Desenvolvimento Front-end com Frameworks

Fundamentos de React

Agenda

Etapa 2: Declaração de variáveis e funções.

- Revisão de Sintaxe.
- Construção de Componentes Básicos.



Revisão de Sintaxe

Template de strings

Template de strings nos fornecem uma alternativa à concatenação.

Essa sintaxe também nos permite inserir variáveis em uma string.

Você os ouvirá ser chamados de template de strings ou literais de template.

```
1 console.log(lastName + ", " + firstName + " " + middleName);
2 console.log(`${lastName}, ${firstName} ${middleName}`);
```

Template de strings

```
const email =
Hello ${firstName},
Thanks for ordering ${qty} tickets to ${event}.
Order Details
$(firstName) $(middleName) $(lastName)
     ${qty} x $${price} = $${qty*price} to ${event}
You can pick your tickets up 30 minutes before
the show.
Thanks,
$ (ticketAgent)
```

Template de strings respeitam os espaços em branco.

Você pode ter uma string que abrange várias linhas sem quebrar seu código

Funções

São blocos nomeados de código projetados para fazer um trabalho específico.

Quando você deseja realizar uma tarefa específica que definiu em uma função, você chama a função responsável por ela.

Se precisar executar essa tarefa várias vezes em todo o programa, não precisará digitar todo o código para a mesma tarefa repetidamente: você apenas chama a função dedicada a lidar com essa tarefa.

Você descobrirá que o uso de funções torna seus programas mais fáceis de escrever, ler, testar e corrigir.

Declaração X Expressão de Funções

```
function logCompliment() {
  console.log("You're doing great!");

logCompliment();
```

```
// Invoking the function before it's declared
hey();
// Function Declaration
function hey() {
   alert("hey!");
}
```

```
const logCompliment = function() {
  console.log("You're doing great!");
2
};
logCompliment();
```

```
// Invoking the function before it's declared
hey();
// Function Expression
const hey = function() {
   alert("hey!");
};
```

Passagem de Parâmetros

```
const logCompliment = function(firstName) {
  console.log('You're doing great, ${firstName}');
logCompliment("Molly");
const logCompliment = function(firstName, message) {
 console.log(`${firstName}: ${message}`);
logCompliment("Molly", "You're so cool");
```

Retorno de Valor

```
const createCompliment = function(firstName, message) {
   return `${firstName}: ${message}`;
};

createCompliment("Molly", "You're so cool");
```

console.log(createCompliment("You're so cool", "Molly"));

Arrow Functions

```
const lordify = function(firstName) {
  return `${firstName} of Canterbury`;
};

console.log(lordify("Dale")); // Dale of Canterbury
console.log(lordify("Gail")); // Gail of Canterbury
```

const lordify = firstName => `\${firstName} of Canterbury`;

Arrow Functions

```
// Typical function
const lordify = function(firstName, land) {
  return '${firstName} of ${land}';
);
// Arrow Function
const lordify = (firstName, land) => `${firstName} of ${land}`;
console.log(lordify("Don", "Piscataway")); // Don of Piscataway
console.log(lordify("Todd", "Schenectady")); // Todd of Schenecta
```

Arrow Functions

```
const lordify = (firstName, land) => {
  if (!firstName) {
    throw new Error ("A firstName is required to lordify");
  if (!land) {
    throw new Error ("A lord must have a land");
  return `${firstName} of ${land} `;
console.log(lordify("Kelly", "Sonoma")); // Kelly of Sonoma
console.log(lordify("Dave")); // ! JAVASCRIPT ERROR
```

Arrow Functions com Objetos

```
const person = (firstName, lastName) =>
{
    first: firstName,
    last: lastName
```

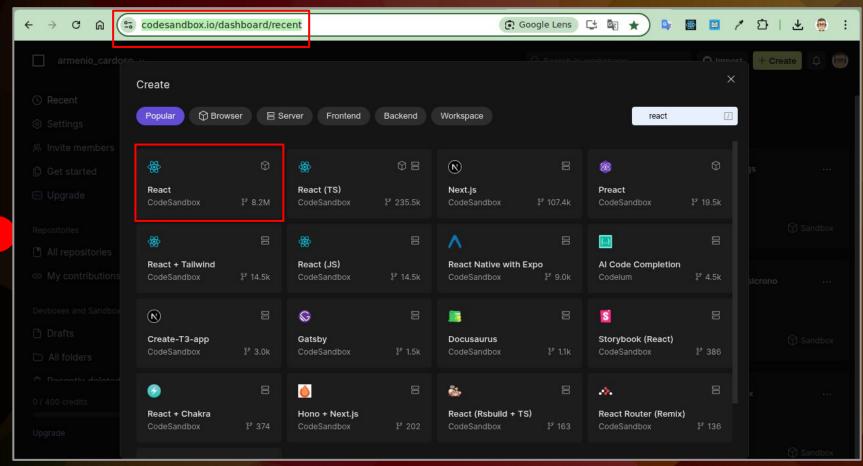
```
console.log(person("Brad", "Janson"));
```

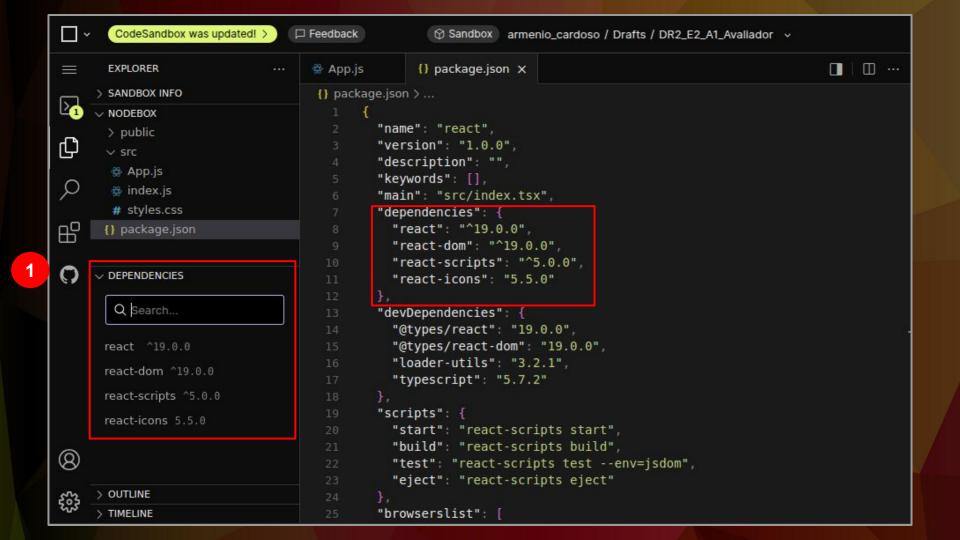
```
const person = (firstName, lastName) => ({
  first: firstName,
  last: lastName
});

console.log(person("Flad", "Hanson"));
```

Esses parênteses ausentes são a fonte de inúmeros bugs em aplicativos JavaScript e React, por isso é importante lembrar disso!

Construção de Componentes Básicos





```
# App.css M x

src > # App.css > ...

1    .container {
        background-color: ■aquamarine;
        border: 1px solid □black;
        margin: 5px;
        padding: 5px;
        6 }
```

```
# App.css M
                    JS App.js M X
     src > JS App.js > ...
            import logo from './logo.svg';
            import './App.css';
            function App() {
3
              return (
       6
                <div>
                  <h3>Componentes Básicos</h3>
                </div>
              );
      11
      12
            export default App;
```

```
# App.css M
               JS App.js M
                              Avaliador.jsx U x
src > Avaliador.isx > ...
      import React from "react";
      import { FaStar } from "react-icons/fa";
      export default function Avaliador() {
        return [
           <FaStar color="grey" />,
          <FaStar color="grey" />,
          <FaStar color="grey" />,
          <FaStar color="grey" />,
          <FaStar color="grey" />
 11
        1;
 12
```

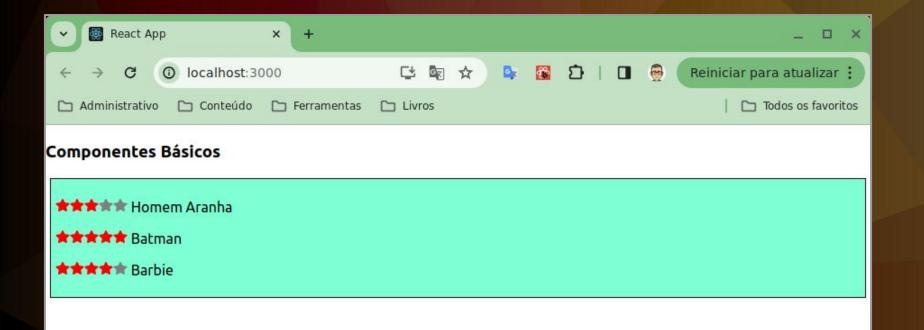
```
# App.css M
              JS App.js M X
                            Avaliador.jsx U
src > JS App.is > ...
      import logo from './logo.svg';
      import './App.css';
      import Avaliador from './Avaliador';
      function App() {
        return (
          <div>
            <h3>Componentes Básicos</h3>
            <div className="container">
              <Avaliador /> Homem Aranha
              <Avaliador /> Batman
 11
 12
              <Avaliador /> Barbie
 13
            </div>
          </div>
        );
      export default App;
```

```
6
```

```
Avaliador.jsx U X
# App.css M
               JS App.js M
src > 👺 Avaliador.jsx > ...
      import React from "react";
      import { FaStar } from "react-icons/fa";
      export default function Avaliador(props) {
          function geraAvaliacao() {
               const avaliacao = [];
               for (let i = 0; i < props.quantidade; i++) {
                   if(i < props.avaliacao) {</pre>
                       avaliacao.push(<FaStar color="red" />);
 11
                     else {
 12
                       avaliacao.push(<FaStar color="grey" />);
 13
               return avaliacao;
           return (geraAvaliacao());
```

```
7
```

```
# App.css M
              JS App.js M X 💮 Avaliador.jsx U
src > JS App.js > ...
      import logo from './logo.svg';
      import './App.css';
      import Avaliador from './Avaliador';
      function App() {
        return (
          <div>
            <h3>Componentes Básicos</h3>
            <div className="container">
             <Avaliador quantidade="5" avaliacao="3" /> Homem Aranha
 11
              <Avaliador quantidade="5" avaliacao="5" /> Batman
 12
              <Avaliador quantidade="5" avaliacao="4" /> Barbie
            </div>
          </div>
        );
      export default App;
```





DESAFIO

Como podemos incluir o título no componente Avaliador?

```
Avaliador.jsx U x
# App.css M
               JS App.js M
src > @ Avaliador.jsx > ...
      import React from "react";
      import { FaStar } from "react-icons/fa";
      export default function Avaliador(props) {
           function geraAvaliacao() {
               const avaliacao = [];
               for (let i = 0; i < props.quantidade; i++) {
                   if(i < props.avaliacao) {</pre>
                       avaliacao.push(<FaStar color="red" />);
                   } else {
                       avaliacao.push(<FaStar color="grey" />);
               return avaliacao;
           return (geraAvaliacao());
```