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

import axios from 'axios';

import Button from 'react-bootstrap/Button';

import Form from 'react-bootstrap/Form';

const FormComponent = () => {

  const [name, setName] = useState("");

  const [email, setEmail] = useState("");

  const [password, setPassword] = useState("");

  const [msg, setMsg] = useState([]);

  const [editingId, setEditingId] = useState(null);

  const handleSubmit = (e) => {

    e.preventDefault();

    const formData = {

      name: name,

      email: email,

      password: password

    };

    if (editingId) {

      editHandle(editingId, formData);

    } else {

      axios

        .post('<http://localhost:8000/frm/create>', formData)

        .then((response) => {

          console.log(response);

          getData();

        })

        .catch((error) => {

          console.error(error);

        });

    }

  };

  useEffect(() => {

    getData();

  }, []);

  const getData = async () => {

    try {

      const response = await axios.get('<http://localhost:8000/frm/list>');

      console.log(response);

      setMsg(response.data);

    } catch (error) {

      console.log(error);

    }

  };

  const editHandle = async (id, formData) => {

    try {

      const response = await axios.put(`<http://localhost:8000/frm/update/>${id}`, formData);

      console.log(response);

      getData();

      setEditingId(null);

    } catch (error) {

      console.log(error);

    }

  };

  const handleEdit = (id) => {

    const entry = msg.find((element) => element.\_id === id);

    if (entry) {

      setName(entry.name);

      setEmail(entry.email);

      setPassword(entry.password);

      setEditingId(id);

    }

  };

  const deleteEntry = async (id) => {

    try {

      await axios.delete(`<http://localhost:8000/frm/delete/>${id}`);

      getData();

    } catch (error) {

      console.log(error);

    }

  };

  return (

    <>

      <Form onSubmit={handleSubmit}>

        <Form.Group className="mb-3">

          <Form.Label>Name</Form.Label>

          <Form.Control type="text" placeholder="Name" value={name} onChange={(e) => setName(e.target.value)} required />

        </Form.Group>

        <Form.Group className="mb-3" controlId="formBasicEmail">

          <Form.Label>Email address</Form.Label>

          <Form.Control type="email" placeholder="Enter email" value={email} onChange={(e) => setEmail(e.target.value)} required />

        </Form.Group>

        <Form.Group className="mb-3" controlId="formBasicPassword">

          <Form.Label>Password</Form.Label>

          <Form.Control type="password" placeholder="Password" value={password} onChange={(e) => setPassword(e.target.value)} required />

        </Form.Group>

        <Button variant="primary" type="submit">

          {editingId ? 'Update' : 'Submit'}

        </Button>

      </Form>

      {msg.map((element) => (

        <div key={element.\_id}>

          <h3>Submitted Data:</h3>

          <p>Name: {element.name}</p>

          <p>Email: {element.email}</p>

          <p>Password: {element.password}</p>

          <Button variant="info" onClick={() => handleEdit(element.\_id)}>Edit</Button>

          <Button variant="danger" onClick={() => deleteEntry(element.\_id)}>Delete</Button>

        </div>

      ))}

    </>

  );

};

export default FormComponent;