**Write a program to design a feedback form (name, email, feedback (radio button options), comments, submit button) and save the submitted form data to database collection “feedback”. Now, by submitting a form display success message on next page "/feedback\_data" and also give a link to display all the submitted feedbacks from the database collection on "/display\_feedback" page .**

e1.js

var expr = require("express");

var app = expr();

const mg = require("mongoose");

// Connect to MongoDB

mg.connect("mongodb://127.0.0.1:27017/lju")

.then(() => { console.log("Successful") })

.catch((err) => { console.error(err) });

mg.pluralize(null);

const myschema = new mg.Schema({

uname: { type: String, required: true },

email: { type: String, required: true },

feedback: { type: String, required: true },

comments: { type: String, required: true }

});

const person = mg.model("feedback", myschema);

app.use(expr.static(\_\_dirname, { index: "e1.html" }));

// Insert data into MongoDB

app.get("/feedback\_data", (req, res) => {

const personData = new person({

uname: req.query.uname,

email: req.query.mail,

feedback: req.query.feedback,

comments: req.query.comments,

});

personData.save()

res.send("Record inserted <a href='/display\_feedback'>Display Feedback</a>")

});

// Fetch data from MongoDB and display in HTML table

app.get("/display\_feedback", async(req, res) => {

const f= await person.find();

res.write(`<table border="1px solid"><tr><th>Name</th><th>feddback</th><th>Comments</th></tr>`)

f.map((f1)=>{

res.write(`<tr><td>${f1.uname}</td><td>${f1.feedback}</td><td>${f1.comments}</td></tr>`)

})

res.write()

res.send(`</table>`);

});

app.listen(4001, () => {

console.log("Server is running on port 4001");

});

e1.html

<html>

<head><title>Feedback Form</title></head>

<body>

<form action="/feedback\_data" method="get">

Name: <input type="text" name="uname"/>

Email: <input type="email" name="mail"/>

Feedback: <select name="feedback">

<option value="bad">Bad</option>

<option value="good">Good</option>

<option value="verygood">Very Good</option>

<option value="excellent"SS>Excellent</option>

</select>

Comments: <textarea name="comments" rows="10" cols="15" placeholder="Comments"></textarea>

<input type="submit" value="Submit"/>

</form>

</body>

</html>

**Find a person by their name in a MongoDB database and update their age using the findByIdAndUpdate method, returning the updated document.**

const mg = require("mongoose");

mg.connect("mongodb://127.0.0.1:27017/lju1")

.then(() => { console.log("success"); })

.catch((err) => { console.error(err); });

const personSchema = new mg.Schema({

name: String,

age: Number,

active: Boolean

});

const Person = mg.model("Person", personSchema);

const updatefunction = async () => {

try {

**// Step 1: Find the person by name**

const person = await Person.findOne({ name:"test" });

if (person) {

**const personId = person.\_id;**

**// Step 2: Use findByIdAndUpdate to update the age**

const updatedPerson = await Person.findByIdAndUpdate(

personId,

{ age: 50 },

{ new: true }

);

console.log("Updated Person:", updatedPerson);

} else {

console.log("Person not found");

}

} catch (err) {

console.error("Error updating person's branch:", err);

}};

updatefunction();