

Roteiro básico para usar o Cypress em um projeto React.js

Passo 1: Inicialização e Configuração do projeto

1. Crie um novo projeto React usando `npx create-react-app bhaskara`, ou navegue até o diretório do projeto existente.
2. Instale o react-bootstrap usando `npm install react-bootstrap bootstrap --save-dev`
3. Instale o Cypress usando o comando `npm install cypress --save-dev`.
4. Após a instalação, execute o comando `npx cypress open` para abrir a interface do Cypress.

Passo 2: Criação dos arquivos de teste

1. Na interface do Cypress, clique em "**Create new file**" para criar um novo arquivo de teste.
2. Renomeie o arquivo para "bhaskara.cy.jsx".
3. Abra o arquivo "bhaskara.cy.jsx" e remova o código gerado automaticamente.
4. Escreva o seguinte código no arquivo de teste:

```

1 import { URL } from './constant'
2
3 describe('Cálculo da Fórmula de Bhaskara', () => {
4   it('Calcula as raízes corretamente', () => {
5     cy.visit(URL); // Visita a página inicial do aplicativo
6
7     // Insere os valores de a, b e c no formulário
8     cy.get('input[name="a"]').type('1');
9     cy.get('input[name="b"]').type('-3');
10    cy.get('input[name="c"]').type('2');
11
12    // Clica no botão "Calcular"
13    cy.get('button[type="submit"]').click();
14
15    // Verifica se as raízes foram calculadas corretamente
16    cy.get('.card-text').should('contain', 'Existem duas raízes reais diferentes:');
17    cy.get('.card-text li').should('have.length', 2);
18  });
19
20  it('Exibe a mensagem para raízes iguais', () => {
21    cy.visit(URL); // Visita a página inicial do aplicativo
22
23    // Insere os valores de a, b e c no formulário
24    cy.get('input[name="a"]').type('1');
25    cy.get('input[name="b"]').type('2');
26    cy.get('input[name="c"]').type('1');
27
28    // Clica no botão "Calcular"
29    cy.get('button[type="submit"]').click();
30
31    // Verifica se a mensagem de raízes iguais é exibida corretamente
32    cy.get('.card-text').should('contain', 'Existem duas raízes iguais:');
33    cy.get('.card-text p').should('have.length', 2);
34  });
35
36  it('Exibe a mensagem para nenhuma raiz real', () => {
37    cy.visit(URL); // Visita a página inicial do aplicativo
38
39    // Insere os valores de a, b e c no formulário
40    cy.get('input[name="a"]').type('1');
41    cy.get('input[name="b"]').type('1');
42    cy.get('input[name="c"]').type('1');
43
44    // Clica no botão "Calcular"
45    cy.get('button[type="submit"]').click();
46
47    // Verifica se a mensagem de nenhuma raiz real é exibida corretamente
48    cy.get('.card-text').should('contain', 'NÃO Existem raízes reais.');
```

Passo 3: Criação dos componentes React

1. Crie um novo arquivo chamado "Bhaskara.jsx" na mesma pasta do arquivo de teste.
2. Escreva o seguinte código no arquivo "Bhaskara.jsx"

```
1 export function calcularDelta(a, b, c) {  
2   return b ** 2 - 4 * a * c;  
3 }  
4  
5 export function calcularRaizes(a, b, c) {  
6   const delta = calcularDelta(a, b, c);  
7   if (delta < 0) {  
8     return ["na"];  
9   } else if (delta === 0) {  
10    const raiz = -b / (2 * a);  
11    return [raiz];  
12   } else {  
13    const raiz1 = (-b + Math.sqrt(delta)) / (2 * a);  
14    const raiz2 = (-b - Math.sqrt(delta)) / (2 * a);  
15    return [raiz1, raiz2];  
16   }  
17 }  
18
```

3. Crie um novo arquivo chamado "App.jsx" na mesma pasta do arquivo de teste.
4. Escreva o seguinte código no arquivo "App.jsx"

```
1 import React, { useState } from 'react';
2 import { calcularRaizes } from './Bhaskara';
3 import { Button, Card, Form } from 'react-bootstrap';
4
5 function App() {
6   const [a, setA] = useState('');
7   const [b, setB] = useState('');
8   const [c, setC] = useState('');
9   const [raizes, setRaizes] = useState([]);
10
11   const handleInputChange = (e) => {
12     const { name, value } = e.target;
13     if (name === 'a') setA(value);
14     if (name === 'b') setB(value);
15     if (name === 'c') setC(value);
16   };
17
18   const handleSubmit = (e) => {
19     e.preventDefault();
20     const raizesCalculadas = calcularRaizes(Number(a), Number(b), Number(c));
21     setRaizes(raizesCalculadas);
22   };
23
24   const renderRaizes = () => {
25     console.log(raizes)
26     if (raizes[0] === "na") {
27       return <p>NÃO Existem raízes reais.</p>;
28     } else if (raizes.length === 1) {
29       return <p>Existem duas raízes iguais: {raizes[0]}</p>;
30     } else {
31       return (
32         <>
33         <p>Existem duas raízes reais diferentes: </p>
34         <ul>
35           {raizes.map((raiz, index) => (
36             <li key={index}>{raiz}</li>
37           ))}
38         </ul>
39         </>
40       );
41     }
42   };
43 }
```

```
1  return (
2    <Card style={{ width: '18rem', margin: '10px' }}>
3      <Card.Body>
4        <h3>Cálculo Bhaskara</h3>
5        <Card.Title>
6          <Form onSubmit={handleSubmit}>
7            <Form.Group controlId="formA">
8              <Form.Label>a:</Form.Label>
9              <Form.Control type="number" name="a" value={a} onChange={handleInputChange} />
10            </Form.Group>
11            <Form.Group controlId="formB">
12              <Form.Label>b:</Form.Label>
13              <Form.Control type="number" name="b" value={b} onChange={handleInputChange} />
14            </Form.Group>
15            <Form.Group controlId="formC">
16              <Form.Label>c:</Form.Label>
17              <Form.Control type="number" name="c" value={c} onChange={handleInputChange} />
18            </Form.Group>
19            <Button variant="primary" type="submit">Calcular</Button>
20          </Form>
21        </Card.Title>
22        <Card.Text>
23          {raizes.length > 0 && (
24            <div>
25              <p>Raízes:</p>
26              {renderRaizes()}
27            </div>
28          )}
29        </Card.Text>
30      </Card.Body>
31    </Card>
32  );
33 }
34
35 export default App;
36
```

Passo 4: Execução dos testes

1. No terminal, execute o comando `npx cypress open` para abrir a interface do Cypress.
2. Na interface do Cypress, clique no arquivo de teste "bhaskara.spec.js".
3. O navegador Cypress será aberto e os testes serão executados automaticamente.

