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

import './App.css';

import Card from './components/card';

import Navbar from './components/navbar';

function App(){

  const [arr,setArr] = useState([]);  // To show data

  const [users, setUsers] = useState([]);  // To patch data

  const [showForm, setShowForm] = useState(false); // To show form

  const [searchStr, setSearchStr] = useState(""); // To search a movie

  const [newArr, setNewArr] = useState([]); // To display the entered content in search box

  const [wholearr, setWholeArr] = useState([]);

  const pageArr = [];

  let i =1;

  while(i <= wholearr.length / 3){

    pageArr.push(i);

    i++;

  };

  const [page, setPage] = useState(1);

  // To show data

  const getData = async() =>{

    await fetch("http://localhost:8080/data",{

      method: "GET"

    })

    .then(res => res.json())

    .then(data => {

      // console.log(data);

      // setArr(data);

      setUsers(data); // For patch operation

      setNewArr(data); // So that No data found is not shown when App.js is started

      setWholeArr(data);

      const end = page \* 4;

      const start = page -1;

      const newData = data.slice(start \* 4, end);

      setArr(newData);

    });

  };

  const handlePage = (el) => {

    setPage(el);

  }

  // To show form when POST button is clicked

  const handlePost = () => {

    setShowForm(true);

  };

  // To submit form when post button is clicked

  const handleSubmit = (e) => {

    e.preventDefault();

    const title = e.target.elements.title.value;

    const posterlink = e.target.elements.posterLink.value;

    const releaseYear = e.target.elements.releaseYear.value;

    const review = e.target.elements.review.value;

    if(title && posterlink && releaseYear && review){

      fetch("http://localhost:8080/data",{

        method: "POST",

        headers:{

          "Content-Type": "application/json"

        },

        body: JSON.stringify({

          Title: title,

          Poster: posterlink,

          Year: releaseYear,

          Review: review

        })

      })

      .then(res => {

        if(res.ok){

          alert("Successfully Posted !!");

          window.location.reload();

        }

      });

      setShowForm(false);

    }

    else{

        alert("Fill in the fields to proceed!!");

    }

  };

  // To update review using PATCH operation

   const handlePatch = (userId) => {

    const newRev = prompt('Enter the new review: ');

    if (newRev) {

      const updatedUsers = users.map(user => {

        if (user.id === userId) {

          return { ...user, Review: newRev };

        }

        return user;

      });

      setUsers(updatedUsers);

      fetch(`http://localhost:8080/data/${userId}`, {

        method: 'PATCH',

        headers: {

          'Content-Type': 'application/json',

        },

        body: JSON.stringify({ Review: newRev }),

      })

        .then(response => response.json())

        .then(() => {

          alert("Review Updated Successfully !!");

          window.location.reload();

        })

        .catch(error => {

          console.error('Error:', error);

        });

    }

  };

  // To delete a movie

  const handleDelete = (userId) =>{

    fetch(`http://localhost:8080/data/${userId}`,{

      method: "DELETE"

    })

    .then(() =>{

      const updatedUsers = users.filter(user => user.id !== userId);

      setUsers(updatedUsers);

      alert("Movie Deleted Successfully");

      window.location.reload();

    });

  };

  // To search a movie through search box

  const handleChange= (e) => {

    setSearchStr(e.target.value);

  };

  const getSearchedData = () => {

    const newData = arr.filter((item) => item.Title.toLowerCase().includes(searchStr.toLowerCase())); // only search the item in that page

    // const newData = wholearr.filter((item) => item.Title.toLowerCase().includes(searchStr.toLowerCase())); // searches items in every page

    setNewArr(newData);

  };

  useEffect(() => {

    getData();

  },[page]);

  useEffect(() => {

  getSearchedData();

  }, [searchStr]);

  return(

    <div className="App">

      <Navbar onSearch={(e) => handleChange(e)} />

      <div>

      <button onClick={handlePost} >POST</button>

      {showForm &&(

        <div id='Form'>

          <form onSubmit={handleSubmit}>

            Movie Title: <input type='text' name='title' placeholder='Enter movie title' /><br />

            Poster Link: <input type='text' name='posterLink' placeholder='Enter poster link' /><br />

            Year of Release: <input type='number' name='releaseYear' placeholder='Enter year of release' /><br />

            Movie Review: <input type='text' name='review' placeholder='Enter movie review' /><br />

            <button type="submit">Submit</button>

          </form>

        </div>

      )}

      </div>

      {searchStr ? (

        <div style={{ display: 'grid', gridTemplateColumns: 'repeat(3,1fr)', marginTop: '5%' }}>

          {newArr.length > 0 ? (

            newArr.map((el) => (

              <Card

                image={el.Poster}

                title={el.Title}

                year={el.Year}

                content={el.Review}

                onPatch={() => handlePatch(el.id)}

                onDelete={() => handleDelete(el.id)}

              />

            ))

          ) : (

            <h1>No Data Found!!!</h1>

          )}

        </div>

      ) : (

        <div style={{ display: 'grid', gridTemplateColumns: 'repeat(3,1fr)', marginTop: '5%' }}>

          {arr.length > 0 ? (

            arr.map((el) => (

              <Card

                image={el.Poster}

                title={el.Title}

                year={el.Year}

                content={el.Review}

                onPatch={() => handlePatch(el.id)}

                onDelete={() => handleDelete(el.id)}

              />

            ))

          ) : (

            <h1>No Data Found!!!</h1>

          )}

        </div>

      )}

      {pageArr.map((el) => {

              return <button onClick={() => handlePage(el)}>{el}</button>

      })}

    </div>

  );

}

export default App;