Name: Gourav Kadyan

Reg No - 12007873

1.

The List component in React is a reusable and efficient way to display a collection of data, such as products or comments, as a list of items. This component accepts an array of data as a prop and generates a list of elements by mapping over the array. Each element can be rendered as a separate list item, which can contain various components. The List component provides reusability, performance, and separation of concerns. It separates the display logic from the data logic, making the code easier to maintain and debug.

Code example:

```

import React from 'react';

const List = ({data}) => {

return (

<ul>

{data.map((item) => (

<li key={item.id}>{item.name}</li>

))}

</ul>

);

};

export default List;

```

In this example, the List component takes an array of `data` as a prop and maps over it to generate a list of `li` elements. Each `li` element displays the `name` property of each item in the array, and the `key` prop is set to the `id` property of each item to help React identify which items have changed. This component can be used to render different lists of data in different parts of the application.

2. Here are some suggested improvements to the List component in React:

1. Fix the PropTypes error in WrappedListComponent by wrapping PropTypes.array in PropTypes.shape, since it is an array of objects with a text property.

2. Swap the setSelectedIndex and selectedIndex variables in WrappedListComponent to ensure that selectedIndex is initialized with null.

3. Optimize the rendering of SingleListItem using React.memo to prevent unnecessary re-renders, since the component doesn't rely on any props other than index, isSelected, onClickHandler, and text.

4. Modify the onClickHandler function in SingleListItem by wrapping it 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;