

# Curso: Front-End do Bem



## AULA 16

- Como adicionar o JavaScript(JS) em uma Página Web
  - Exemplos
- O Console do Navegador
  - Exemplos
- Sugestões de Prática para essa aula
  - O que vamos aprender na aula 17



# Como adicionar o JavaScript em uma Página Web

Na aula 15 vimos como configurar o nosso ambiente com a Instalação do VSCode e do Live Server, além de outro conceito importantíssimo sobre o que é um Servidor.

Dentro do VSCode, vamos usar nesse início o mesmo código HTML o qual digitamos na aula 15, ah e não podemos esquecer de inicializar o Live Server.

Agora que o nosso ambiente está preparado, chegou o momento de vermos como adicionamos o JavaScript em uma Página Web.

A primeira coisa que temos que saber é em qual lugar podemos adicionar o JavaScript dentro de uma página HTML.

Podemos utilizar o JavaScript em qualquer local dentro das tags:

1 - `<head>Fica como Exercício para Pesquisar</head>`

2 - `<body></body>`

Nesse primeiro momento não vamos nos preocupar com mais nada, tudo o que vamos fazer é especificar uma tag `<script>` que contem abertura e fechamento para que possamos escrever o tal do conteúdo JavaScript dentro dessa tag.

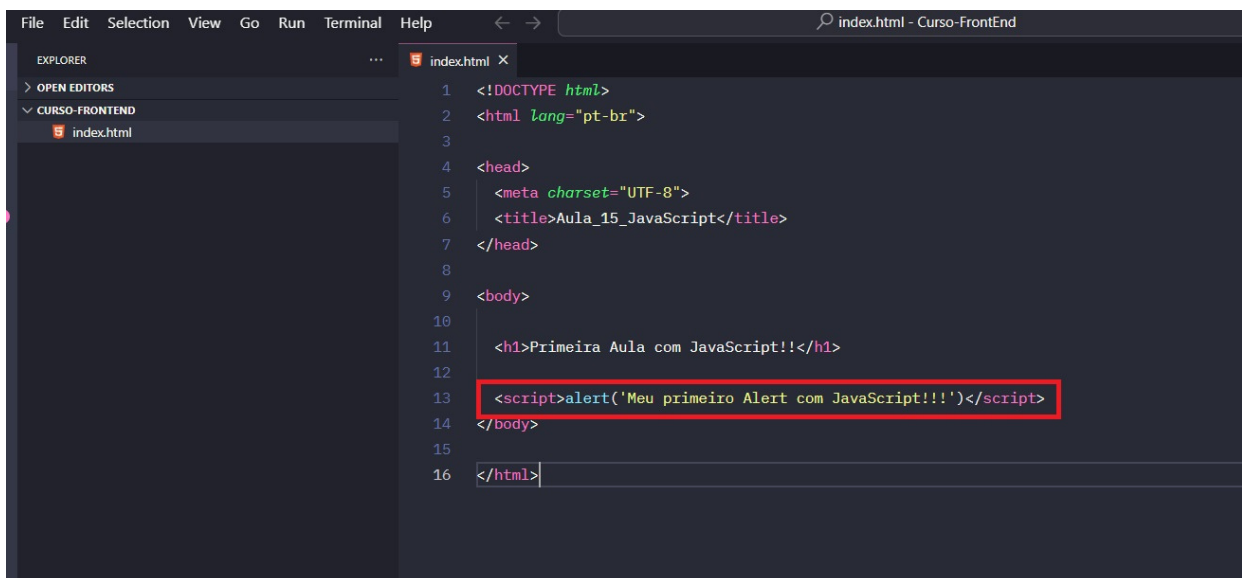


# Como adicionar o JavaScript em uma Página Web

Vamos colocar a tag `<script></script>` como a ultima tag dentro da tag `<body></body>`

Então assim podemos, declarar um alert que vai exibir a mensagem ('Meu primeiro Alert com JavaScript!!!').

O código vai ficar desse jeito, conforme imagem abaixo:



```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3
4 <head>
5   <meta charset="UTF-8">
6   <title>Aula_15_JavaScript</title>
7 </head>
8
9 <body>
10
11   <h1>Primeira Aula com JavaScript!!</h1>
12
13   <script>alert('Meu primeiro Alert com JavaScript!!!')</script>
14 </body>
15
16 </html>
```

# Como adicionar o JavaScript em uma Página Web

Igor, esse tal de alert com a mensagem dentro dos parênteses, está meio confuso o que seria isso?

Mais acima, eu comentei que não era para nos preocuparmos com mais nada que não fosse entender o local aonde iríamos colocar o JavaScript, pois bem, isso vale aqui também, não se preocupe agora em entender o que esse código significa, mas vou deixar uma breve explicação aqui: ele é apenas um Método que vai gerar um pop-up (uma caixa de diálogo) na tela com uma mensagem e um botão onde podemos clicar em OK (explicaremos isso com mais detalhes nas próximas aulas).

Se salvarmos esse documento que acabamos de alterar com o bom e velho CTRL + S maroto, o código dessa tag JavaScript será executado e vai aparecer uma mensagem com um botão para clicarmos em OK e o restante da página atrás desse pop-up também será carregada.



# Como adicionar o JavaScript em uma Página Web

O Exemplo visto na página anterior o qual colocamos a tag `<script>` no `<body>` é conhecido como código JavaScript inline. Esse meio é válido quando temos pouquíssimo código, já se tivermos uma quantidade maior de código JavaScript o ideal é que seja feito por um outro meio.

Igor, estava pensando aqui, se esse não é o meio recomendado quando temos uma quantidade maior de código JavaScript dentro da tag `<body>` então podemos criar um arquivo separado em JavaScript assim como fizemos com o CSS quando estudamos sobre CSS?

Beleza, tá legal, o Igor vai embora, vocês não precisam mais desse professor aqui.. vou vender minha arte na praia.

Ai que orgulho!!!!!!!!!!!!!!!!!!!!!!... é isso mesmo pessoal, vocês estão certíssimos... Podemos isolar esse código em um arquivo separado e só fazemos uma referência desse arquivo JavaScript o qual fora criado separado lá dentro do arquivo `index.html`, chamamos isso de (Linkar).

Já falei que eu estou orgulhoso, né? Fica mais uma vez o registro!!!

Então bora criar esse arquivo JavaScript separado...

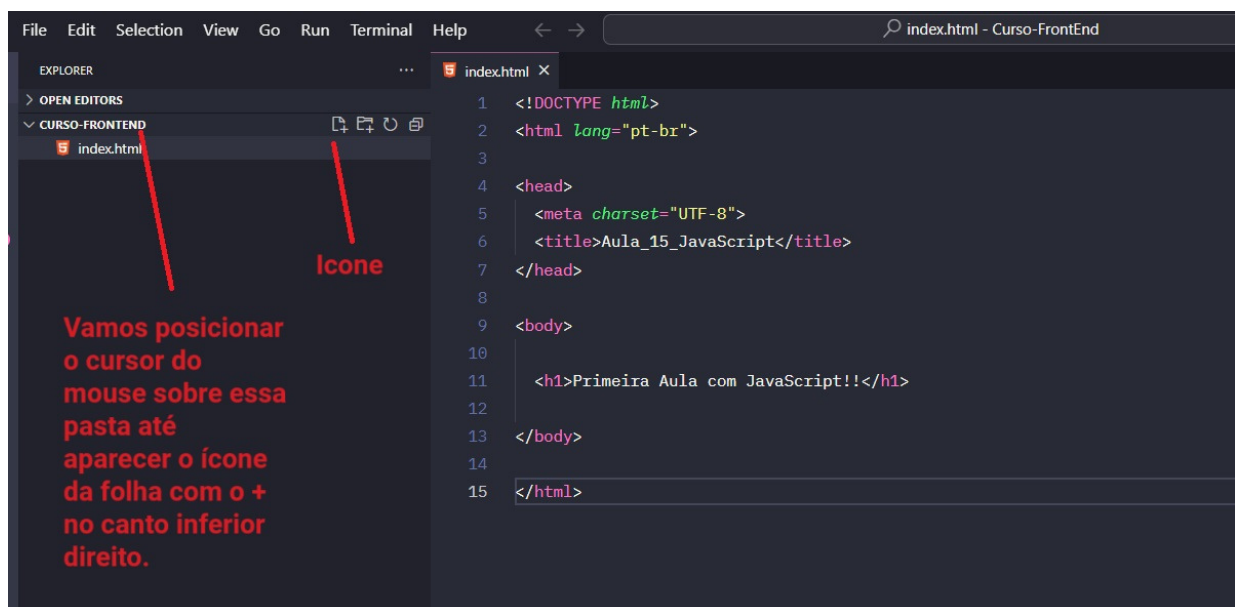


# Como adicionar o JavaScript em uma Página Web

A nomenclatura do arquivo é composta por: nome.extensão onde nome(vamos chama-lo de: app) e a extensão depois do . temos que usar a extensão javascript(representada por: js). Então com base nessa definição o nome do nosso arquivo javascript vai ficar assim:

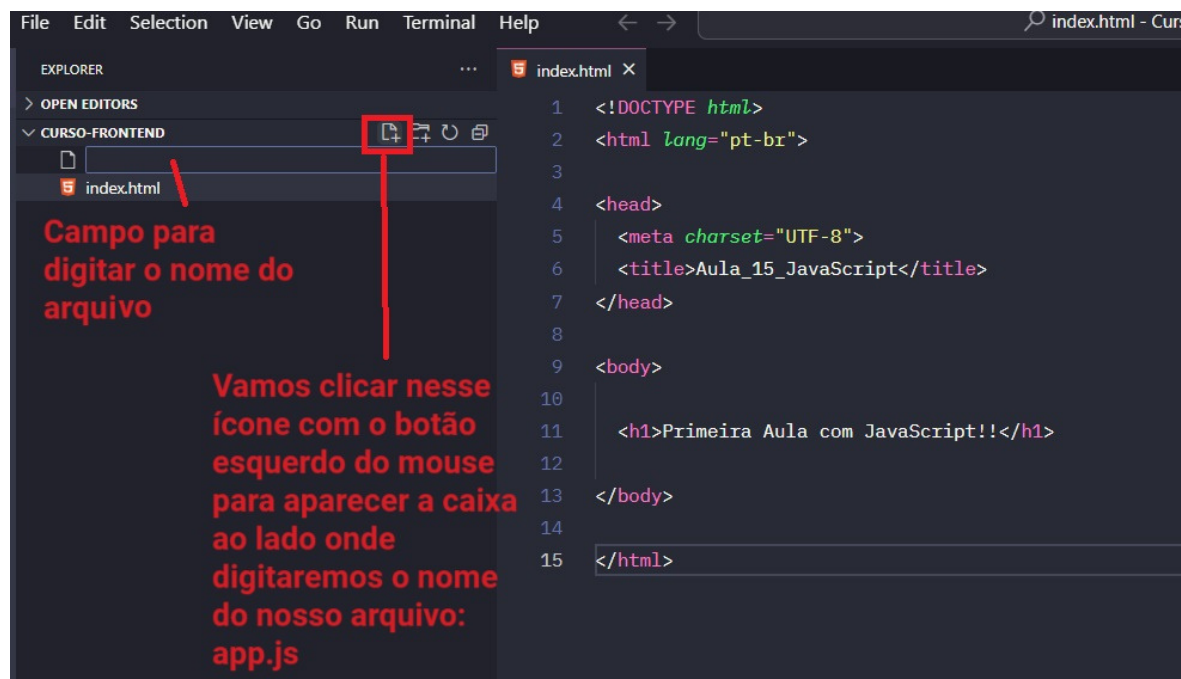
app.js

Agora que definimos o nome do nosso arquivo JavaScript, vamos na lateral esquerda dentro do VSCode, posicionaremos o cursor do mouse sobre a pasta: CURSO-FRONTEND até aparecer alguns ícones a direita.



# Como adicionar o JavaScript em uma Página Web

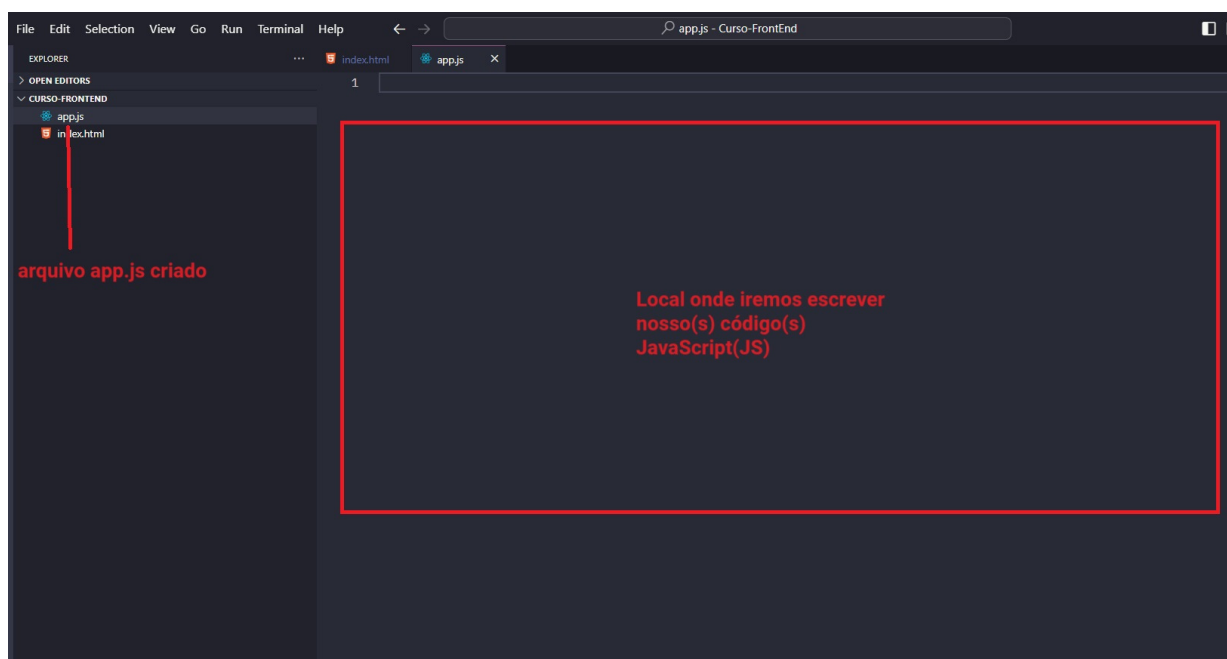
E clicaremos no primeiro item (o desenho de uma folha com um sinal de + no canto inferior direito com o botão direito sobre ela e selecionaremos criar novo arquivo



Digitaremos app.js e pressionaremos enter.

# Como adicionar o JavaScript em uma Página Web

Notem que assim como criamos o arquivo index.html, o arquivo app.js consta abaixo da nossa pasta Curso-FrontEnd e do lado direito abriu um local sem nada escrito, onde vamos escrever todo o nosso código JavaScript.



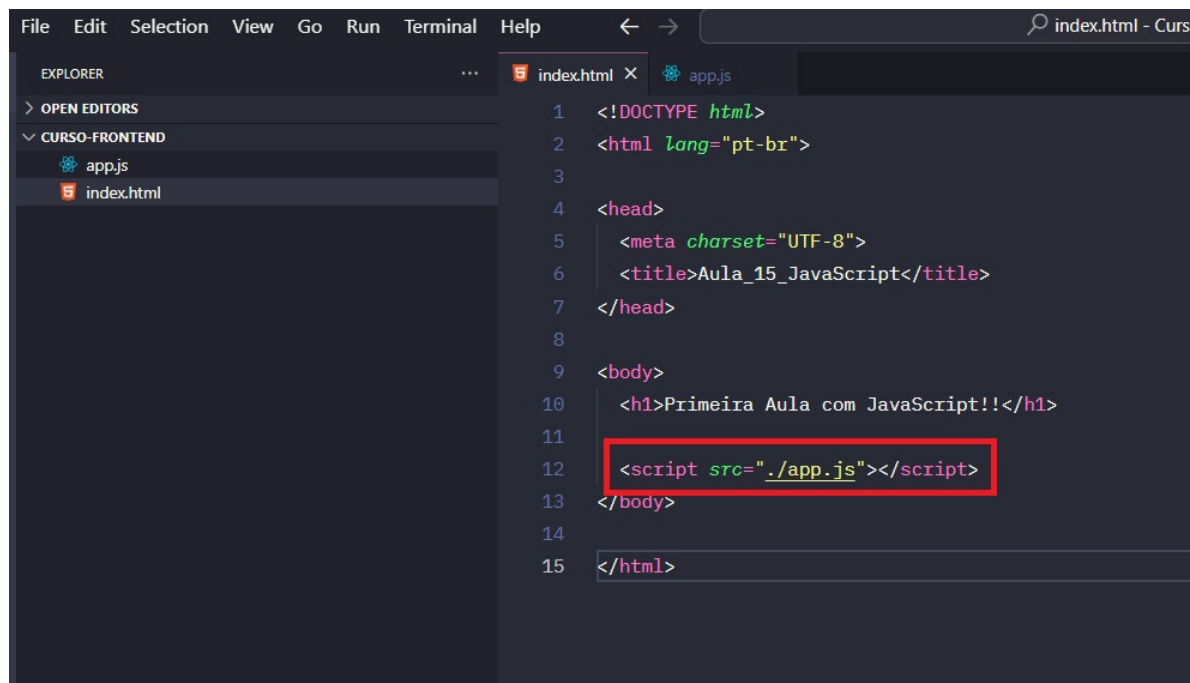
Lembra que para que o arquivo app.js(Agora temos o nome) funcione dentro do index.html temos que fazer um link(Linkar), pois bem vamos fazer isso agora

Abriremos o arquivo index.html, e antes do fechamento da tag `</body>`, nós vamos declarar uma nova Tag `<script></script>`, só que aqui ela não vai ter conteúdo igual quando definimos o alert com a mensagem no exemplo mais acima, e sim declararemos um atributo src que vai receber o caminho do arquivo o qual queremos fazer referência(linkar) que é o app.js. Ficando conforme imagem da próxima página.





# Como adicionar o JavaScript em uma Página Web



```
File Edit Selection View Go Run Terminal Help
index.html - Cursor

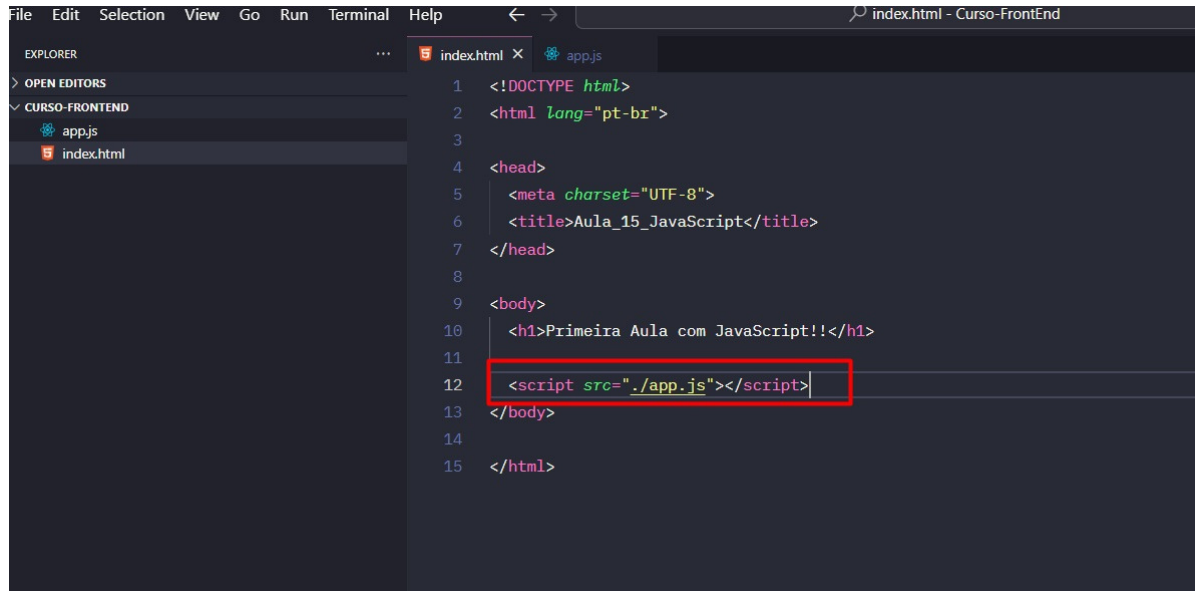
EXPLORER
OPEN EDITORS
CURSO-FRONTEND
  app.js
  index.html

index.html
1 <!DOCTYPE html>
2 <html lang="pt-br">
3
4 <head>
5   <meta charset="UTF-8">
6   <title>Aula_15_JavaScript</title>
7 </head>
8
9 <body>
10   <h1>Primeira Aula com JavaScript!!</h1>
11   <script src="./app.js"></script>
12 </body>
13
14
15 </html>
```

Com o app.js devidamente referenciado(Linkado) dentro do arquivo index.html, se agora formos escrever código javascript dentro do arquivo app.js, esse código será executado quando rodarmos o arquivo index.html no navegador(Chrome).

Digitaremos: `alert('Meu primeiro Alert com JavaScript!!!')` e salvaremos esse arquivo mais uma vez com o CTRL + S maroto.

# Como adicionar o JavaScript em uma Página Web



```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3
4 <head>
5   <meta charset="UTF-8">
6   <title>Aula_15_JavaScript</title>
7 </head>
8
9 <body>
10  <h1>Primeira Aula com JavaScript!!</h1>
11
12  <script src="./app.js"></script>
13 </body>
14
15 </html>
```



É desse jeito que adicionamos o JavaScript(JS) dentro de uma Página WEB.

# O Console do Navegador

O Console é como uma linha de comando inteligente e rica dentro do DevTools e é uma ótima ferramenta complementar para usar com outras ferramentas.

O Console fornece uma maneira poderosa de criar scripts, inspecionar a página da Web atual e manipular a página da Web atual usando JavaScript.

Fonte: <https://learn.microsoft.com/pt-br/microsoft-edge/devtools-guide-chromium/console/>

Nós estamos usando o Google Chrome como navegador padrão aqui no curso, a definição acima foi retirada da página do Microsoft Edge, e isso foi proposital, para vocês verem que essa definição vale tanto para o: Chrome, Edge e FireFox.

Para abrirmos essa ferramenta temos 2 jeitos para fazermos:

- 1 - Pressionar F12 no teclado com o Navegador(Chrome) aberto
- 2 - Clicar com o botão direito em qualquer lugar que não tenha nada escrito dentro do Navegador(Chrome) e clicar com o botão esquerdo do mouse em Inspecionar e selecionaremos a aba Console

Mostraremos a segunda forma para abrirmos a Ferramenta



# O Console do Navegador

Igor, não entendi nada do conceito técnico sobre o Console do Navegador (DevTools, Inspeccionar, Manipular e etc...) poderia explicar de uma outra forma?

É pra já e é ai que vem a parte Lúdica da coisa:

Podemos considerar a aba Console dentro do DevTools(Ferramentas do Desenvolvedor) como um parque de diversões do Desenvolvedor o qual temos disponível no Navegador(Chrome) para testar, inspecionar e manipular o código JavaScript.

O DevTools nós usaremos tanto mais tanto e mais tanto(eu já disse mais tanto né) que ele vai ser quase como um membro da família, por um pequeno detalhe que nós não poderemos declara-lo no imposto de renda como dependente.

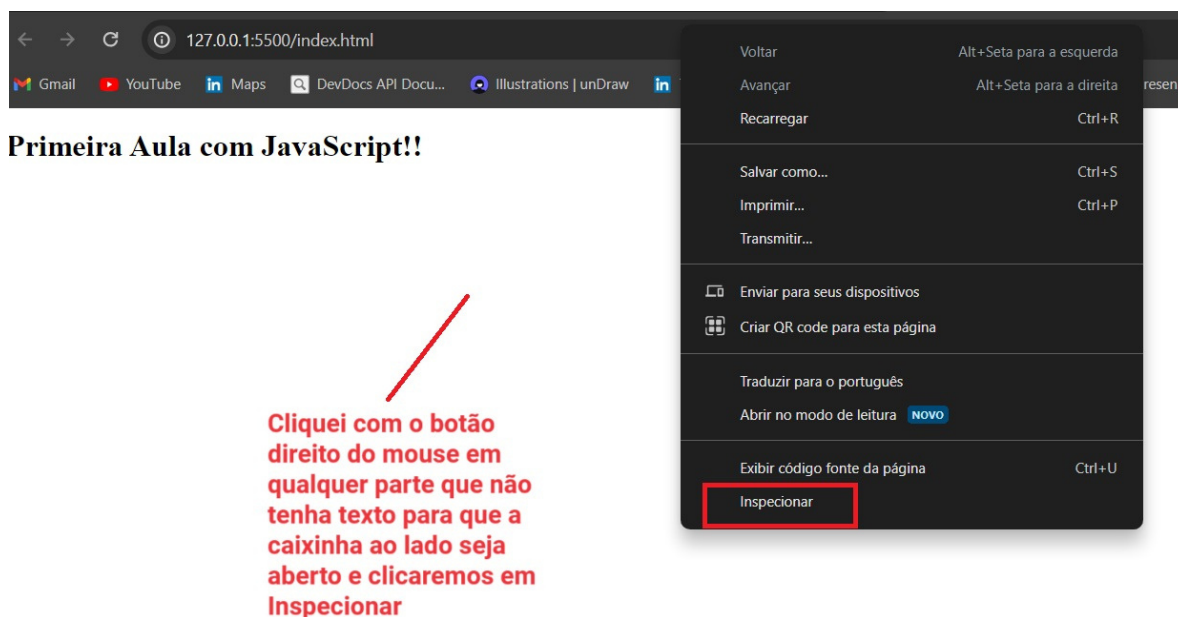
DevTools esse é o Pessoal, Pessoal esse é o DevTools... Agora que vocês já foram devidamente apresentados, vamos ver como esse carinho funciona na prática.

Ah, ops... já ia me esquecendo... antes precisamos abrir esse carinho....

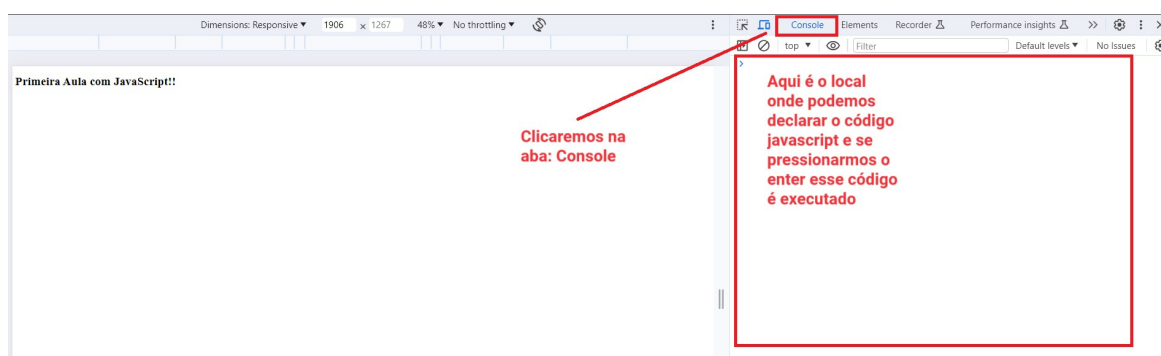
Clicar com o botão direito em qualquer lugar que não tenha nada escrito dentro do Navegador(Chrome) e clicar com o botão esquerdo do mouse em Inspeccionar e selecionaremos a aba Console



# O Console do Navegador



Dentro da Aba Console, podemos declarar código JavaScript e ao pressionarmos o Enter esse código JavaScript será executado.



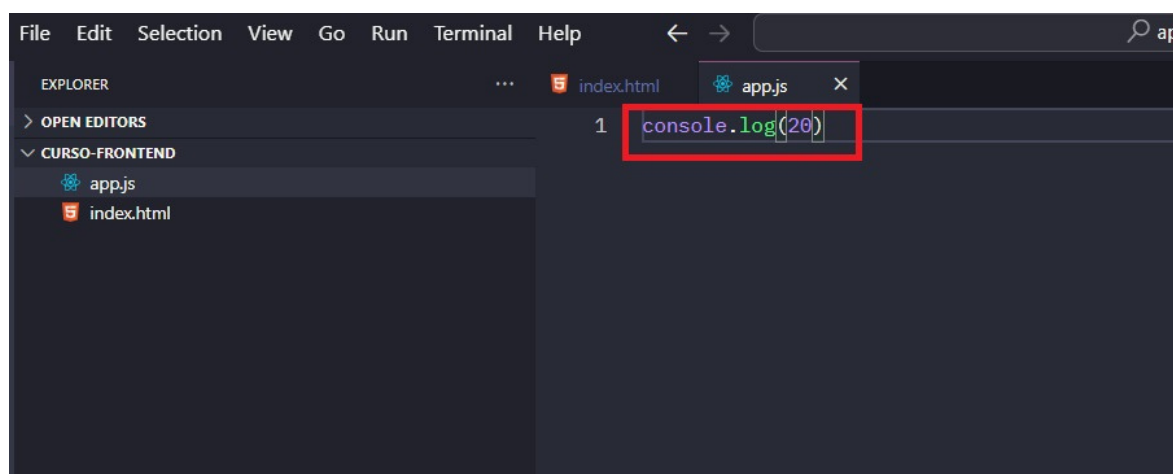
Ao declararmos o mesmo comando o qual usamos no exemplo um pouquinho mais acima: `alert('Meu primeiro Alert com JavaScript!!!')` e pressionarmos enter a mensagem com o botão OK, será mostrada no Navegador(Chrome).

# O Console do Navegador



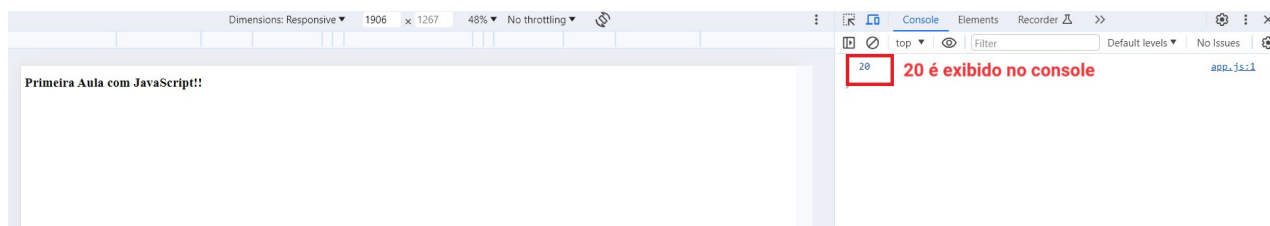
Além de fazermos isso, também é possível que seja mostrado algo desse Console que vem laaaaaaa do nosso arquivo `app.js`

Removeremos o `alert` que inserimos no exemplo anterior, vamos digitar `console.log()` e aqui dentro dos parênteses colocaremos o número 20.

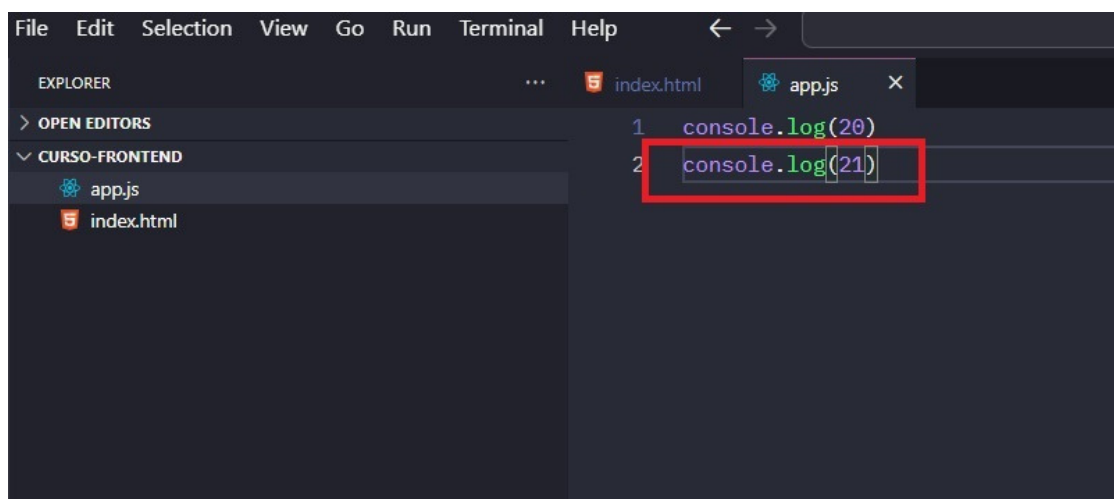


E ao salvarmos com aquele CTRL + S maroto, o número 20 é exibido no console.

# O Console do Navegador

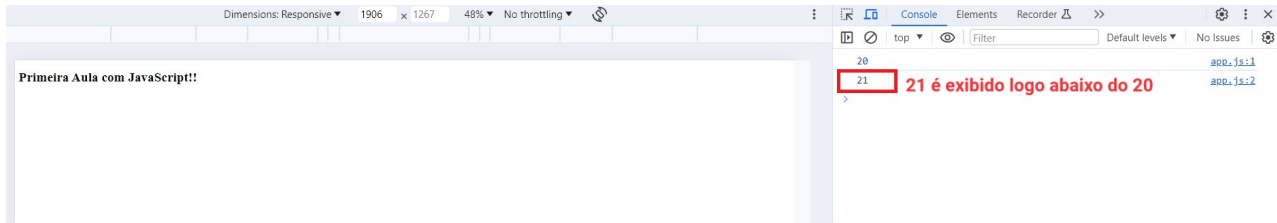


Se logo abaixo do `console.log(20)` digitarmos `console.log(21)`, esse código será executado de modo sequencial, pois o JavaScript é executado (Top-Down) ou seja de cima para baixo.



Então o que acontecerá aqui quando salvarmos esse arquivo e executarmos no navegador(Chrome), a primeira linha será executada e na sequência a segunda linha. Conforme imagem abaixo (onde o console do DevTools vai exibir 20 e logo abaixo 21)

# O Console do Navegador



Só reforçando mais uma vez, não vamos ficar preocupados nesse momento com o que significa esse `console.log`, a sua estrutura como `ponto(.)` e parênteses(), nas próximas aulas veremos certinho a definição, exemplos e como ele funciona(entre várias outras coisas).

O que eu gostaria que vocês soubessem agora, é que esse `console.log` é um método que podemos usar para exibir valores (caracteres, números...) no console do DevTools e isso vai ser muito útil nesse momento em que vocês estão estudando sobre a linguagem(JavaScript).



# Sugestões de prática para essa aula:

1. Refazer e Entender os exemplos dessa aula;
2. Na página 02 dessa aula eu deixei a tag `<head>` como pesquisa, peço a gentileza que essa pesquisa seja efetuada, se eu posso colocar a tag `script` aqui dentro.
3. Alterar os números dentro do `console.log(20)` e `console.log(21)` com números da Vossa preferência e verificar o comportamento dessa alteração dentro do DevTools na aba Console.
4. Abrir o DevTools dentro do Navegador(Chrome) pressionando a tecla F12.

Programação é prática, curiosidade e repetição!!!!



# O que vamos aprender na aula 17:

Na décima sétima aula do curso(Terceira aula de JavaScript) vamos entender sobre: Variáveis, Comentários e Constantes.

**Feedbacks dessa décima sexta aula são bem-vindos.**

Nos vemos na décima sétima aula que será disponibilizada no dia 29/02/2024 às 8h30 (Horário de Brasília) – Dentro do meu perfil no LinkedIn.

