Compare and contrast the terms "state" and "props." What is the purpose of each? Where does a component get its props? Where does it get its state? How are they modified?

In React, both "state" and "props" are fundamental concepts that play a crucial role in building dynamic and interactive user interfaces. They are used to manage and control different aspects of a component's behavior and appearance.

Props (Properties):

Props are a way to pass data from a parent component to a child component. They allow a parent component to configure and customize a child component's behavior, appearance, or content. Props are provided to a component by its parent component when it is being rendered. Props are immutable. They are meant to be read-only and should not be modified directly within the child component. If changes are needed, they must be made in the parent component and passed down again as new props.

State:

State is used to manage internal data and dynamic aspects of a component. It allows a component to maintain and react to changes in its own data over time, causing re-rendering when the state is updated. Each component can have its own state, which is managed within the component itself. State can be modified using the setState method provided by React. This method ensures that the component's state is updated properly, triggering a re-render and reflecting the changes in the user interface.

import React, { Component } from 'react';

class Counter extends Component {

constructor(props) {

super(props);

this.state = {

count: 0,

};

}

incrementCount = () => {

this.setState({ count: this.state.count + 1 });

};

render() {

return (

<div>

<p>Count: {this.state.count}</p>

<button onClick={this.incrementCount}>Increment</button>

</div>

);

}

}

export default Counter;

The initial count value is stored in the component's state. When the "Increment" button is clicked, the state is updated using **setState**, causing the component to re-render with the updated count value.

References:

Eygi, C. (2021). React.js for beginners — props and state explained. freeCodeCamp.org. https://www.freecodecamp.org/news/react-js-for-beginners-props-state-explained/

GeeksforGeeks. (2023c). ReactJS State vs Props. GeeksforGeeks. https://www.geeksforgeeks.org/reactjs-state-vs-props/

React State vs Props [Differences and Similarities]. (n.d.). https://www.knowledgehut.com/blog/web-development/react-state-vs-props

Subhash, V. (2022, January 6). ReactJS State and Props Explained In Detail | Medium. Medium. https://medium.com/@VinithaSubhash/lets-create-our-first-react-app-f49d40075a12