

MySQL with Node.js

1. MySql Node.js : Connecting to the database
2. Inserting data
3. Update data
4. Delete data
5. Read data
6. Calling Stored Procedures

Node.js Connection to the MySQL Database:

```
C:\Users\Anil Kumar-DSPS>npm install mysql
npm WARN saveError ENOENT: no such file or directory, open 'C:\Users\Anil Kumar-DSPS\package.json'
npm WARN enoent ENOENT: no such file or directory, open 'C:\Users\Anil Kumar-DSPS\package.json'
npm WARN Anil Kumar-DSPS No description
npm WARN Anil Kumar-DSPS No repository field.
npm WARN Anil Kumar-DSPS No README data
npm WARN Anil Kumar-DSPS No license field.

+ mysql@2.18.1
updated 1 package and audited 17 packages in 1.618s
found 0 vulnerabilities
```

Open the MySQL and execute below commands for changing the privileges

```
create database testDb;
```

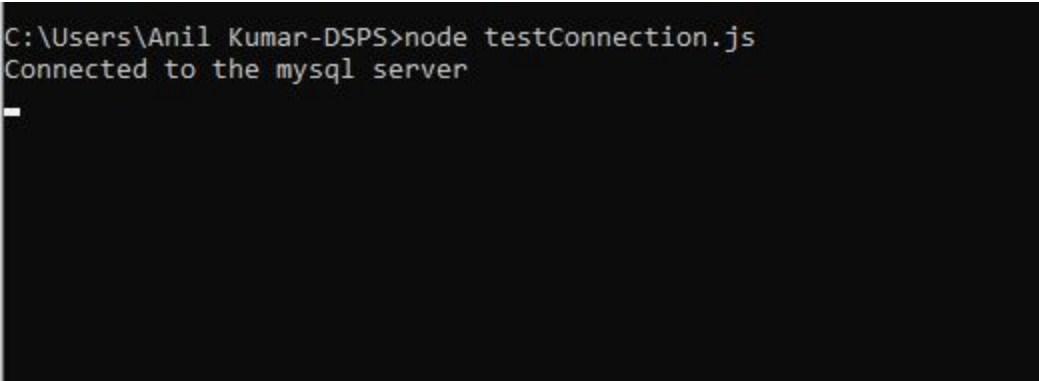
```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'dsps@123';
```

```
flush privileges;
```

Write the below program and save into same folder

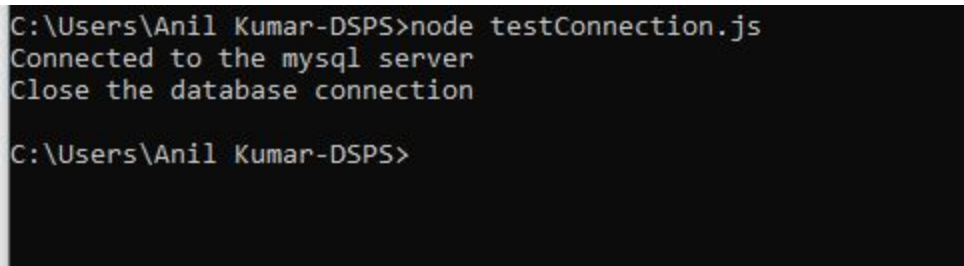
```
let mysql = require('mysql');
let connection = mysql.createConnection({
  host : 'localhost',
  user : 'root',
  password : 'dsps@123',
  database : 'testDb'

});
connection.connect(function(err){
  if(err){
    return console.error('error' + err.message);
  }
  console.log('Connected to the mysql server');
});
```

A terminal window with a black background and white text. The prompt is 'C:\Users\Anil Kumar-DSPS>'. The command 'node testConnection.js' has been executed, and the output is 'Connected to the mysql server'.

```
C:\Users\Anil Kumar-DSPS>node testConnection.js
Connected to the mysql server
```

```
connection.end(function(err){
  if(err){
    return console.error('error' + err.message);
  }
  console.log('Close the database connection');
});
```

A terminal window with a black background and white text. The prompt is 'C:\Users\Anil Kumar-DSPS>'. The command 'node testConnection.js' has been executed, and the output is 'Connected to the mysql server' followed by 'Close the database connection' on the next line.

```
C:\Users\Anil Kumar-DSPS>node testConnection.js
Connected to the mysql server
Close the database connection

C:\Users\Anil Kumar-DSPS>
```

Insert Data to the table:

```
let mysql = require('mysql');
let connection = mysql.createConnection({
  host : 'localhost',
  user : 'root',
  password : 'dsps@123',
  database : 'testDb'

});
let sqlCom = "insert into userData(Username,Password) values('Ajay','ajay')";
connection.query(sqlCom);
connection.end();
```

```
let mysql = require('mysql');
let connection = mysql.createConnection({
  host : 'localhost',
  user : 'root',
  password : 'dsps@123',
  database : 'testDb'

});
let sqlCom = "insert into userData(Username,Password) values(?,?)";
let vals = ['anil','anil'];
connection.query(sqlCom,vals);
connection.end();
```

```
C:\Users\Anil Kumar-DSPS>node insertRecord.js
```

Let's make the Connection as Local Module



conn - Notepad

File Edit Format View Help

```
let conn = {  
    host : 'localhost',  
    user : 'root',  
    password : 'dsps@123',  
    database : 'testDb'  
};  
module.exports = conn;
```

```
let mysql = require('mysql');  
let connect = require('./conn.js');  
let connection = mysql.createConnection(connect);  
let sqlCom = "Insert into userData(Username,Password) values('Gitam','Gitam');";  
connection.query(sqlCom,(err,result) => {  
    if(err) {  
        return console.error(err.message);  
    }  
    console.log('Record is inserted');  
});  
connection.end();
```

Insert the Multiple records in a single Insert statement

```
let mysql = require('mysql');  
let connect = require('./conn.js');  
let connection = mysql.createConnection(connect);  
let sqlComm = "insert into userData(Username,Password) values(?,?);";  
let vals = [['Dinesh','Dinesh-dsps'],['Arvind','Arvind-dsps'],['Harsha','Harsha-dsps']];  
  
connection.query(sqlComm,vals,(err,result) =>{  
    if(err){  
        return console.error(err.message);  
    }  
    console.log('records are inserted');  
});  
connection.end();
```

