**Install mysql package with npm as below**

npm install mysql

create a database.js file in the project folder and test the below code. We can give any file name

We have 2 ways to create connection using below methods

1. createConnection
   1. We have to manage the connections from outside means we have to do the connections and we have to close the connection
2. createPool - We can create a poll of a connection not a single connection and we can execute the query parallelly

const mysql = require('mysql');

// Create connection

const connection = mysql.createConnection({

host: 'localhost',

user: 'username',

password: 'password',

database: 'database\_name'

});

// Connect

connection.connect();

// Create table

connection.query('CREATE TABLE IF NOT EXISTS users (id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(255), email VARCHAR(255))', (err, result) => {

if (err) throw err;

console.log('Table created successfully');

});

// Insert data

const userData = { name: 'John', email: 'john@example.com' };

connection.query('INSERT INTO users SET ?', userData, (err, result) => {

if (err) throw err;

console.log('Data inserted successfully');

});

// Update data

connection.query('UPDATE users SET email = ? WHERE name = ?', ['john.doe@example.com', 'John'], (err, result) => {

if (err) throw err;

console.log('Data updated successfully');

});

// Delete table

connection.query('DROP TABLE IF EXISTS users', (err, result) => {

if (err) throw err;

console.log('Table deleted successfully');

});

//Bulk insert

const bulkInsertQuery = `

INSERT INTO users (username, email) VALUES ?

`;

const values = [

['Alice', 'alice@example.com'],

['Bob', 'bob@example.com'],

['Charlie', 'charlie@example.com']

];

connection.query(bulkInsertQuery, [values], (err, results) => {

if (err) throw err;

console.log('Bulk data inserted successfully');

});

//Subqueries

const subQuery = `

SELECT \* FROM users WHERE email IN (SELECT email FROM other\_table WHERE condition)

`;

connection.query(subQuery, (err, results) => {

if (err) throw err;

console.log('Subquery executed:', results);

});

//Inner queries

const innerQuery = `

SELECT \* FROM users WHERE id IN (SELECT MAX(id) FROM users)

`;

connection.query(innerQuery, (err, results) => {

if (err) throw err;

console.log('Inner query executed:', results);

});

// Close connection

connection.end();