# Department of Computing

**Muhammad Nouman**

**BSCS 7B 218663**

**CS-213: Advanced Programming**

**Class: BSCS 7AB**

# Lab 5: Node.js MySQL

**Date: 3rd October, 2019**

**Time: 10:00-01:00pm & 02:00-05:00pm**

# Instructor: Dr. Sidra Sultana

**Lab Engineer: Ms. Ayesha Asif**

# 

# Lab 5: Node.js MySQL

**Introduction**

Node.js can be used in database applications. One of the most popular databases is MySQL.client.

**Objectives**

This lab will get you familiar with the node.js mysql environment.

**Tools/Software Requirement**

Node.js, Notepad

**Description**

## **MySQL Database**

To be able to experiment with the code examples, you should have MySQL installed on your computer.

You can download a free MySQL database at <https://www.mysql.com/downloads/>.

**Install MySQL Driver**

Once you have MySQL up and running on your computer, you can access it by using Node.js.

To access a MySQL database with Node.js, you need a MySQL driver. This tutorial will use the "mysql" module, downloaded from NPM.

To download and install the "mysql" module, open the Command Terminal and execute the following:

C:\Users\Your Name>npm install mysql

Now you have downloaded and installed a mysql database driver.

Node.js can use this module to manipulate the MySQL database:

var mysql = require('mysql');

**Helping Material**

Slides of Lecture 6

<https://www.w3schools.com/nodejs/nodejs_mysql.asp>

**Lab Tasks**

**Task 1:** Start by creating a connection to the database. Use the username and password from your MySQL database.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql'); var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '' }); conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!"); });  Task Output Screenshot: |

**Task 2:** Use SQL statements to read from (or write to) a MySQL database. The query method takes an sql statements as a parameter and returns the result.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "SELECT \* FROM `customer`";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result);   });  });  Task Output Screenshot: |

**Task 3:** Create a database named "mydb". Save the code in a file called "demo\_create\_db.js" and run the file.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: ''  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  conn.query("CREATE DATABASE mydb", function (err, result) {  if(err) throw err;  *console*.log("Database Created!");   });  });  Task Output Screenshot: |

**Task 4:** Create a table named "customers". Save the code above in a file called "demo\_create\_table.js" and run the file

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  conn.query("CREATE TABLE customers (name VARCHAR(255), address VARCHAR(255))", function (err, result) {  if(err) throw err;  *console*.log("table Created!");   });  });  Task Output Screenshot: |

**Task 5:** Create primary key when creating the table. If the table already exists, use the ALTER TABLE keyword.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  conn.query("ALTER TABLE customers ADD COLUMN id INT AUTO\_INCREMENT PRIMARY KEY", function (err, result) {  if(err) throw err;  *console*.log("table altered!");   });  });  Task Output Screenshot: |

**Task 6:** Insert a record in the "customers" table. Save the code above in a file called "demo\_db\_insert.js", and run the file.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  conn.query("INSERT INTO customers(id, name, address) VALUES (NULL, 'john nash', 'nust attar 1')", function (err, result) {  if(err) throw err;  *console*.log("record inserted");   });  });  Task Output Screenshot: |

**Task 7:** Fill the "customers" table with multiple data. Save the code above in a file called "demo\_db\_insert\_multple.js", and run the file. Return the number of affected rows

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "INSERT INTO `customers` (`name`, `address`, `id`) VALUES ('test1', 'test1', NULL), ('test2', 'test2', NULL);";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log("Affected rows:"+result.affectedRows);   });  });  Task Output Screenshot: |

**Task 8:** Select all records from the "customers" table, and display the result object. Save the code above in a file called "demo\_db\_select.js" and run the file

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "SELECT \* FROM `customers`";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result);   });  });  Task Output Screenshot: |

**Task 9:** Select name and address from the "customers" table, and display the return object. Save the code above in a file called "demo\_db\_select2.js" and run the file

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "SELECT name, address FROM `customers`";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result);   });  });  Task Output Screenshot: |

**Task 10:** Select all records from the "customers" table, and display the fields object. Save the code above in a file called "demo\_db\_select\_fields.js" and run the file

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "SELECT \* FROM `customers`";  conn.query(sql , function (err, result, fields) {  if(err) throw err;  *console*.log(fields);   });  });  Task Output Screenshot: |

**Task11:** Select record(s) with the address "Park Lane 38". Save the code above in a file called "demo\_db\_where.js" and run the file

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "SELECT \* FROM `customers` WHERE address LIKE 'Park Lane 38'";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result);  });  });  Task Output Screenshot: |

**Task 12:** Select records where the address starts with the letter 'S'. Save the code above in a file called "demo\_db\_where\_s.js" and run the file.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "SELECT \* FROM `customers` WHERE address LIKE 'S%'";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result);   });  });  Task Output Screenshot: |

**Task 13:** Sort the result alphabetically by name. Save the code above in a file called "demo\_db\_orderby.js" and run the file

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "SELECT \* FROM `customers`ORDER BY name ASC";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result);   });  });  Task Output Screenshot: |

**Task 14:** Delete any record with the address "Mountain 21". Save the code above in a file called "demo\_db\_delete.js" and run the file.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "DELETE FROM `customers`WHERE address = 'Mountain 21'";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result.affectedRows);   });  });  Task Output Screenshot: |

**Task 15:** Delete the table "customers". Save the code above in a file called "demo\_db\_drop\_table\_if.js" and run the file.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "DROP TABLE `customers`";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result);   });  });  Task Output Screenshot: |

**Task 16:** Overwrite the address column from "Valley 345" to "Canyon 123". Save the code above in a file called "demo\_db\_update.js" and run the file

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "UPDATE `customers` SET address = 'VCanyon 123' WHERE address = 'Valley 345'";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result.affectedRows);   });  });  Task Output Screenshot: |

**Task 17:** Select the 5 first records in the "customers" table. Save the code above in a file called "demo\_db\_limit.js" and run the file. Now Start from position 3, and return the next 5 records.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'mydb'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "SELECT \* FROM `customers` LIMIT 5, 3";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result);   });  });  Task Output Screenshot: |

**Task 18:** Practice the Join operations on different tables.

|  |
| --- |
| Solution |
| Task Code:  var mysql = require('mysql');    var conn = mysql.createConnection({  host: 'localhost',  user: 'root',  pass: '',  database: 'prestashop'  });   conn.connect(function (err) {  if(err) throw err;  *console*.log("Connected!");  var sql = "SELECT \* FROM `ps\_address` JOIN `ps\_address\_format` using(id\_country)";  conn.query(sql , function (err, result) {  if(err) throw err;  *console*.log(result);   });  });  Task Output Screenshot: |

### Deliverables

Compile a single word document by filling in the solution part and submit this Word file on LMS. This lab grading policy is as follows: The lab is graded between 0 to 10 marks. The submitted solution can get a maximum of 5 marks. At the end of each lab or in the next lab, there will be a viva related to the tasks. The viva has a weightage of 5 marks. Insert the solution/answer in this document. You must show the implementation of the tasks in the designing tool, along with your complete Word document to get your work graded. You must also submit this Word document on the LMS. In case of any problems with submissions on LMS, submit your Lab assignments by emailing it to Ms. Ayesha Asif: [ayesha.asif@seecs.edu.pk](mailto:ayesha.asif@seecs.edu.pk).