Name: Ravi Mishra

1. In React, a simple List component is used to render a list of items in a more efficient and reusable way. It allows developers to display a collection of data, such as a list of products, comments, or contacts, as a series of list items.

The simple List component accepts an array of data as a prop, and then maps over the array to generate a list of elements. Each element in the array can be rendered as a separate list item, which can contain various components like text, images, or buttons.

The simple List component provides several benefits, such as:

Reusability: The same component can be used to render different lists of data in different parts of the application.

Performance: React's virtual DOM allows the simple List component to efficiently update the DOM when changes occur, minimizing the number of re-renders needed.

Separation of concerns: The simple List component separates the display logic from the data logic, making the code easier to maintain and debug.

Overall, the simple List component is a powerful tool in React that enables developers to quickly and easily display a list of data in a scalable and efficient manner.

2. Here are some fixes, optimizations, and modifications to the component:

Fix the PropTypes error in WrappedListComponent: The PropTypes.array needs to be wrapped in PropTypes.shape, as it is an array of objects with a text property.

Fix the setSelectedIndex and selectedIndex variables in WrappedListComponent: The variables need to be swapped in order for selectedIndex to be initialized with the value null.

Optimize the rendering of SingleListItem using React.memo: Since SingleListItem doesn't rely on any props other than index, isSelected, onClickHandler, and text, we can use React.memo to memoize the component and prevent unnecessary re-renders.

Modify the onClickHandler function in SingleListItem: The onClickHandler function needs to be wrapped in a function to prevent it from being called immediately on render.

3. Here is the Updated Code

import React, { useState, useEffect, memo } from 'react';

import PropTypes from 'prop-types';

// Single List Item

const SingleListItem = memo(({ index, isSelected, onClickHandler, text }) => {

return (

<li

style={{ backgroundColor: isSelected ? 'green' : 'red'}}

onClick={() => onClickHandler(index)}

>

{text}

</li>

);

});

SingleListItem.propTypes = {

index: PropTypes.number.isRequired,

isSelected: PropTypes.bool.isRequired,

onClickHandler: PropTypes.func.isRequired,

text: PropTypes.string.isRequired,

};

// List Component

const List = memo(({ items }) => {

const [selectedIndex, setSelectedIndex] = useState(null);

useEffect(() => {

setSelectedIndex(null);

}, [items]);

const handleClick = index => {

setSelectedIndex(index);

};

return (

<ul style={{ textAlign: 'left' }}>

{items.map((item, index) => (

<SingleListItem

key={index}

onClickHandler={handleClick}

text={item.text}

index={index}

isSelected={index === selectedIndex}

/>

))}

</ul>

);

});

List.propTypes = {

items: PropTypes.arrayOf(

PropTypes.shape({

text: PropTypes.string.isRequired,

})

),

};

List.defaultProps = {

items: null,

};

export default List;