

1.Explain what the simple List component does.

sol: The simple list component is a react component which renders a list of selectable items highlights the selected item and is composed of two sub-components `WrappedSingleListItem` and `WrappedListComponent`. Depending on whether the item is selected or not, the `WrappedSingleListItem` renders individual list items and changes the style of the `li` element. By iterating through the array of items and giving each item to the `SingleListItem` component, which is a memoized version of `WrappedSingleListItem`, the `WrappedListComponent` renders the list.

2.What problems / warnings are there with code?

- The "PropTypes.array" and "PropTypes.shapeOf" declaration for "items" in the "WrappedListComponent" component should actually be "PropTypes.arrayOf" and "PropTypes.shape"  

```
WrappedListComponent.propTypes = {  
  items: PropTypes.arrayOf(PropTypes.shape({  
    text: PropTypes.string.isRequired,  
  })),  
};
```
- `setSelectedIndex` not defined properly.  
it should be  

```
const [selectedIndex, setSelectedIndex] = useState(null);
```
- `useState()` is defined wrong in this code it should be changed to `const [selectedIndex, setSelectedIndex] = useState(null);`
- When the `map()` method was used on an array that had no initial values and produced a null value, an error was raised. It is advised to initialise the array items with default values in order to avoid this problem.

```
WrappedListComponent.defaultProps = {  
  items: [  
    { text: 'List 1', itemId: 1},  
    { text: 'List 2', itemId: 2},  
    { text: 'List 3', itemId: 3},  
    { text: 'List 4', itemId: 4},  
    { text: 'List 5', itemId: 5},  
  ],  
};
```

- An error has raised, indicating that `WrappedSingleListItem` has been given an invalid prop of type string. The actual value supplied to `isSelected` as a string does not match the defined propType, which is boolean, which causes this issue. it'll changed to  

```
<SingleListItem  
  key={index}  
  onClickHandler={() => handleClick(index)}  
  text={item.text}  
  isSelected={selectedIndex === index}  
>
```

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

```
import React, { useState, memo } from 'react';
import PropTypes from 'prop-types';

// Single List Item
const WrappedSingleListItem = ({
  isSelected,
  onClickHandler,
  text,
}) => {
  return (
    <li
      style={{ backgroundColor: isSelected ? 'green' : 'red' }}
      onClick={onClickHandler}
    >
      {text}
    </li>
  );
};

WrappedSingleListItem.propTypes = {
  isSelected: PropTypes.bool,
  onClickHandler: PropTypes.func,
  text: PropTypes.string.isRequired,
};

const SingleListItem = memo(WrappedSingleListItem);

// List Component
const WrappedListComponent = ({
  items,
}) => {
  const [selectedIndex, setSelectedIndex] = useState(-1);

  const handleClick = index => {
    setSelectedIndex(index);
  };

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

```
/>  
)))
```

```
);  
};
```

```
WrappedListComponent.propTypes = {  
  items: PropTypes.arrayOf(  
    PropTypes.shape({  
      text: PropTypes.string.isRequired,  
      itemId: PropTypes.number.isRequired  
    }),  
  ),  
};
```

```
WrappedListComponent.defaultProps = {  
  items: [  
    { text: 'List 1', itemId: 1},  
    { text: 'List 2', itemId: 2},  
    { text: 'List 3', itemId: 3},  
    { text: 'List 4', itemId: 4},  
    { text: 'List 5', itemId: 5},  
  ],  
};
```

```
const List = memo(WrappedListComponent);
```

```
export default List;
```