demo1-use-context: Example using useContext() Hook

demo1-use-context: Files Required:

1. index.html	2. index.js
3. studentContext.ts	4. sourseContext.ts
5. App.js	6. StudentComponent.js
7. CourseComponent.js	8. TestComponent.js

1. index.html

2. index.js

```
import React from "react";
import ReactDOM from "react-dom";
import App from "./App";

ReactDOM.render(<App />, document.getElementById("myroot"));
```

3. studentContext.js

```
import React from "react";

//1.Create the Context
const studentContext = React.createContext();

 //2. Define the initialState
const initialState = {
   studentId: 101,
   studentName: "Dandes",
   email: "dandes@jlc.com",
   phone: 55555,
};

export const MyStudentProvider = (props) => {

     return(
         //3.Specify the Provider
         <studentContext.Provider value={initialState}>
         {props.children}
         </studentContext.Provider>
      );
   }

//4. Export the Cntext
//export const MyStudentContext = studentContext;

//4. Export the Consumer
export const MyStudentConsumer = studentContext.Consumer;
```

4. courseContext.js

```
import React from "react";
import { useState } from "react";

//1.Create the Context
const courseContext = React.createContext();

export const MyCourseProvider = (props) => {

  //2. Define the State
  const [courseInfo,setCourseInfo] = useState(
    {
      courseId: "C-11",
      courseName: "Spring MicroServices",
      cost: 18000,
      trainer: "Srinivas Dande",
    }
  );

    return(
      //3.Specify the Provider
      <courseContext.Provider value={courseInfo}>
      {props.children}
      </courseContext.Provider>
    );
  }

//4. Export the Context
export const MyCourseContext = courseContext;

//4. Export the Consumer
//export const MyCourseConsumer = courseContext.Consumer;
```

5. App.js

```
import React, { Component } from "react";

import "bootstrap/dist/css/bootstrap.min.css";

import { MyStudentProvider } from "./StudentContext";
import { MyCourseProvider } from "./CourseContext";

import  StudentComponent  from "./StudentComponent";
import  CourseComponent  from "./CourseComponent";
import  TestComponent  from "./TestComponent";
```

```
class App extends Component {
 render() {
  return (
   <MyStudentProvider>
    <MyCourseProvider>
   <div className="container">
     <StudentComponent/>
    <CourseComponent/>
    <hr/>
    <TestComponent/>
   </div>
   </MyCourseProvider>
   </MyStudentProvider>
  );
 }
}

export default App;
```

6. StudentComponent.js

```
import React from "react";
import { useContext } from "react";
import { MyStudentConsumer } from "./studentContext";

const StudentComponent = (props) => {

 const mystudentData = useContext(MyStudentConsumer);
 const { studentId, studentName, email, phone } = mystudentData;

  return (
   <div className="container">
        <br />  <b><u>
          <h2 className="text-center"> Student Info</h2>  </u> </b>
        <br />
        <div>
          <h3> Student ID : {studentId}</h3>
          <h3> Student Name : {studentName}</h3>
          <h3> Email Id: {email}</h3>
          <h3> Phone No : {phone}</h3>
        </div>
       </div>
       );
  }

export default StudentComponent;
```

7. CourseComponent.js

```
import React from "react";
import { useContext } from "react";
import { MyCourseContext } from "./courseContext";

const CourseComponent = () => {

 const mycourseData = useContext(MyCourseContext);
 const { courseId, courseName, cost, trainer } = mycourseData;

      return(
         <div className="container">
         <br /> <b><u>
            <h2 className="text-center"> Course Info</h2>  </u> </b>
         <br />
         <div>
         <h3> Course Id: {courseId}</h3>
             <h3> Course Name : {courseName}</h3>
             <h3> Cost : {cost}</h3>
             <h3> Trainer : {trainer}</h3>
         </div>
        </div>
       );
      }

export default CourseComponent;
```

8. TestComponent.js

```
import React, { useContext } from "react";
import { MyStudentConsumer } from "./studentContext";
import { MyCourseContext } from "./courseContext";

const TestComponent = () => {

const mystudentData = useContext(MyStudentConsumer);
const mycourseData = useContext(MyCourseContext);

const { studentId, studentName, email, phone } = mystudentData;
const { courseId, courseName, cost, trainer } = mycourseData;
return (
<div className="container">
<br /> <b><u>
<h2 className="text-center"> Student Info</h2>  </u> </b>
<br />
<div>
<h3> Student ID : {studentId}</h3>
<h3> Student Name : {studentName}</h3>
<h3> Email Id: {email}</h3>
<h3> Phone No : {phone}</h3>
<h3> Course Id: {courseId}</h3>
<h3> Course Name : {courseName}</h3>
<h3> Cost : {cost}</h3>
<h3> Trainer : {trainer}</h3>
</div>
</div>
)
}

export default TestComponent;
```

demo2-use-context-reducer: Example using useContext() and useReducer() Hooks

demo2-use-context-reducer: Files Required:

1. index.html	2. index.js
3. courseTypes.js	4. courseContext.ts
5. App.js	6. CourseComponent.js

1. index.html

2. index.js

```
import React from "react";
import ReactDOM from "react-dom";
import App from "./App";

ReactDOM.render(<App />, document.getElementById("myroot"));
```

3. courseTypes.js

```
export const UPDATE_CNAME="UPDATE_CNAME";
export const UPDATE_PRICE="UPDATE_PRICE";
export const UPDATE_DURATION="UPDATE_DURATION";
export const UPDATE_TRAINER="UPDATE_TRAINER";
```

4. courseContext.js

```
import React, { useReducer } from "react";

import {UPDATE_CNAME,UPDATE_PRICE} from './courseTypes';
import {UPDATE_DURATION,UPDATE_TRAINER} from './courseTypes';

//1.Create the Context
const courseContext = React.createContext(null);

//2.Define Initial State
const initialState=  {
   courseId: "C-101",
   courseName: "Master React ",
   price: 10000,
   duration:"50 Hrs",
   trainer: "SD"
}

//3. Define the Reducer
const myreducer = (currentState,action) => {
console.log("C","[courseContext]",action.type,action.payload);
switch(action.type){
   case UPDATE_CNAME:
      return {
         ...currentState,
         courseName:action.payload
      };
   case UPDATE_PRICE:
      return {
         ...currentState,
          price:action.payload
      };
   case UPDATE_DURATION:
      return {
         ...currentState,
          duration:action.payload
      };
   case UPDATE_TRAINER:
      return {
         ...currentState,
          trainer:action.payload
     };
     default:
      return currentState
}
}
```

```
export const MyCourseProvider = (props) => {

    //4.use the useReducer() Hook
  const [courseData,courseDispatch] = useReducer(myreducer,initialState);

     return(
        //5.Specify the Provider
        <courseContext.Provider value={
           {courseData:courseData,
            courseDispatch:courseDispatch
           }
           }>
        {props.children}
        </courseContext.Provider>
     );
   }

//6. Export the Context
export const MyCourseContext = courseContext;
```

5. App.js

```
import React, { Component } from "react";
import "bootstrap/dist/css/bootstrap.min.css";

import { MyCourseProvider } from "./CourseContext";
import  CourseComponent  from "./CourseComponent";

class App extends Component {
  render() {

    return (
     <MyCourseProvider>
     <div className="container">

     <CourseComponent/>

     </div>
     </MyCourseProvider>
    );
  }
}

export default App;
```

6. CourseComponent.js

```
import React, { useContext } from "react";

import { MyCourseContext } from "./courseContext";

import { UPDATE_CNAME, UPDATE_PRICE } from './courseTypes';
import { UPDATE_DURATION, UPDATE_TRAINER } from './courseTypes';

const CourseComponent = (props) => {

const mycourseConsumer = useContext(MyCourseContext);

const { courseData, courseDispatch } = mycourseConsumer;
const { courseId, courseName, price, duration, trainer } = courseData;

const updateCourseName = (updatedCname) => {
console.log("A", "CourseComponent - updateCourseName", updatedCname);
courseDispatch({
type: UPDATE_CNAME,
payload: updatedCname
});
}
const updatePrice = (updatedPrice) => {
console.log("A", "CourseComponent - updatePrice", updatedPrice);
courseDispatch({
type: UPDATE_PRICE,
payload: updatedPrice
});
}

const updateDuration = (updatedDuration) => {
console.log("A", "CourseComponent - updateDuration", updatedDuration);
courseDispatch({
type: UPDATE_DURATION,
payload: updatedDuration
});
}

const updateTrainer = (updatedTrainer) => {
console.log("A", "CourseComponent - updateTrainer", updatedTrainer);
courseDispatch({
type: UPDATE_TRAINER,
payload: updatedTrainer
});
}
```

```
return (
<div className="container">
<br />
<b><u>
<h3 className="text-center"> Course Info</h3>  </u> </b>
<br />
<div>

<table className="table">
<tbody>
<tr>
<td><h4> Course Id: {courseId}</h4> </td>
<td> </td>
</tr>

<tr>
<td><h4> Course Name : {courseName}</h4> </td>
<td>
<button onClick={updateCourseName.bind(this, "Master React 16")} className="btn
btn-success">
Update Course Name
</button>
</td>
</tr>

<tr>
<td><h4> Price : {price}</h4> </td>
<td>
<button onClick={updatePrice.bind(this, 13000)} className="btn btn-success">
Update Price
</button>
</td>
</tr>

<tr>
<td><h4> Duration : {duration}</h4> </td>
<td>
<button onClick={updateDuration.bind(this, "75 Hrs")} className="btn btn-success">
Update Duration
</button>
</td>
</tr>
<tr>
<td><h4> Trainer : {trainer}</h4> </td>
<td>
<button onClick={updateTrainer.bind(this, "Srinivas Dande")} className="btn btn-success">
Update Trainer
</button>
</td>
</tr>
</tbody>
</table>

</div>
</div>

);
}

export default CourseComponent;
```

demo3-use-ref-hook : Example using useRef() Hook

demo3-use-ref-hook : Files Required:

1. index.html	2. index.js
3. App.js	4. LoginComponent.js

1. index.html

2. index.js

3. App.js

```
import React from "react";
import "bootstrap/dist/css/bootstrap.min.css";
import { LoginComponent } from "./LoginComponent";

const App = (props) => {
  return (
    <div className="App container">
      <LoginComponent />
    </div>
  );
}

export default App;
```

4. LoginComponent.js

```
import React, { useEffect, useRef } from "react";

const LoginComponent = () => {

const usernameInput= useRef();
const passowrdInput= useRef();

const verifyUser = () => {
console.log("verifyUser()");
console.log("UN: ",usernameInput.current.value);
console.log("PW: ",passowrdInput.current.value);
}

useEffect( () => {
//usernameInput.current.focus();
passowrdInput.current.focus();
},[]);

return(
<div  className="container">
<table className="table">
<tbody>
```

```
    <tr>
    <td> Username </td>
    <td> <input ref={usernameInput} type="text" name="username"/> </td>
    </tr>
    <tr>
    <td> Password </td>
    <td> <input ref={passowrdInput} type="password" name="password"/></td>
    </tr>
    <tr>
    <td>  </td>
    <td> <button onClick={verifyUser.bind(this)}> Login Now </button>  </td>
    </tr>
    </tbody>
    </table>
    </div>
    );
    }

export default LoginComponent;
```

demo4-memo-function: Example using React.memo() function

demo4-memo-function: Files Required:

1. index.html	2. index.js
3. App.js	4. PriceComponent.js
5. OfferComponent.js	

1. index.html

2. index.js

3. App.js

```
import React from "react";

import "bootstrap/dist/css/bootstrap.min.css";
import { useState } from "react";

import  PriceComponent  from "./PriceComponent";

const App = () =>  {

console.log("[App] - Rendered");

const [courseName,setCourseName] = useState("React JS");
const [coursePrice,setCoursePrice] = useState(10000);
const updateName = () => {
if(courseName === "React JS"){
setCourseName("React JS 16");
}else{
setCourseName("React JS");
}
}
const updatePrice = () => {
if(coursePrice===10000){
setCoursePrice(12000)
}else{
setCoursePrice(10000)
}
}
return (
<div className="container">
<table className="table">
<tbody>
<tr>
<td> Course Name </td>
<td> {courseName} </td>
<td>
<button onClick={updateName.bind(this)} className="btn btn-success">
 Update Name
</button>
</td>
</tr>
<tr>
<td> Course Price </td>
<td> {coursePrice} </td>
<td>
<button onClick={updatePrice.bind(this)} className="btn btn-success">
Update Price
</button>
</td>
</tr>
</tbody>
</table>

<PriceComponent price={coursePrice}/>

</div>
);
}
export default App;
```

4. **PriceComponent.js**

```
import React from "react";

import OfferComponent from "./OfferComponent";

const PriceComponent = (props) => {

console.log("[PriceComponent] - Rendered");

return(
<div className="container">
<br />
<h5> I am Price Component </h5>
<h5>Price : {props.price} </h5>
<OfferComponent price={props.price}/>
</div>
);
}

export default React.memo(PriceComponent);
```

5. **OfferComponent.js**

```
import React from "react";

const OfferComponent = (props) => {

console.log("[OfferComponent] - Rendered");

return(
<div className="container">
<br />
<h5> I am Offer Component </h5>
<h5>Price : {props.price} </h5>
</div>
;
}

export default React.memo(OfferComponent);
```

demo5-use-callback-hook : Example using useCallback() Hook

demo5-use-callback-hook : Files Required:

1. index.html	2. index.js
3. App.js	4. TestComponent.js
5. HeaderComponent.js	6. CourseComponent.js
7. ButtonComponent.js	

1. index.html

2. index.js

3. App.js

```
import React from "react";
import "bootstrap/dist/css/bootstrap.min.css";

import TestComponent from "./TestComponent";

const App = (props) => {
 return (
  <div className="App container text-center">
   <TestComponent />
  </div>
 );
}
export default App;
```

4. TestComponent.js

```
import React, { useState,useCallback} from "react";

import HeaderComponent from './HeaderComponent';
import CourseComponent from './CourseComponent';
import ButtonComponent from './ButtonComponent';

const TestComponent = (props) => {

console.log("[TestComponent] - Rendered");

const [reactPrice,setReactPrice] = useState(10000)
const [angularPrice,setAngularPrice] = useState(10000);
const [devOpsPrice,setDevOpsPrice] = useState(10000);

/*
const updateReactPrice = () => {
console.log("updateReactPrice");
setReactPrice(reactPrice+2000);
}
```

```
const updateAngularPrice = () => {
console.log("updateAngularPrice");
setAngularPrice(angularPrice+2000);
}

const updateDevOpsPrice = () => {
console.log("updateDevOpsPrice");
setDevOpsPrice(devOpsPrice+2000);
}
*/

const updateReactPrice = useCallback(
() => {
console.log("updateReactPrice");
setReactPrice(reactPrice+2000);
} ,[reactPrice]
);

const updateAngularPrice = useCallback(
() => {
console.log("updateAngularPrice");
setAngularPrice(angularPrice+2000);
},[angularPrice]
);

const updateDevOpsPrice = useCallback(
() => {
console.log("updateDevOpsPrice");
setDevOpsPrice(devOpsPrice+2000);
},[devOpsPrice]
);

return(
<div className="container">
<HeaderComponent />
<br/>
<table className="table">
<tbody>
<tr>
<td> <CourseComponent   courseName="React"  price={reactPrice}/> </td>
<td>
<ButtonComponent clickHandler={updateReactPrice} courseName="React">
Update React Price
</ButtonComponent>
</td>
</tr>
<tr>
<td> <CourseComponent courseName="Angular" price={angularPrice}/> </td>
<td>
<ButtonComponent clickHandler={updateAngularPrice} courseName="Angular">
Update Angular Price
</ButtonComponent>
</td>
</tr>
<tr>
<td> <CourseComponent  courseName="DevOps" price={devOpsPrice}/> </td>
<td>
<ButtonComponent clickHandler={updateDevOpsPrice} courseName="DevOps">
Update DevOps Price
</ButtonComponent>
</td>
</tr>
</tbody>
</table>
</div>
);
}

export default TestComponent;
```

5. HeaderComponent.js

```
import React from "react";

const HeaderComponent = (props) => {

  console.log("[HeaderComponent] - Rendered");

  return (
    <div className="container">
      <h5> useCallback() Hook Demo</h5>
    </div >
  );
}
// export default HeaderComponent;
export default React.memo(HeaderComponent);
```

6. CourseComponent.js

```
import React from "react";

const CourseComponent = (props) => {

 console.log(props.courseName,"[CourseComponent] - Rendered");

 return(
   <div className="container">
  <h3> {props.courseName}  - { props.price} </h3>
  </div>
   );
 }

//export default CourseComponent;
export default React.memo(CourseComponent);
```

7. ButtonComponent.js

```
import React from "react";

const ButtonComponent = (props) => {

 console.log(props.courseName,"[ButtonComponent] - Rendered");

 return(
   <div className="container">

  <button onClick={props.clickHandler} className="bg-success">
  {props.children}
  </button>

    </div>


         );
       }
//export default ButtonComponent;
export default React.memo(ButtonComponent);
```

demo6-use-memo-hook : Example using useMemo() Hook

demo6-use-memo-hook: Files Required:

1. index.html	2. index.js
3. App.js	4.

1. index.html

2. index.js

3. App.js

```
import React from 'react';
import { useState, useMemo } from 'react';

function App() {

  const [numbers, setNumbers] = useState({ firstNum: 10, secondNum: 10 })

  const result = useMemo(() => {
    console.log("Function called for doing Math");
    return numbers.firstNum + numbers.secondNum;
  }, [numbers.firstNum, numbers.secondNum]);

  /*
   const result = () => {
     console.log("Function called for doing Math");
     return numbers.firstNum + numbers.secondNum;
   };
  */

  const changeNumbers = () => {
    setNumbers({
      firstNum: numbers.firstNum + 5,
      secondNum: numbers.secondNum
    });
  }

  return (
    <div className="container">
      <h2>Sum : {numbers.firstNum} + {numbers.secondNum} = {result}</h2>
      <div>
        <button onClick={() => changeNumbers()}>Change Numbers</button>
      </div>
    </div>
  );
}

export default App;
```

demo7-custom-hook-1 : Writing Custom Hooks

demo7-custom-hook-1 : Files Required:

1. index.html	2. index.js
3. App.js	4. useDocTitle.js
5. HelloComponent.js	6. HaiComponent.js

1. index.html

2. index.js

3. App.js

```
import React, { Component } from "react";
import "bootstrap/dist/css/bootstrap.min.css";
import HelloComponent from "./HelloComponent";
import HaiComponent from "./HaiComponent";

class App extends Component {
  render() {
    return (
      <div className="App container">
        <HelloComponent />
        <HaiComponent />
      </div>
    );
  }
}
export default App;
```

4. useDocTitle.js

```
import { useEffect } from "react";

const useDocTitle = (color, count) => {

    useEffect(() => {

        document.title = `You clicked ${color} button ${count} times`;

    }, [count]);

}

export default useDocTitle;
```

5. HelloComponent.js

```
import React, { useState } from "react";
import useDocTitle from "./useDocTitle";

const HelloComponent = () => {

  const [count, setCount] = useState(0);

  const updateCounter = () => {
    setCount(count + 1);
  }

  useDocTitle("Green", count);

  return (
    <div className="container">
      <br />
      <h2> I am Hello Component  - {count}</h2>
      <br />
      <button onClick={updateCounter} className="bg-success">
        You clicked me {count} times
      </button>
    </div >
  );
}

export default HelloComponent;
```

6. HaiComponent.js

```
import React, { useState } from "react";
import useDocTitle from "./useDocTitle";

const HaiComponent = () => {

  const [count, setCount] = useState(0);

  const updateCounter = () => {
    setCount(count + 1);
  }

  useDocTitle("Red", count);

  return (
    <div className="container">
```

```
            <br />
            <h2> I am Hai Component  - {count}</h2>
            <br />
            <button onClick={updateCounter} className="bg-danger">
              You clicked me {count} times
            </button>
          </div >
      );
  }

  export default HaiComponent;
```

demo8-custom-hook-2 : Writing Custom Hooks

demo8-custom-hook-2 : Files Required:

1. index.html	2. index.js
3. App.js	4. useCounter.js
5. HelloComponent.js	6. HaiComponent.js

1. index.html

2. index.js

3. App.js

```
import React, { Component } from "react";
import "bootstrap/dist/css/bootstrap.min.css";

import HelloComponent from "./HelloComponent";
import HaiComponent from "./HaiComponent";

class App extends Component {
  render() {
    return (
      <div className="App container">
        <HelloComponent />
        <HaiComponent />
      </div>
    );
  }   }
export default App;
```

4. useCounter.js

```
import { useState } from "react";
const useCounter = (initialCount = 0, value) => {

  const [count, setCount] = useState(initialCount);

  const doIncrement = () => {
    setCount(count + value);
  }

  const doDecrement = () => {
    setCount(count - value);
  }

  const doReset = () => {
    setCount(initialCount);
  }

  return [count, doIncrement, doDecrement, doReset];
}
export default useCounter;
```

5. HelloComponent.js

```
import React from "react";
import useCounter from "./useCounter";

const HelloComponent = () => {

  const [count, doIncrement, doDecrement, doReset] = useCounter(0, 1);

  return (
    <div className="container">
      <br />
      <h2> I am Hello Component  - {count}</h2>
      <br />
      <button onClick={doIncrement}> Increment</button> &nbsp;
      <button onClick={doDecrement}> Decrement</button> &nbsp;
      <button onClick={doReset}> Reset </button>
    </div >

  );
}

export default HelloComponent;
```

6. HaiComponent.js

```
import React from "react";
import useCounter from "./useCounter";

const HaiComponent = () => {

  const [count, doIncrement, doDecrement, doReset] = useCounter(10, 10);

  return (
    <div className="container">
      <br />
      <h2> I am Hai Component  - {count}</h2>
      <br />
      <button onClick={doIncrement}> Increment</button> &nbsp;
      <button onClick={doDecrement}> Decrement</button> &nbsp;
      <button onClick={doReset}> Reset </button>
    </div >
  );
}
export default HaiComponent;
```

demo9-custom-hook-3 : Writing Custom Hooks

demo9-custom-hook-3 : Files Required:

1. index.html	2. index.js
3. App.js	4. useAxios.js
5. CourseList.js	6. BooksList.js
7. AddBook.js	8. MyBook.js
9. DeleteBook.js	

1. index.html

2. index.js

3. App.js

```
import React from "react";
import "bootstrap/dist/css/bootstrap.min.css";

import { BookList } from './BookList';

//import { CourseList } from './ CourseList;
//import { AddBook } from "./AddBook";
//import { MyBook } from "./MyBook";
//import { DeleteBook } from "./DeleteBook";

export const App = () => {
 return (
    <div className="container">
      <br />
    <CourseList />
    {  /*     Test these One by One
     <BookList />
     <AddBook/>
      <MyBook  mybookId="400"/>
      <MyBook  mybookId="400"/>
      <MyBook  mybookId="400"/>
    <DeleteBook mybookId="388"/>
    <DeleteBook mybookId="388"/>
     */    }
     </div >
  );
}
```

4. useMyAxios.js

```
import { useState, useEffect } from 'react';
import axios from 'axios';

axios.defaults.baseURL = "https://www.coursecube.com/jlc-bookstore";

export const useMyAxios = (axiosParams) => {

  const [response, setResponse] = useState([]);
  const [error, setError] = useState('');
  const [loading, setLoading] = useState(true);

  const processRequest = async (params) => {
    try {
      const result = await axios.request(params);
      setResponse(result.data);
      console.log("useMyAxios : Result : ", result.data);
    } catch (error) {
      setError(error);
    } finally {
      setLoading(false);
    }
  };
      useEffect(() => {
        processRequest(axiosParams);
      }, []);  // execute only once

  return { response, error, loading };
};
```

5. CourseList.js

```
import React from "react";
import "bootstrap/dist/css/bootstrap.min.css";
import { useMyAxios } from "./useMyAxios";

export const CourseList = () => {

const { response, loading, error } = useMyAxios({
    method: 'GET',
    url: '/mini-courses',
});

const mycourseList = response.map(
    mycourse => {
    return (
    <tr key={mycourse.courseId}>
    <td> {mycourse.courseId} </td>
    <td> {mycourse.courseName} </td>
    <td> {mycourse.duration} </td>
    <td> {mycourse.trainer} </td>
    <td> {mycourse.enrollments} </td>
    <td>
    <button className="btn btn-success"> View Details </button>
    </td>
    </tr>
    );
    }
);

const mycourseListToDisplay = (
    <table className="table">
    <thead>
    <tr>
    <th> Course ID</th>
    <th> Course Name</th>
    <th> Duration</th>
    <th> Trainer</th>
    <th> Enrollments</th>
    <th> </th>
    </tr>
    </thead>
    <tbody>
    {mycourseList}
    </tbody>
    </table>
);
const messageToDisplay = <h3>{error.message} </h3>;

let toDisplay = null;

if (loading) {
        toDisplay = (<h3>Loading...Please Wait...</h3>);
}
else {
    if (error) {
        toDisplay = messageToDisplay;
    } else {
        toDisplay = mycourseListToDisplay;
    }
}
return (
<div className="container">
<br />
{toDisplay}
</div >
);
}
```

6. BookList.js

```
import React from "react";

import "bootstrap/dist/css/bootstrap.min.css";

import { useMyAxios } from "./useMyAxios";

export const BookList = () => {

const { response, loading, error } = useMyAxios({
method: 'GET',
url: 'https://www.coursecube.com/jlc-bookstore/mybooks',
data: {
},
});

const mybooksList = response.map((mybook) => {
return (
    <tr key={mybook.bookId}>
    <td> {mybook.bookId} </td>
    <td> {mybook.bookName} </td>
    <td> {mybook.author} </td>
    <td> {mybook.price} </td>
    <td> {mybook.category} </td>
    <td> {mybook.publications} </td>
    <td>
    <button className="btn btn-success"> View Details </button>
    </td>
    </tr>
);
});

const mybooksListToDisplay = (
    <table className="table">
    <thead>
    <tr>
    <th> Book ID</th>
    <th> Book Name</th>
    <th> Author</th>
    <th> Price</th>
    <th> Category</th>
    <th> Publications</th>
    <th> </th>
    </tr>
    </thead>
    <tbody>{mybooksList}</tbody>
    </table>
);

const messageToDisplay = <h3>{error.message} </h3>;

let toDisplay = null;
if (loading) {
    toDisplay = (<h3>Loading...Please Wait...</h3>);
}
else {
    if (error) {
    toDisplay = messageToDisplay;

    } else {
    toDisplay = mybooksListToDisplay;
    }
}
return (
<div className="container">
<br />
{toDisplay}
</div >
);
}
```

7. AddBook.js

```
import React from "react";
import "bootstrap/dist/css/bootstrap.min.css";

import { useMyAxios } from "./useMyAxios";

export const AddBook = () => {

    const { response, loading, error } = useMyAxios({
        method: 'POST',
        url: '/addBook',
       headers: {
         accept: '*/*'
        },
        data: {
            bookName: "Learn Java15",
            author: "Dande",
            price: 999,
            category: "Java",
            publications: "JLC"
        },
    });

    console.log("Response : ", response);

    return (
        <div className="container">
            {response}
        </div >
    );

}
```

8. MyBook.js

```
import React from "react";
import "bootstrap/dist/css/bootstrap.min.css";

import { useMyAxios } from "./useMyAxios";

export const MyBook = (props) => {

    const { response, loading, error } = useMyAxios({
        method: 'GET',
        url: `/mybook/${props.mybookId}`,
    });
console.log("MyBook - Response : ", response);
const mybook= response;
return (
    <div className="container">
    <table className="table">
    <tbody>
    <tr key={mybook.bookId}>
    <td> {mybook.bookId} </td>
    <td> {mybook.bookName} </td>
    <td> {mybook.author} </td>
    <td> {mybook.price} </td>
    <td> {mybook.category} </td>
    <td> {mybook.publications} </td>
    </tr>
    </tbody>
    </table>
    </div >
);
}
```

9. DeleteBook.js

```
import React from "react";
import "bootstrap/dist/css/bootstrap.min.css";

import { useMyAxios } from "./useMyAxios";

export const DeleteBook = (props) => {

  const { response, loading, error } = useMyAxios({
    method: 'DELETE',
    url: `/deleteBook/${props.mybookId}`,
  });

  console.log("MyBook - Response : ", response);

  return (
    <div className="container">
      {response}
    </div >
  );

}
```