NodeJS capstone project

Dec 05, 2024

4NM21CS071

Question 01:

```
const express = require("express");
const mongoose = require("mongoose");

const app = express();
app.use(express.json());

mongoose
  .connect("mongodb://localhost:27017/shoppingstore", {
    useNewUrlParser: true,
    useUnifiedTopology: true,
  })
  .then(() => console.log("MongoDB Connected"))
  .catch(console.error);

const productSchema = new mongoose.Schema({
  productId: { type: String, unique: true },
  productTitle: { type: String, required: true },
  productPrice: { type: Number, required: true },
  productDesc: { type: String, required: true },
});

productSchema.pre("save", async function (next) {
  const count = await mongoose.models.Product.countDocuments();
  this.productId = `P${count + 1}`;
  next();
});

const Product = mongoose.model("Product", productSchema);

app.post("/products", async (req, res) => {
  try {
    const product = new Product(req.body);
    await product.save();
    res.status(201).json({ message: "Product added successfully", product });
  } catch (error) {
    res.status(500).json({ error: error.message });
  }
});

app.get("/products", async (req, res) => {
  try {
    const products = await Product.find(
```

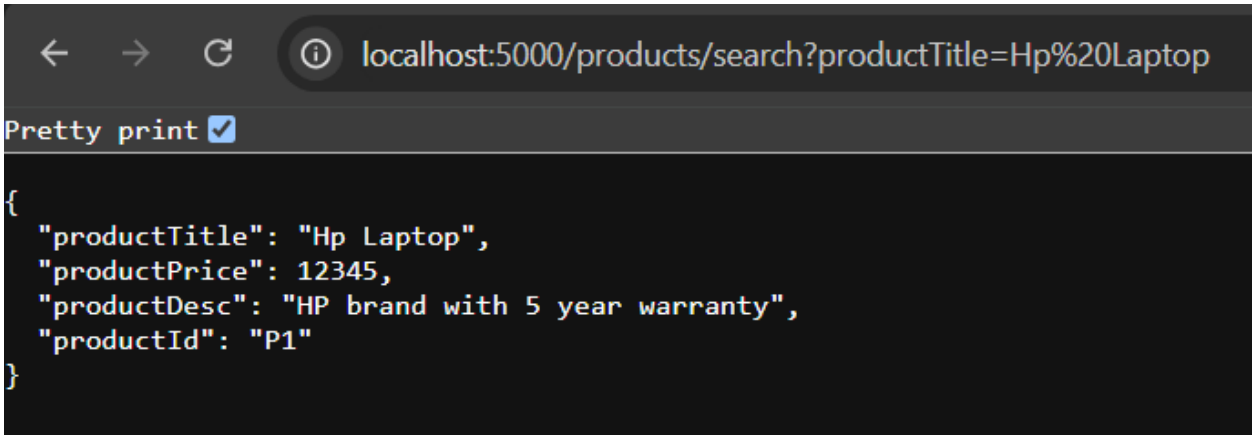
```
      {},
      { _id: 0, productTitle: 1, productPrice: 1, productDesc: 1 }
    );
    res.status(200).json(products);
  } catch (error) {
    res.status(500).json({ error: error.message });
  }
});

app.get("/products/search", async (req, res) => {
  try {
    const product = await Product.findOne(
      { productTitle },
      { _id: 0, productTitle: 1, productPrice: 1, productDesc: 1 }
    );
    if (product) res.status(200).json(product);
    else res.status(404).json({ message: "Product not found" });
  } catch (error) {
    res.status(500).json({ error: error.message });
  }
});

app.put("/products/:productId", async (req, res) => {
  try {
    const product = await Product.findOneAndUpdate(
      { productId: req.params.productId },
      { productPrice: req.body.productPrice },
      { new: true, fields: "productId productTitle productPrice" }
    );
    if (product)
      res.status(200).json({ message: "Price updated successfully", product });
    else res.status(404).json({ message: "Product not found" });
  } catch (error) {
    res.status(500).json({ error: error.message });
  }
});

app.delete("/products/:productId", async (req, res) => {
  try {
    const product = await Product.findOneAndDelete({
      productId: req.params.productId,
    });
    if (product)
      res.status(200).json({ message: "Product removed successfully" });
```

```
    else res.status(404).json({ message: "Product not found" });
  } catch (error) {
    res.status(500).json({ error: error.message });
  }
});

const PORT = 5000;
app.listen(PORT, () => console.log(`Server running on port ${PORT}`));
```

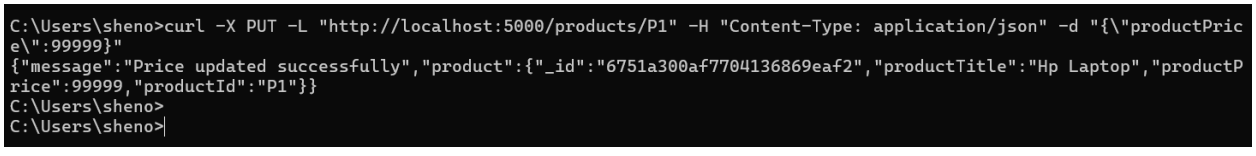
```
C:\Users\sheno>curl -X POST -L "http://localhost:5000/products" -H "Content-Type: application/json" -d "{\"productTitle\
":\"Hp Laptop\",\"productPrice\":12345,\"productDesc\":\"HP brand with 5 year warranty\"}"
{"message":"Product added successfully","product":{"productTitle":"Hp Laptop","productPrice":12345,"productDesc":"HP bra
nd with 5 year warranty","_id":"6751a300af7704136869eaf2","productId":"P1","__v":0}}
C:\Users\sheno>curl -X POST -L "http://localhost:5000/products" -H "Content-Type: application/json" -d "{\"productTitle\
":\"samsung mobile\",\"productPrice\":45678,\"productDesc\":\"Samsung brand with 5 year warranty\"}"
{"message":"Product added successfully","product":{"productTitle":"samsung mobile","productPrice":45678,"productDesc":"S
amsung brand with 5 year warranty","_id":"6751a315af7704136869eaf5","productId":"P2","__v":0}}
C:\Users\sheno>curl -X POST -L "http://localhost:5000/products" -H "Content-Type: application/json" -d "{\"productTitle\
":\"Dell mouse\",\"productPrice\":123,\"productDesc\":\"Dell mouse with 1 year warranty\"}"
{"message":"Product added successfully","product":{"productTitle":"Dell mouse","productPrice":123,"productDesc":"Dell mo
use with 1 year warranty","_id":"6751a31aaf7704136869eaf8","productId":"P3","__v":0}}
C:\Users\sheno>curl -X POST -L "http://localhost:5000/products" -H "Content-Type: application/json" -d "{\"productTitle\
":\"Hp speaker\",\"productPrice\":5623,\"productDesc\":\"HP brand with 5 year warranty\"}"
{"message":"Product added successfully","product":{"productTitle":"Hp speaker","productPrice":5623,"productDesc":"HP bra
nd with 5 year warranty","_id":"6751a324af7704136869eafb","productId":"P4","__v":0}}
C:\Users\sheno>
```

```
C:\Users\sheno>curl -X GET -L "http://localhost:5000/products"
[{"productTitle":"Hp Laptop","productPrice":12345,"productDesc":"HP brand with 5 year warranty","productId":"P1"},{"prod
uctTitle":"samsung mobile","productPrice":45678,"productDesc":"Samsung brand with 5 year warranty","productId":"P2"},{"p
roductTitle":"Dell mouse","productPrice":123,"productDesc":"Dell mouse with 1 year warranty","productId":"P3"},{"product
Title":"Hp speaker","productPrice":5623,"productDesc":"HP brand with 5 year warranty","productId":"P4"}]
C:\Users\sheno>
```

```
[
  {
    "productTitle": "Hp Laptop",
    "productPrice": 12345,
    "productDesc": "HP brand with 5 year warranty",
    "productId": "P1"
  },
  {
    "productTitle": "samsung mobile",
    "productPrice": 45678,
    "productDesc": "Samsung brand with 5 year warranty",
    "productId": "P2"
  },
  {
    "productTitle": "Dell mouse",
    "productPrice": 123,
    "productDesc": "Dell mouse with 1 year warranty",
    "productId": "P3"
  },
  {
    "productTitle": "Hp speaker",
    "productPrice": 5623,
    "productDesc": "HP brand with 5 year warranty",
    "productId": "P4"
  }
]
```


```
{
  "productTitle": "Hp Laptop",
  "productPrice": 12345,
  "productDesc": "HP brand with 5 year warranty",
  "productId": "P1"
}
```

```
{
  "productTitle": "Hp Laptop",
  "productPrice": 99999,
  "productDesc": "HP brand with 5 year warranty",
  "productId": "P1"
}
```


```
C:\Users\sheno>
C:\Users\sheno>curl -X DELETE -L "http://localhost:5000/products/P1"
{"message":"Product removed successfully"}
C:\Users\sheno>
```


```
[
  {
    "productTitle": "samsung mobile",
    "productPrice": 45678,
    "productDesc": "Samsung brand with 5 year warranty",
    "productId": "P2"
  },
  {
    "productTitle": "Dell mouse",
    "productPrice": 123,
    "productDesc": "Dell mouse with 1 year warranty",
    "productId": "P3"
  },
  {
    "productTitle": "Hp speaker",
    "productPrice": 5623,
    "productDesc": "HP brand with 5 year warranty",
    "productId": "P4"
  }
]
```

Question 02:

2. *shoppers.com* is happy to see a huge number of products getting added to their inventory. Now they want us to write some queries to fetch the products based on the given requirements from shoppers.com.-15M

1. Write a query to fetch all the products in shoppers.com inventory

2. Write a query to display all the details of products under the category Mobiles

3. Write a query to display all the details of product with productId *P1001*

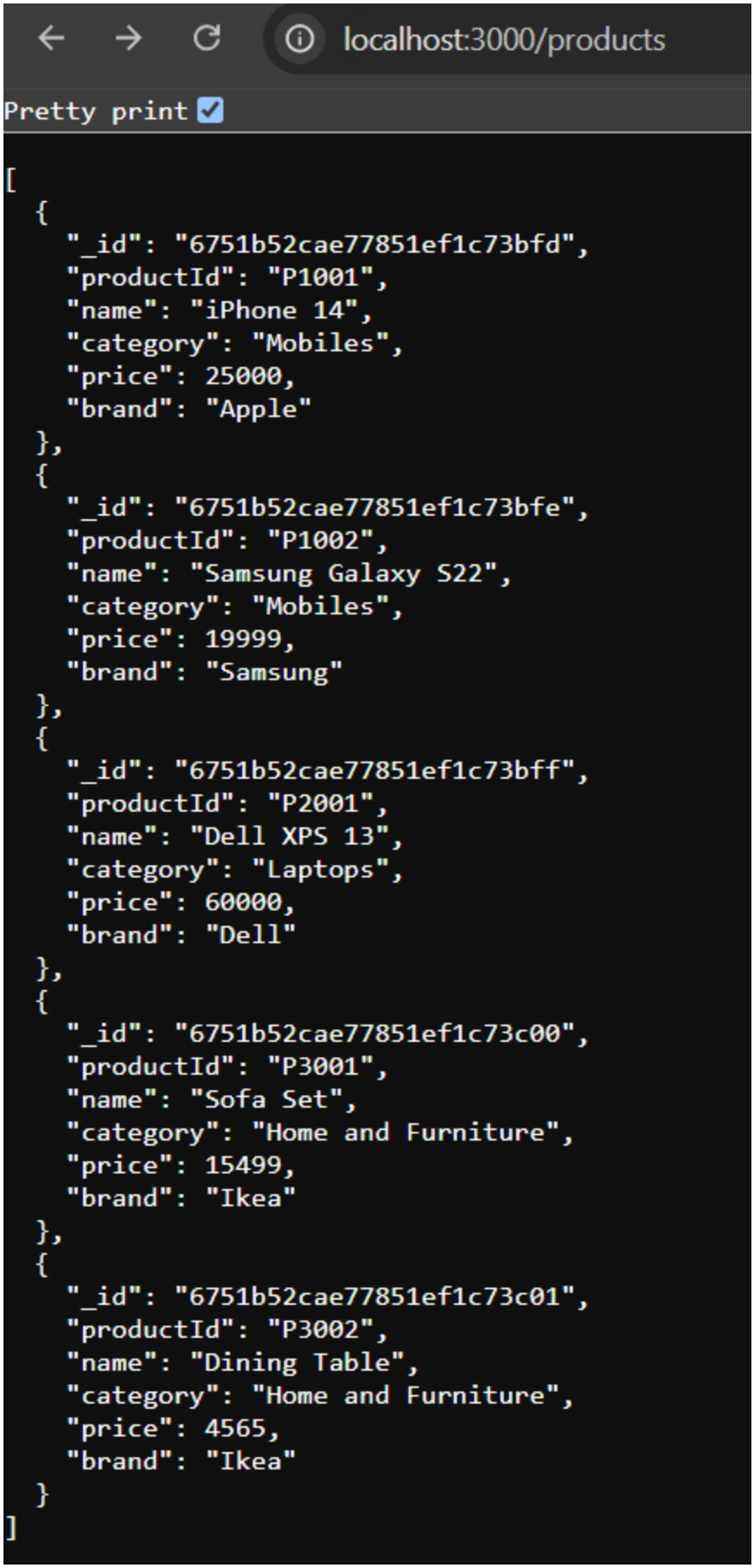
4. Write a query to display the name of all the laptops

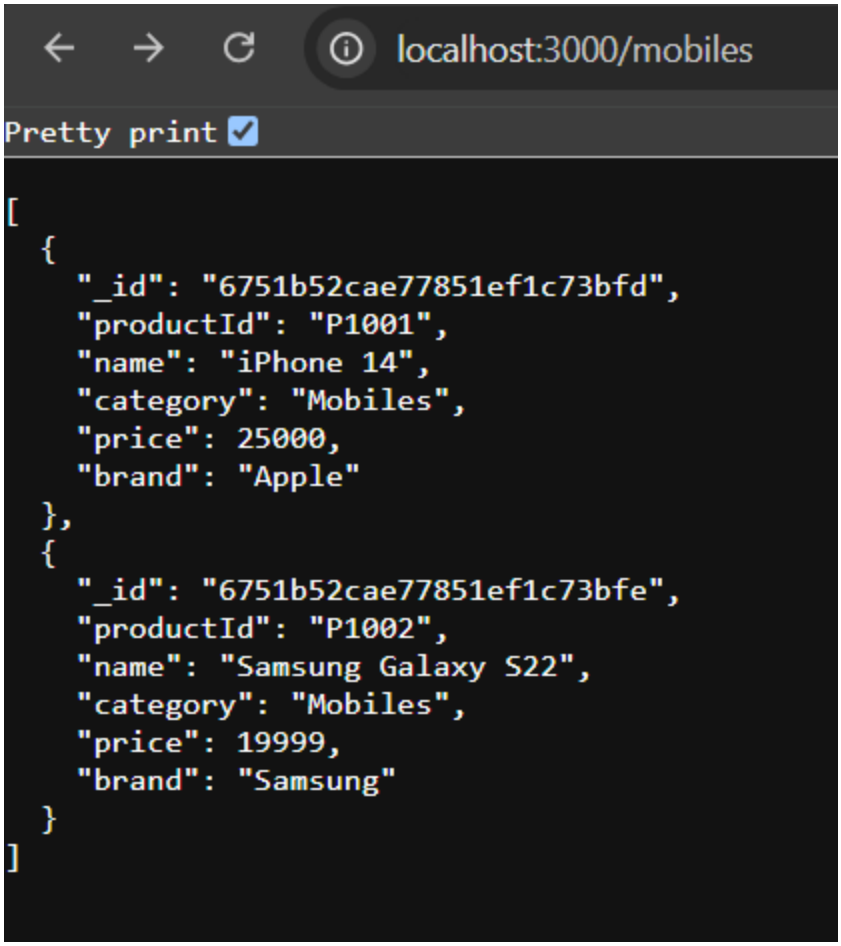
5. Write a query to fetch all *Home and Furnitures* products.

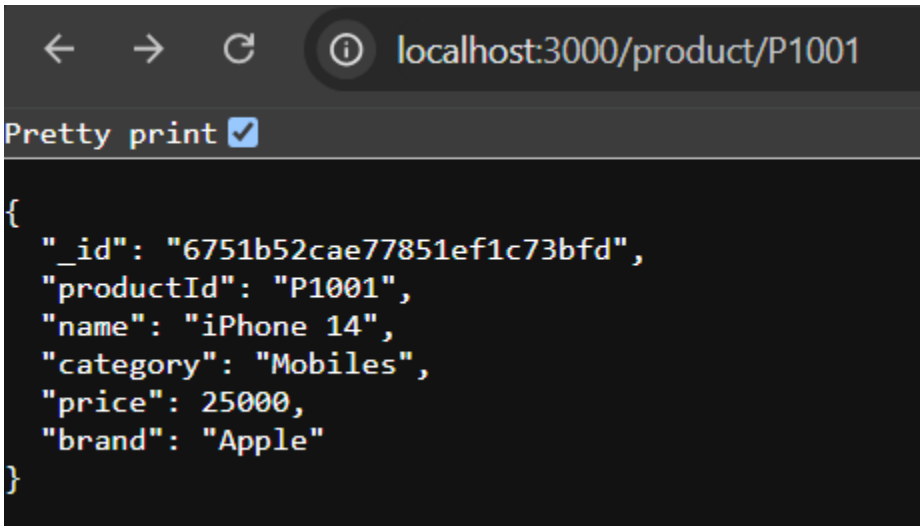
```
const express = require("express");
const { MongoClient } = require("mongodb");

const app = express();
const port = 3000;
const uri = "mongodb://localhost:27017";
const client = new MongoClient(uri);

app.use(express.json());

async function connectToDatabase() {
  await client.connect();
  const database = client.db("shoppersonline");
  return database.collection("products");
}

app.get("/products", async (req, res) => {
  try {
    const products = await connectToDatabase();
    const allProducts = await products.find({}).toArray();
    res.json(allProducts);
  } catch (error) {
    console.error(error);
    res.status(500).send("Error fetching products");
  }
```

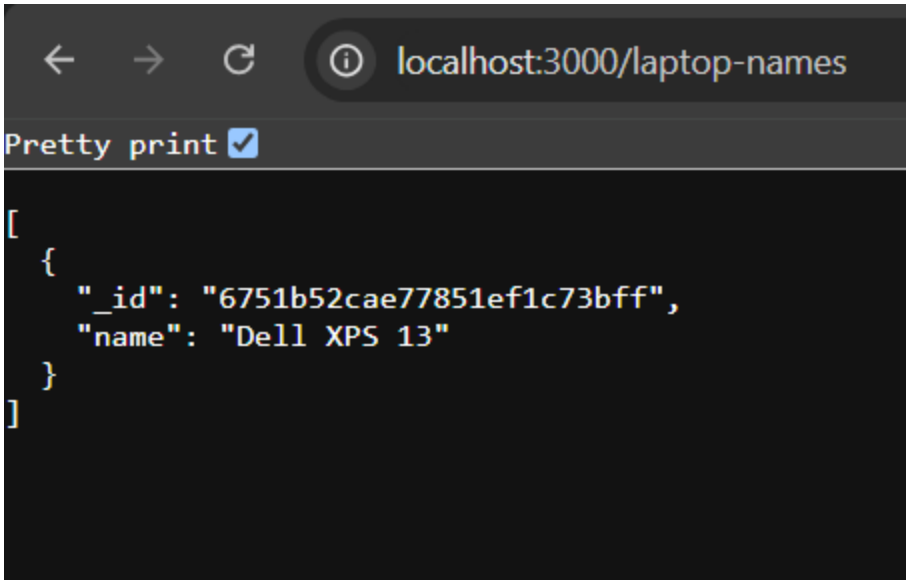
```
});

app.get("/mobiles", async (req, res) => {
  try {
    const products = await connectToDatabase();
    const mobiles = await products.find({ category: "Mobiles"
}).toArray();
    res.json(mobiles);
  } catch (error) {
    console.error(error);
    res.status(500).send("Error fetching mobiles");
  }
});


app.get("/product/:id", async (req, res) => {
  try {
    const productId = req.params.id;
    const products = await connectToDatabase();
    const product = await products.findOne({ productId: productId });
    if (product) {
      res.json(product);
    } else {
      res.status(404).send("Product not found");
    }
  } catch (error) {
    console.error(error);
    res.status(500).send("Error fetching product");
  }
});

app.get("/laptop-names", async (req, res) => {
  try {
    const products = await connectToDatabase();
    const laptops = await products
      .find({ category: "Laptops" })
      .project({ name: 1 })
      .toArray();
```

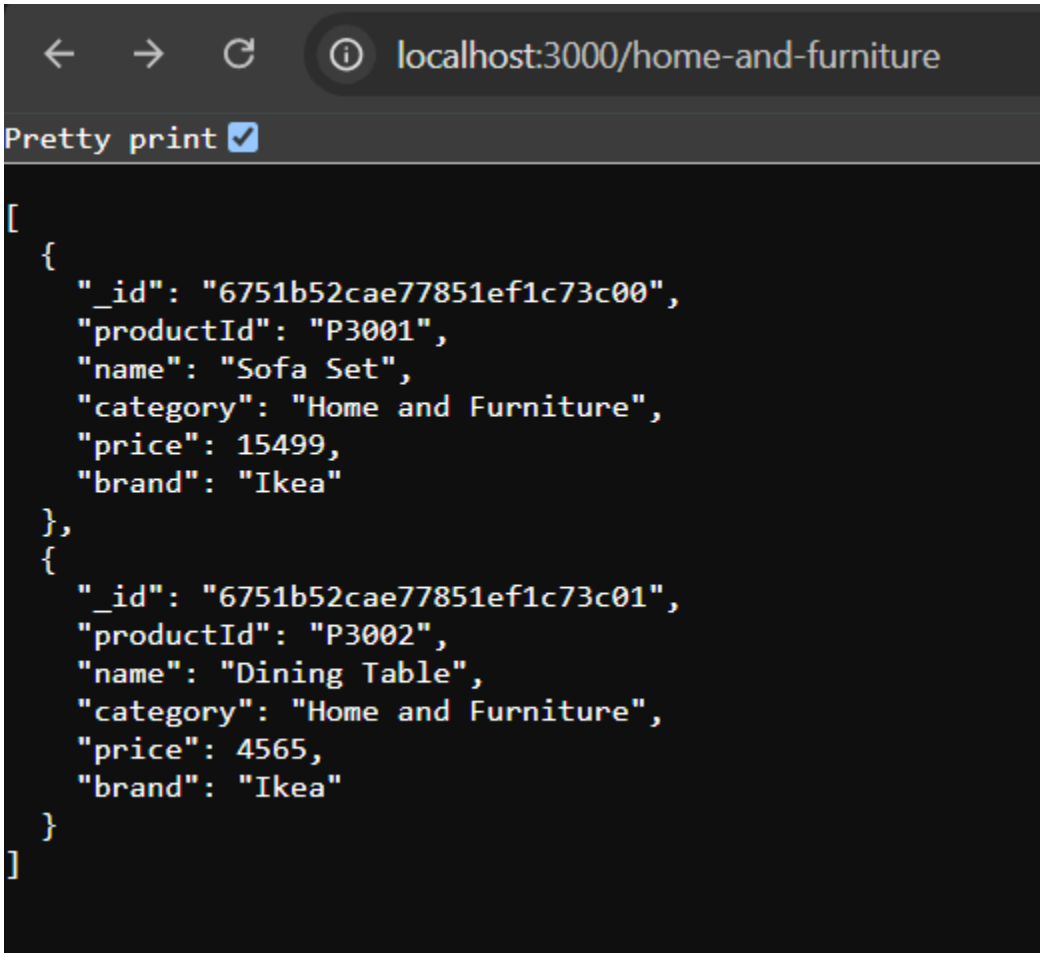
```
    res.json(laptops);
  } catch (error) {
    console.error(error);
    res.status(500).send("Error fetching laptop names");
  }
});

app.get("/home-and-furniture", async (req, res) => {
  try {
    const products = await connectToDatabase();
    const homeAndFurniture = await products
      .find({ category: "Home and Furniture" })
      .toArray();
    res.json(homeAndFurniture);
  } catch (error) {
    console.error(error);
    res.status(500).send("Error fetching home and furniture items");
  }
});

app.listen(port, () => {
  console.log(`Server running on port: ${port}`);
});
```


localhost:3000/products
Pretty print
[
{
"_id": "6751b52cae77851ef1c73bfd",
"productId": "P1001",
"name": "iPhone 14",
"category": "Mobiles",
"price": 25000,
"brand": "Apple"
},
{
"_id": "6751b52cae77851ef1c73bfe",
"productId": "P1002",
"name": "Samsung Galaxy S22",
"category": "Mobiles",
"price": 19999,
"brand": "Samsung"
},
{
"_id": "6751b52cae77851ef1c73bff",
"productId": "P2001",
"name": "Dell XPS 13",
"category": "Laptops",
"price": 60000,
"brand": "Dell"
},
{
"_id": "6751b52cae77851ef1c73c00",
"productId": "P3001",
"name": "Sofa Set",
"category": "Home and Furniture",
"price": 15499,
"brand": "Ikea"
},
{
"_id": "6751b52cae77851ef1c73c01",
"productId": "P3002",
"name": "Dining Table",
"category": "Home and Furniture",
"price": 4565,
"brand": "Ikea"
}
]


localhost:3000/mobiles
Pretty print
[
{
"_id": "6751b52cae77851ef1c73bfd",
"productId": "P1001",
"name": "iPhone 14",
"category": "Mobiles",
"price": 25000,
"brand": "Apple"
},
{
"_id": "6751b52cae77851ef1c73bfe",
"productId": "P1002",
"name": "Samsung Galaxy S22",
"category": "Mobiles",
"price": 19999,
"brand": "Samsung"
}
]

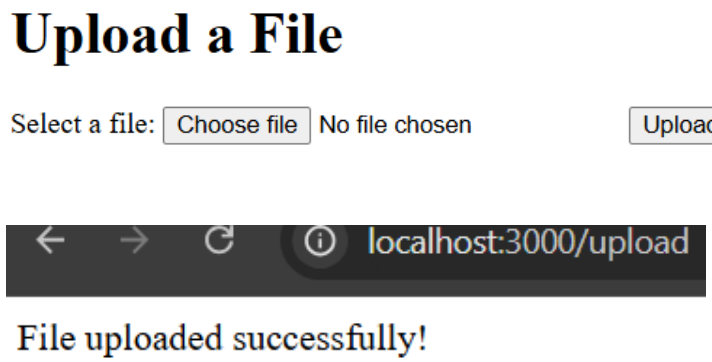

localhost:3000/product/P1001
Pretty print
{
"_id": "6751b52cae77851ef1c73bfd",
"productId": "P1001",
"name": "iPhone 14",
"category": "Mobiles",
"price": 25000,
"brand": "Apple"
}


localhost:3000/laptop-names
Pretty print
[
{
"_id": "6751b52cae77851ef1c73bff",
"name": "Dell XPS 13"
}
]

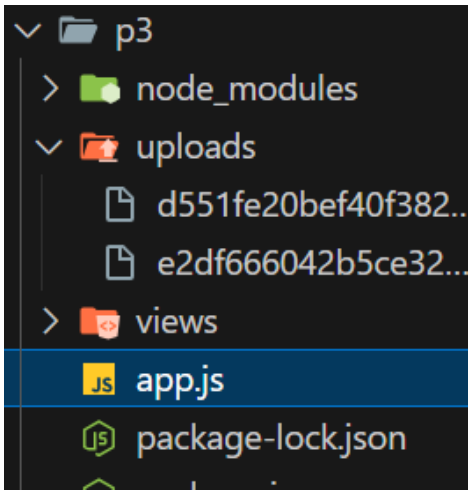

localhost:3000/home-and-furniture
Pretty print
[
{
"_id": "6751b52cae77851ef1c73c00",
"productId": "P3001",
"name": "Sofa Set",
"category": "Home and Furniture",
"price": 15499,
"brand": "Ikea"
},
{
"_id": "6751b52cae77851ef1c73c01",
"productId": "P3002",
"name": "Dining Table",
"category": "Home and Furniture",
"price": 4565,
"brand": "Ikea"
}
]

Question 03:

```
const multer = require("multer");
const path = require("path");
const express = require("express");
const app = express();

app.set("view engine", "ejs");
app.set("views", path.join(__dirname, "views"));

const upload = multer({ dest: "uploads/" });

app.post("/upload", upload.single("file"), (req, res) => {
  if (!req.file) {
    return res.status(400).send("No file uploaded.");
  }
  console.log("Uploaded file:", req.file);
  res.send("File uploaded successfully!");
});

app.get("/", (req, res) => {
  res.render("main");
});

const PORT = 3000;
app.listen(PORT, () => {
  console.log(`Server running on ${PORT}`);
});
```



```
PS D:\Finacle Training\capstone\p3> node .\app.js
Server running on 3000
Uploaded file: {
  fieldname: 'file',
  originalname: 'Screenshot 2024-12-05 180625.png',
  encoding: '7bit',
  mimetype: 'image/png',
  destination: 'uploads/',
  filename: 'e2df666042b5ce322c8e7ce2021ae8f9',
  path: 'uploads\\e2df666042b5ce322c8e7ce2021ae8f9',
  size: 42810
}
```



Question 04:

1.

```
const os = require("os");

console.log("OS Type: ", os.type());
console.log("OS Platform: ", os.platform());
console.log("CPU Architecture: ", os.arch());
console.log("Total Memory: ", os.totalmem());
console.log("Free Memory: ", os.freemem());
console.log("Home Directory: ", os.homedir());
```

```
PS D:\Finacle Training\capstone\p4> node .\p1.js
OS Type: Windows_NT
OS Platform: win32
CPU Architecture: x64
Total Memory: 16906022912
Free Memory:  5269151744
Home Directory: C:\Users\sheno
PS D:\Finacle Training\capstone\p4>
```

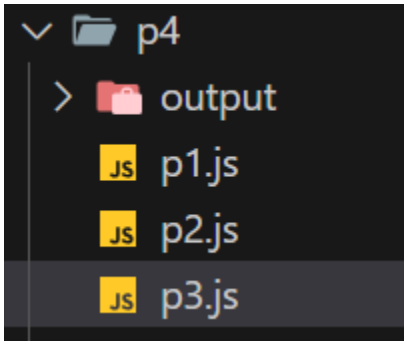
2.

```
const path = require("path");

const filePath = "D:\Finacle Training\capstone\p4\p2.js";

console.log(`Directory Name: ${path.dirname(filePath)}`);
console.log(`Base Name: ${path.basename(filePath)}`);
console.log(`Extension: ${path.extname(filePath)}`);
console.log(`Is Absolute Path: ${path.isAbsolute(filePath)}`);
```

```
PS D:\Finacle Training\capstone\p4> node .\p2.js
Directory Name: D:
Base Name: Finacle Trainingcapstonep4p2.js
Extension: .js
Is Absolute Path: false
PS D:\Finacle Training\capstone\p4>
```

3.

```
const fs = require("fs");
const path = require("path");

const directoryPath = path.join(__dirname, "output");

fs.mkdir(directoryPath, (err) => {
  if (err) {
    return console.log(err);
  }
  console.log("Directory created successfully");
});
```

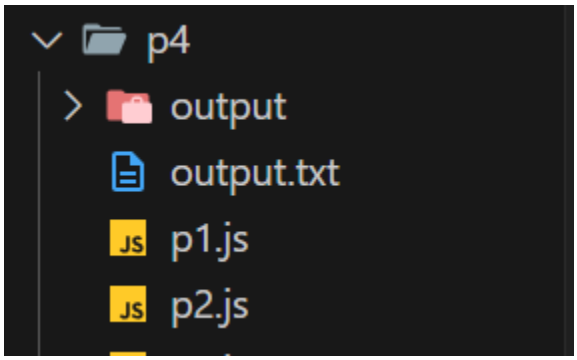



```
PS D:\Finacle Training\capstone\p4> node .\p3.js
Directory created successfully
```

4.

```
const fs = require("fs");
const path = require("path");

const filePath = path.join(__dirname, "output.txt");
const fileContent = "NodeJS capstone project,Dec 05 2024";

fs.writeFile(filePath, fileContent, (err) => {
  if (err) {
    return console.log(err);
  }
```

```
  console.log(`File created successfully`);
});
```




```
PS D:\Finacle Training\capstone\p4> node .\p4.js
File created successfully
PS D:\Finacle Training\capstone\p4>
```