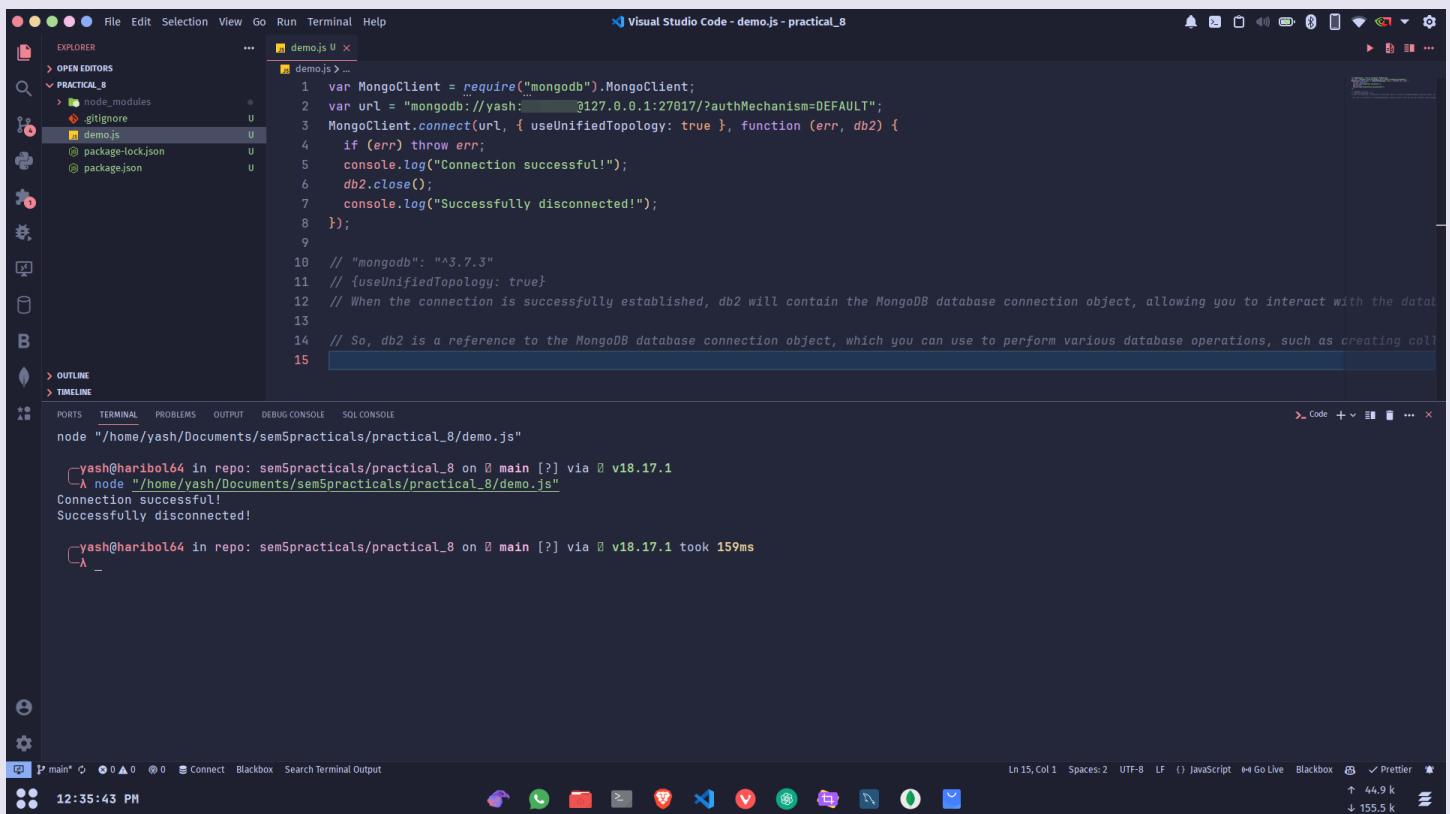


Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

Scenario : Implement an application with CRUD operation using MongoDB with NodeJS. Sympoms Pvt Ltd company wants to manage their employee records and this task is given to you. They are seeking functionalities:

Tasks, Code and Screenshots :

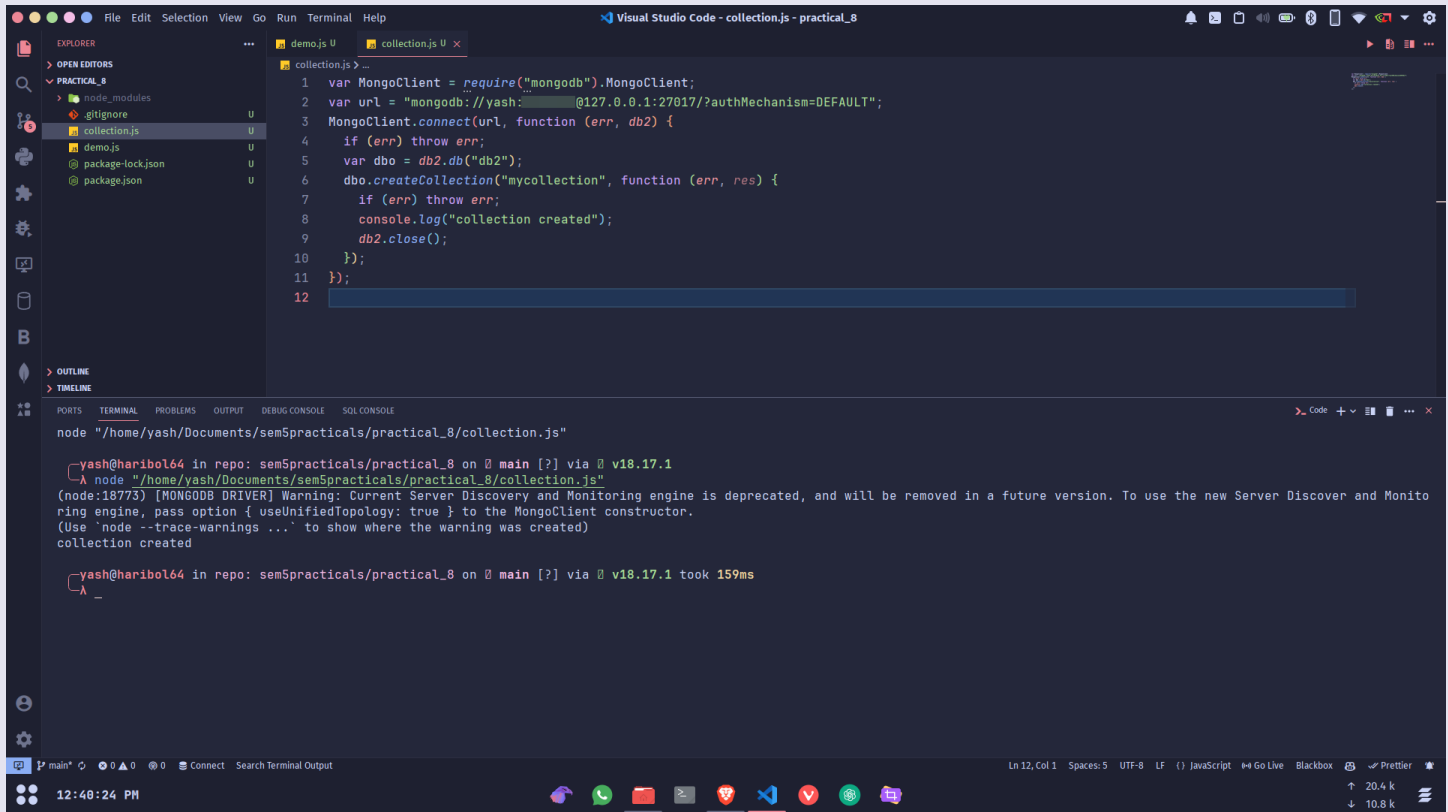
Initializing and connection, creating database, collection, etc.



```
1 var MongoClient = require("mongodb").MongoClient;
2 var url = "mongodb://yash: @127.0.0.1:27017/?authMechanism=DEFAULT";
3 MongoClient.connect(url, { useUnifiedTopology: true }, function (err, db2) {
4   if (err) throw err;
5   console.log("Connection successful!");
6   db2.close();
7   console.log("Successfully disconnected!");
8 });
9
10 // "mongodb": "4.3.7.3"
11 // {useUnifiedTopology: true}
12 // When the connection is successfully established, db2 will contain the MongoDB database connection object, allowing you to interact with the data
13
14 // So, db2 is a reference to the MongoDB database connection object, which you can use to perform various database operations, such as creating coll
15
```

```
node "/home/yash/Documents/sem5practicals/practical_8/demo.js"
yash@haribol64 in repo: sem5practicals/practical_8 on main [?] via v18.17.1
└─ node "/home/yash/Documents/sem5practicals/practical_8/demo.js"
Connection successful!
Successfully disconnected!
yash@haribol64 in repo: sem5practicals/practical_8 on main [?] via v18.17.1 took 159ms
└─
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8



The screenshot shows the Visual Studio Code editor with a file named `collection.js` open. The file contains the following code:

```
1 var MongoClient = require("mongodb").MongoClient;
2 var url = "mongodb://yash: @127.0.0.1:27017/?authMechanism=DEFAULT";
3 MongoClient.connect(url, function (err, db2) {
4   if (err) throw err;
5   var dbo = db2.db("db2");
6   dbo.createCollection("mycollection", function (err, res) {
7     if (err) throw err;
8     console.log("collection created");
9     db2.close();
10  });
11 });
12
```

The terminal output shows the command `node "/home/yash/Documents/sem5practicals/practical_8/collection.js"` being executed. The output is:

```
yash@haribol64 in repo: sem5practicals/practical_8 on 0 main [?] via 0 v18.17.1
A node "/home/yash/Documents/sem5practicals/practical_8/collection.js"
(node:18773) [MONGODB DRIVER] Warning: Current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discover and Monito
ring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
(Use 'node --trace-warnings ...' to show where the warning was created)
collection created

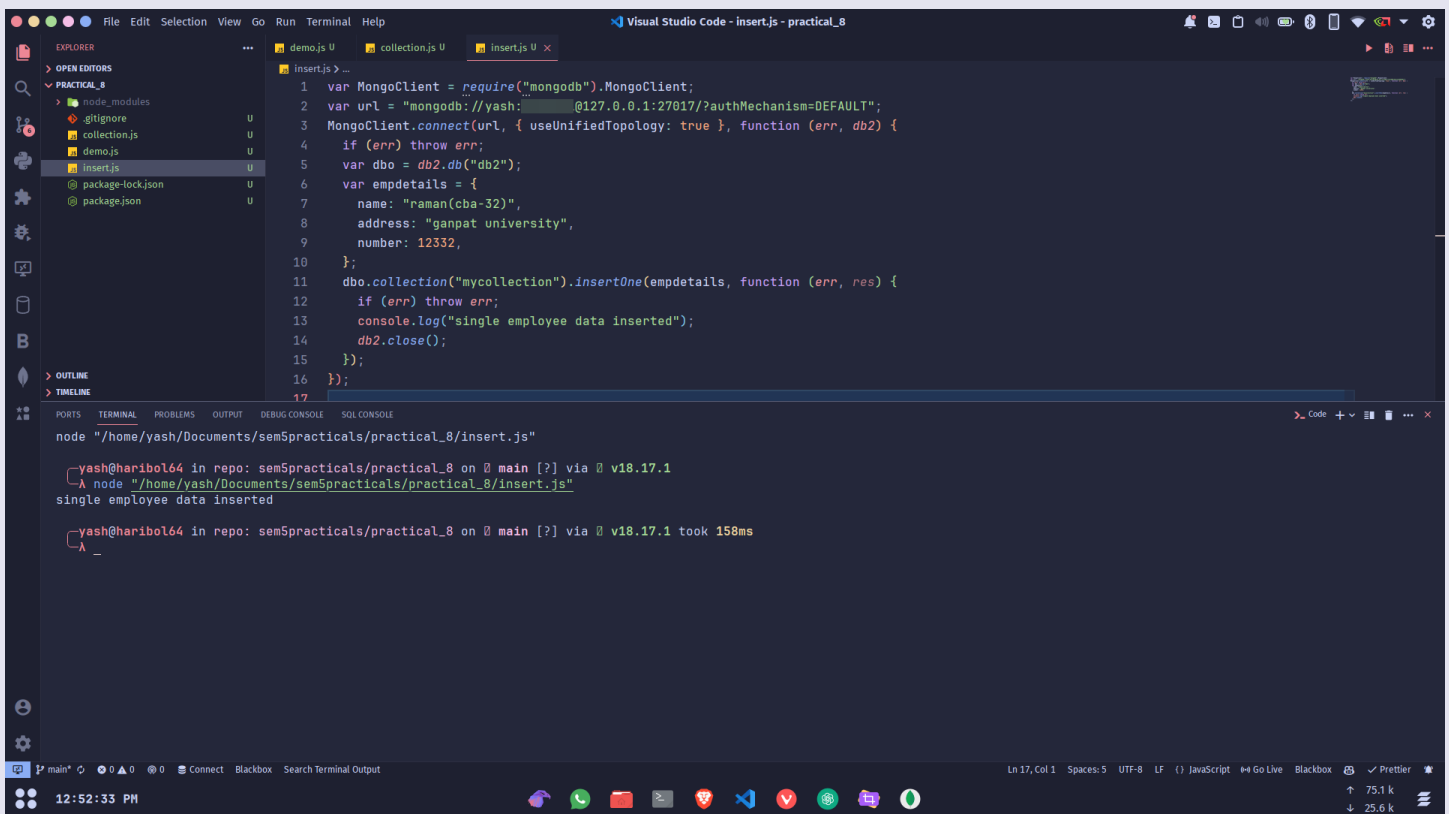
yash@haribol64 in repo: sem5practicals/practical_8 on 0 main [?] via 0 v18.17.1 took 159ms
_
```

Practical 8.1: New joiner data should be stored in DB.

```
var MongoClient = require("mongodb").MongoClient;
var url = "mongodb://yash@127.0.0.1:27017/?authMechanism=DEFAULT";
MongoClient.connect(url, { useUnifiedTopology: true }, function (err, db2)
{
if (err) throw err;
var dbo = db2.db("db2");
var empdetails = {
name: "raman(cba-32)",
address: "ganpat university",
number: 12332,
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

```
};  
  
dbo.collection("mycollection").insertOne(empdetails, function (err, res) {  
if (err) throw err;  
console.log("single employee data inserted");  
db2.close();  
});  
});
```



The screenshot displays the Visual Studio Code interface. The Explorer sidebar on the left shows the project structure with files like .gitignore, collection.js, demo.js, insert.js, package-lock.json, and package.json. The main editor area shows the content of insert.js, which is a Node.js script connecting to a MongoDB database and inserting a single employee record. The script includes the following code:

```
1 var MongoClient = require("mongodb").MongoClient;  
2 var url = "mongodb://yash:123456789@127.0.0.1:27017/?authMechanism=DEFAULT";  
3 MongoClient.connect(url, { useUnifiedTopology: true }, function (err, db2) {  
4   if (err) throw err;  
5   var dbo = db2.db("db2");  
6   var empdetails = {  
7     name: "Raman(cba-32)",  
8     address: "ganpat university",  
9     number: 12332,  
10  };  
11  dbo.collection("mycollection").insertOne(empdetails, function (err, res) {  
12    if (err) throw err;  
13    console.log("single employee data inserted");  
14    db2.close();  
15  });  
16 });  
17
```

The Terminal at the bottom shows the command to run the script: `node "/home/yash/Documents/sem5practicals/practical_8/insert.js"`. The output indicates that the script executed successfully, logging "single employee data inserted" and completing in 158ms.

```
node "/home/yash/Documents/sem5practicals/practical_8/insert.js"  
yash@haribol64 in repo: sem5practicals/practical_8 on  main [?] via  v18.17.1  
node "/home/yash/Documents/sem5practicals/practical_8/insert.js"  
single employee data inserted  
yash@haribol64 in repo: sem5practicals/practical_8 on  main [?] via  v18.17.1 took 158ms
```

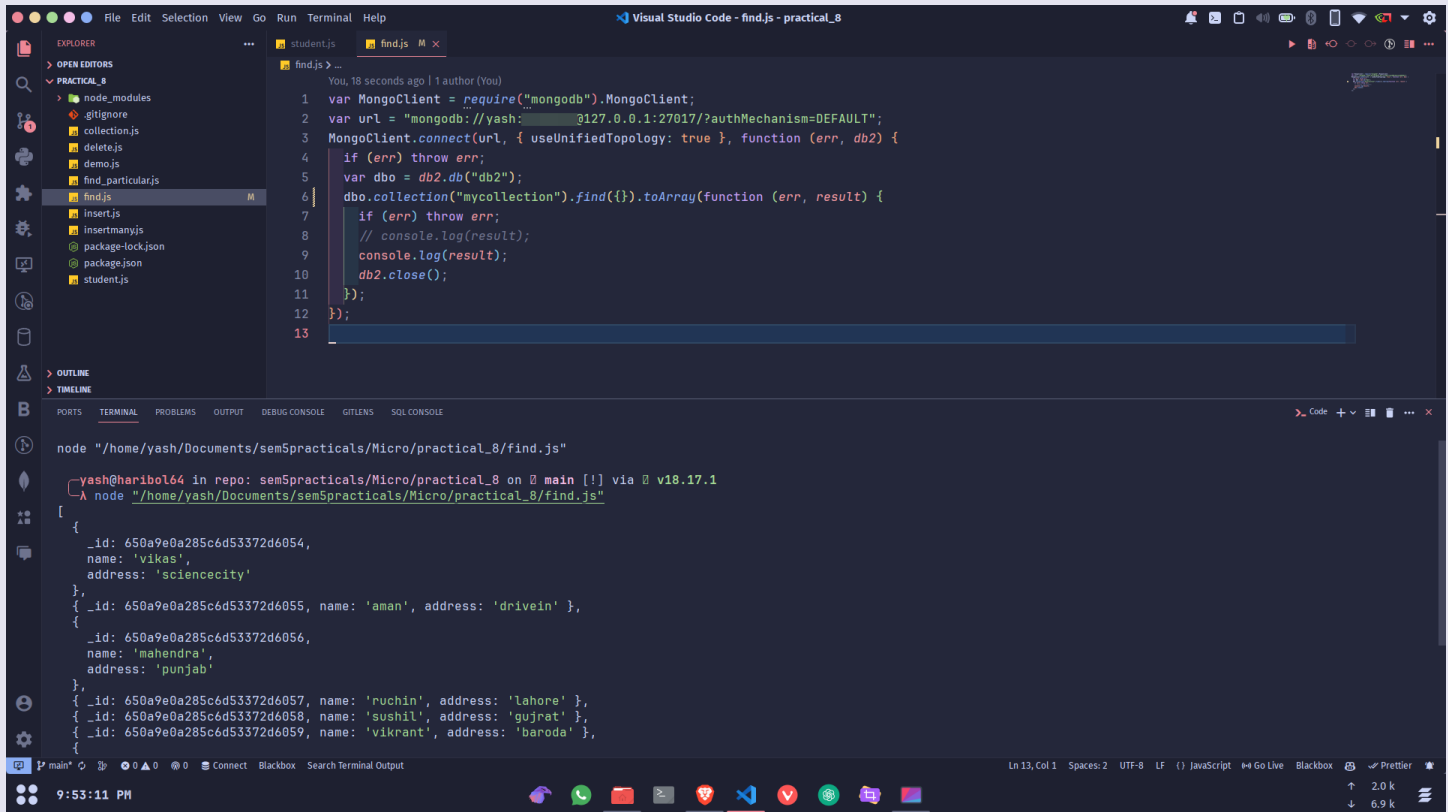
The status bar at the bottom shows the file is named insert.js, located in the practical_8 directory, with 17 lines of code, 1 column, and 1 space. The encoding is UTF-8, and the language is JavaScript.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

Practical 8.2: The admin can see the whole DB

```
var MongoClient = require("mongodb").MongoClient;
var url = "mongodb://yash@127.0.0.1:27017/?authMechanism=DEFAULT";
MongoClient.connect(url, { useUnifiedTopology: true }, function (err, db2)
{
  if (err) throw err;
  var dbo = db2.db("db2");
  dbo.collection("mycollection").find({}).toArray(function (err, result) {
    if (err) throw err;
    // console.log(result);
    console.log(result);
    db2.close();
  });
});
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8



The screenshot shows the Visual Studio Code editor with a file named `find.js` open. The file contains a JavaScript script that connects to a MongoDB database and finds a document in a collection named `mycollection`. The terminal output shows the command `node "/home/yash/Documents/sem5practicals/Micro/practical_8/find.js"` being executed, and the output is a JSON array of documents. The documents are:

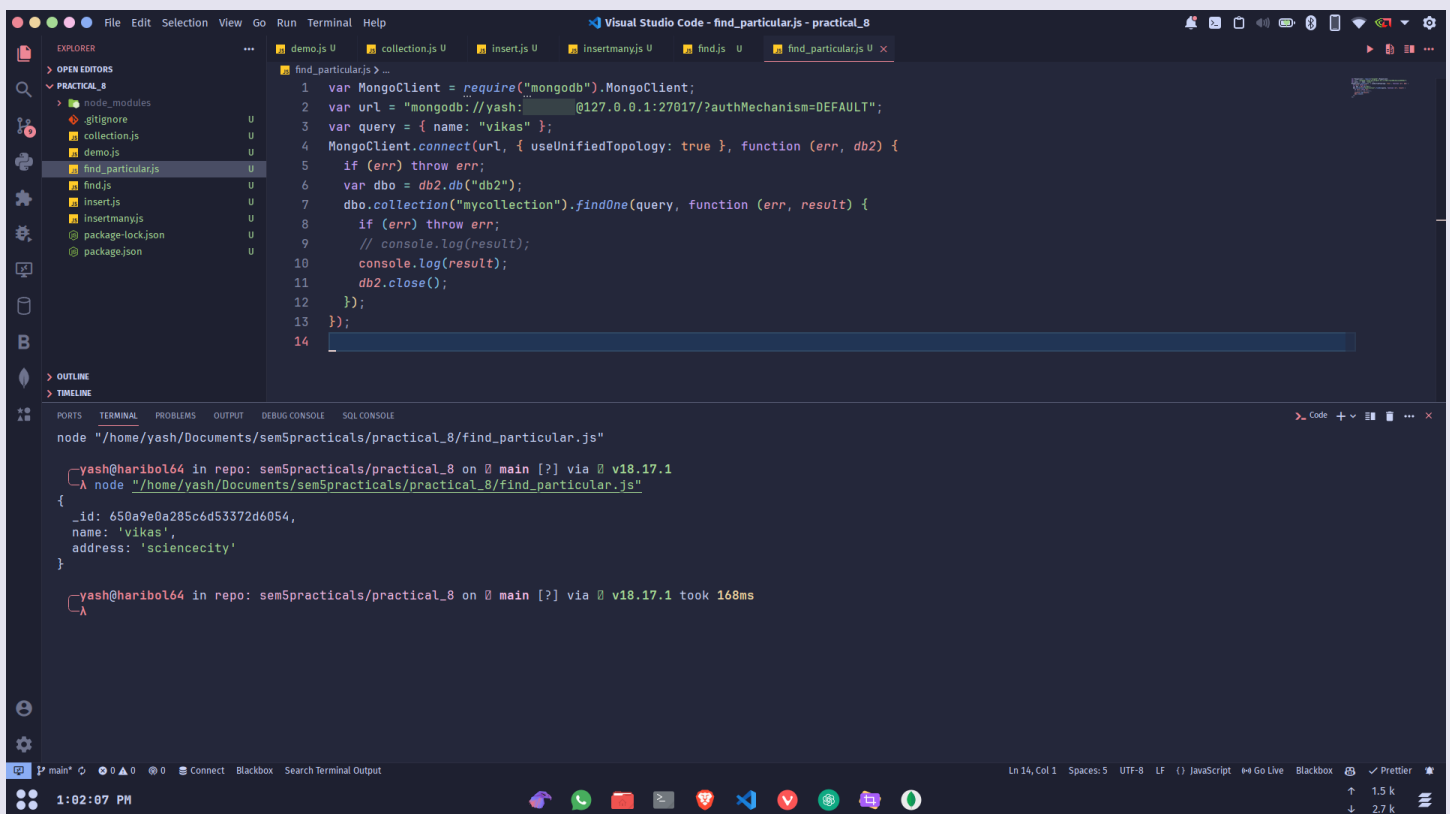
```
[
  {
    _id: 650a9e0a285c6d53372d6054,
    name: 'vikas',
    address: 'sciencecity'
  },
  {
    _id: 650a9e0a285c6d53372d6055,
    name: 'aman',
    address: 'drivein'
  },
  {
    _id: 650a9e0a285c6d53372d6056,
    name: 'mahendra',
    address: 'punjab'
  },
  {
    _id: 650a9e0a285c6d53372d6057,
    name: 'ruchin',
    address: 'lahore'
  },
  {
    _id: 650a9e0a285c6d53372d6058,
    name: 'sushil',
    address: 'gujrat'
  },
  {
    _id: 650a9e0a285c6d53372d6059,
    name: 'vikrant',
    address: 'baroda'
  },
  {
    _id: 650a9e0a285c6d53372d6059,
    name: 'vikrant',
    address: 'baroda'
  }
]
```

Practical 8.3: The admin can filter out the DB.

```
var MongoClient = require("mongodb").MongoClient;
var url = "mongodb://yash@127.0.0.1:27017/?authMechanism=DEFAULT";
var query = { name: "vikas" };
MongoClient.connect(url, { useUnifiedTopology: true }, function (err, db2) {
  if (err) throw err;
  var dbo = db2.db("db2");
  dbo.collection("mycollection").findOne(query, function (err, result) {
    if (err) throw err;
    // console.log(result);
  });
});
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

```
console.log(result);  
  
db2.close();  
  
});  
  
});
```



The screenshot shows the Visual Studio Code editor with a file named `find_particular.js` open. The file contains the following code:

```
1 var MongoClient = require("mongodb").MongoClient;  
2 var url = "mongodb://yash:12345678@127.0.0.1:27017/?authMechanism=DEFAULT";  
3 var query = { name: "vikas" };  
4 MongoClient.connect(url, { useUnifiedTopology: true }, function (err, db2) {  
5   if (err) throw err;  
6   var dbo = db2.db("db2");  
7   dbo.collection("mycollection").findOne(query, function (err, result) {  
8     if (err) throw err;  
9     // console.log(result);  
10    console.log(result);  
11    db2.close();  
12  });  
13 });  
14
```

The terminal output shows the command `node "/home/yash/Documents/sem5practicals/practical_8/find_particular.js"` being executed. The output is:

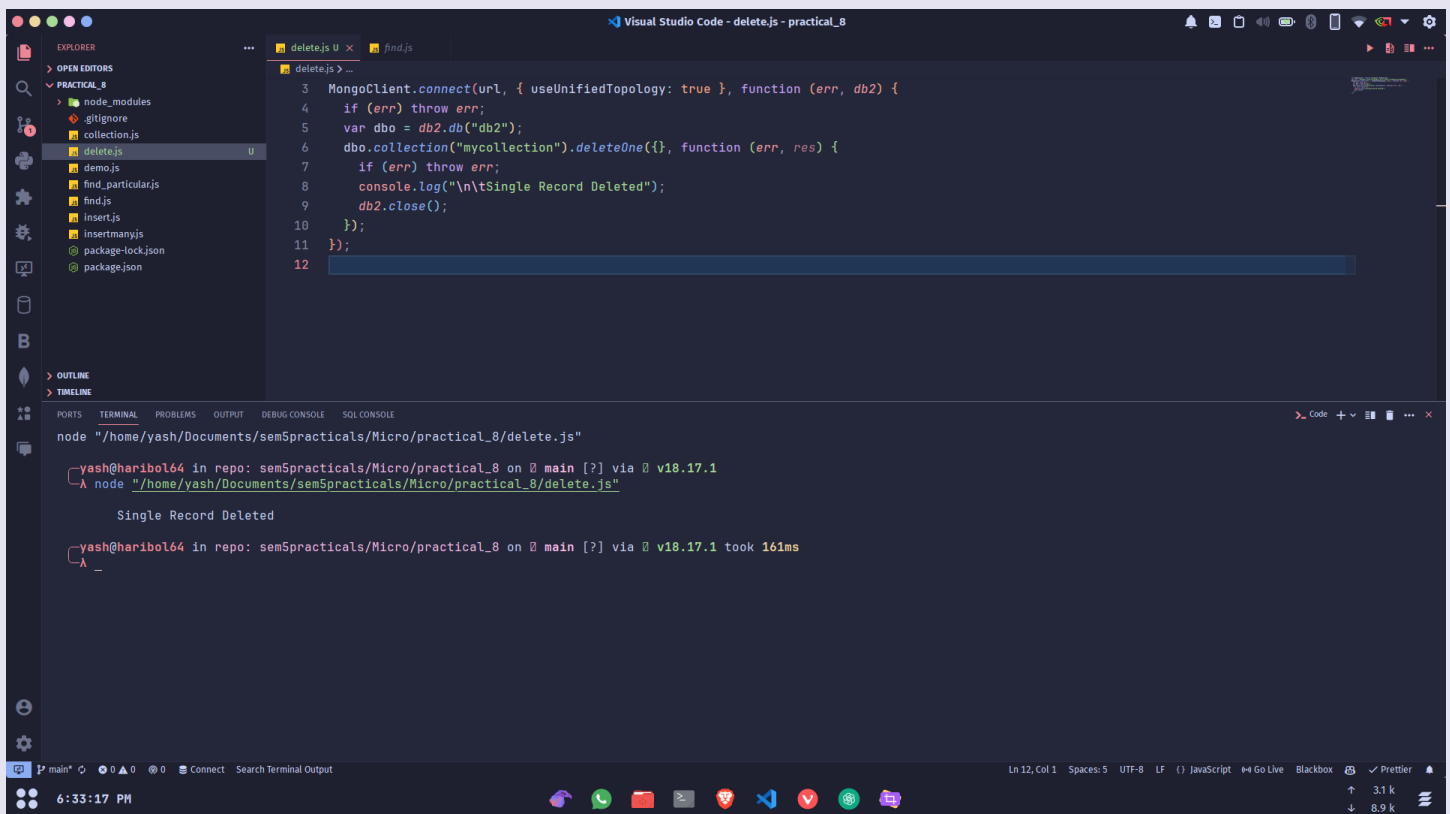
```
yash@haribol64 in repo: sem5practicals/practical_8 on 0 main [?] via 0 v18.17.1  
node "/home/yash/Documents/sem5practicals/practical_8/find_particular.js"  
{  
  _id: 650a9e0a285c6d53372d6054,  
  name: 'vikas',  
  address: 'sciencecity'  
}  
  
yash@haribol64 in repo: sem5practicals/practical_8 on 0 main [?] via 0 v18.17.1 took 168ms
```

Practical 8.4: The admin can delete the record from DB.

```
var MongoClient = require("mongodb").MongoClient;  
  
var url = "mongodb://yash@127.0.0.1:27017/?authMechanism=DEFAULT";  
  
MongoClient.connect(url, { useUnifiedTopology: true }, function (err, db2)  
{  
  
if (err) throw err;
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

```
var dbo = db2.db("db2");  
  
dbo.collection("mycollection").deleteOne({}, function (err, res) {  
  
if (err) throw err;  
  
console.log("\n\tSingle Record Deleted");  
  
db2.close();  
  
});  
  
});
```



The screenshot shows the Visual Studio Code editor with the file `delete.js` open. The code in the editor is as follows:

```
3 MongoClient.connect(url, { useUnifiedTopology: true }, function (err, db2) {  
4   if (err) throw err;  
5   var dbo = db2.db("db2");  
6   dbo.collection("mycollection").deleteOne({}, function (err, res) {  
7     if (err) throw err;  
8     console.log("\n\tSingle Record Deleted");  
9     db2.close();  
10  });  
11  });  
12
```

The terminal output shows the command being executed and the result:

```
node "/home/yash/Documents/sem5practicals/Micro/practical_8/delete.js"  
  
yash@haribol64 in repo: sem5practicals/Micro/practical_8 on  main [?] via  v18.17.1  
λ node "/home/yash/Documents/sem5practicals/Micro/practical_8/delete.js"  
  
Single Record Deleted  
  
yash@haribol64 in repo: sem5practicals/Micro/practical_8 on  main [?] via  v18.17.1 took 161ms  
λ _
```

The status bar at the bottom indicates the file is `delete.js`, line 12, column 1, using UTF-8 encoding, and the editor is configured for JavaScript with the Blackbox theme and Prettier formatter.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

Practical 8.5: Create a database with the name "student". Create a collection named "cba". Perform the following tasks:

- Insert the details of 10 students with the following details: Name, age, address, phone number, and email.
- Filter the students whose age is greater than 16.
- Delete the student record whose phone number contains the digit '6'

```
let MongoClient = require("mongodb").MongoClient;
let url = "mongodb://yash@127.0.0.1:27017/?authMechanism=DEFAULT";
let details = [
  {
    name: "John Doe",
    age: 20,
    address: "123 Main Street",
    phone: "555-123-4597",
    email: "john.doe@example.com",
  },
  {
    name: "Jane Smith",
    age: 15,
    address: "456 Elm Street",
    phone: "555-234-5078",
    email: "jane.smith@example.com",
  },
  {
    name: "Robert Johnson",
```


Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

```
age: 21,  
address: "789 Oak Street",  
phone: "555-345-2789",  
email: "robert.johnson@example.com",  
},  
{  
name: "Emily Davis",  
age: 16,  
address: "101 Pine Street",  
phone: "555-456-7890",  
email: "emily.davis@example.com",  
},  
{  
name: "Michael Wilson",  
age: 19,  
address: "202 Cedar Street",  
phone: "555-567-8901",  
email: "michael.wilson@example.com",  
},  
{  
name: "Sarah Brown",  
age: 14,  
address: "303 Maple Street",  
phone: "555-678-9012",
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

```
email: "sarah.brown@example.com",
},
{
  name: "William Lee",
  age: 21,
  address: "404 Birch Street",
  phone: "555-789-0123",
  email: "william.lee@example.com",
},
{
  name: "Olivia Clark",
  age: 18,
  address: "505 Redwood Street",
  phone: "555-890-1234",
  email: "olivia.clark@example.com",
},
{
  name: "David Turner",
  age: 15,
  address: "606 Spruce Street",
  phone: "555-901-2345",
  email: "david.turner@example.com",
},
{
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

```
name: "Sophia Rodriguez",
age: 24,
address: "707 Pine Avenue",
phone: "555-012-3456",
email: "sophia.rodriguez@example.com",
},
];

MongoClient.connect(url, { useUnifiedTopology: true }, function (err,
client) {
if (err) throw err;
let database = client.db("student");
console.log("\n\tCreated database - student");

database.createCollection("cba", function (err, res) {
if (err) throw err;
console.log("\tCreated collection - cba");

database.collection("cba").insertMany(details, function (err, res) {
if (err) throw err;
console.log(res);
console.log(`\n\tInserted ${res.insertedCount} records`);
});
});
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

```
database.collection("cba").find({ age: { $gt: 16 } }).toArray(function
(err, result) {
if (err) throw err;
console.log("\n\tStudent(s) with age > 16 is/are : \n");
console.log(result);
});

database.collection("cba").deleteMany({ phone: { $regex: /6/ } }, function
(err, result) {
if (err) throw err;
console.log(`\n\tDeleted ${result.deletedCount} student(s) with phone
numbers containing '6'`);
client.close();
});
});
});
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

```

1  let details = [
2
3      {
4          name: "John Doe",
5          age: 20,
6          address: "123 Main Street",
7          phone: "555-123-4597",
8          email: "john.doe@example.com",
9      },
10     {
11         name: "Jane Smith",
12         age: 15,
13         address: "456 Elm Street",
14         phone: "555-234-5078",
15         email: "jane.smith@example.com",
16     },
17     {
18         name: "Robert Johnson",
19         age: 21,
20         address: "789 Oak Street",
21         phone: "555-345-2789",
22         email: "robert.johnson@example.com",
23     },
24     {
25         name: "Emily Davis",
26         age: 16,
27         address: "101 Pine Street",
28         phone: "555-456-7890",
29         email: "emily.davis@example.com",
30     },
31     {
32         name: "Michael Wilson",
33         age: 19,
34         address: "202 Cedar Street",
35         phone: "555-567-8901",
36         email: "michael.wilson@example.com",
37     },
38 ]

```

```

node "/home/yash/Documents/sem5practicals/Micro/practical_8/student.js"

yash@haribol64 in repo: sem5practicals/Micro/practical_8 on 0 main (?) via
v18.17.1
A node "/home/yash/Documents/sem5practicals/Micro/practical_8/student.js"

Created database - student
Created collection - cba

{
  result: { ok: 1, n: 10 },
  ops: [
    {
      name: 'John Doe',
      age: 20,
      address: '123 Main Street',
      phone: '555-123-4597',
      email: 'john.doe@example.com',
      _id: 65105d9dd3a59c912ba99f06
    },
    {
      name: 'Jane Smith',
      age: 15,
      address: '456 Elm Street',
      phone: '555-234-5078',
      email: 'jane.smith@example.com',
      _id: 65105d9dd3a59c912ba99f07
    },
    {
      name: 'Robert Johnson',
      age: 21,
      address: '789 Oak Street',
      phone: '555-345-2789',
      email: 'robert.johnson@example.com',
      _id: 65105d9dd3a59c912ba99f08
    },
    {
      name: 'Emily Davis',
      age: 16,
      address: '101 Pine Street',
      phone: '555-456-7890',
      email: 'emily.davis@example.com',
      _id: 65105d9dd3a59c912ba99f09
    },
    {
      name: 'Michael Wilson',
      age: 19,
      address: '202 Cedar Street',
      phone: '555-567-8901',
      email: 'michael.wilson@example.com',
      _id: 65105d9dd3a59c912ba99f0a
    }
  ]
}

```

```

node "/home/yash/Documents/sem5practicals/Micro/practical_8/student.js"

yash@haribol64 in repo: sem5practicals/Micro/practical_8 on 0 main (?) via
v18.17.1
A node "/home/yash/Documents/sem5practicals/Micro/practical_8/student.js"

Created database - student
Created collection - cba

{
  result: { ok: 1, n: 10 },
  ops: [
    {
      name: 'John Doe',
      age: 20,
      address: '123 Main Street',
      phone: '555-123-4597',
      email: 'john.doe@example.com',
      _id: 65105d9dd3a59c912ba99f06
    },
    {
      name: 'Jane Smith',
      age: 15,
      address: '456 Elm Street',
      phone: '555-234-5078',
      email: 'jane.smith@example.com',
      _id: 65105d9dd3a59c912ba99f07
    },
    {
      name: 'Robert Johnson',
      age: 21,
      address: '789 Oak Street',
      phone: '555-345-2789',
      email: 'robert.johnson@example.com',
      _id: 65105d9dd3a59c912ba99f08
    },
    {
      name: 'Emily Davis',
      age: 16,
      address: '101 Pine Street',
      phone: '555-456-7890',
      email: 'emily.davis@example.com',
      _id: 65105d9dd3a59c912ba99f09
    },
    {
      name: 'Michael Wilson',
      age: 19,
      address: '202 Cedar Street',
      phone: '555-567-8901',
      email: 'michael.wilson@example.com',
      _id: 65105d9dd3a59c912ba99f0a
    }
  ]
}

```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 8

The screenshot shows a Visual Studio Code editor with a file named `student.js` open. The file contains a JavaScript array of student objects. The terminal output shows a message: "Deleted 4 student(s) with phone numbers containing '6'".

```
student.js
1  let details = [
2
3
4      {
5          name: "John Doe",
6          age: 20,
7          address: "123 Main Street",
8          phone: "555-123-4597",
9          email: "john.doe@example.com",
10     },
11     {
12         name: "Jane Smith",
13         age: 15,
14         address: "456 Elm Street",
15         phone: "555-234-5078",
16         email: "jane.smith@example.com",
17     },
18     {
19         name: "Robert Johnson",
20         age: 21,
21         address: "789 Oak Street",
22         phone: "555-345-2789",
23         email: "robert.johnson@example.com",
24     },
25     {
26         name: "Emily Davis",
27         age: 16,
28         address: "101 Pine Street",
29         phone: "555-456-7890",
30         email: "emily.davis@example.com",
31     },
32     {
33         name: "Michael Wilson",
34         age: 19,
35         address: "202 Cedar Street",
36         phone: "555-567-8901",
37         email: "michael.wilson@example.com",
38     }
39 ]
```

```
Deleted 4 student(s) with phone numbers containing '6'
```