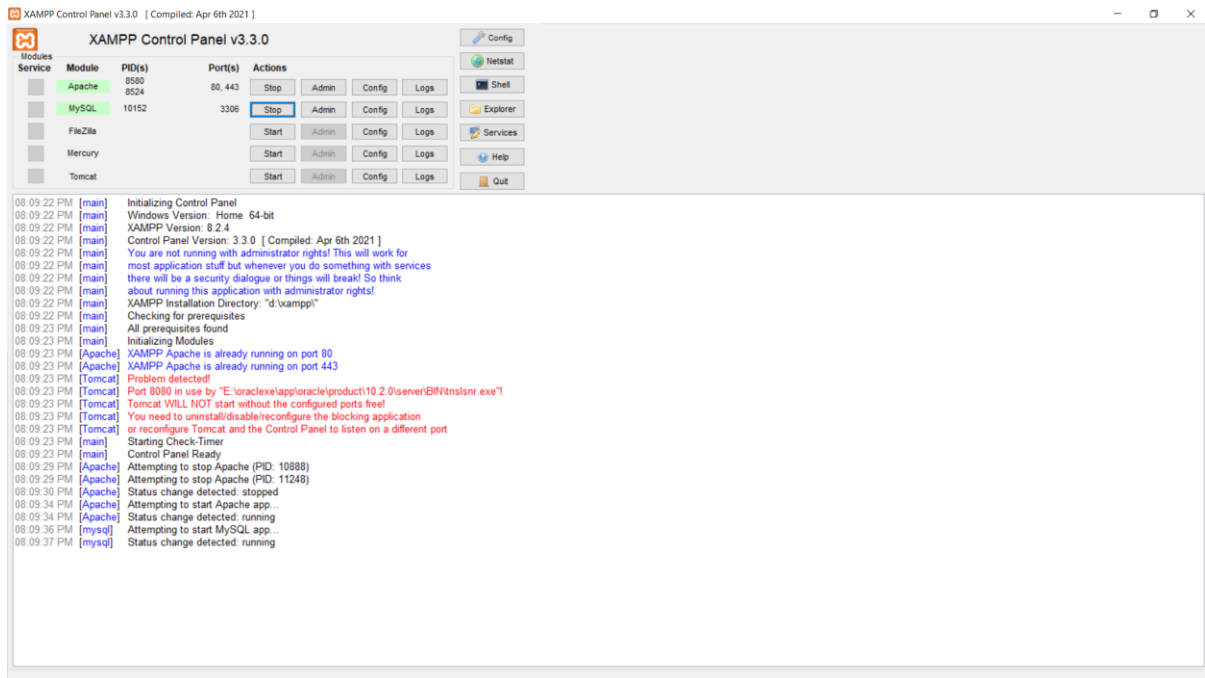


11. Handling SQL Databases from NodeJS

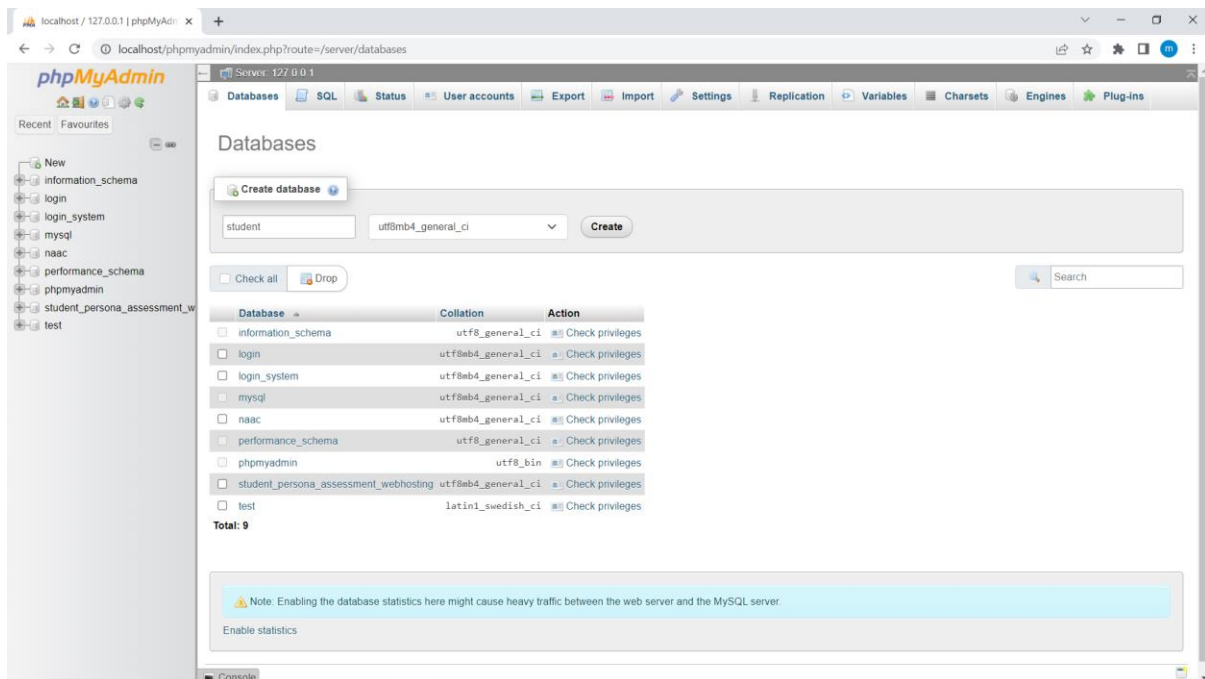
Procedure:

Step 1: Create Database

Open XAMPP Control Panel and start Apache and MySQL.



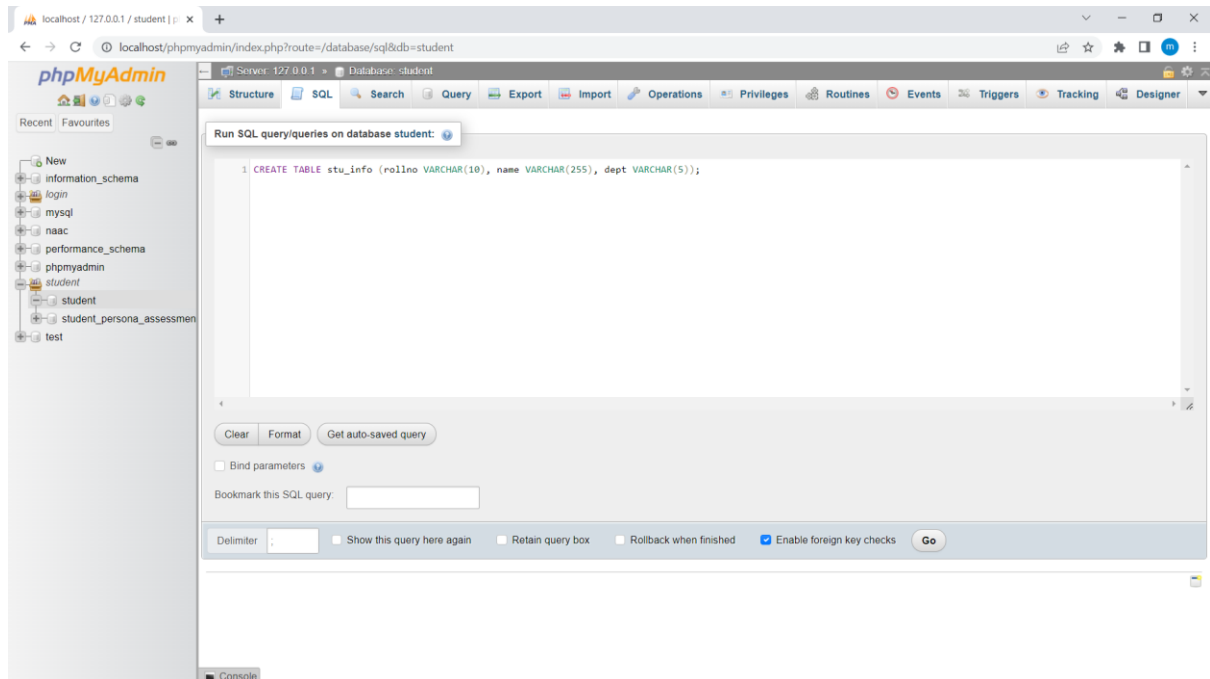
Then Click Admin corresponding to the MySQL Module.



Step 2: Create Table

Type the following query in SQL tab to create MySQL table.

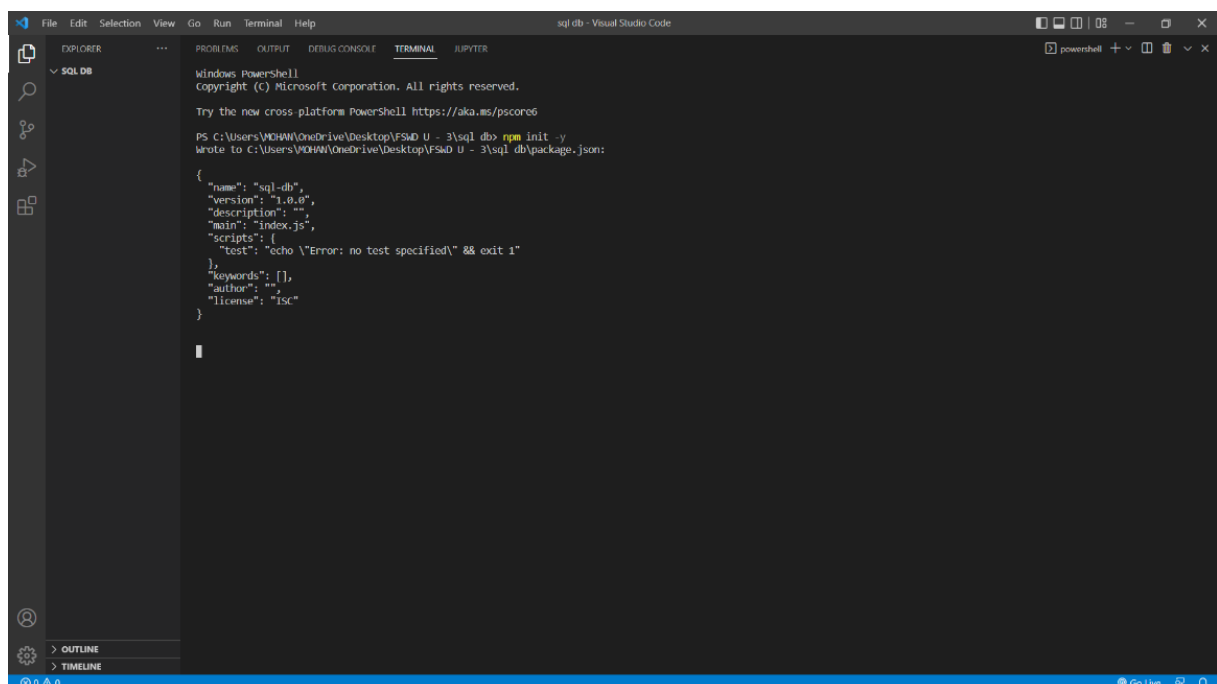
CREATE TABLE stu_info (rollno VARCHAR(10), name VARCHAR(255), dept VARCHAR(5));



Step 3: Setup Project

Create new folder and open in visual studio code, then create NodeJS Project by typing the following command in terminal.

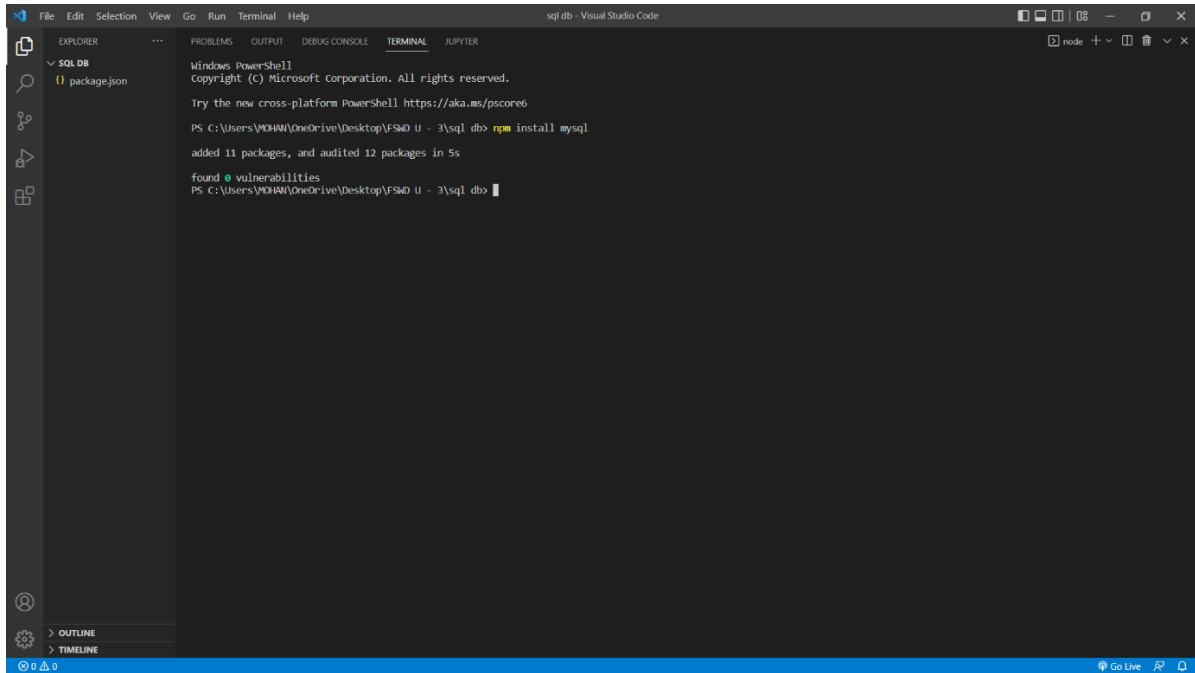
npm init -y



Step 4: Download MySQL Driver

To access a MySQL database with Node.js, you need a MySQL driver. To download and install the "mysql" module, open the Command Terminal and execute the following command.

npm install mysql



Step 5: Insert Data

Create new file named as insert.js and type the following code to insert data in MySQL Database.

```
var mysql = require('mysql');

var con = mysql.createConnection({
  host: "localhost",
  user: "root",
  password: "",
  database: "student"
});

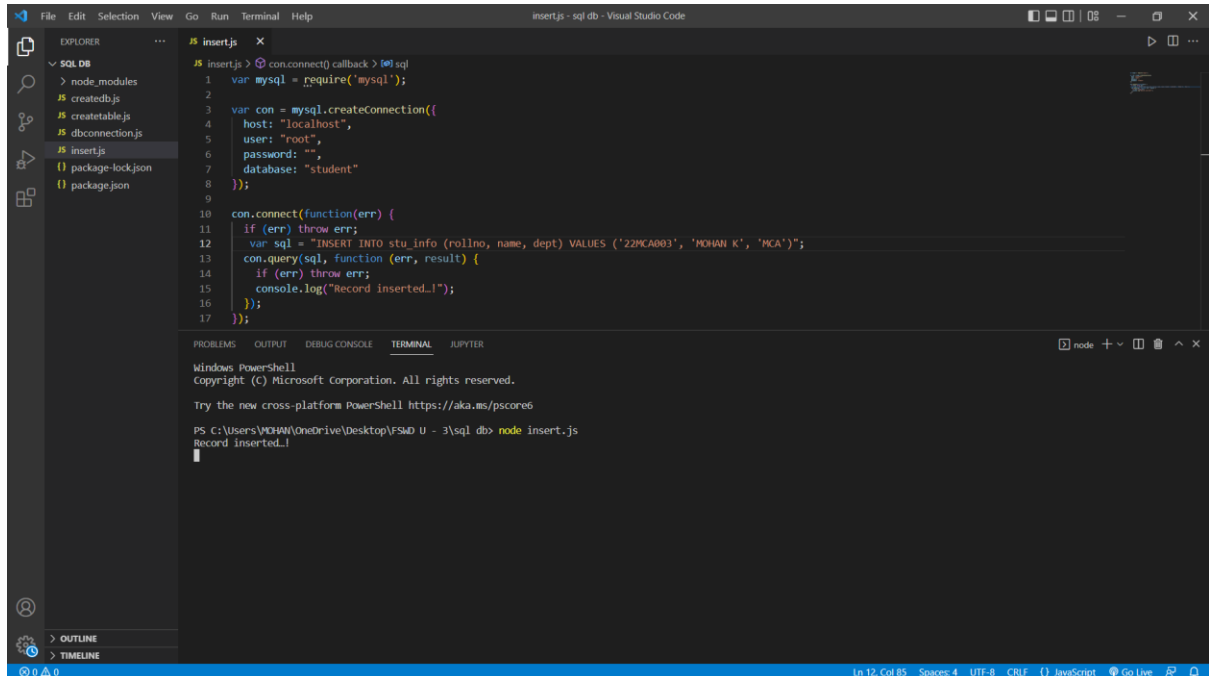
con.connect(function(err) {
  if (err) throw err;
  var sql = "INSERT INTO stu_info (rollno, name, dept) VALUES ('22MCA002', 'AJAY V', 'MCA')";
  con.query(sql, function (err, result) {
    if (err) throw err;
    console.log("Record inserted...!");
  });
});
```

```
});  
});
```

Type the following command in terminal to run the code.

node insert.js

Output:



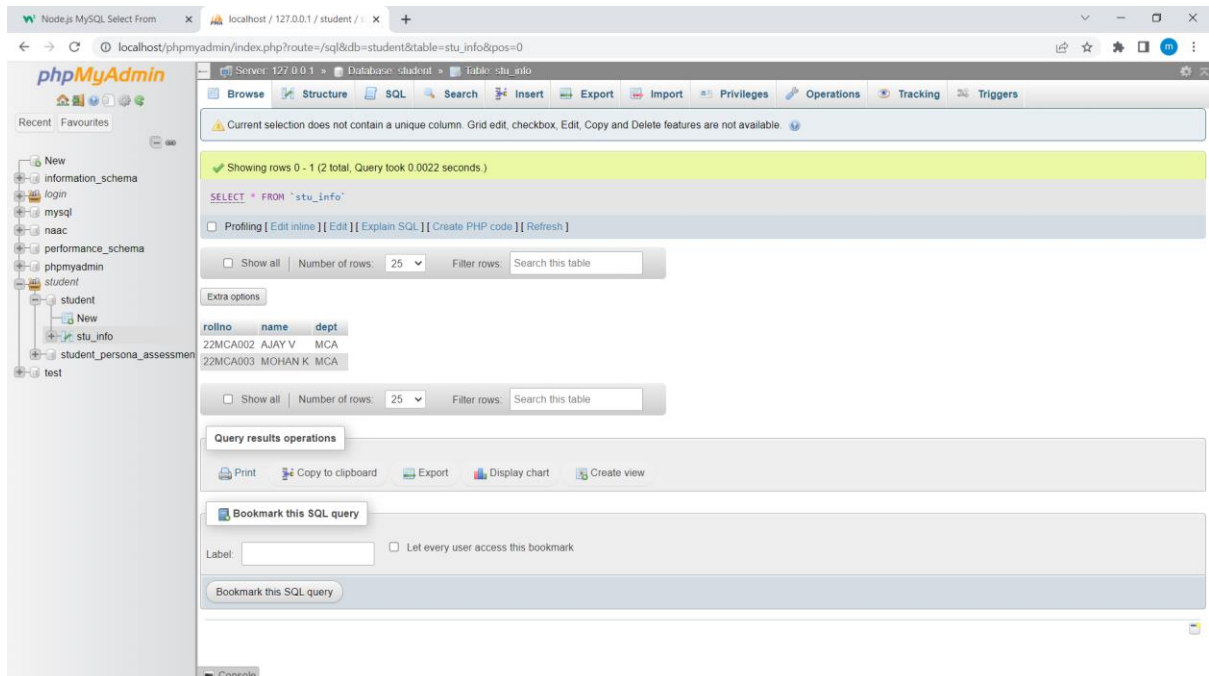
The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows the project structure with files like node_modules, createdb.js, createtable.js, dbconnection.js, insert.js, package-lock.json, and package.json. The main editor shows the content of insert.js, which is a JavaScript file that connects to a MySQL database and inserts data into the stu_info table. The terminal at the bottom shows the command 'node insert.js' being executed, resulting in the output 'Record inserted.'.

```
insert.js  
1 var mysql = require('mysql');  
2  
3 var con = mysql.createConnection({  
4   host: "localhost",  
5   user: "root",  
6   password: "",  
7   database: "student"  
8 });  
9  
10 con.connect(function(err) {  
11   if (err) throw err;  
12   var sql = "INSERT INTO stu_info (rollno, name, dept) VALUES ('22MCA003', 'MOHAN K', 'MCA')";  
13   con.query(sql, function (err, result) {  
14     if (err) throw err;  
15     console.log("Record inserted.");  
16   });  
17 });
```

Windows PowerShell
Copyright (c) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\MOHAN\OneDrive\Desktop\FSMD U - 3\sql db> node insert.js
Record inserted.
|



Step 6: Retrieve data

Create new file named as find.js and type the following code to retrieve data from MySQL Database.

```
var mysql = require('mysql');

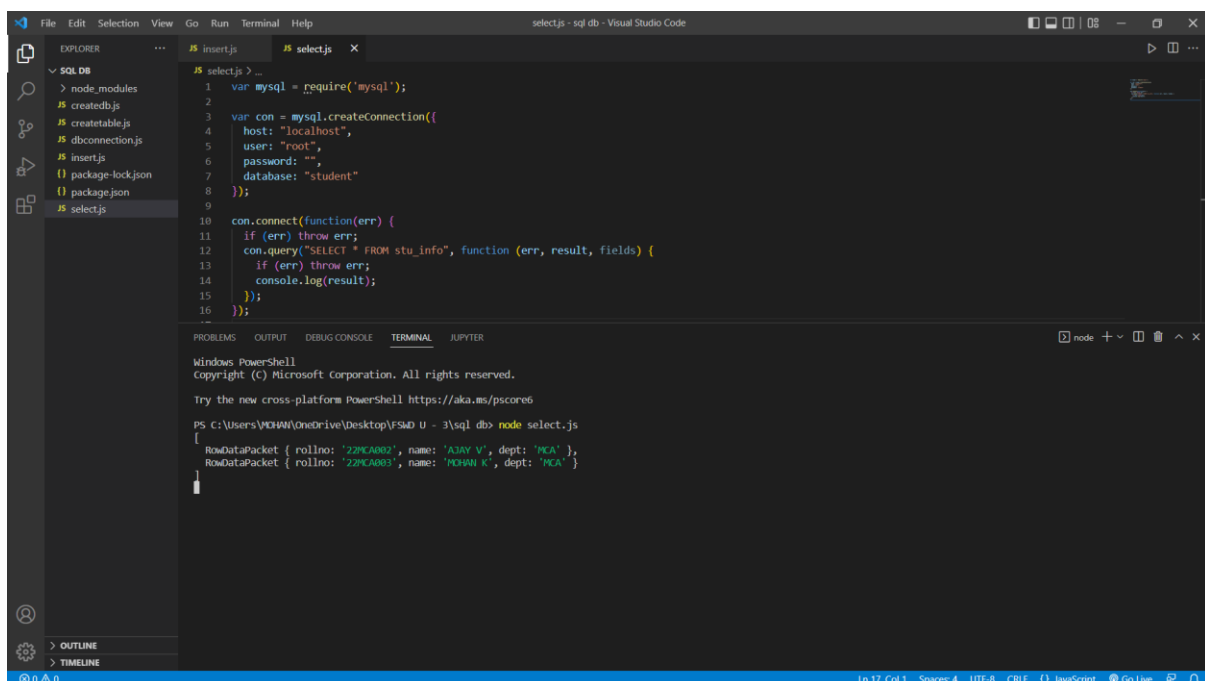
var con = mysql.createConnection({
  host: "localhost",
  user: "root",
  password: "",
  database: "student"
});

con.connect(function(err) {
  if (err) throw err;
  con.query("SELECT * FROM stu_info", function (err, result,
fields) {
    if (err) throw err;
    console.log(result);
  });
});
```

Type the following command in terminal to run the code.

node find.js

Output:

The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar shows a file named 'select.js' selected. The main editor area displays the code for 'select.js', which is identical to the code provided in the previous blocks. Below the editor, the TERMINAL panel is open, showing the output of running 'node select.js'. The terminal output displays two rows of data from the 'stu_info' table: one for 'AJAY V' with rollno '22PCA002' and one for 'POHNI K' with rollno '22PCA003'.

```
1 var mysql = require('mysql');
2
3 var con = mysql.createConnection({
4   host: "localhost",
5   user: "root",
6   password: "",
7   database: "student"
8 });
9
10 con.connect(function(err) {
11   if (err) throw err;
12   con.query("SELECT * FROM stu_info", function (err, result, fields) {
13     if (err) throw err;
14     console.log(result);
15   });
16 });
```

```
Windows PowerShell
Copyright (c) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\POHNI\OneDrive\Desktop\VS\sql db> node select.js
[
  RowDataPacket { rollno: '22PCA002', name: 'AJAY V', dept: 'PCA' },
  RowDataPacket { rollno: '22PCA003', name: 'POHNI K', dept: 'PCA' }
]
```

```
[  
  RowDataPacket { rollno: '22MCA002', name: 'AJAY V', dept: 'MCA' },  
  RowDataPacket { rollno: '22MCA003', name: 'MOHAN K', dept: 'MCA' }  
]
```

Step 7: Update Data

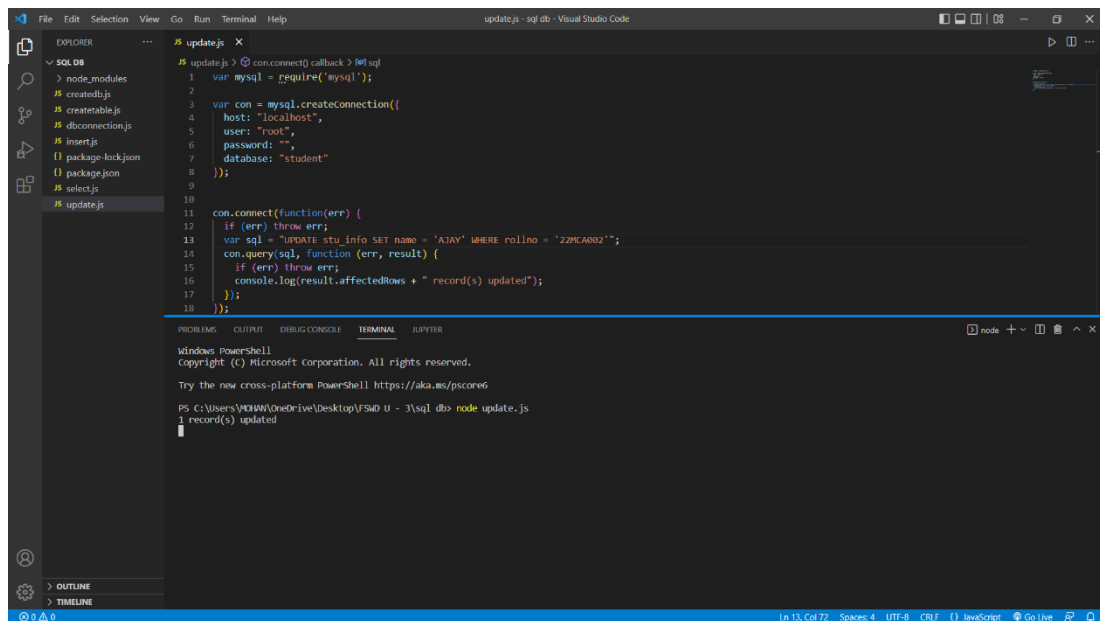
Create new file named as update.js and type the following code to update data in MySQL Database.

```
var mysql = require('mysql');  
  
var con = mysql.createConnection({  
  host: "localhost",  
  user: "root",  
  password: "",  
  database: "student"  
});  
  
con.connect(function(err) {  
  if (err) throw err;  
  var sql = "UPDATE stu_info SET name = 'AJAY' WHERE rollno =  
  '22MCA002'";  
  con.query(sql, function (err, result) {  
    if (err) throw err;  
    console.log(result.affectedRows + " record(s) updated");  
  });  
});
```

Type the following command in terminal to run the code.

node update.js

Output:



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

```
1  update.js > con.connect(callback > test.sql
2  var mysql = require('mysql');
3  var con = mysql.createConnection({
4  host: "localhost",
5  user: "root",
6  password: "",
7  database: "student"
8  });
9
10
11 con.connect(function(err) {
12   if (err) throw err;
13   var sql = "UPDATE stu_info SET name = 'AJAY' WHERE rollno = '22MCA002'";
14   con.query(sql, function (err, result) {
15     if (err) throw err;
16     console.log(result.affectedRows + " record(s) updated");
17   });
18 });
```

The terminal output shows the command `node update.js` being executed, resulting in the output: `1 record(s) updated`.

Step 8: Delete Data

Create new file named as `delete.js` and type the following code to delete data in MySQL Database.

```
var mysql = require('mysql');

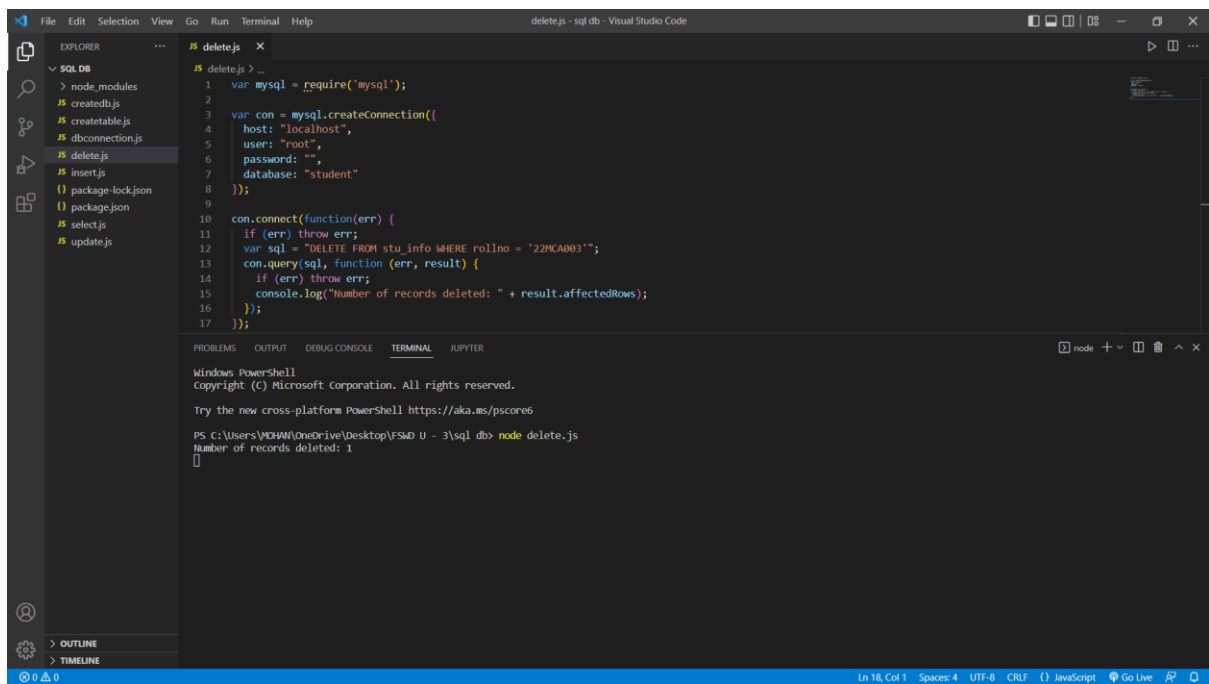
var con = mysql.createConnection({
  host: "localhost",
  user: "root",
  password: "",
  database: "student"
});

con.connect(function(err) {
  if (err) throw err;
  var sql = "DELETE FROM stu_info WHERE rollno = '22MCA003'";
  con.query(sql, function (err, result) {
    if (err) throw err;
    console.log("Number of records deleted: " +
result.affectedRows);
  });
});
```

Type the following command in terminal to run the code.

```
node delete.js
```

Output:



The screenshot displays the Visual Studio Code interface with a file named `delete.js` open in the editor. The file contains a Node.js script that connects to a MySQL database and deletes records. The terminal shows the output of running `node delete.js`, which reports that 1 record was deleted.

```
delete.js - sql db - Visual Studio Code

EXPLORER
  > SQL DB
    > node_modules
    > createdb.js
    > createtable.js
    > dbconnection.js
    > delete.js
    > insert.js
    > package-lock.json
    > package.json
    > select.js
    > update.js

delete.js X
1 var mysql = require('mysql');
2
3 var con = mysql.createConnection({
4   host: "localhost",
5   user: "root",
6   password: "",
7   database: "student"
8 });
9
10 con.connect(function(err) {
11   if (err) throw err;
12   var sql = "DELETE FROM stu_info WHERE rollno = '22MCA003'";
13   con.query(sql, function (err, result) {
14     if (err) throw err;
15     console.log("Number of records deleted: " + result.affectedRows);
16   });
17 });

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
node
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\MOHAMM\OneDrive\Desktop\FSD U - 3\sql db> node delete.js
Number of records deleted: 1
```