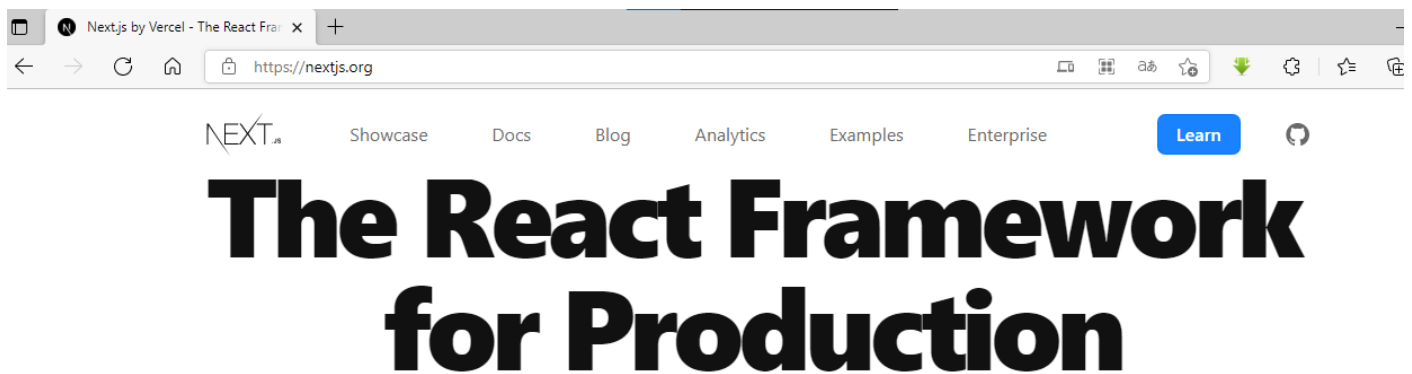


React não é um framework. React é considerada uma biblioteca, a biblioteca mais famosa para construção de aplicações front-end com Javascript.

E o modelo de aplicação que acabamos criando com React é um tipo de aplicação conhecido como SPA, ou seja, aplicação de página única.

Significa que você terá um arquivo HTML para sua aplicação inteira e todo o resto da aplicação é gerado pelo JS.

Sabendo que o React é a biblioteca mais famosa, o NextJS veio trazer a ideia de um framework, uma estrutura para que você construa a aplicação.



A ideia é dar uma melhor experiência no desenvolvimento de aplicações React, vários recursos que você precisa para produção, como por exemplo a produção de conteúdos estáticos, renderização do lado do servidor, suporte ao TypeScript, agrupamento inteligente, pré busca de rotas e muito mais.

O NextJS é uma excelente opção para desenvolver aplicações baseada em React. A estrutura desse framework traz muitos recursos para o rápido desenvolvimento.

Nossa primeira aplicação. Criar a pasta curso-nextjs e dar o comando:

`create-next-app exercicios`

```
Prompt de Comando
Microsoft Windows [versão 10.0.19042.1348]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\Alexandre>node -v
v16.13.1

C:\Users\Alexandre>cd\

C:\>cd C:\curso-nextjs

C:\curso-nextjs>npx create-next-app exercicios
```

```

C:\curso-nextjs>npx create-next-app exercicios
Need to install the following packages:
  create-next-app
Ok to proceed? (y) y
Creating a new Next.js app in C:\curso-nextjs\exercicios.

Using npm.

Installing dependencies:
- react
- react-dom
- next

[██████████] / idealTree:styled-jsx: sill fetch manifest supports-color@^7.1.0

```

Entrar na pasta e digitar code .

```

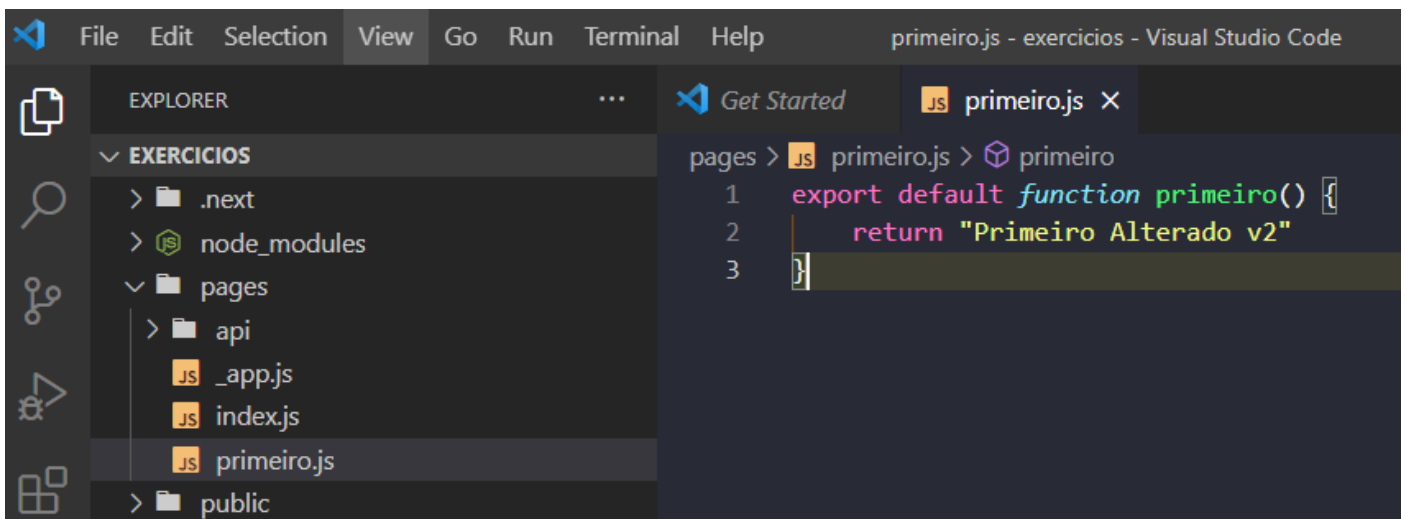
MINGW64:/c/curso-nextjs/exercicios

Alexandre@DESKTOP-78RVV7R MINGW64 /c/curso-nextjs/exercicios
$ code .

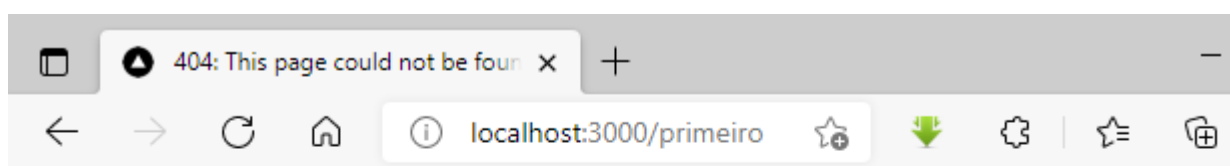
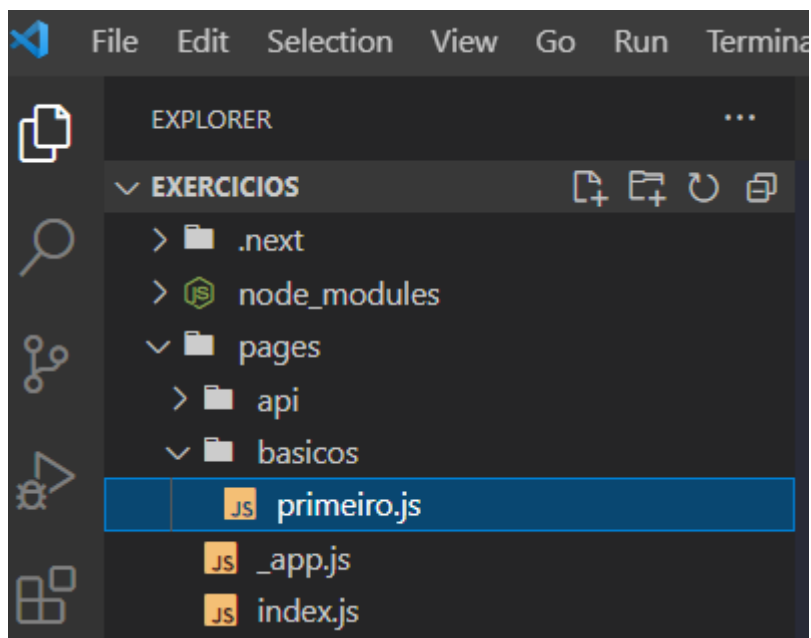
Alexandre@DESKTOP-78RVV7R MINGW64 /c/curso-nextjs/exercicios
$

```

No VSCode criar a o arquivo primeiro.js



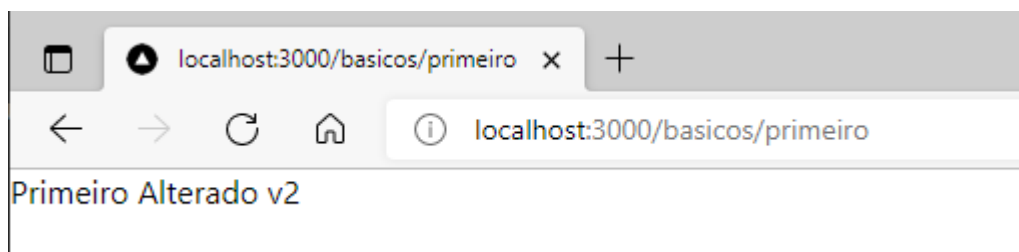
Mover o arquivo para a pasta básicos



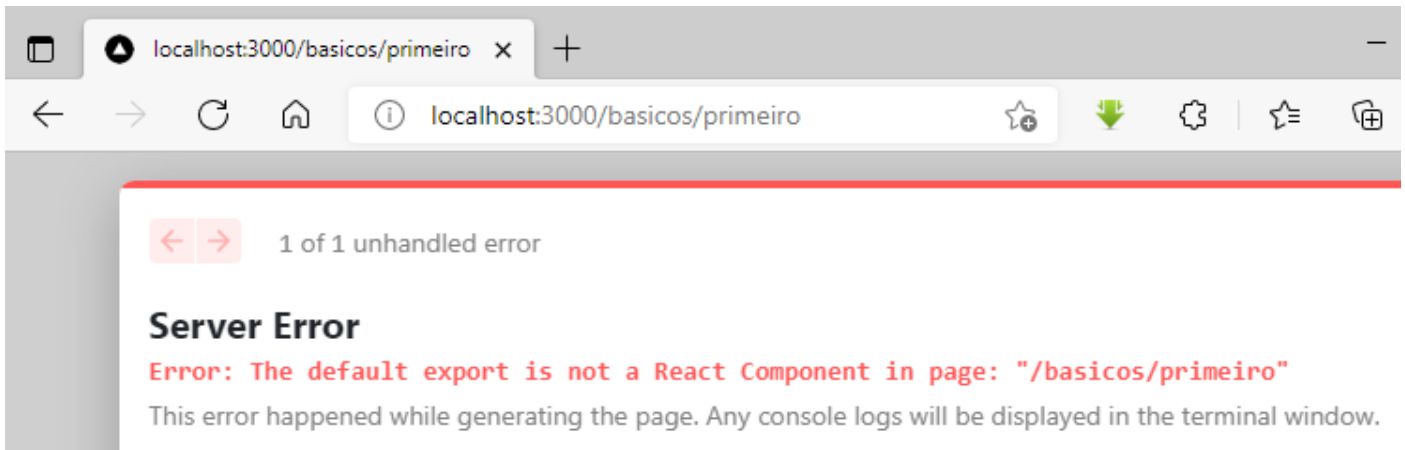
404

This page could not be found.

Alterando o caminho

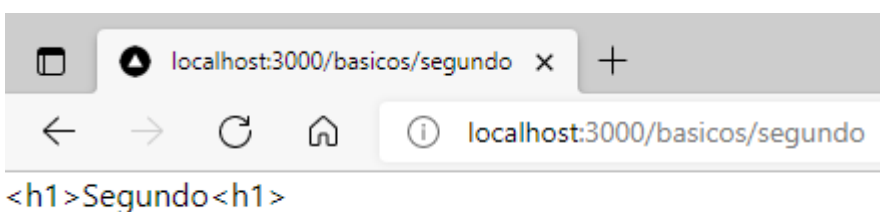
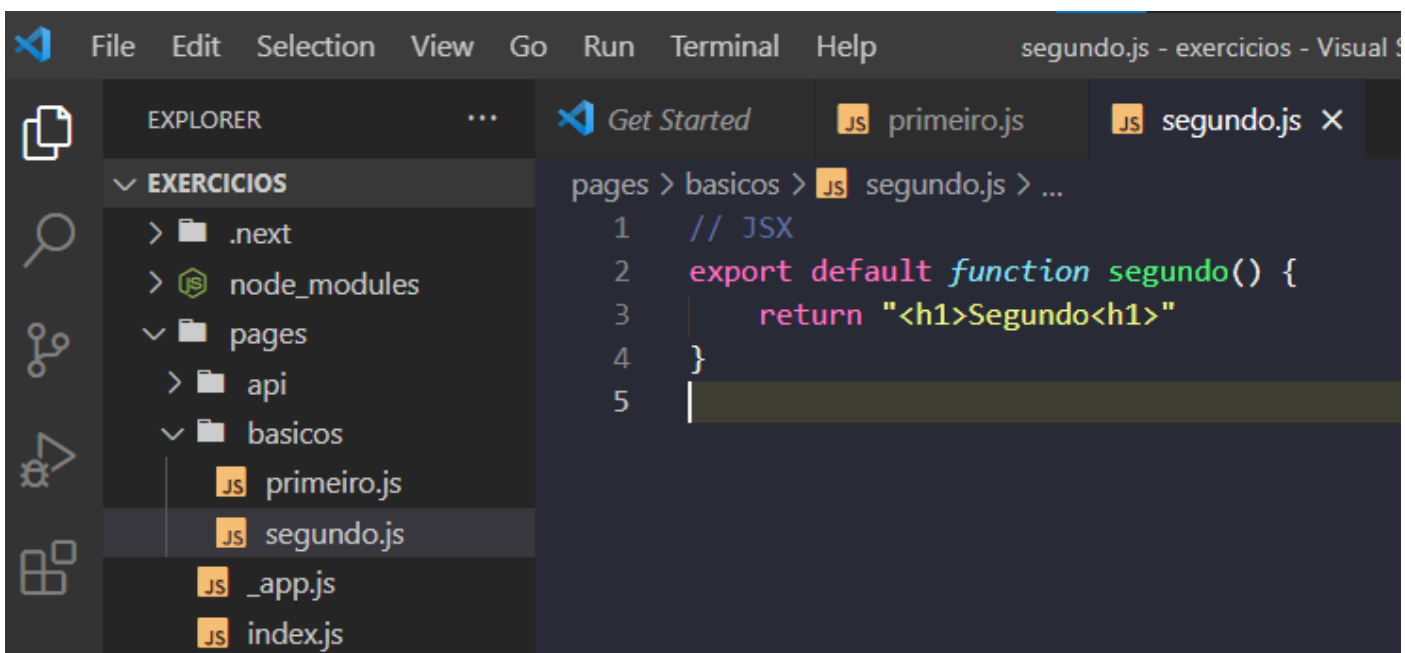


Se retirarmos o default



Fizemos nosso primeiro componente e movemos ele de lugar.

Faremos o segundo.

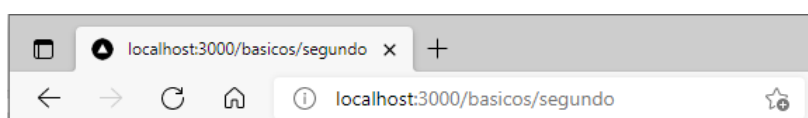


Visual Studio Code interface showing the file explorer on the left with the project structure: EXERCICIOS > .next > node\_modules > pages > api > basics > primeiro.js > segundo.js. The editor displays the content of 'segundo.js':

```

1 // JSX
2 export default function segundo() {
3   return <h1>Segundo</h1>
4 }
5

```

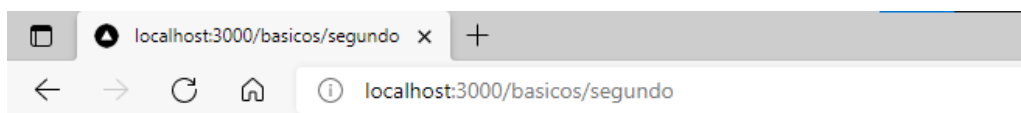


## Segundo

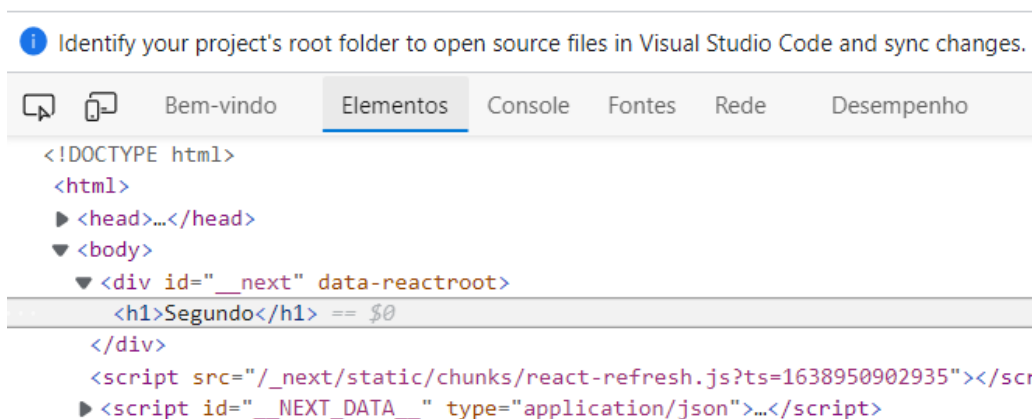
Ocorre é que internamente o React vai manipular a página para criar esses elementos dinamicamente.

Consigo trabalhar misturando HTML com JS.

Inspecionando o elemento:



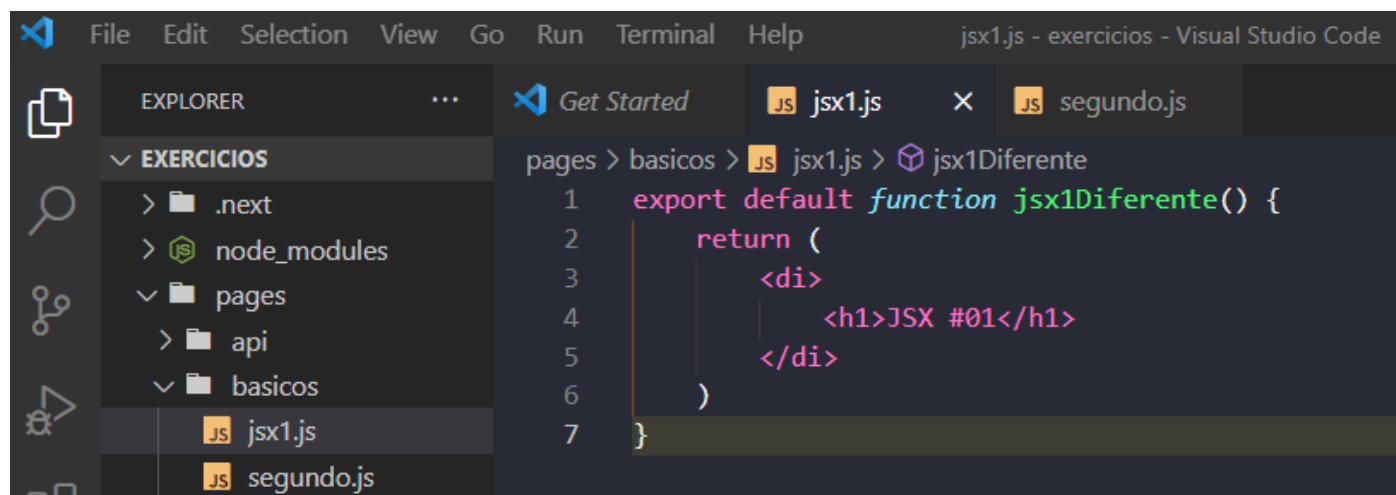
## Segundo



O JSX é como se fosse uma extensão ao JS normal que permite que você escreva HTML diretamente dentro de um código JS sem uso das aspas, e você consegue criar um código muito simples, em que é feita uma tradução para JS puro para que o navegador consiga interpretar JS.

Essa é a facilidade do React, conseguir escrever código HTML dentro do JS.

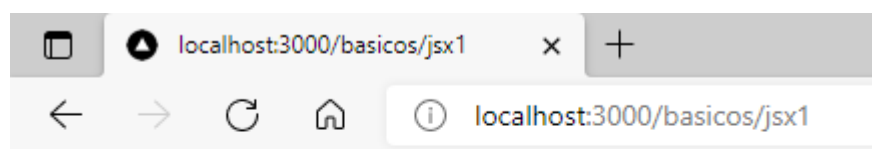
Mudarei o código e a função e você verá que não reflete na página, exceto o endereço.



```

1 export default function jsx1Diferente() {
2   return (
3     <div>
4       <h1>JSX #01</h1>
5     </div>
6   )
7 }

```



## JSX #01

Podemos armazenar esse HTML em uma constante e chama-la



```

1 export default function jsx2() {
2   const conteudo = (
3     <div>
4       <h1>JSX #02 Alterado</h1>
5     </div>
6   )
7   return conteudo
8 }

```

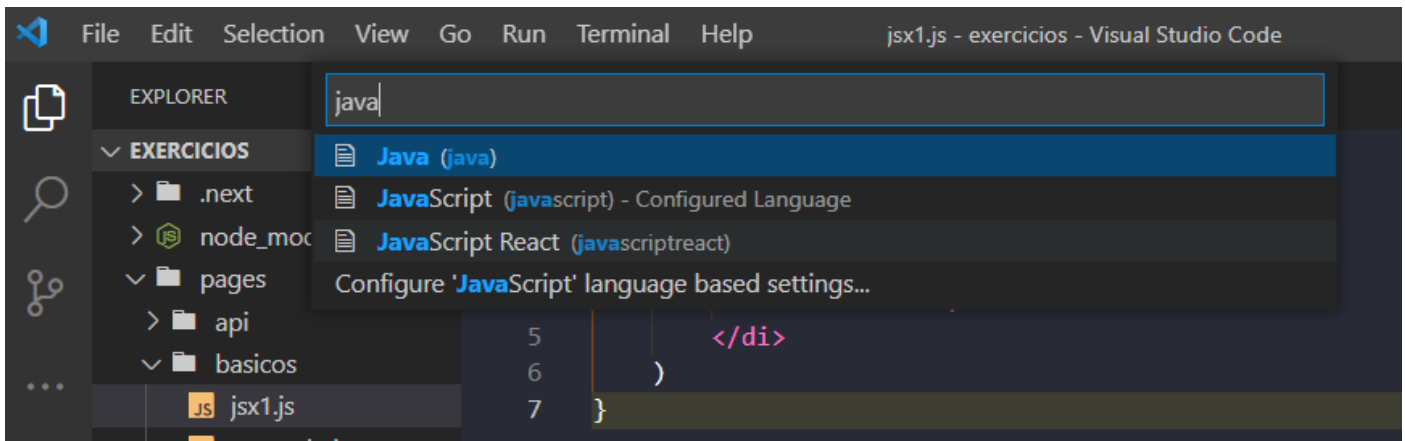
## Antigamente

```

10 // HTML, CSS e JS - Tecnologias
11
12 // Componentes (HTML, JS)

```

## Mudando a linguagem



```

<di>
  h1
  x h1

```

```

<di>
  <h1></h1>

```

```

<di>
  <h1></h1>
  div#legal
  x div#legal

```

```

<di>
  <h1></h1>
  <div id="legal"></div>

```

```

span.opa
  x span.opa

```

```

<span className="opa"></span>

```

EXPLORER

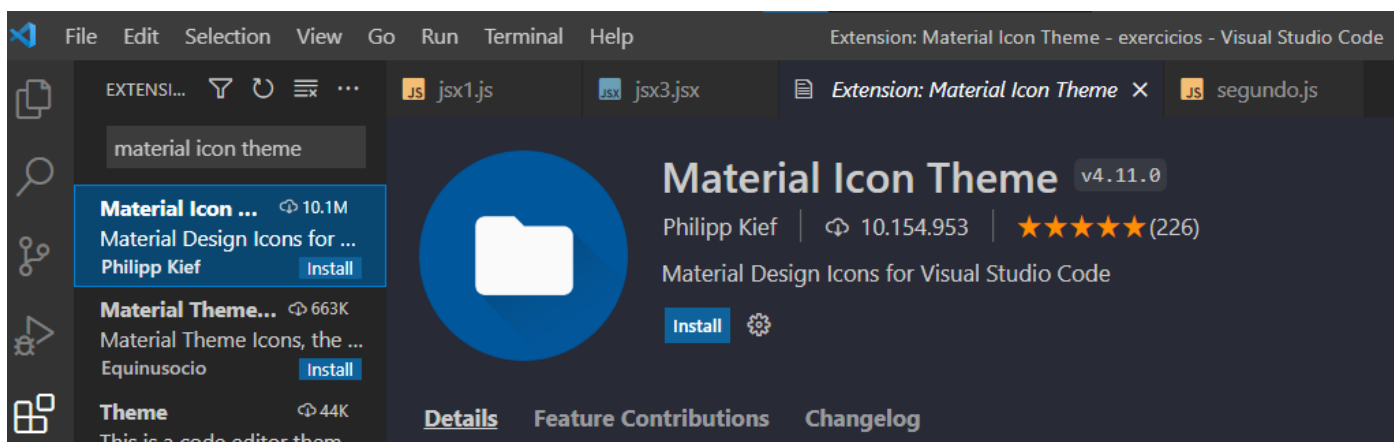
- EXERCICIOS
  - .next
  - node\_modules
  - pages
    - api
    - basicos
      - jsx1.js
      - jsx3.jsx
      - segundo.js

pages > basicos > jsx3.jsx > ...

```

1  export default function jsx3() {
2      return (
3          <div>
4              <h1>JSX #03</h1>
5          </div>
6      )
7  }
8
9

```



## Comunicando o JS com o JSX

EXPLORER

- EXERCICIOS
  - .next
  - node\_modules
  - pages
    - api
    - basicos
      - jsx1.jsx
      - jsx2.jsx
      - jsx3.jsx
      - jsx4.jsx
      - primeiro.jsx
      - \_app.js
      - index.jsx

pages > basicos > jsx4.jsx > jsx4

```

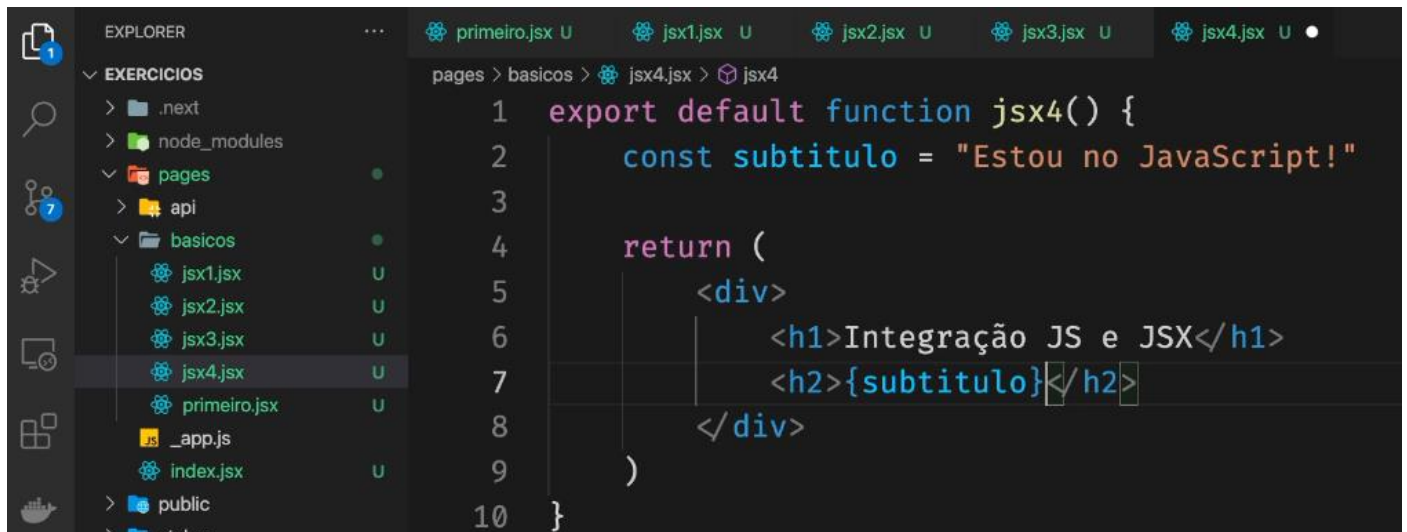
1  export default function jsx4() {
2      const subtítulo = "Estou no JavaScript!"
3
4      return (
5          <div>
6              <h1>Integração JS e JSX</h1>
7              <h2>subtítulo</h2>
8          </div>
9      )
10 }

```





Fazendo a ligação do JS com o JSX



```
return (  
  <div>  
    <h1>Integração JS e JSX</h1>  
    <h2>{subtitulo}</h2>  
    <h3>{3 * 3}</h3>  
    <h4>{Math.max(13, 39)}</h4>  
  </div>  
)
```