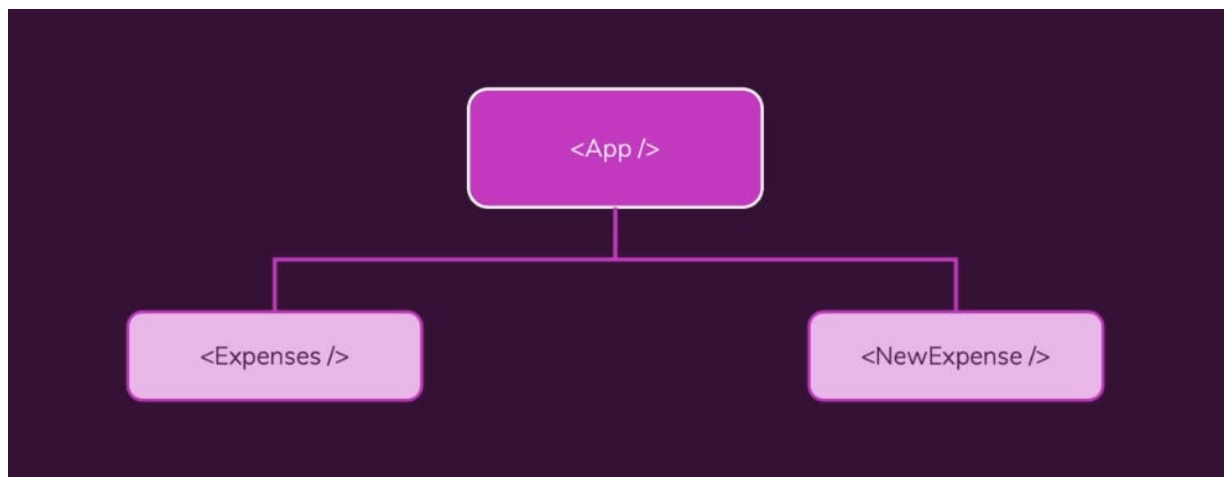


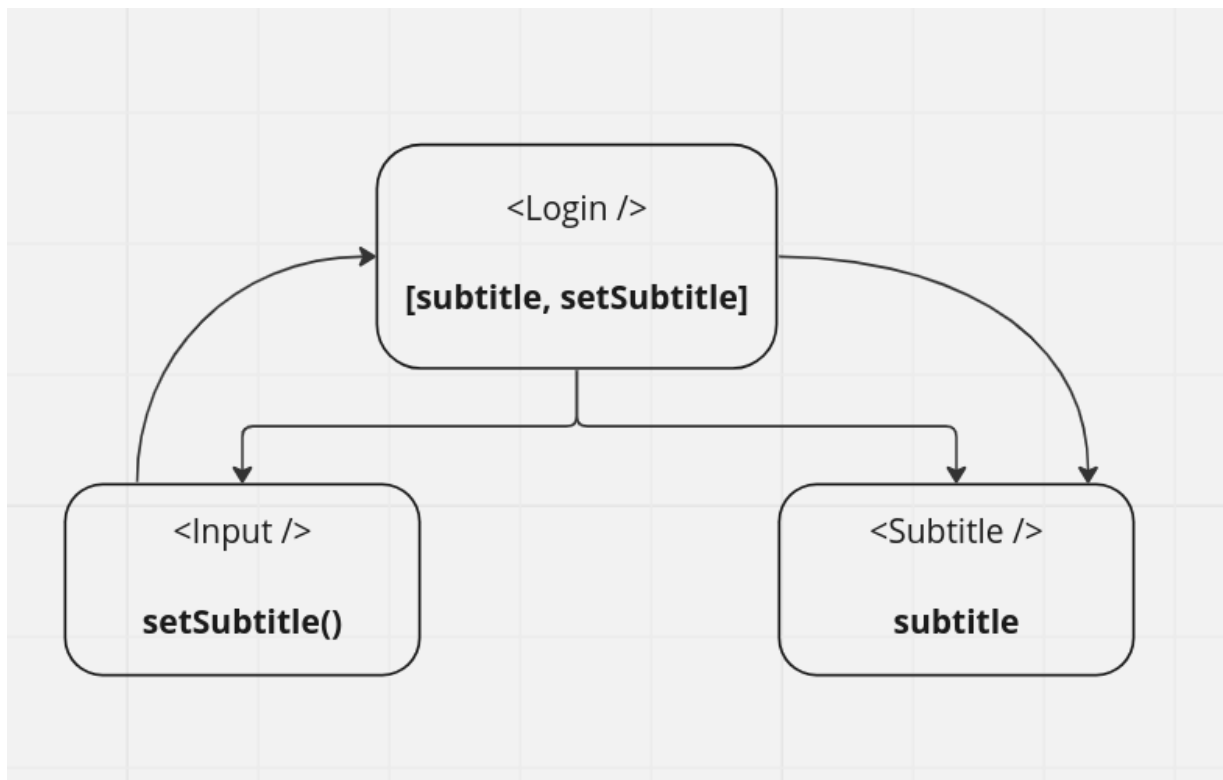
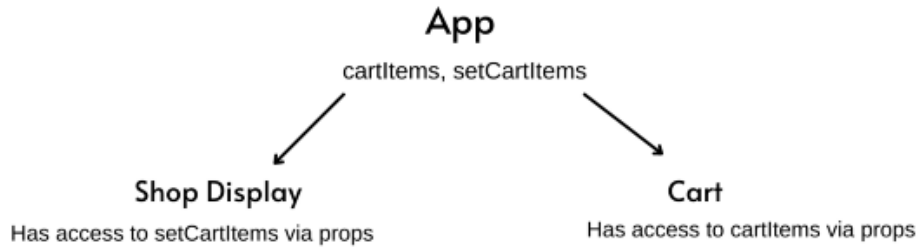
# Aula 06 - Oi Devs

- ✓ CRA
- ✓ Modelagem baseada em componentes
- ✓ JSX (doc oficial)
- ✓ ES6
- ✓ React Router
- ✓ Props
- ✓ Arrow Function
- ✓ Proptypes
- ✓ React State

## State Lifting

Elevando o Estado (state)





No React, o compartilhamento do state é alcançado ao movê-lo para o elemento pai comum aos componentes que precisam dele. Isso se chama “elevanto o state” (state lift)

`find()`

# Renderização Condicional

```
1  const Greeting = ({ isLoggedIn }) => {  
2    if (isLoggedIn) {  
3      return <UserGreeting />;  
4    }  
5    return <GuestGreeting />;  
6  };
```

```
1  const Greeting = ({ isLoggedIn }) => {  
2    let button;  
3    if (isLoggedIn) {  
4      button = <LogoutButton onClick={handleLogoutClick} />;  
5    } else {  
6      button = <LoginButton onClick={handleLoginClick} />;  
7    }  
8    return (  
9      <div>  
10       <Greeting isLoggedIn={isLoggedIn} />  
11       {button}  
12     </div>  
13   );  
14 };
```

```
1  const Mailbox = ({ unreadMessages }) => {  
2    return (  
3      <div>  
4        <h1>Hello!</h1>  
5        {unreadMessages.length > 0 && (  
6          <h2>You have {unreadMessages.length} unread message  
7          s.</h2>  
8        )}  
9      </div>  
10   );  
11 }
```

```
8     </div>
9   );
10  };
```

```
1  isLoggedIn = false
2  const Greeting = ({ isLoggedIn }) => {
3    return (
4      <div>
5        0 usuário <b> {isLoggedIn ? "está" : "não está"} </b>
6        logado.
7      </div>
8    );
9  };
```

```
1  isLoggedIn = true
2  const Greeting = ({ isLoggedIn }) => {
3    <div>
4      {isLoggedIn ? (
5        <LogoutButton onClick={this.handleLogoutClick} />
6      ) : (
7        <LoginButton onClick={this.handleLoginClick} />
8      )}
9    </div>;
10 }
```

## Renderização de Listas

Dado o código abaixo, nós usamos a função `map()` para receber um array de `números` e dobrar o valor de cada um deles. Atribuímos o novo array retornado pela função `map()` para a variável `doubled` e imprime no console:

```
1  const numbers = [1, 2, 3, 4, 5];
2  const doubled = numbers.map((number) => number * 2);
3  console.log(doubled);
```

```
1 const TodoList = ({ todos }) => {
2   return (
3     <ul>
4       {todos.map((todo) => (
5         <li key={todo}>{todo}</li>
6       ))}
7     </ul>
8   );
9 }
```

```
1 const NumberList = ({ numbers }) => {
2   const listItems = numbers.map((number) => (
3     <ListItem key={number.toString()} value={number} />
4   ));
5   return <ul>{listItems}</ul>;
6   };
```

```
1 const NumberList = ({ numbers }) => {
2   return (
3     <ul>
4       {numbers.map((number) => (
5         <ListItem key={number.toString()} value={number} />
6       ))}
7     </ul>
8   );
9   };
```

- ☐ Formulários
- ☐ Binds
- ☐ Composição de Componentes (children)
- ☐ Eventos
- ☐ Métodos do Ciclo de Vida