**Based on the code below answer the following queries:**

**Code:**

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'}}

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,

}) => {

const [setSelectedIndex, selectedIndex] = 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}

/>

))}

</ul>

)

};

WrappedListComponent.propTypes = {

items: PropTypes.array(PropTypes.shapeOf({

text: PropTypes.string.isRequired,

})),

};

WrappedListComponent.defaultProps = {

items: null,

};

const List = memo(WrappedListComponent);

export default List;

1. **Explain what the simple List component does.**

Ans.

We render lists inside a component.

Lists are used to show the data in sequential and an ordered format.Mostly we use in the menus in the websites.It can be created in simpler ways in regular javascript.

Using map functions we used to traverse in the list in react. To create a list in react we had use the map() function for traversing the list element, and for updates, we enclosed them between curly braces{}. To include this new list inside <ul> </ul> elements and render it to the DOM.

Let us just look at how to generate a list with React. To accomplish this, researchers will traverse the column element using the map() code and will surround modifications in square brackets. The parts and components are then assigned to listItems. Finally, display this list on the Web by putting it among the ul> and /ul> components.

**React's components list creation work:**

In order to create lists of pieces, students must use the character characteristic "key," as specified in the React documentation. To determine whether items in a list had already been modified, added, or destroyed, React need so many keys. An individual identifier is required for each object. The id about an entity usually works nicely for all of that.

But we should render the lists inside a component.

**import** React from 'react';

**import** ReactDOM from 'react-dom';

function NameList(props) {

**const** myLists = props.myLists;

**const** listItems = myLists.map((myList) =>

    <li>{myList}</li>

  );

**return** (

    <div>

        <h2>Rendering Lists inside component</h2>

              <ul>{listItems}</ul>

    </div>

  );

}

**const** myLists = ['Peter', 'Sachin', 'Kevin', 'Dhoni', 'Alisa'];

ReactDOM.render(

  <NameList myLists={myLists} />,

  document.getElementById('app')

);

export **default** App;

1. **What problems / warnings are there with code?**

Ans.

Failed to resolve import "prop-types" from "src/App.jsx". Does the file exist?

We got also the warnings like

1. at formatError (file:///home/damner/code/node\_modules/vite/dist/node/chunks/dep-561c5231.js:35003:46)
2. atTransformContext.error (file:///home/damner/code/node\_modules/vite/dist/node/chunks/dep-561c5231.js:34999:19)
3. at normalizeUrl (file:///home/damner/code/node\_modules/vite/dist/node/chunks/dep-561c5231.js:40109:33)
4. at processTicksAndRejections (internal/process/task\_queues.js:95:5)
5. at async TransformContext.transform (file:///home/damner/code/node\_modules/vite/dist/node/chunks/dep-561c5231.js:40243:47)
6. at async Object.transform (file:///home/damner/code/node\_modules/vite/dist/node/chunks/dep-561c5231.js:35252:30)
7. at async loadAndTransform (file:///home/damner/code/node\_modules/vite/dist/node/chunks/dep-561c5231.js:39740:29

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

Ans.

import React, { useState, useEffect, memo } from 'react';

import ReactDOM from 'react-dom';

import PropTypes from 'prop-types';

// Single List Item

const WrappedSingleListItem = ({

  index,

  isSelected,

  onClickHandler,

  text,

}) => {

  return (

    <li

      style={{ backgroundColor: isSelected ? 'green' : 'red'}}

      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,

}) => {

  const [setSelectedIndex, selectedIndex] = 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}

        />

      ))}

    </ul>

  )

};

WrappedListComponent.propTypes = {

  items: PropTypes.array(PropTypes.shapeOf({

    text: PropTypes.string.isRequired,

  })),

};

WrappedListComponent.defaultProps = {

  items: null,

};

const List = memo(WrappedListComponent);

export default WrappedListComponent;