

App.css

```
/* Styling for App component */
.App {
  text-align: center;
  font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
  background-color: #81c9e6; /* Light gray background */
  padding: 20px;
  min-height: 100vh; /* Minimum full viewport height */
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
}

/* Styling for the header */
h1 {
  color: #007bff; /* Blue header text */
  font-size: 36px;
  margin-bottom: 20px;
}

/* Styling for the counter */
.counter {
  background-color: #fff; /* White background */
  border-radius: 4px;
  padding: 20px;
  box-shadow: 0px 4px 6px rgba(0, 0, 0, 0.1);
}

/* Styling for the counter value */
h2 {
  font-size: 48px;
  color: #333; /* Dark text color */
  margin-bottom: 20px;
}

/* Styling for the increment and decrement buttons */
button {
  padding: 10px 20px;
  background-color: #007bff; /* Blue button background */
  color: white;
  border: none;
}
```

```

border-radius: 4px;
font-size: 18px;
cursor: pointer;
margin: 0 10px;
transition: background-color 0.2s ease-in-out;
}

button:hover {
  background-color: #0056b3; /* Darker blue on hover */
}

```

App.js

```

import React, { Component } from 'react';
import RandomNumber from './RandomNumber';

class App extends Component {
  constructor() {
    super();
    this.state = {
      randomNumber: null,
      min: 1, // Minimum value for the random number
      max: 100, // Maximum value for the random number
    };
  }

  generateRandomNumber = () => {
    const { min, max } = this.state;
    const randomNum = Math.floor(Math.random() * (max - min + 1)) + min;
    this.setState({ randomNumber: randomNum });
  };

  render() {
    return (
      <div className="App">
        <h1>Random Number Generator</h1>
        <RandomNumber
          randomNumber={this.state.randomNumber}
          generateRandomNumber={this.generateRandomNumber}
        />
      </div>
    );
  }
}

export default App;

```

App.test.js

```
import { render, screen } from '@testing-library/react';
import App from './App';

test('renders learn react link', () => {
  render(<App />);
  const linkElement = screen.getByText(/learn react/i);
  expect(linkElement).toBeInTheDocument();
});
```

Index.css

```
body {
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto',
'Oxygen',
  'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
  sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}

code {
  font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
  monospace;
}
```

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
```

```
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals  
reportWebVitals();
```

OUTPUT:

Random Number Generator

Random Number: 37

Generate

Random Number Generator

Random Number: 72

Generate
