**WebTech Lab DA-5**

**Binayak Bishnu**

**20BIT0155**

**Qs1**



<!DOCTYPE html>

<html>

<head>

    <style>

        body {

            background-color: #c4c4c4;

        }

        .card {

            box-shadow: 0 0 20px 0 rgba(0, 0, 0, 0.2);

            max-width: 20%;

            margin: auto;

            text-align: center;

            padding: 10% 0%;

            background-image: linear-gradient(to bottom, blue, rgba(0, 0, 0, 0));

        }

    </style>

</head>

<body>

    <h1 style="text-align:center">My Profile</h1>

    <div class="card">

        <h1>Binayak Bishnu</h1>

        <p>2nd Year BTech</p>

        <p>VIT, Vellore</p>

        <a>

            <input type="button" value="Contact" style="width:80%; height:30px; background-color:blue;" />

        </a>

    </div>

</body>

</html>

var http = require('http');

var fs = require('fs');

http.createServer(function (req, res) {

    fs.readFile('Qs-1.html', function (err, data) {

        res.writeHead(200, { 'Content-Type': 'text/html' });

        res.write(data);

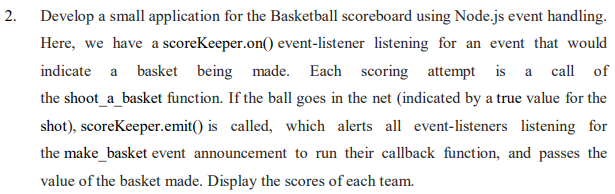
        res.end();

    });

}).listen(8080);

**Output**

**Qs2**



var events = require('events');

var scoreKeeper = new events.EventEmitter();

VIT\_V = [false, 0];

VIT\_C = [false, 0];

var shoot\_a\_basket = function () {

    if (VIT\_V[0] == true) {

        console.log('VIT-V has scored');

        VIT\_V[0] = false

        VIT\_V[1] += 1

    }

    if (VIT\_C[0] == true) {

        console.log('VIT-C has scored');

        VIT\_C[0] = false

        VIT\_C[1] += 1

    }

    console.log('VIT-V :' + VIT\_V[1]);

    console.log('VIT-C :' + VIT\_C[1]);

}

scoreKeeper.on('basket', shoot\_a\_basket);

scoreKeeper.emit('basket', VIT\_V[0] = true);

scoreKeeper.emit('basket', VIT\_C[0] = true);

scoreKeeper.emit('basket', VIT\_V[0] = false);

scoreKeeper.emit('basket', VIT\_V[0] = true);

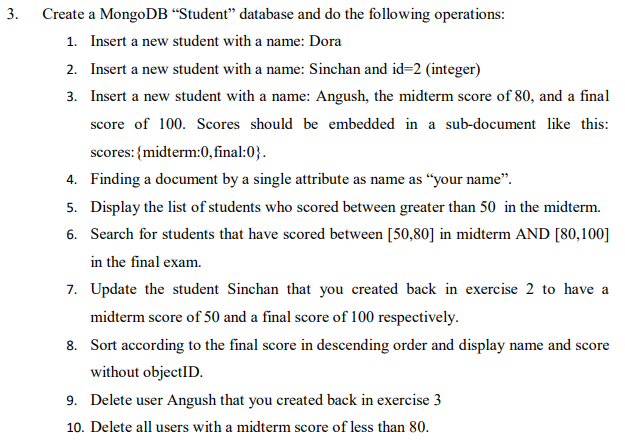
scoreKeeper.emit('basket', VIT\_C[0] = true);

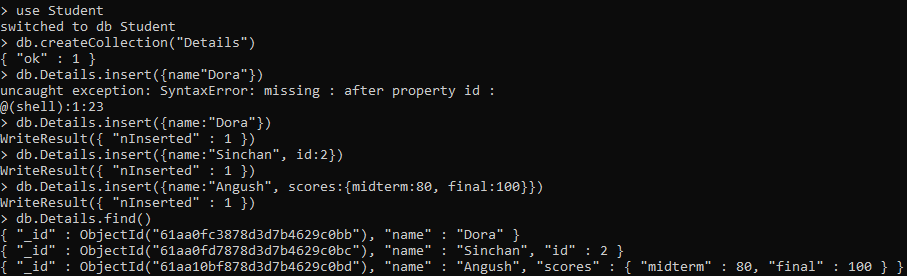
scoreKeeper.emit('basket', VIT\_V[0] = true);

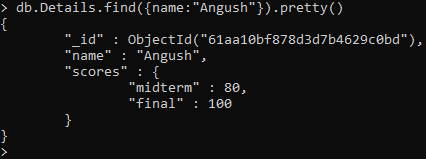
**Output**

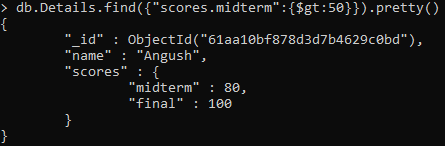


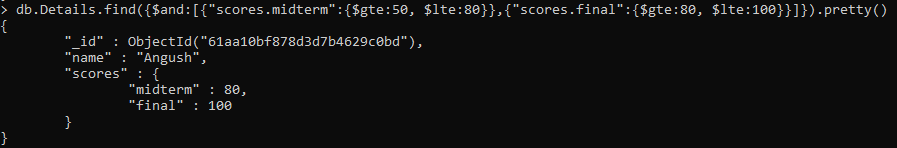
**Qs3**

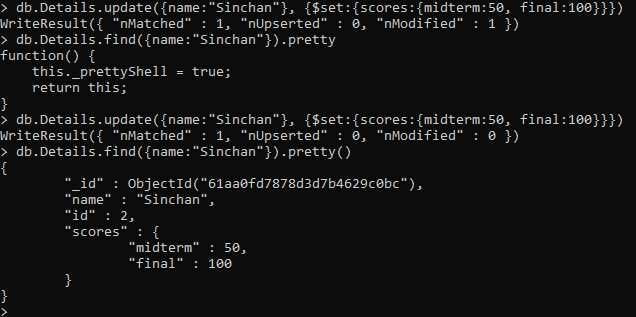


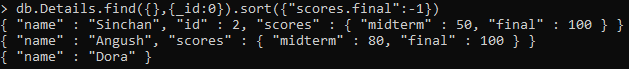


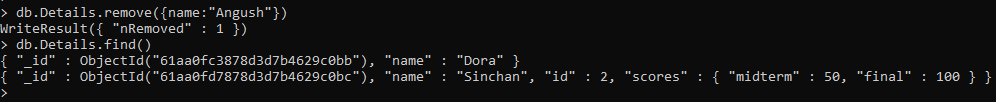
D:\VIT\Semester_III\Web_Tech_A2;L17,18\WebTech_L17_18\Assessment-5\Qs-3_part4a.png 

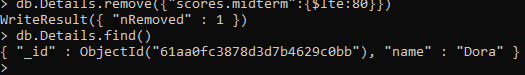




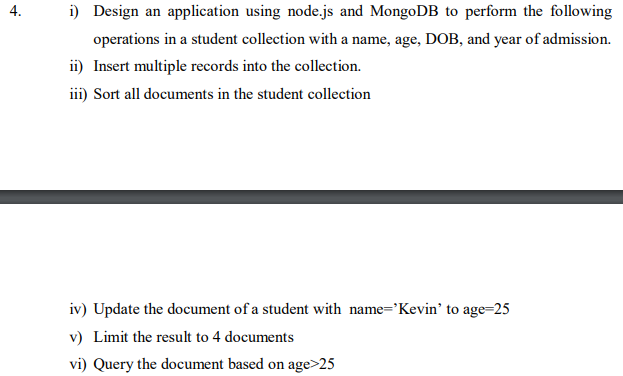








**Qs4**



const mongoose = require("mongoose")

mongoose.connect("mongodb://localhost:27017/Qs\_4corrected", {

    useNewUrlParser: true

});

const detailsSchema = new mongoose.Schema({

    name: String,

    age: Number,

    dob: Date,

    yearofadm: Number

});

const Student = mongoose.model("Student", detailsSchema)

const Binayak = new Student({

    name: "Binayak",

    age: 19,

    dob: "2002-10-11",

    yearofadm: 2020,

});

const Bishnu = new Student({

    name: "Bishnu",

    age: 18,

    dob: "2001-09-11",

    yearofadm: 2020,

});

const Neo = new Student({

    name: "Neo",

    age: 20,

    dob: "2000-10-11",

    yearofadm: 2021,

});

const Neil = new Student({

    name: "Neil",

    age: 30,

    dob: "1992-10-15",

    yearofadm: 2009,

});

const ABC = new Student({

    name: "ABC",

    age: 26,

    dob: "1995-09-11",

    yearofadm: 2012,

});

Student.insertMany([Binayak, Bishnu, Neo, Neil, ABC], function (err) {

    if (err) {

        console.log(err);

    } else {

        console.log("All records added");

    }

});

Student.find(function (err, student) {

    if (err) {

        console.log(err)

    } else {

        student.forEach((student) => {

            console.log(student);

        });

    }

})

var sortage = { age: 1 };

Student.find({}, function (err, result) {

    if (err) {

        console.log("error query")

    } else {

        console.log(result)

    }

}).sort(sortage)

Student.updateOne({ name: "Neil" }, { age: 25 }, function (err) {

    if (err) {

        console.log(err)

    } else {

        console.log('Updated');

    }

})

Student.find(function (err, student) {

    if (err) {

        console.log(err)

    } else {

        student.forEach((student) => {

            console.log(student);

        });

    }

}).limit(4)

var sortage = { age: 1 };

Student.find({age:{$gte:25}}, function (err, result) {

    if (err) {

        console.log("error query")

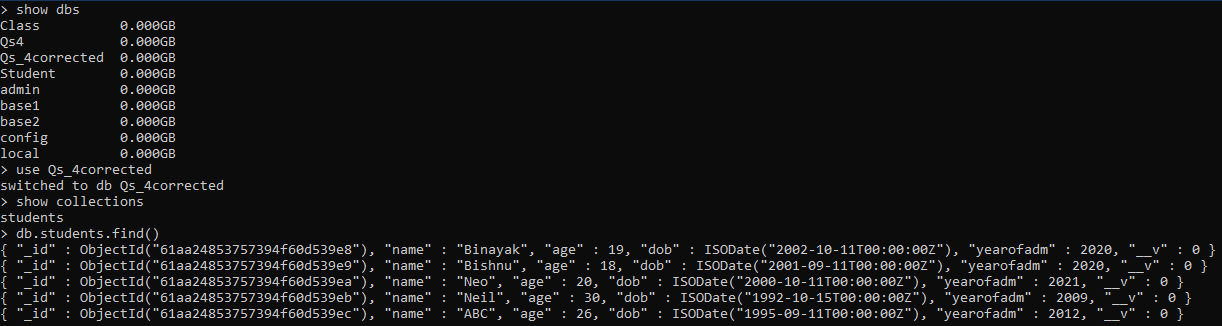
    } else {

        console.log(result)

    }

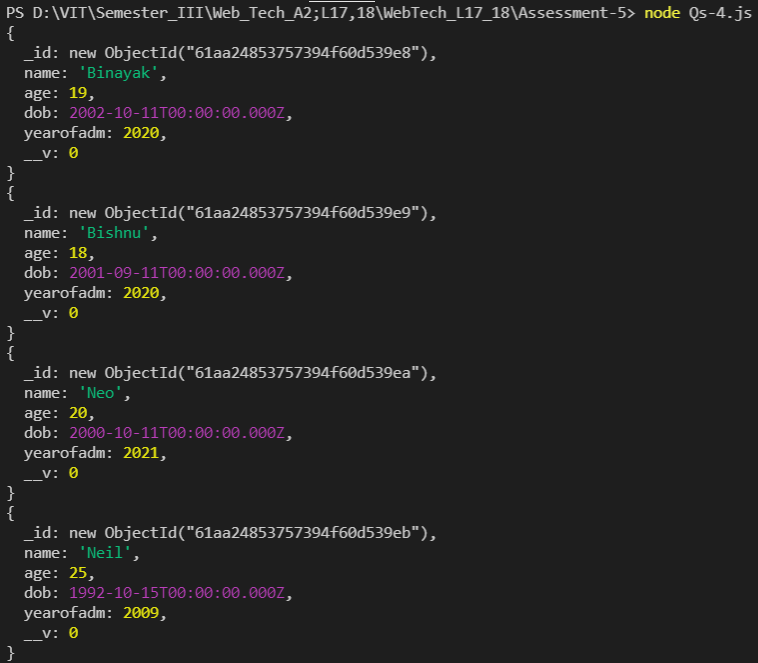
}).sort(sortage)

**Output**



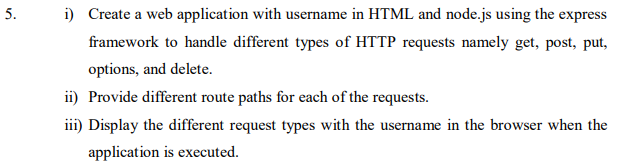


D:\VIT\Semester_III\Web_Tech_A2;L17,18\WebTech_L17_18\Assessment-5\Qs-4_part4.png





**Qs5**



<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

    <title>Exercise 5</title>

</head>

<body>

    <form action="/submitform" method="POST">

        First Name: <input name="firstName" type="text" /><br />

        Last Name: <input name="lastName" type="text" /><br />

        <input type="submit" />

    </form>

    <form action="/deletedata" method="POST">

        First Name: <input name="firstName" type="text" /><br />

        Last Name: <input name="lastName" type="text" /><br />

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

    </form>

</body>

</html>

var express = require('express');

const path = require('path');

var app = express();

var bodyParser = require('body-parser');

app.use(bodyParser.urlencoded({ extended: false }));

app.get('/', function (req, res) {

    res.sendFile(path.join(\_\_dirname, 'Qs-5.html'));

});

app.post('/submitform', function (req, res) {

    var name = req.body.firstName + ' ' + req.body.lastName;

    res.send(name + ' POSTed!!!');

});

app.put('/updatedata', function (req, res) {

    res.send('PUT Request');

});

app.delete('/deletedata', function (req, res) {

    res.send('DELETE Request');

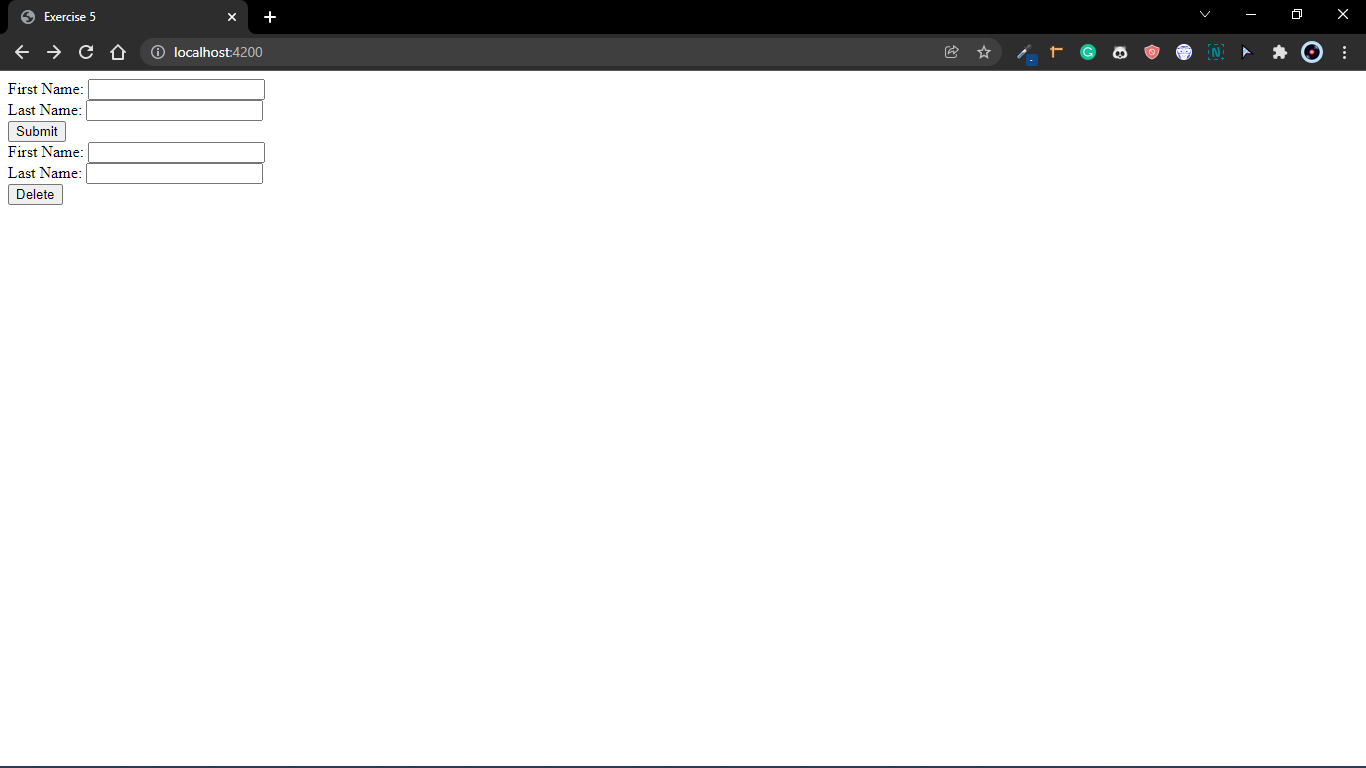
});

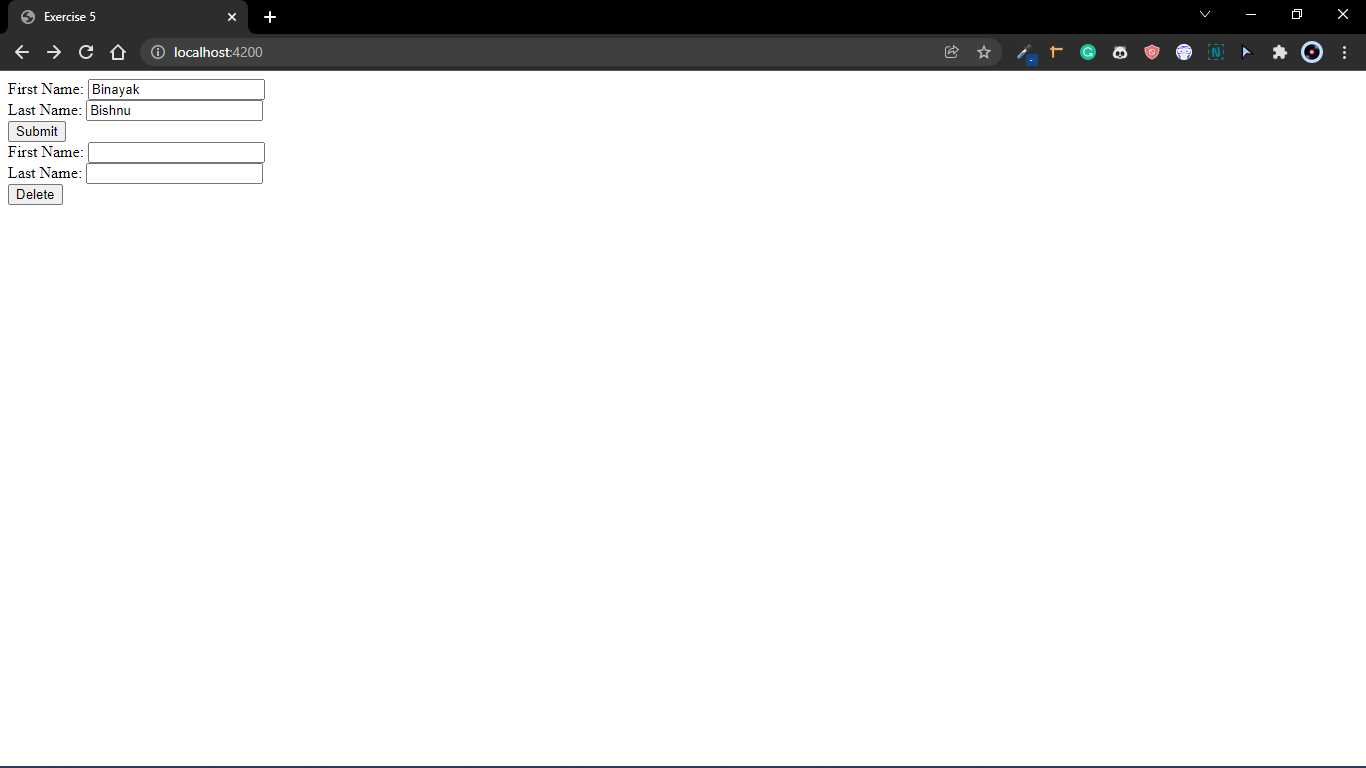
app.listen(4200, () => {

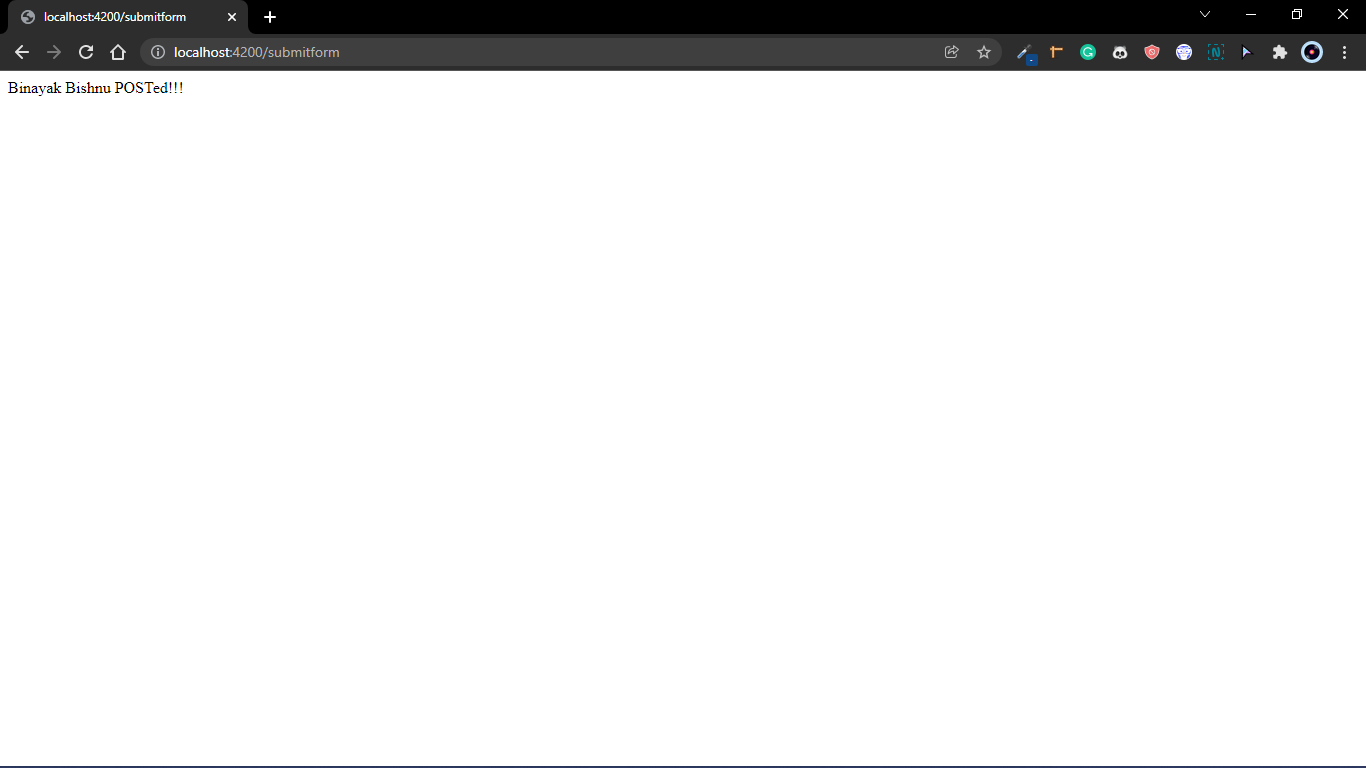
    console.log('Running...');

});

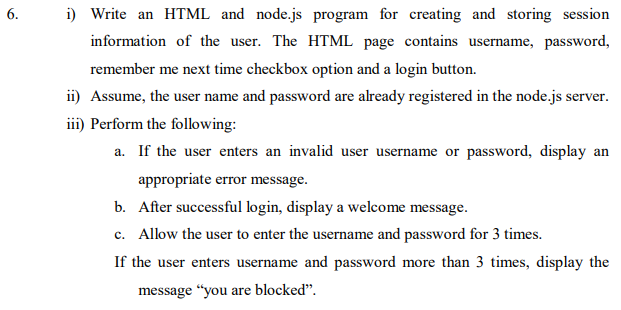
**Output**







**Qs6**



<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Exercise 6</title>

</head>

<body>

    <form id="form1" name="myForm1" method="post" action="/signup">

        <h2>Sign Up</h2>

        <div>

            <label for="username">Username</label>

            <input type="text" name="username" name="username" />

        </div>

        <div>

            <label for="password">Password</label>

            <input type="password" name="password" name="password" />

        </div>

        <input type="checkbox" name="remember" />

        <input type="submit" name="submitbtn" value="Login" />

    </form>

    <form id="form2" name="myForm2" method="post" action="/login">

        <h2>Login</h2>

        <div>

            <label for="username">Username</label>

            <input type="text" name="username" name="username" />

        </div>

        <div>

            <label for="password">Password</label>

            <input type="password" name="password" name="password" />

        </div>

        <input type="checkbox" name="remember" />

        <input type="submit" name="submitbtn" value="Login" />

    </form>

</body>

</html>

var express = require('express');

const path = require('path');

var app = express();

var bodyParser = require('body-parser');

var u = bodyParser.urlencoded({

    extended: false

})

app.use(express.static('public'));

app.get('/', function (req, res) {

    res.sendFile(path.join(\_\_dirname, 'Qs-6.html'));

})

app.post('/signup', u, function (req, res) {

    var response = {

        username: req.body.username,

        password: req.body.password,

    };

    var usernameobj = {

        username: req.body.username,

    }

    var MongoClient = require('mongodb').MongoClient, format = require('util').format;

    MongoClient.connect('mongodb://127.0.0.1:27017/', function (err, db) {

        if (err) {

            throw err;

        }

        else {

            console.log("Connected");

        }

        var myDB = db.db("Qs-6");

        myDB.collection('users').findOne(usernameobj, function (err, result) {

            console.log(result);

            if (err) {

                throw err;

            }

            else if (result) {

                console.log(usernameobj.username + " already exists");

                res.sendFile(path.join(\_\_dirname, 'Qs-6.html'));

            }

            else if (!result) {

                myDB.collection('users').insertOne(response, function (err, result) {

                    if (err) {

                        throw err;

                    }

                    else {

                        console.log("New account created");

                        console.log(response);

                        res.sendFile(path.join(\_\_dirname, 'Qs-6b.html'));

                    }

                });

            }

        });

    });

})

app.post('/login', u, function (req, res) {

    var response = {

        username: req.body.username,

        password: req.body.password

    };

    var usernameobj = {

        username: req.body.username

    }

    var passwordobj = {

        password: req.body.password

    }

    var MongoClient = require('mongodb').MongoClient, format = require('util').format;

    MongoClient.connect('mongodb://127.0.0.1:27017/Qs-6', function (err, db) {

        if (err) {

            throw err;

        }

        console.log("Connected");

        var myDB = db.db("Qs-6");

        myDB.collection('users').findOne(usernameobj, function (err, result) {

            console.log(usernameobj);

            console.log(result);

            if (err) throw err;

            else if (!result) {

                console.log("No user found");

                res.sendFile(path.join(\_\_dirname, 'Qs-6.html'));

            }

            else if (result) {

                myDB.collection('users').findOne(passwordobj, function (err, result2) {

                    if (err) {

                        throw err;

                    }

                    else if (!result2) {

                        console.log("Password is incorrect for " + response.username);

                        res.sendFile(path.join(\_\_dirname, 'Qs-6.html'));

                    }

                    else if (result2) {

                        console.log(response.username + " logged in");

                        res.sendFile(path.join(\_\_dirname, 'Qs-6b.html'));

                    }

                });

            }

        });

    });

    console.log(response);

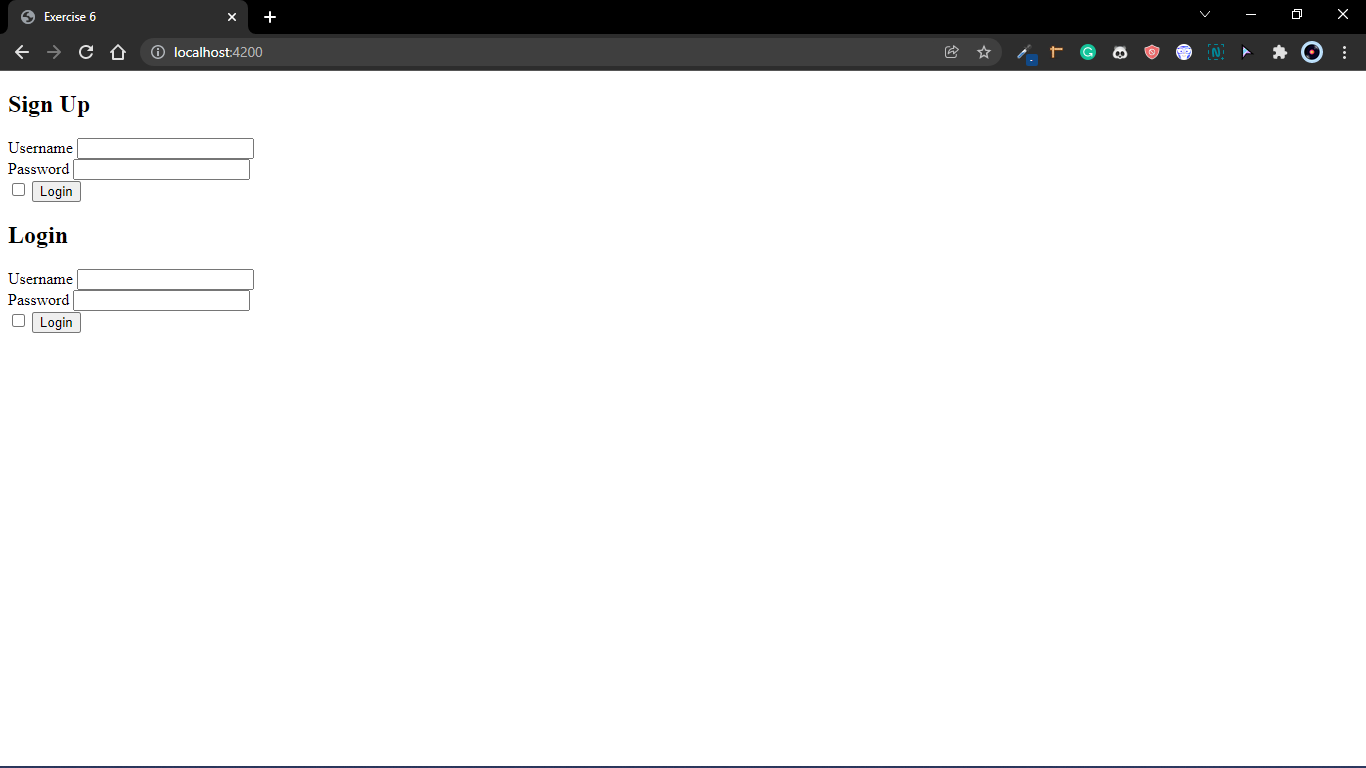
})

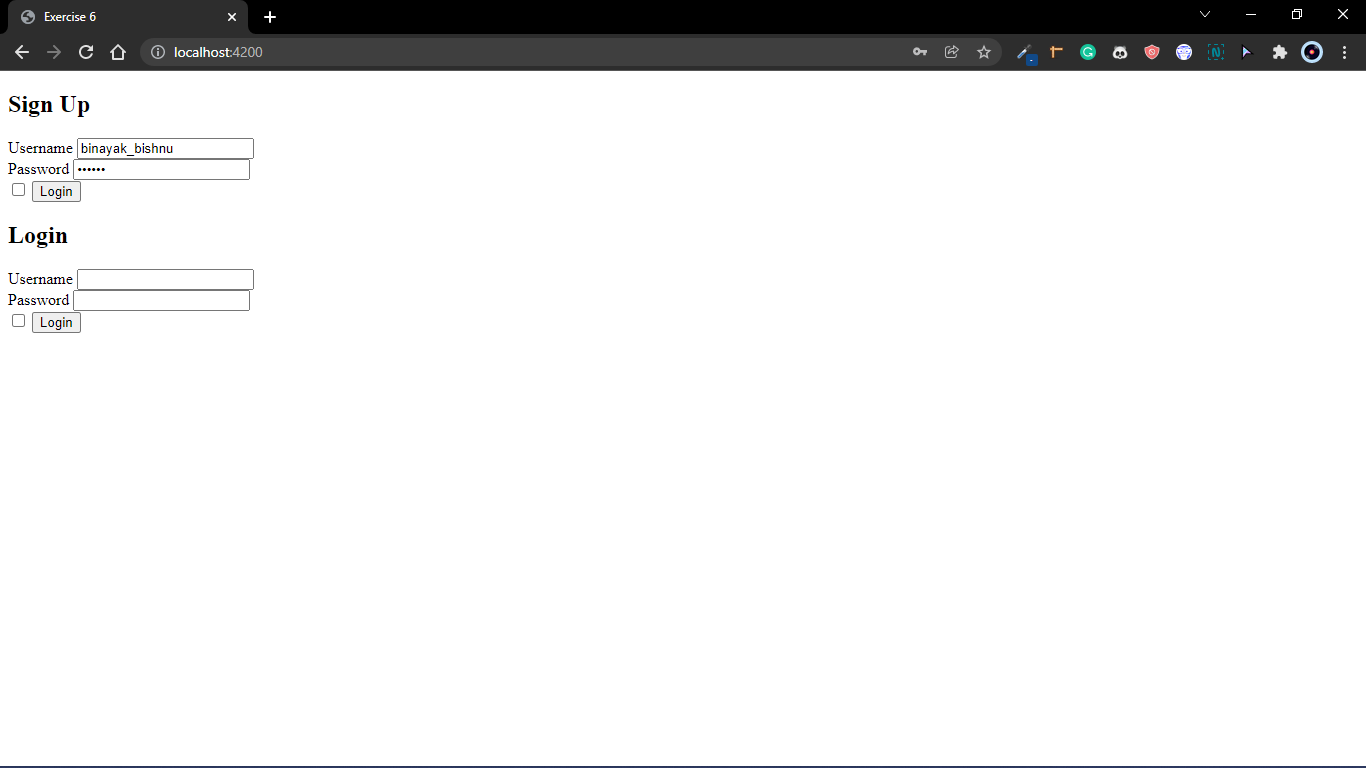
app.listen(4200, () => {

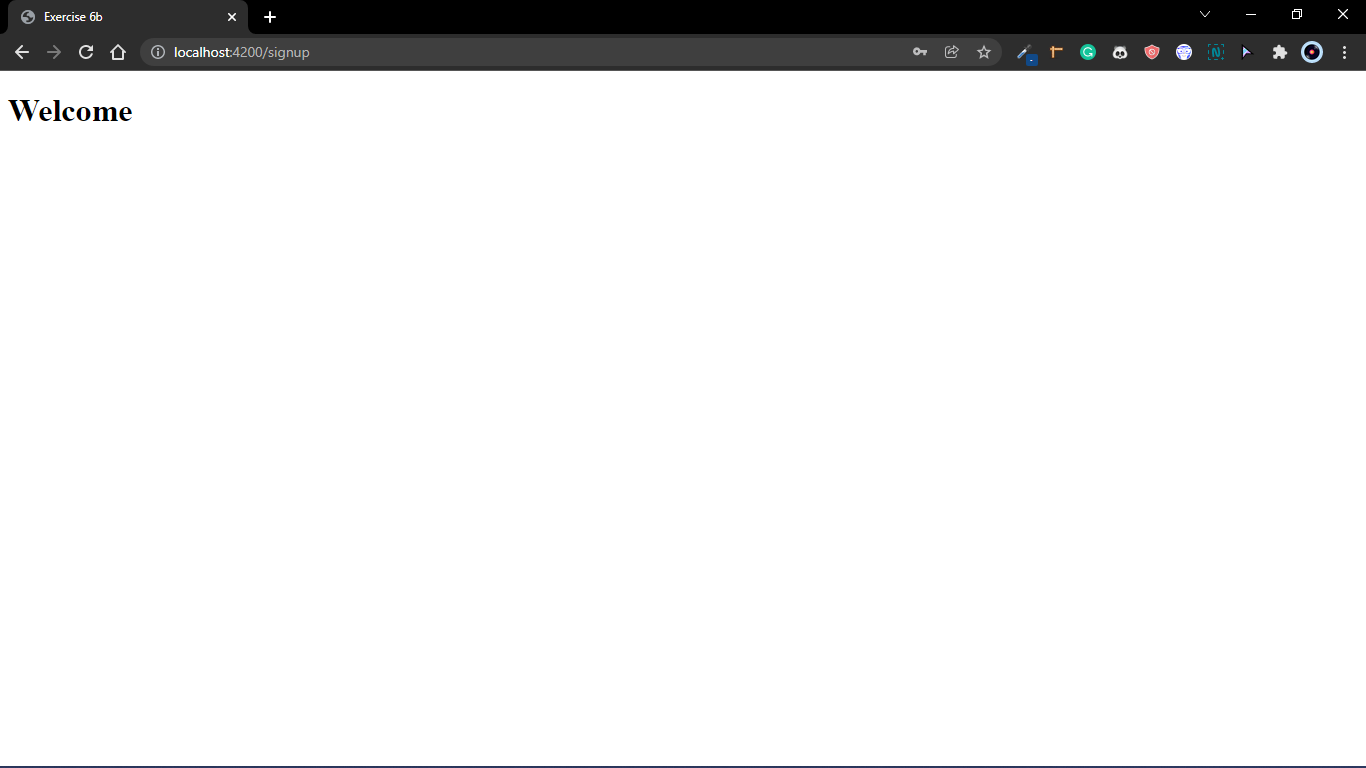
    console.log('Running...');

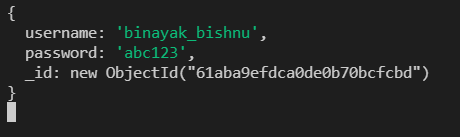
});

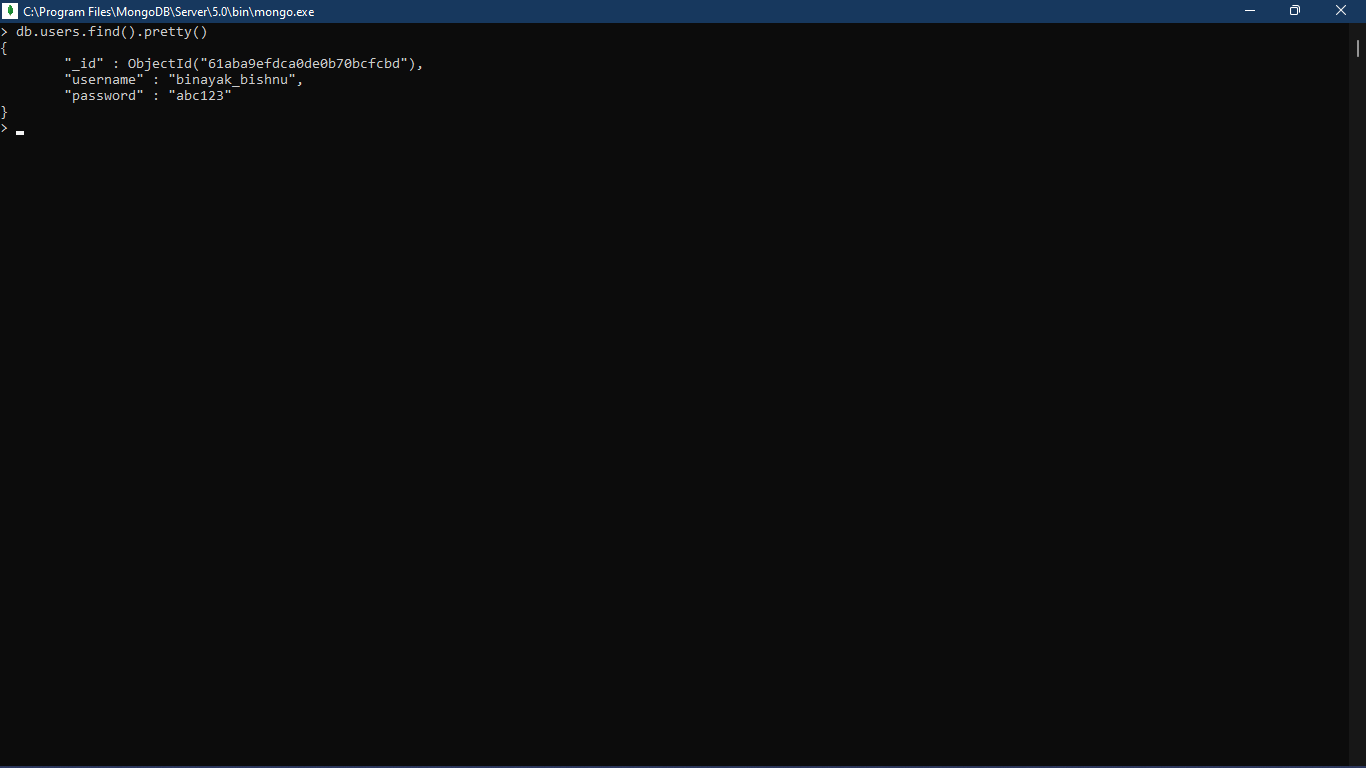
**Output**

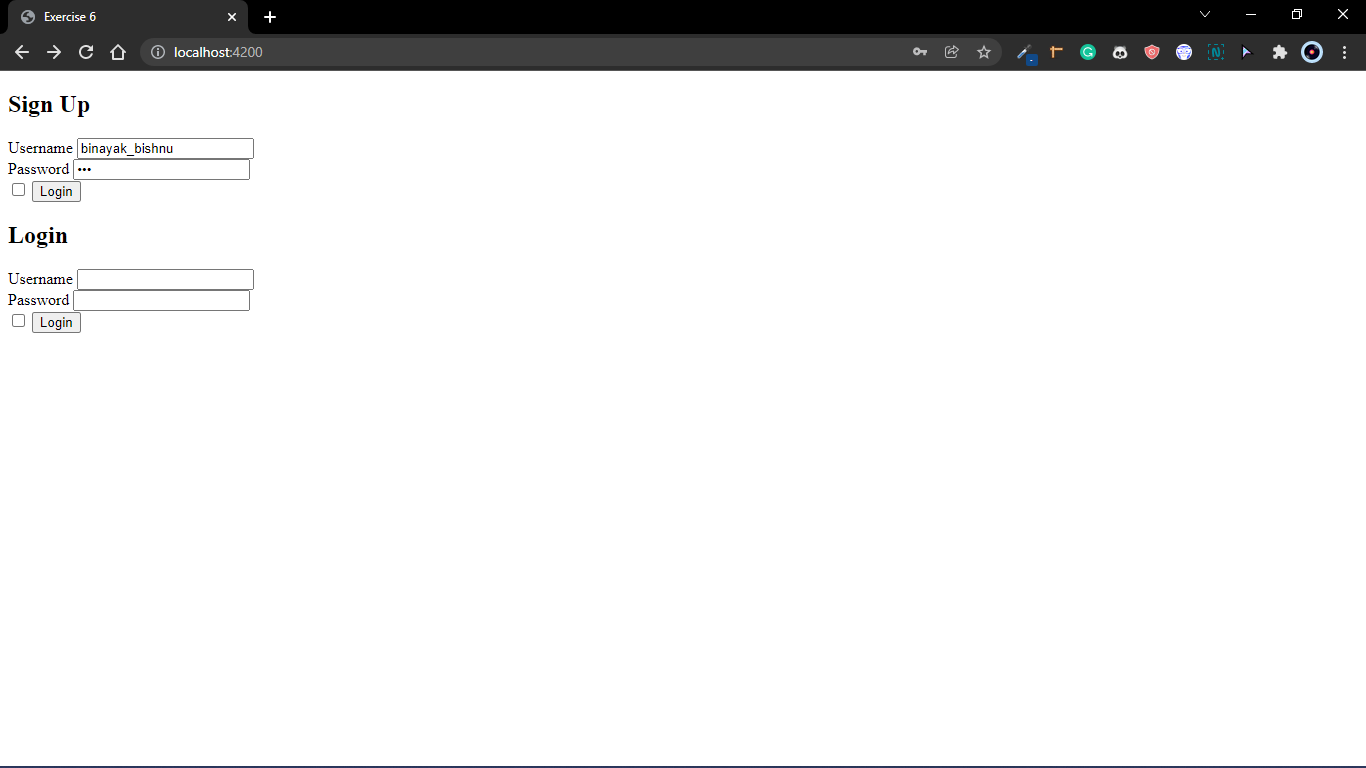


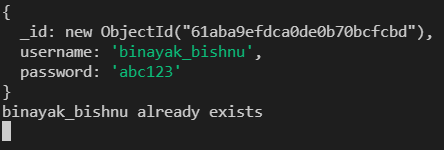


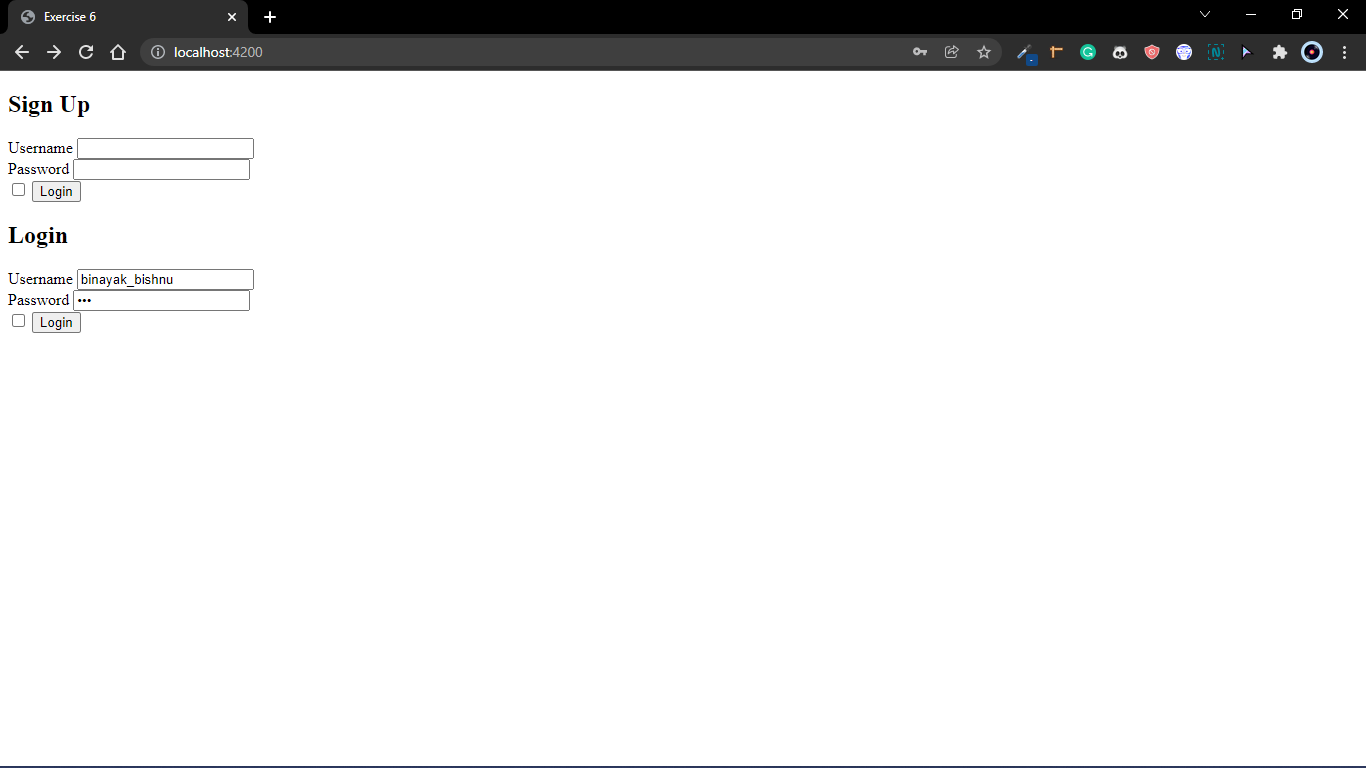


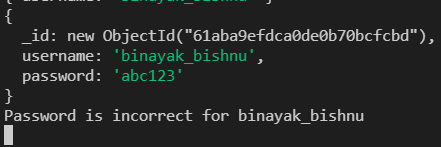
****



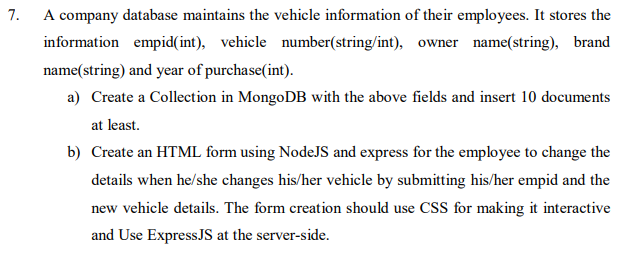


****



****

**Qs7**



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>UpdateInfo Form</title>

<style>

body {

padding: 200px;

background: linear-gradient(to bottom, #33ccff 0%, #ff99cc 100%);

}

div {

background-color: rgba(255, 155, 25, 0.5);

text-align: center;

border-radius: 30px;

padding: 10px;

box-shadow: 5px 10px #888888;

}

table {

width: 300px;

}

table td {

border-radius: 20px;

}

button {

background-color: white;

color: black;

border: 2px solid #e7e7e7;

padding: 5px 15px;

border-radius: 20px;

}

button:hover {

background-color: #e7e7e7;

}

</style>

</head>

<body>

<h1>Update your vehicle Info</h1>

<form action="/" method="post">

<table>

<tr>

<td>Employee ID:</td>

<td><input type="number" name="eid"><br></td>

</tr>

<tr>

<td>Vehicle Number:</td>

<td><input type="number" name="vnum"><br></td>

</tr>

<tr>

<td>Brand:</td>

<td><input type="text" name="brnd"><br></td>

</tr>

<tr>

<td>Year of Purchasing:</td>

<td><input type="number" name="yop"><br></td>

</tr>

<tr>

<td rowspan="2"><button type="submit">Update</button></td>

</tr>

</table>

</form>

</body>

</html>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>20BIT0195</title>

<style>

body {

padding: 200px;

background: linear-gradient(to bottom, #33ccff 0%, #ff99cc 100%);

}

div {

background-color: rgba(255, 155, 25, 0.5);

text-align: center;

border-radius: 30px;

padding: 10px;

box-shadow: 5px 10px #888888;

}

table {

width: 300px;

}

table td {

border: 2px Solid black;

border-radius: 20px;

}

</style>

</head>

<body>

<h1>Updated vehicle Info</h1>

<table>

<tr>

<td>Employee ID:</td>

<td>

<%= empid %>

</td>

</tr>

<tr>

<td>Vehicle Number:</td>

<td>

<%= vehnum %>

</td>

</tr>

<tr>

<td>Brand:</td>

<td>

<%= band %>

</td>

</tr>

<tr>

<td>Year of Purchasing:</td>

<td>

<%= poy %>

</td>

</tr>

</table>

</body>

</html>

const express = require("express");

const bodyParser = require("body-Parser");

const app = express();

app.set("view engine", "ejs");

app.use(bodyParser.urlencoded({ extended: true }));

app.use(express.static("public"));

var MongoClient = require("mongodb").MongoClient;

var url = "mongodb://localhost:27017/mydb";

app.listen(3000, function () {

console.log("Server is running on port 3000");

});

app.get("/", function (req, res) {

res.sendFile(\_\_dirname + "/form7.html");

});

var myobj = [

{

\_id: 1,

veh\_num: 1324,

name: "Parikshit",

brand: "Nexa",

YOP: 2020,

},

{

\_id: 2,

veh\_num: 2124,

name: "Meghna",

brand: "Hyundai",

YOP: 2018,

},

{

\_id: 3,

veh\_num: 5814,

name: "Umang",

brand: "Ford",

YOP: 2017,

},

{

\_id: 4,

veh\_num: 6474,

name: "Mithila",

brand: "Tata",

YOP: 2021,

},

{

\_id: 5,

veh\_num: 7451,

name: "Dhruv",

brand: "Nexa",

YOP: 2020,

},

{

\_id: 6,

veh\_num: 3521,

name: "Phil",

brand: "Toyota",

YOP: 2015,

},

{

\_id: 7,

veh\_num: 4175,

name: "Claire",

brand: "Honda",

YOP: 2014,

},

{

\_id: 8,

veh\_num: 6684,

name: "Luke",

brand: "Volkswagen",

YOP: 2012,

},

{

\_id: 9,

veh\_num: 9145,

name: "Manny",

brand: "Kia",

YOP: 2019,

},

{

\_id: 10,

veh\_num: 8419,

name: "Gloria",

brand: "Ferrari",

YOP: 2021,

},

];

var n, v, b, y;

app.post("/", function (req, res) {

var n = Number(req.body.eid);

var v = req.body.vnum;

var b = req.body.brnd;

var y = req.body.yop;

res.render("updated", { empid: n, vehnum: v, band: b, poy: y });

MongoClient.connect(url, function (err, db) {

if (err) throw err;

dbo = db.db("company");

var myquery = { \_id: n };

console.log(n);

var newvalues = { $set: { veh\_num: v, brand: b, YOP: y } };

dbo

.collection("vehicle")

.updateMany(myquery, newvalues, function (err, res) {

if (err) throw err;

console.log(res);

console.log("Document Updated");

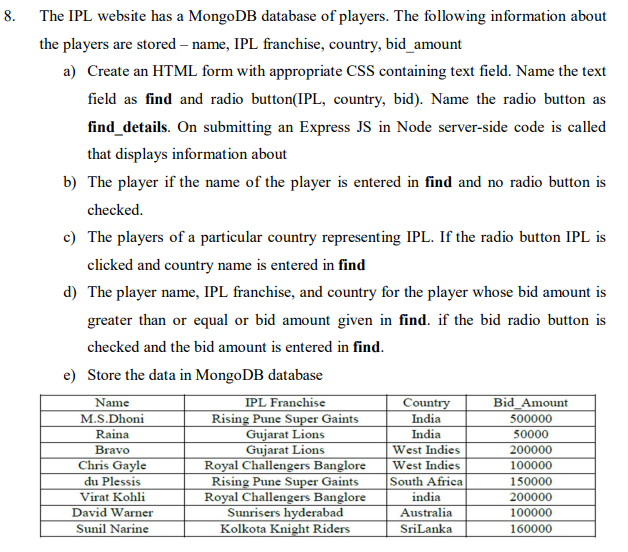
db.close();

});

});

});

**Qs8**



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Search Player Info Form</title>

<style>

body {

padding: 200px;

background: linear-gradient(to bottom, #33ccff 0%, #ff99cc 100%);

}

div {

background-color: rgba(255, 255, 255, 0.5);

text-align: left;

border-radius: 30px;

padding: 10px;

box-shadow: 5px 10px #888888;

width: 300px;

}

button {

background-color: white;

color: black;

border: 2px solid #e7e7e7;

padding: 5px 15px;

border-radius: 20px;

}

button:hover {

background-color: #e7e7e7;

}

</style>

</head>

<body>

<h1>Search Players Info</h1>

<form action="/" method="post">

<table>

<tr>

<td>

<input name="finddetails" type="radio" value="1">

<label for="rname"><input type="text" name="name" placeholder="Player Name"></label>

</td>

</tr>

<tr>

<td>

<input name="finddetails" type="radio" value="2">

<label for="rfranchise">

<select name="Franchise">

<option name="none" value="">Select Franchise</option>

<option name="RPS" value="Rising Pune Super Giants">Rising Pune Super Giants</option>

<option name="GL" value="Gujrat Lions">Gujrat Lions</option>

<option name="RCB" value="Royal Challengers Banglore">Royal Challengers Banglore</option>

<option name="SRH" value="Sunrisers Hyderabad">Sunrisers Hyderabad</option>

<option name="KKR" value="Kolkata Night Riders">Kolkata Night Riders</option>

<option name="RR" value="Rajasthan Royals">Rajasthan Royals</option>

<option name="PKBS" value="Punjab Kings">Punjab Kings</option>

<option name="DC" value="Delhi Capitals">Delhi Capitals</option>

</select>

</label>

</td>

</tr>

<tr>

<td>

<input name="finddetails" type="radio" value="3">

<label for="rcountry">

<select name="country">

<option value="nope">Select Country</option>

<option value="India">India</option>

<option value="West Indies">West Indies</option>

<option value="South Africa">South Africa</option>

<option value="Australia">Australia</option>

<option value="Sri Lanka">Sri Lanka</option>

</select>

</label>

</td>

</tr>

<tr>

<td>

<input name="finddetails" type="radio" value="4">

<label for="rbid">

<input type="number" name="bida" placeholder="Bid\_Amount">

</label>

</td>

</tr>

<tr>

<td rowspan="2"><button type="submit">Search</button></td>

</tr>

</table>

</form>

</body>

</html>

const express = require("express");

const bodyParser = require("body-Parser");

const app = express();

app.set("view engine", "ejs");

app.use(bodyParser.urlencoded({ extended: true }));

app.use(express.static("public"));

app.listen(3000, function () {

console.log("Server is running on port 3000");

});

var MongoClient = require("mongodb").MongoClient;

var url = "mongodb://localhost:27017/";

app.get("/", function (req, res) {

res.sendFile(\_\_dirname + "/form8.html");

});

var myobj = [

{

\_id: 1,

name: "MS Dhoni",

IPL\_Franchise: "Rising Pune Super Giants",

Country: "India",

Bid\_Amount: 500000,

},

{

\_id: 2,

name: "Raina",

IPL\_Franchise: "Gujrat Lions",

Country: "India",

Bid\_Amount: 50000,

},

{

\_id: 3,

name: "Bravo",

IPL\_Franchise: "Gujrat Lions",

Country: "West Indies",

Bid\_Amount: 200000,

},

{

\_id: 4,

name: "Chris Gayle",

IPL\_Franchise: "Royal Challengers Banglore",

Country: "West Indies",

Bid\_Amount: 100000,

},

{

\_id: 5,

name: "Du Plessis",

IPL\_Franchise: "Rising Pune Super Giants",

Country: "South Africa",

Bid\_Amount: 150000,

},

{

\_id: 6,

name: "Virat Kohli",

IPL\_Franchise: "Royal Challengers Banglore",

Country: "India",

Bid\_Amount: 200000,

},

{

\_id: 7,

name: "David Warner",

IPL\_Franchise: "Sunrisers Hyderabad",

Country: "Australia",

Bid\_Amount: 100000,

},

{

\_id: 8,

name: "Sunil Narine",

IPL\_Franchise: "Kolkata Night Riders",

Country: "SriLanka",

Bid\_Amount: 160000,

},

];

app.post("/", function (req, res) {

var n = req.body.name;

var f = req.body.Franchise;

var c = req.body.country;

var b = Number(req.body.bida);

var r = Number(req.body.finddetails);

MongoClient.connect(url, function (err, db) {

if (err) throw err;

dbo = db.db("IPL");

var myquery;

function mymongo(myquery) {

dbo.collection("iplinfo").find(myquery).toArray(function (err, res) {

if (err) throw err;

console.log(res);

db.close();

});

}

switch (r) {

case 1: //Find by name

myquery = { name: n };

mymongo(myquery);

break;

case 2: //Find by franchise

myquery = { IPL\_Franchise: f };

mymongo(myquery);

break;

case 3: //Find by Country

myquery = { Country: c };

mymongo(myquery);

break;

case 4: //Find by Bid\_Amount

myquery = { Bid\_Amount: { $gte: b } };

mymongo(myquery);

break;

}

});

});