Folder EAD9

6 printable files

(file list disabled)

EAD9\.gitignore

```
1  node_modules
2
```

EAD9\readme.md

TERMINAL1

```
start mongod --replSet m101 -logpath \data\rs1\1.log --dbpath \data\rs1 --port 27020
```

TERMINAL2

```
start mongod --replSet m101 -logpath \data\rs2\2.log --dbpath \data\rs2 --port 27021
```

TERMINAL3

```
start mongod --replSet m101 -logpath \data\rs3\3.log --dbpath \data\rs3 --port 27022
```

TERMINAL1

```
mongosh --port 27020

config = { _id: "m101", members:[
        { _id : 0, host : "localhost:27020"},
        { _id : 1, host : "localhost:27021"},
        { _id : 2, host : "localhost:27022"} ]
        };

rs.initiate(config);
rs.status();

use cbit
db.it3.find();
```

TERMINAL2

```
mongosh --port 27021
```

```
rs.secondaryOk()
db.products.find();
```

TERMINAL3

```
mongosh --port 27022
rs.secondaryOk()
db.products.find();
```

TERMINAL1

```
use admin;
db.shutdownServer();
```

open a TERMINAL2 & TERMINAL3

```
rs.status();
```

EAD9\server\app.js

```
1  const express = require('express')
2  const router = require('./routes/students')
3  const mongoose = require('mongoose')
4  const app = express()
5  const url = "mongodb://127.0.0.1:27020,127.0.0.1:27021,127.0.0.1:27022/cbitit1?
   replicaSet=m101";
6
7  mongoose.connect(url)
8  const conn = mongoose.connection
9  conn.on('open', () => {
10     console.log("connection started")
11 })
12
13 app.use(express.json());
14 app.use('/students', router)
15
16 app.listen(3000, () => {
17     console.log('server is running on port 3000')
18 })
```

EAD9\server\models\student.js

```
1  const mongoose = require('mongoose')
2  const studentSchema = new mongoose.Schema({
3      name: {
4          type: String,
5          required: true
6      },
```

```javascript
 7        rollNo: {
 8            type: Number,
 9            required: true
10        },
11        isPassed: {
12            type: Boolean,
13            default: false
14
15        }
16 })
17
18 module.exports = mongoose.model("Student", studentSchema)
```

EAD9\server\package.json

```json
 1 {
 2    "name": "server",
 3    "version": "1.0.0",
 4    "description": "",
 5    "main": "app.js",
 6    "scripts": {
 7      "start": "node app.js",
 8      "dev": "nodemon app.js",
 9      "test": "echo \"Error: no test specified\" && exit 1"
10    },
11    "author": "Anvesh Nalimela <nalimelaanvesh@gmail.com>",
12    "license": "ISC",
13    "dependencies": {
14      "express": "^4.19.2",
15      "mongodb": "^6.8.0",
16      "mongoose": "^8.5.2"
17    },
18    "devDependencies": {
19      "nodemon": "^3.1.4"
20    }
21 }
22
```

EAD9\server\routes\students.js

```javascript
 1 const express = require('express')
 2 const router = express.Router()
 3
 4 const Student = require('../models/student')
 5
 6
 7 router.get('/', async (req, res) => {
 8     try {
 9         const result = await Student.find({})
10         res.json(result)
11     } catch {
12         console.log("error while fetching the data through get request")
13
14     }
15 })
```

```
16
17  //Route the post the route
18  router.post('/', async (req, res) => {
19      const student = new Student({
20          name: req.body.name,
21          rollNo: req.body.rollNo,
22          isPassed: req.body.isPassed
23      })
24      try {
25          console.log(student)
26
27          const s = await student.save()
28          res.json(s)
29      } catch {
30          console.log("error while fetching the data through post request")
31      }
32  })
33
34  //Route to find student with speciic id
35  router.get('/:id', async (req, res) => {
36      const student = await Student.findById(req.params.id)
37      res.json(student)
38  })
39
40  //Route to update the data
41  router.delete('/:id', async (req, res) => {
42      const student = await Student.findByIdAndDelete(req.params.id)
43      res.json(student)
44  })
45
46  router.patch('/:id', async (req, res) => {
47      try {
48          const student = await Student.findById(req.params.id)
49          student.isPassed = true
50          const s1 = await student.save()
51          res.json(sq)
52      }
53      catch (err) {
54          res.send('Error')
55      }
56
57  })
58
59  module.exports = router
60
61
```