
1  body {
2      background-color:  #bbbbbb;
3      font-family: Calibri;
4      font-size: 20px;
5      color:  #333333;
6  }
```