1.Explain what does the simple List component does.

React.memo(WrappedListComponent) returns a new memoized component List.

List outputs the same content as the original WrappedListComponent component,

but with one difference — List render is memoized. React reuses the memoized content

as long as title and releaseDate props are the same between renderings.

2.What problems / warnings are there with code?

When we are setting propsTypes of WrappedListComponent then we are using depricated way

to call PropTypes validator's directly which is inappropriate after react v15.

And second thing we can't pass any state or method or handler directly as the developer

did in the code.we can't map null values.

eg:

WrappedListComponent.defaultProps = {

items: [items1, items2, items3],

};

The syntax of propTypes in List component is wrong it should be like this

WrappedListComponent.propTypes = {

items: PropTypes.arrayOf(

PropTypes.shape({

```
    text: PropTypes.string.isRequired,

  })

),

};
```

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

```
import React, { useState, useEffect, memo } from "react";

import PropTypes from "prop-types";


// Single List Item

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

return (


{text}

);

};


WrappedSingleListItem.propTypes = {

index: PropTypes.number,

isSelected: PropTypes.bool,

onClickHandler: PropTypes.func.isRequired,

text: PropTypes.string.isRequired,

};
```

```jsx
const SingleListItem = memo(WrappedSingleListItem);


// List Component

const WrappedListComponent = ({ items }) => {

const [selectedIndex, setSelectedIndex] = useState();


useEffect(() => {

setSelectedIndex(null);

}, [items]);


const handleClick = (index) => {

setSelectedIndex(index);

};


return (

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

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

<SingleListItem

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

text={item.text}

index={index}

isSelected={selectedIndex}

/>
```

```
))}


    );
  };


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


WrappedListComponent.defaultProps = {
  items: [{text: "Archana", index:1}, {text: "Roy", index: 2}],
};


const List = memo(WrappedListComponent);


export default List;
```