Steeleye Project

Name-Himesh Maurya

RegNo-12011077

Frontend Project

College-Lovely Professional University

Explain what the simple List component does.

Ans-List component renders the items array which is an array of object

which has a text as string .

or we can say

List component is a simple component that renders an unordered list of items, where each item is a SingleListItem component.

Example:

```
items=[
  {
     "text":"Raj"
  },
  {
   "text":"Pooja"
  }
]
```

Its gets items array via props .Then it stores the selected item index in

setSelectedIndex.

we are passing onClickHandler, text, index, isSelected to renders the item

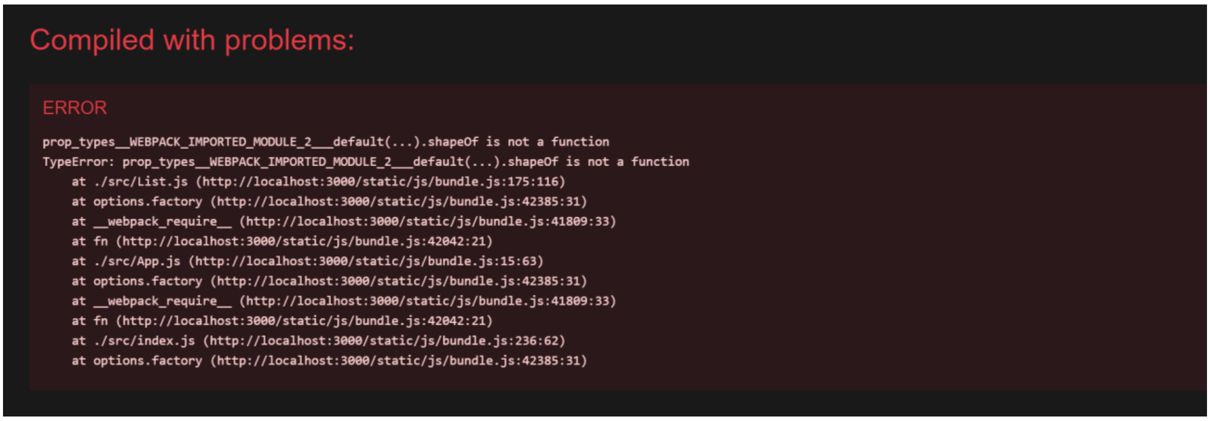
of array with help of map on clicking the any item

its color changes to green and all others to red.

What problems / warnings are there with code?

1)Problem-shapeOf is not a function

solution-we change the PropTypes.array to PropTypes.arrayOf and PropTypes.shapeOf to PropTypes.shape



```
Compiled with problems:

ERROR

prop_types__WEBPACK_IMPORTED_MODULE_2___default(...).shapeOf is not a function
TypeError: prop_types__WEBPACK_IMPORTED_MODULE_2___default(...).shapeOf is not a function
    at ./src/List.js (http://localhost:3000/static/js/bundle.js:175:116)
    at options.factory (http://localhost:3000/static/js/bundle.js:42385:31)
    at __webpack_require__ (http://localhost:3000/static/js/bundle.js:41809:33)
    at fn (http://localhost:3000/static/js/bundle.js:42042:21)
    at ./src/App.js (http://localhost:3000/static/js/bundle.js:15:63)
    at options.factory (http://localhost:3000/static/js/bundle.js:42385:31)
    at __webpack_require__ (http://localhost:3000/static/js/bundle.js:41809:33)
    at fn (http://localhost:3000/static/js/bundle.js:42042:21)
    at ./src/index.js (http://localhost:3000/static/js/bundle.js:236:62)
    at options.factory (http://localhost:3000/static/js/bundle.js:42385:31)
```

2)Problem-onClickHandler directly render when we render component without clicking **button (LOGICAL ERRROR)**

solution-onClickHandler should be call with anonymous function to prevent

it from being immediately called when the component is rendered.

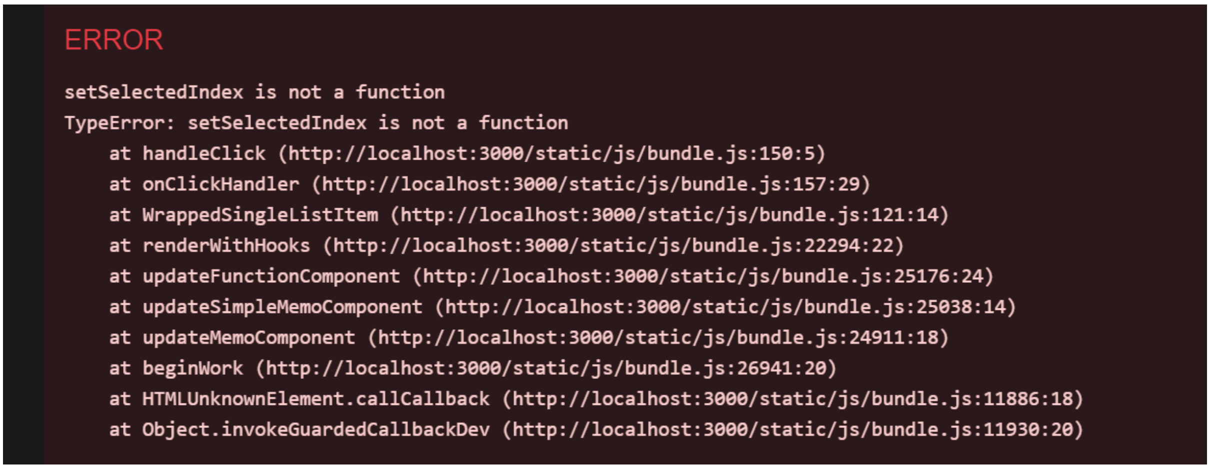
so we change onClick={onClickHandler(index)} to onClick={() => onClickHandler(index)}

3)Problem-setSelectedIndex is not a function

solution-swap setSelectedIndex to selectedIndex, to make the variable name match the state it represents.

and intialize with null .

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



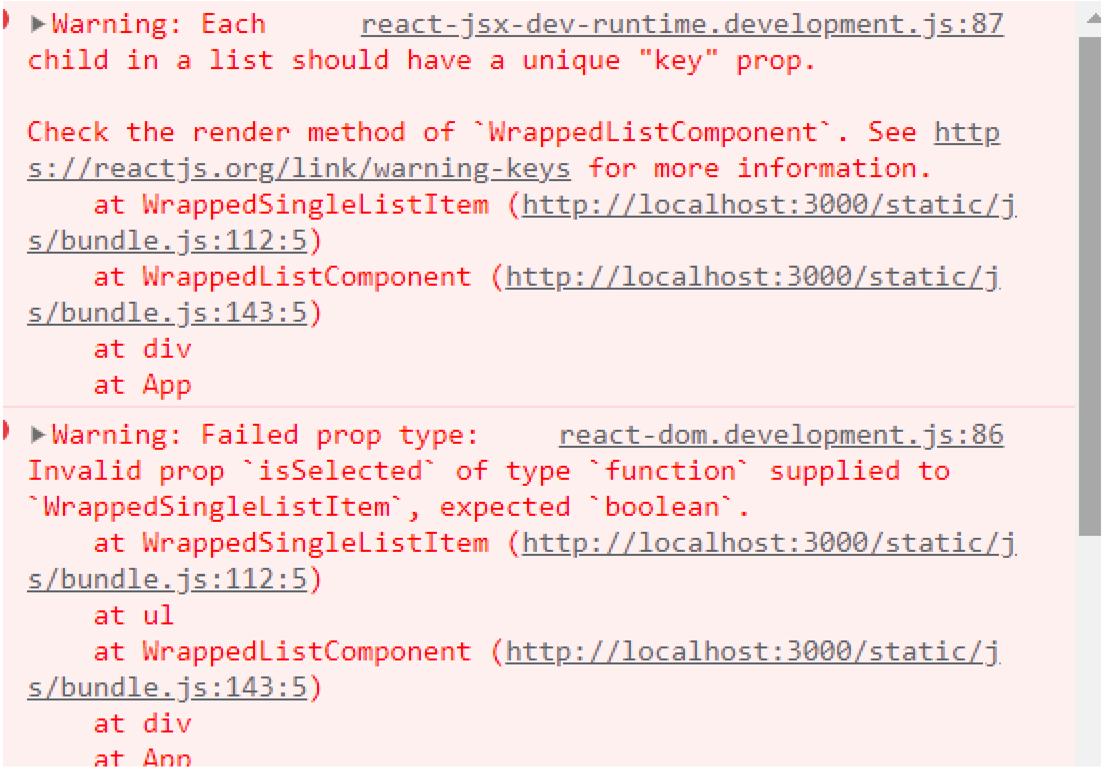
```
ERROR

setSelectedIndex is not a function
TypeError: setSelectedIndex is not a function
    at handleClick (http://localhost:3000/static/js/bundle.js:150:5)
    at onClickHandler (http://localhost:3000/static/js/bundle.js:157:29)
    at WrappedSingleListItem (http://localhost:3000/static/js/bundle.js:121:14)
    at renderWithHooks (http://localhost:3000/static/js/bundle.js:22294:22)
    at updateFunctionComponent (http://localhost:3000/static/js/bundle.js:25176:24)
    at updateSimpleMemoComponent (http://localhost:3000/static/js/bundle.js:25038:14)
    at updateMemoComponent (http://localhost:3000/static/js/bundle.js:24911:18)
    at beginWork (http://localhost:3000/static/js/bundle.js:26941:20)
    at HTMLUnknownElement.callCallback (http://localhost:3000/static/js/bundle.js:11886:18)
    at Object.invokeGuardedCallbackDev (http://localhost:3000/static/js/bundle.js:11930:20)
```

4)Problem-child in a list should have unique property

solution-provide key to map to distinguish the elements from others

like- key={index}



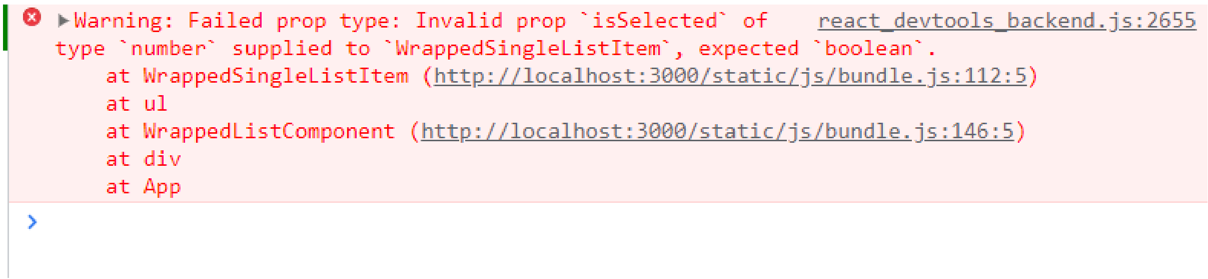
```
▶Warning: Each      react-jsx-dev-runtime.development.js:87
child in a list should have a unique "key" prop.

Check the render method of `WrappedListComponent`. See https://reactjs.org/link/warning-keys for more information.
    at WrappedSingleListItem (http://localhost:3000/static/js/bundle.js:112:5)
    at WrappedListComponent (http://localhost:3000/static/js/bundle.js:143:5)
    at div
    at App
▶Warning: Failed prop type:     react-dom.development.js:86
Invalid prop `isSelected` of type `function` supplied to
`WrappedSingleListItem`, expected `boolean`.
    at WrappedSingleListItem (http://localhost:3000/static/js/bundle.js:112:5)
    at ul
    at WrappedListComponent (http://localhost:3000/static/js/bundle.js:143:5)
    at div
    at App
```

5)Problem-isSelected expected to be boolean

solution-isSelected take boolean value to change the color so we have to make it boolean

so we change isSelected={selectedIndex} to isSelected={selectedIndex===index}



```
▶Warning: Failed prop type: Invalid prop `isSelected` of    react_devtools_backend.js:2655
type `number` supplied to `WrappedSingleListItem`, expected `boolean`.
    at WrappedSingleListItem (http://localhost:3000/static/js/bundle.js:112:5)
    at ul
    at WrappedListComponent (http://localhost:3000/static/js/bundle.js:146:5)
    at div
    at App
>
```

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

Install the dependency prop-types in your reactjs app

IN List.js

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

// Single List Item
const WrappedSingleListItem = ({
  index,
  isSelected,
  onClickHandler,
  text,
}) => {
  return (
    <li
      style={{ backgroundColor: isSelected ? 'green' : 'red'}}
      //onClickHandler should be call with anonymous function to prevent it
      // from being immediately called when the component is rendered.
      onClick={() => onClickHandler(index)}
    >
      {text}
    </li>
  );
};

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

const SingleListItem = memo(WrappedSingleListItem);
// List Component
const WrappedListComponent = ({
  items,
}) => {
 //write selectedIndex at its correct position and initialize with null
const [selectedIndex,setSelectedIndex] = useState(null);
  useEffect(() => {
    setSelectedIndex(null);
  }, [items]);
  const handleClick = index => {
    setSelectedIndex(index);
  };
  return (
    <ul style={{ textAlign: 'left' }}>
    {/* we provide key to map to distinguish the elements from others */}
      {items.map((item, index) => (
        <SingleListItem
          onClickHandler={() => handleClick(index)}
          key={index}
          text={item.text}
          index={index}
          isSelected={selectedIndex===index}
          // as isSelected is boolean so make it boolean equilizing with index
        />
      ))}
    </ul>
  )
};
//error replace shapeOf with shape for proper syntax as shapeOf is not a function
//replace array with arrayOf
WrappedListComponent.propTypes = {
  items: PropTypes.arrayOf(PropTypes.shape({
    text: PropTypes.string.isRequired,
  })),
};
WrappedListComponent.defaultProps = {
  items: null,
};
const List = memo(WrappedListComponent);
export default List;
```

In App.js

```
import List from "./List";
var items=[
  {
       "text":"Raj"

   },
   {
    "text":"Pooja"

}, {
  "text":"Deep"

}
]

function App() {
  return (
    <div className="App">
      <List items={items}/>
    </div>
  );
}

export default App;
```