Name : Mohammad Adnan Shariff
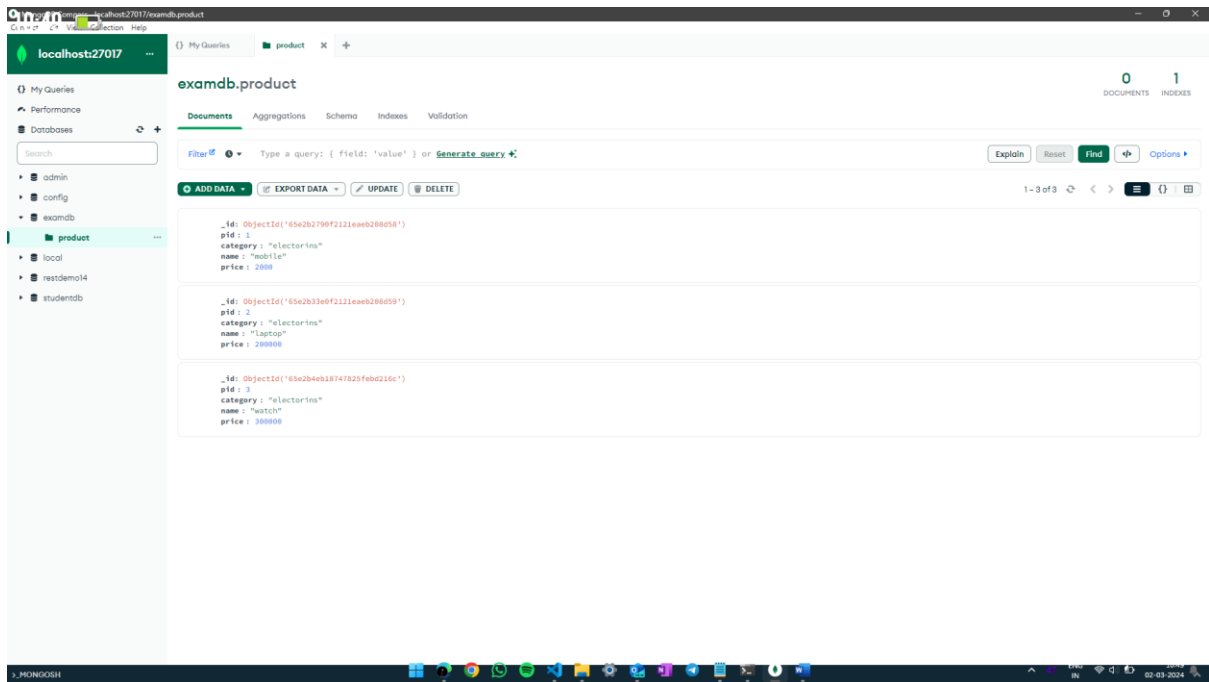
ID :2200090035

Question No: 13
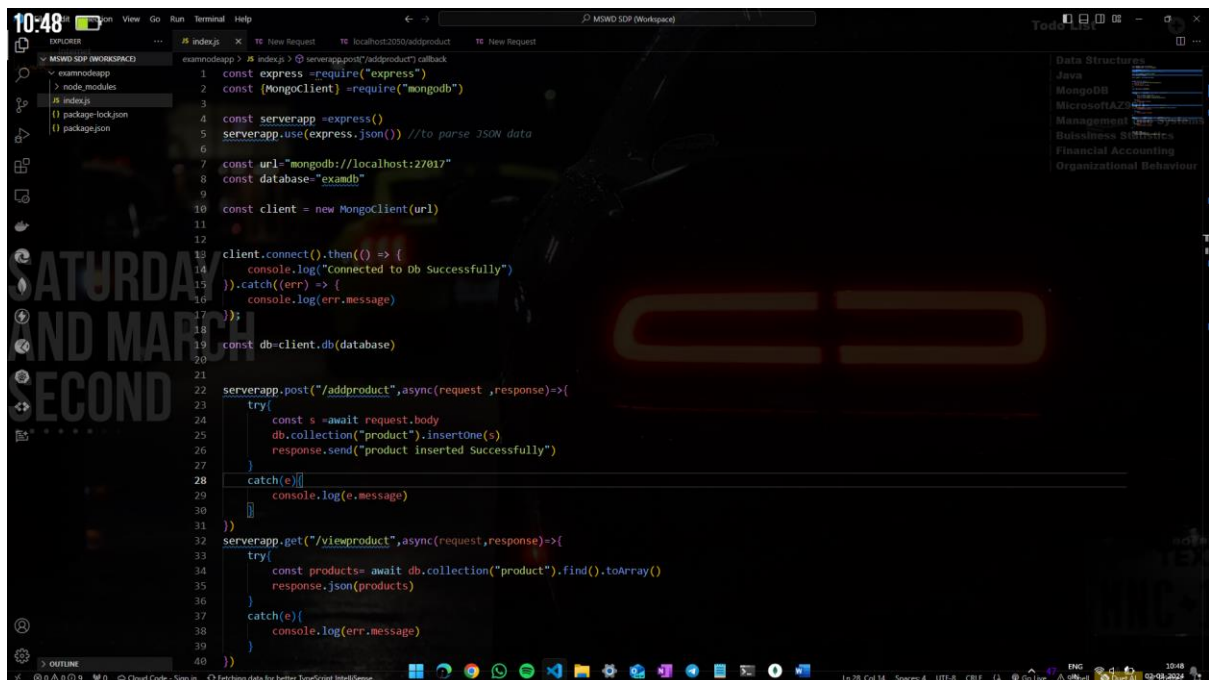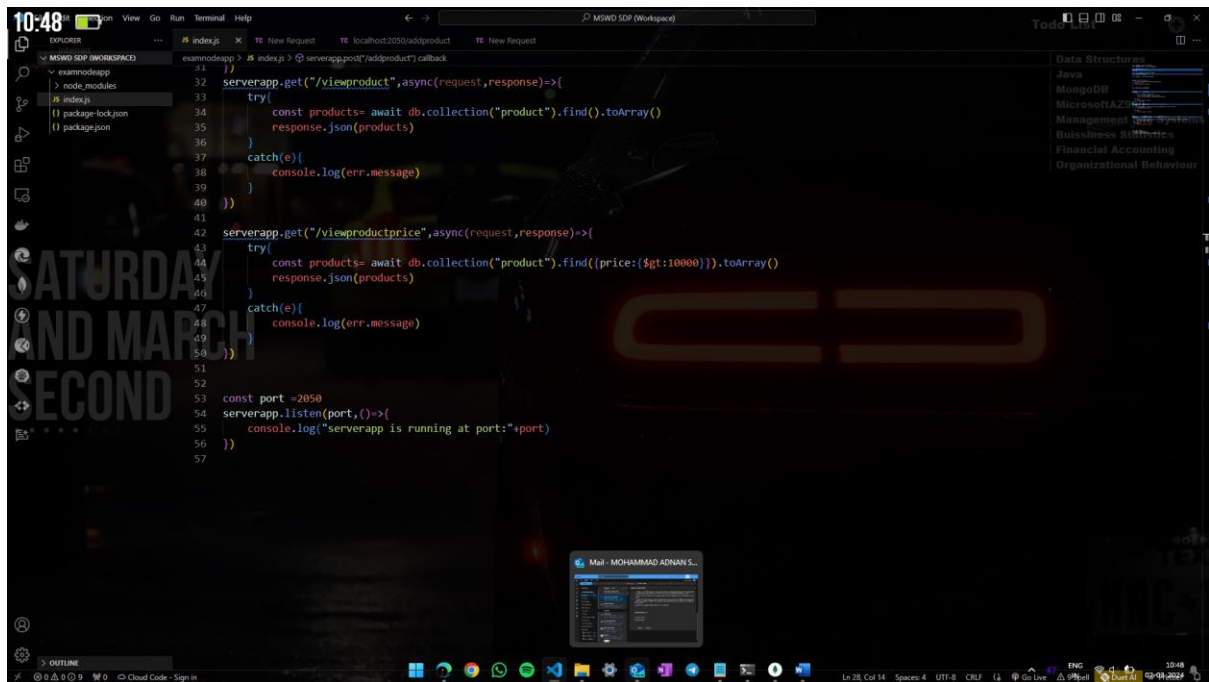**Question:**
i. Set up a new Node js project (examnodeapp) & create package.json file
ii. Install all required packages
iii. Create index.js file in examnodeapp
iv. Create an Express server & Connect your Node js application to a MongoDB database (examdb) & collection (product)
v. Implement two routes
  - Route 1: HTTP POST request to insert product data into the MongoDB database. The route should accept parameters such as product id, category, name, price and insert this data into the database.
  - Route 2: HTTP GET request to fetch products data from the MongoDB database and display it in JSON format.
  - Route 3: HTTP GET request to fetch products data with price greater than 10000 from the MongoDB database and display it in JSON format. Take price input through URL. Ensure proper error handling for this operation.
vi. Test above 3 requests using thunder client / postman api.
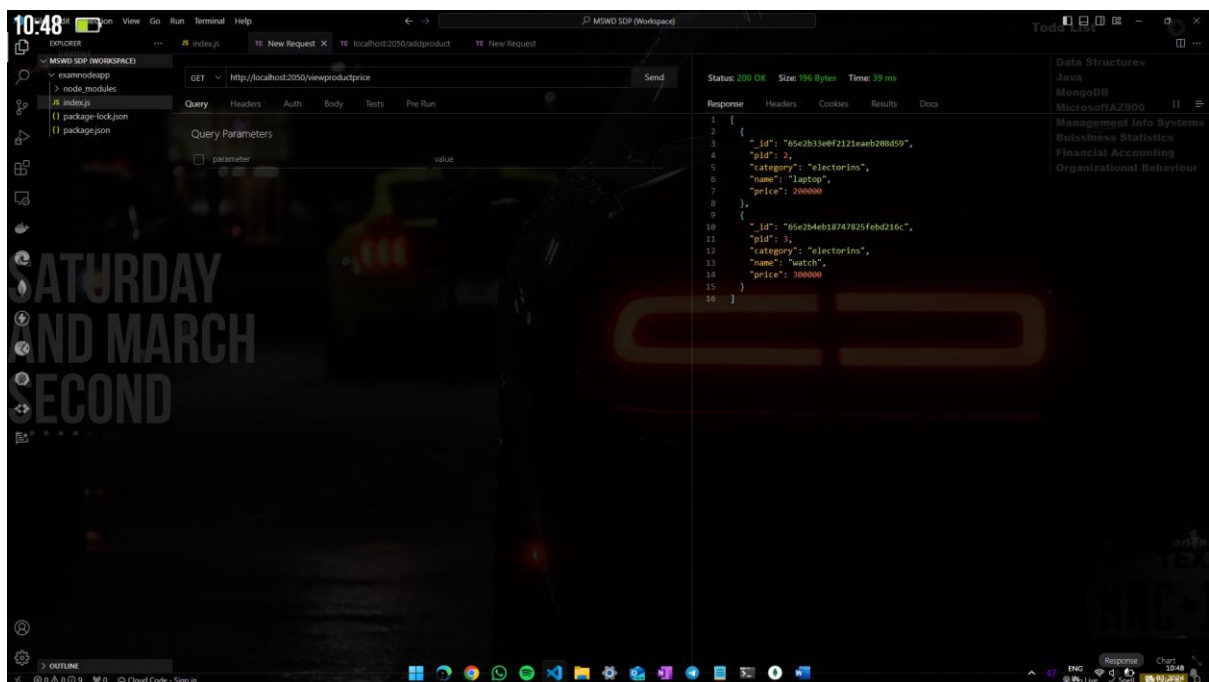
1.inserted the data into the mongodb database by routes

Index.js file


```javascript
const express =require("express")
const {MongoClient} =require("mongodb")

const serverapp =express()
serverapp.use(express.json()) //to parse JSON data

const url="mongodb://localhost:27017"
const database="examdb"

const client = new MongoClient(url)


client.connect().then(() => {
    console.log("Connected to Db Successfully")
}).catch((err) => {
    console.log(err.message)
});

const db=client.db(database)


serverapp.post("/addproduct",async(request ,response)=>{
    try{
        const s =await request.body
        db.collection("product").insertOne(s)
        response.send("product inserted Successfully")
    }
    catch(e){
        console.log(e.message)
    }
})
serverapp.get("/viewproduct",async(request,response)=>{
    try{
        const products= await db.collection("product").find().toArray()
        response.json(products)
    }
    catch(e){
        console.log(err.message)
    }
})
```

```javascript
serverapp.get("/viewproduct",async(request,response)=>{
    try{
        const products= await db.collection("product").find().toArray()
        response.json(products)
    }
    catch(e){
        console.log(err.message)
    }
})

serverapp.get("/viewproductprice",async(request,response)=>{
    try{
        const products= await db.collection("product").find({price:{$gt:10000}}).toArray()
        response.json(products)
    }
    catch(e){
        console.log(err.message)
    }
})


const port =2050
serverapp.listen(port,()=>{
    console.log("serverapp is running at port:"+port)
})
```

View product price based on the condition price greater than 10000


```
GET  http://localhost:2050/viewproductprice        Send

Status: 200 OK   Size: 196 Bytes   Time: 39 ms

[
  {
    "_id": "65e2b33e0f2121eaeb208d59",
    "pid": 2,
    "category": "electorins",
    "name": "laptop",
    "price": 200000
  },
  {
    "_id": "65e2b4eb18747825febd216c",
    "pid": 3,
    "category": "electorins",
    "name": "watch",
    "price": 300000
  }
]
```

View all the products
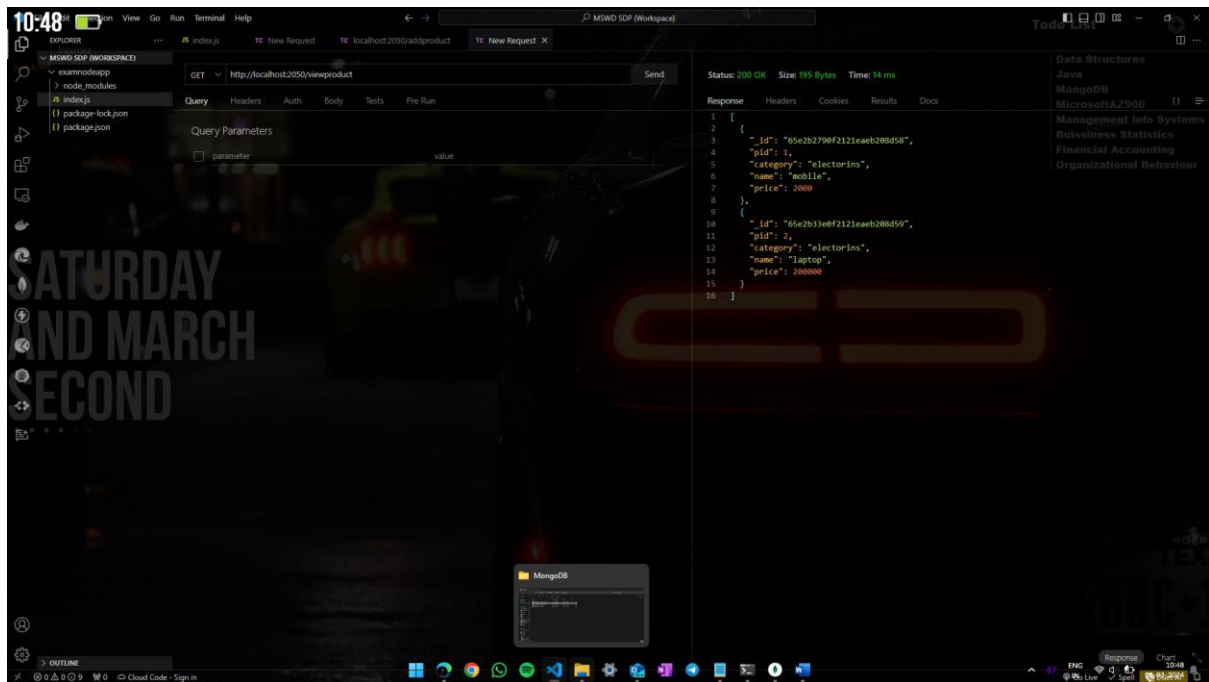
Adds the product data

Code:

```javascript
const express =require("express")
const {MongoClient} =require("mongodb")

const serverapp =express()
serverapp.use(express.json()) //to parse JSON data

const url="mongodb://localhost:27017"
const database="examdb"

const client = new MongoClient(url)


client.connect().then(() => {
    console.log("Connected to Db Successfully")
}).catch((err) => {
    console.log(err.message)
});

const db=client.db(database)


serverapp.post("/addproduct",async(request ,response)=>{
    try{
        const s =await request.body
        db.collection("product").insertOne(s)
        response.send("product inserted Successfully")
    }
    catch(e){
        console.log(e.message)
    }
})
serverapp.get("/viewproduct",async(request,response)=>{
    try{
        const products= await
db.collection("product").find().toArray()
        response.json(products)
    }
    catch(e){
```

```
            console.log(err.message)
    }
})

serverapp.get("/viewproductprice",async(request,response)=>{
    try{
        const products= await
db.collection("product").find({price:{$gt:10000}}).toArray()
        response.json(products)
    }
    catch(e){
        console.log(err.message)
    }
})


const port =2050
serverapp.listen(port,()=>{
    console.log("serverapp is running at port:"+port)
})
```