Frontend Engineer Assignment

React component code review

Q1. Explain what the simple List component does.

ANSWER: -

The List component is a React component that renders an unordered list of items, where each item can be selected by clicking on it. When an item is selected, its background color changes to green. The component consists of two sub-components: SingleListItem and ListComponent.

SingleListItem is a memoized functional component that renders a single list item. It receives four props:

index: the index of the item in the list

isSelected: a boolean indicating whether the item is currently selected

onClickHandler: a callback function that is called when the item is clicked

text: the text content of the item

ListComponent is also a memoized functional component that renders the entire list. It receives one prop:

items: an array of objects, where each object represents a list item and has a "text" property containing the text content of the item.

The ListComponent maintains a state variable "selectedIndex" using the useState hook, which keeps track of the index of the currently selected item. When the "items" prop changes, the useEffect hook resets the selected index to null. The handleClick function is called when an item is clicked, and it updates the selectedIndex state variable to the index of the clicked item.

Q2. What problems / warnings are there with code?

ANSWER: -

There are several problems/warnings with this code:

The PropTypes for the "items" prop in ListComponent is incorrect. It should be PropTypes.arrayOf(PropTypes.shape({text: PropTypes.string.isRequired})) instead of PropTypes.array(PropTypes.shapeOf({text: PropTypes.string.isRequired})).

The isSelected prop in SingleListItem is supposed to be a boolean, but it is being set to the selectedIndex state variable, which is a number. This should be changed to isSelected={selectedIndex === index}.

The onClickHandler prop in SingleListItem is not being passed correctly. It should be onClick={() => onClickHandler(index)} instead of onClick={onClickHandler(index)}.

The setSelectedIndex function call in useEffect hook of ListComponent is missing a depency array. The second argument to useEffect should be [items] to correctly reset the selected index when the items prop changes.

Q3. Please fix, optimize, and/or modify the component as much as you think is necessary.

ANSWER: -

```
import React, { useState, useEffect, useCallback, 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}
    >
      {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 = useCallback(
    (index) => {
      setSelectedIndex(index);
    },
```

```jsx
    [setSelectedIndex]
  );

  return (
   <ul style={{ textAlign: 'left' }}>
     {items &&
      items.map((item, index) => (
        <SingleListItem
          key={index}
          onClickHandler={() => handleClick(index)}
          text={item.text}
          index={index}
          isSelected={selectedIndex === index}
        />
      ))}
   </ul>
  );
});

List.propTypes = {
  items: PropTypes.arrayOf(
    PropTypes.shape({
      text: PropTypes.string.isRequired,
    })
  ),
};

List.defaultProps = {
  items: null,
};

export default List;
```