Unit – 6 Node JS

Working with database

introduction

- Nodejs can be used with any type of database.
- Nodejs can be use with mysql, oracle, mariadb, mongodb etc.
- Which database should be used with node depends upon requirement of project.
- For simple web application nodejs is used with MYSQL.

Working with mysql & nodejs

- 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.
- To download and install the "mysql" module, open the Command Terminal and execute the following:
- npm install mysql
- Now let us see some examples.s

connecting with NODE

```
var mysql = require('mysql');
var con = mysql.createConnection({
  host: "localhost",
  user: "root",
  port : 3308,
  password: "",
  database: "node"
});
con.connect(function(error) {
  if (error)
    console.log(error)
  else
  {
    console.log("connection established.....");
  }
});
module.exports.con = con;
// con.end();
```

insert record

```
var connection = require("./connection.js")
var sql = "insert into category (title,photo) values ('shampoo','shampoo.jpg')";
connection.con.query(sql, function (error, result) {
    if (error)
        console.log(error)
    else
    {
        console.log("Category Added " + result.insertId);
        console.log("no of rows added " + result.affectedRows);
        console.log(result);
    }
  });
connection.con.end();
```

delete record

```
var connection = require("./connection.js")
var sql = "delete from category where id=10";
connection.con.query(sql, function (error, result) {
    if (error)
        console.log(error)
    else
    {
        console.log("no of rows deleted " + result.affectedRows);
    }
  });
connection.con.end();
```

update record

```
var connection = require("./connection.js")
var sql = "update category set title='shaving cream', photo='shaving_cream.jpg' where id=9";
connection.con.query(sql, function (error, result) {
    if (error)
        console.log(error)
    else
    {
        console.log("no of rows updated " + result.affectedRows);
    }
  });
connection.con.end();
```

select record

```
var connection = require("./connection.js")
var sql = "select * from category order by title";
connection.con.query(sql, function (error, result,fields) {
    if (error)
        console.log(error);
    else
    {
        console.log(fields); //fields is Array that has information about fields fetched in result

        size = result.length; //length property return size of the list
        for(let index=0;index<size;index=index+1)
        {
            console.log(result[index].id," ",result[index].title," ",result[index].photo);
        }
    }
  });
connection.con.end();
```

```
const express = require("express");
const mysql = require("./connection");
const app = express();
// parse requests of content-type - application/json
app.use(express.json());
// parse requests of content-type - application/x-www-form-urlencoded
app.use(express.urlencoded({ extended: true }));
// simple route
app.get("/", (request, response) => {
  response.json({ message: "ready to send response " });
});
app.get("/categories", function (request, response) {
  var sql = "select * from category order by title";
  mysql.con.query(sql, function (error, result, fields) {
    if (error)
    {
      response.json({ error: "error occured" });
    }
    else {
      var output = JSON.parse(JSON.stringify(result));
      response.send(output);
    }
    mysql.con.end();
  });
});
```

```
app.get("/categories/:id", function (request, response) {
  var sql = "select * from category where id=" + request.params.id;
  mysql.con.query(sql, function (error, result, fields) {
    if (error)
    {
      response.json({ error: "error occured" });
    }
    else {
      var output = JSON.parse(JSON.stringify(result));
      response.send(output);
    }
    mysql.con.end();
  });
});
app.delete("/category/:id", function (request, response) {
  var sql = "delete from category where id=" + request.params.id;
  mysql.con.query(sql, function (error, result, fields) {
    if (error)
    {
      response.json({ error: "error occured" });
    }
    else {
      response.json({ message: "category deleted successfully" });
    }
    mysql.con.end();
  });
});
```

```
app.get("/category/:title/:photo", function (request, response) {
  var title = request.params.title;
  var photo = request.params.photo;
  var sql = `insert into category (title,photo) values('${title}','${photo}')`;
  mysql.con.query(sql, function (error, result, fields) {
    if (error)
    {
      response.json({ error: "error occured" });
    }
    else
    {
      response.json({ message: "category inserted successfully" });
    }
    mysql.con.end();
  });
});
```

```
app.get("/category/:title/:photo/:id", function (request, response) {
  var title = request.params.title;
  var photo = request.params.photo;
  var id = request.params.id;
  var sql = `update category set title='${title}' , photo='${photo}' where id=${id} `;
  mysql.con.query(sql, function (error, result, fields) {
    if (error)
    {
      response.json({ error: error.message });
    }
    else
    {
      response.json({ message: "category updated successfully" });
    }
    mysql.con.end();
  });
});
// set port, listen for requests
const PORT = 5000;
app.listen(PORT, function (error) {
  console.log(`Server is running on port ${PORT}.`);
});
```