Assignment 9

1.Using Json placeholder display the data on the console using Dispatch

Using payload axios.get(https://jsonplaceholder.typicode.com/users)

import React, { useEffect } from "react";

import axios from "axios";

export default function App() {

  const getData = async () => {

    const response = await axios.get(

      "https://jsonplaceholder.typicode.com/users"

    );

    console.log(response.data);

  };

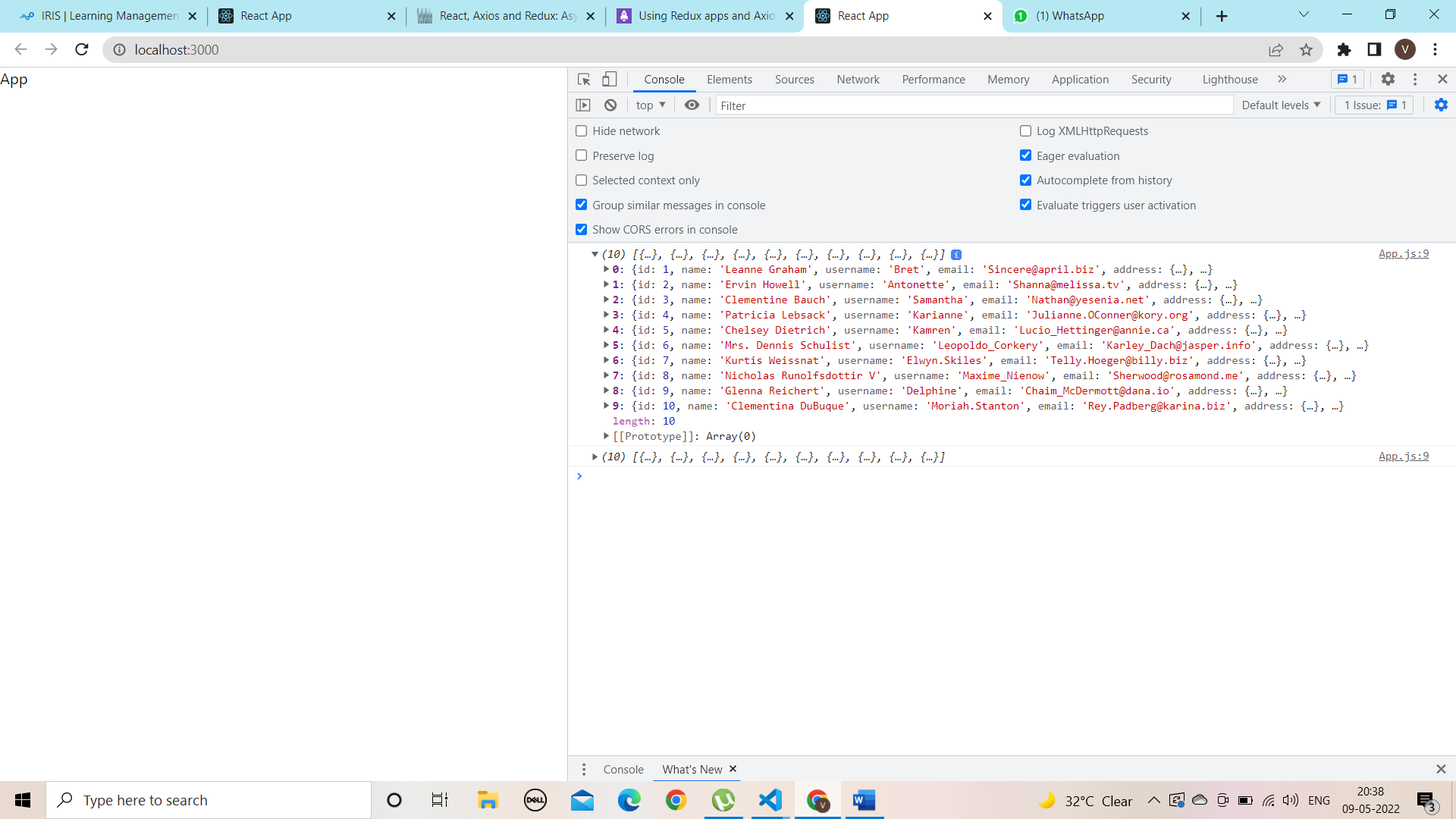
  useEffect(() => {

    getData();

  }, []);

  return <div>App</div>;

}



2. Using Json placeholder add the data to the backend using post Request

Using Json placeholder delete the data from the backend using delete Request

App.js:-

import React from "react";

import { Button, Form } from "react-bootstrap";

import { useForm } from "react-hook-form";

import axios from "axios";

export default function App() {

  const {

    register,

    handleSubmit,

    formState: { errors },

  } = useForm();

  const onSubmit = async (data) => {

    console.log(data);

    const response = await axios.post(

      "http://localhost:5000/api/userController",

      data

    );

    console.log(response.data);

  };

  return (

    <Form className="divClass" onSubmit={handleSubmit(onSubmit)}>

      <Form.Group className="mb-3 pl-5 pr-5" controlId="formBasicEmail">

        <Form.Label>Name</Form.Label>

        <Form.Control

          type="text"

          placeholder="Enter your name"

          {...register("name", {

            required: true,

          })}

        />

        {errors.name && <p>name is required.</p>}

      </Form.Group>

      <Form.Group className="mb-3 pl-5 pr-5" controlId="formBasicEmail">

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

        <Form.Control

          type="email"

          placeholder="Enter email"

          {...register("mail", {

            required: true,

          })}

        />

        <Form.Text className="text-muted">

          We'll never share your email with anyone else.

        </Form.Text>

        {errors.mail && <p>invalid E-mail please provide a valid Email</p>}

      </Form.Group>

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

        Register

      </Button>

    </Form>

  );

}

Backend code:-

Routes:-

const express = require("express");

const router = express.Router();

const userController = require("../../controller/user");

function addRoutes(app) {

  app.all("\*", (req, res, next) => {

    console.log(req.method + " " + req.url);

    next();

  });

  app.get("/test-url", (req, res, next) => {

    res.send({

      success: true,

    });

  });

  app.post("/api/userController", userController.addName);

}

const routes = {

  addRoutes,

};

module.exports = routes;

controllers:-

// const express = require("express");

// var app = express();

const User = require("../models/userModel");

exports.addName = async (req, res) => {

  const errors = [];

  const { name, email } = req.body;

  console.log(req.body);

  if (!name) {

    res.status(422).send("Please enter name");

  } else {

    User.create({

      name: req.body.name,

    }).then((doc) => {

      if (doc) {

        console.log("doc" + doc);

        res.send(doc);

      }

    });

  }

  if (!email) {

    res.status(422).send("Please enter email");

  } else {

    User.create({

      email: req.body.email,

    }).then((doc) => {

      if (doc) {

        console.log("doc" + doc);

        res.send(doc);

      }

    });

  }

};

3. Using Json placeholder edit the data from the backend and display the edit data on the console

// const express = require("express");

// var app = express();

const User = require("../models/userModel");

exports.addName = async (req, res) => {

  const errors = [];

  const { name, email } = req.body;

  console.log(req.body);

  if (!name) {

    res.status(422).send("Please enter name");

  } else {

    User.create({

      name: req.body.name,

    }).then((doc) => {

      if (doc) {

        console.log("doc" + doc);

        res.send(doc);

      }

    });

  }

  if (!email) {

    res.status(422).send("Please enter email");

  } else {

    User.create({

      email: req.body.email,

    }).then((doc) => {

      if (doc) {

        console.log("doc" + doc);

        res.send(doc);

      }

    });

  }

};

exports.userEdit = function (req, res) {

  var condition = { \_id: req.params.\_id };

  User

    .findOneAndUpdate(condition, req.body)

    .then((doc) => {

      if (!doc) {

        res.status(404).send("No active user");

      } else {

        res.status(200).send("Updated Successfully");

      }

    })

    .catch((err) => res.json("Id not found"));

};

Routes:-

router.route("/edit-user/:\_id").put(useeController.editUser);

App.js:-

const editUser = async () => {

    console.log(id);

    const res = await axios.put(

      "http://localhost:5000/api/edit-user/" + id,

      data);

  }

  useEffect(() => {

    editUser();

  }, []);