Render props is a technique for sharing code between React components using a prop whose value is a function. It’s a pattern where a component uses a function as a prop to know what to render.

Let's go through an example to illustrate how render props work:

1. Create a component that provides some data and a render prop:

jsx

import React, { Component } from 'react';

class DataProvider extends Component {

state = {

data: 'Hello, render props!',

};

render() {

return this.props.render(this.state.data);

}

}

export default DataProvider;

1. Use the DataProvider component and provide a render prop:

jsx

import React from 'react';

import DataProvider from './DataProvider';

const App = () => {

return (

<div>

<DataProvider render={(data) => (

<h1>{data}</h1>

)} />

</div>

);

};

export default App;

In this example:

* DataProvider is a component that manages some state (data) and uses a render prop (this.props.render) to decide what to render.
* The App component uses DataProvider and provides a function to the render prop. This function receives the data state from DataProvider and renders it inside an <h1> element.

This technique allows you to pass functions as props to a component and use them to determine what to render, making it easy to share and reuse code in different parts of your application.