Name: Sudharshan R

Reg : 185001173

Class : CSE-C

Exercise8: Programs using Node.js

AIM:

a. Write a Node.js program that reads all the greetings from the file greetings.txt, asks the user "What is your name?", then prints a random greeting followed by the given name. Make sure to check for the case where the file doesn't exist! For example, if the greeting is "Hey", then the program will print "Hey, Joe" to the console, then pick some other greeting and do the same until finished. Use Non-blocking I/O.

Code:

```
var fs=require('fs');
const readline = require("readline");
const rl = readline.createInterface({
    input: process.stdin,
    output: process.stdout
});
rl.question("What is your name ? ", function(name) {
    fs.readFile('../input.txt', function (err, data) {
        if (err) return console.log("File Not Found");
        data+='';
        var x=data.split(/\r?\n/)
        var greet = x[Math.floor(Math.random()*x.length)]
        console.log(greet,name);
    });
    rl.close();
});
```

Output:

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

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

PS F:\SEM6\IPlab\NODEJS\assignment8> cd A
PS F:\SEM6\IPlab\NODEJS\assignment8\A> node main.js
What is your name ? Sudharshan
What's up Sudharshan
PS F:\SEM6\IPlab\NODEJS\assignment8\A> node main.js
What is your name ? Sudhan
What's up Sudhan
PS F:\SEM6\IPlab\NODEJS\assignment8\A> node main.js
What is your name ? jack
Hello jack
PS F:\SEM6\IPlab\NODEJS\assignment8\A> node main.js
What is your name ? Mary
Hello Mary
PS F:\SEM6\IPlab\NODEJS\assignment8\A>
```

AIM:

b) Write a Node.js program that reads all the greetings as before. When all the greetings are loaded, it creates a server listening on port number 8080. On request, it checks for whether there is a name value in the query string. If there isn't, the value of query.name will be undefined. In other words, if you access http://localhost:8080/?name=Mike, then your browser should just display something like "Hello, Mike" when the page loads

Code:

```
var fs=require('fs');
var http = require("http");
var url = require('url');
http.createServer(function (request, response) {
    const queryObject = url.parse(request.url,true).query;
    response.writeHead(200, {'Content-Type': 'text/plain'});
    fs.readFile('../input.txt', function (err, data) {
        if (err) return console.log("File Not Found");
        data+='';
        var x=data.split(/\r?\n/)
        var greet = x[Math.floor(Math.random()*x.length)]
        response.write(greet);
        response.write(" ");
        response.end(queryObject['name']);
    });
}).listen(8080);
```

Output:

AIM:

c. Create a web server using node.js which listens for clients request. Once the client request the server, the server returns a web page which contains a list of books and its details in table format.

CODE:

Server/main.js:

```
var http = require('http');
var fs = require('fs');
var url = require('url');
http.createServer( function (request, response) {
    var pathname = url.parse(request.url).pathname;
    console.log("Request for " + pathname + " received.");
    fs.readFile('.'+pathname, function (err, data) {
        if (err) {
            console.log(err);
            response.writeHead(404, {'Content-Type': 'text/html'});
        }
        else {
            response.writeHead(200, {'Content-Type': 'text/html'});
            response.write(data.toString());
        }
        response.end();
    });
}).listen(8080);
console.log('Server running at http://127.0.0.1:8080/');
```

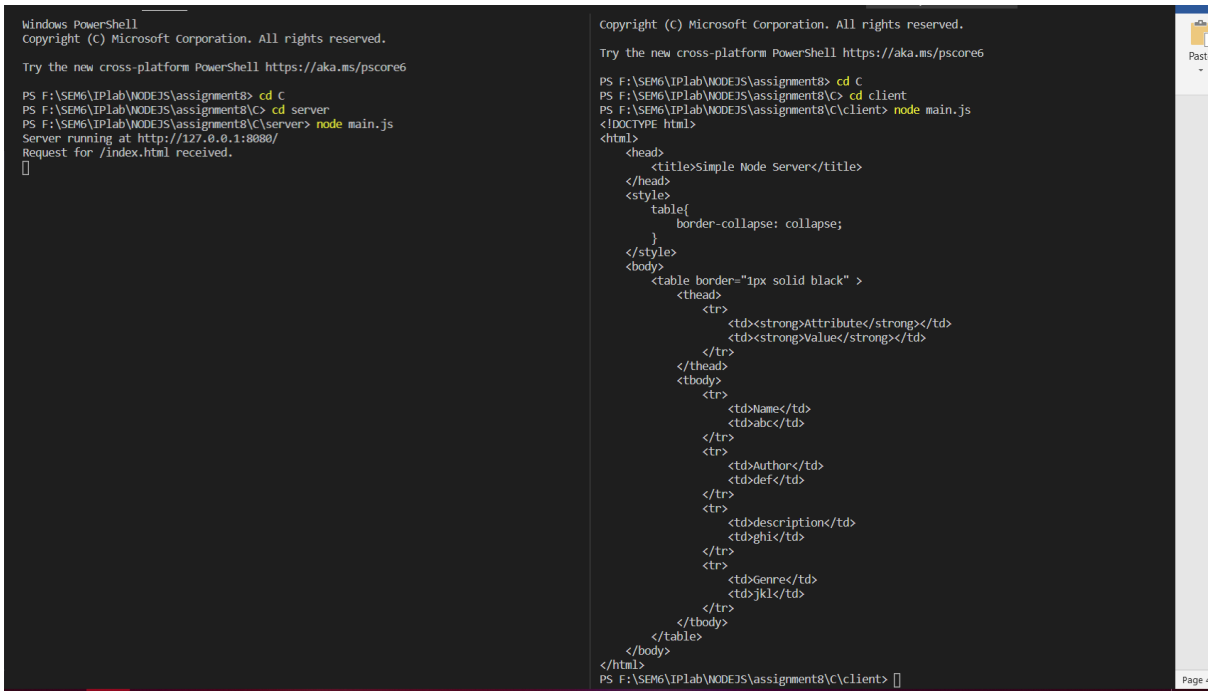
Client/main.js

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

var options = { host: 'localhost', port: '8080', path: '/index.html' };

var callback = function(response) {

    var body = '';
    response.on('data', function(data) {
        body += data;
    });
    response.on('end', function() {
        console.log(body);
    });
}
var req = http.request(options, callback);
req.end();
```

Index.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Simple Node Server</title>
    </head>
    <style>
        table{
            border-collapse: collapse;
        }
    </style>
    <body>
        <table border="1px solid black" >
            <thead>
                <tr>
                    <td><strong>Attribute</strong></td>
                    <td><strong>Value</strong></td>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td>Name</td>
                    <td>abc</td>
                </tr>
                <tr>
                    <td>Author</td>
                    <td>def</td>
                </tr>
                <tr>
                    <td>description</td>
                    <td>ghi</td>
                </tr>
                <tr>
                    <td>Genre</td>
                    <td>jkl</td>
                </tr>
            </tbody>
        </table>
    </body>
</html>
```

OUTPUT:



Attribute	Value
Name	abc
Author	def
description	ghi
Genre	jkl

AIM:

d. Create a DB with the following details using Mongodb:

Database Name: Patient_Details

Table Schema: Name, age, ID, gender, address, marital status, Date of Visit

Write a node.js program to do the following operations:

Add, Delete, Update, Search.

CODE:

Main.js

```
var MongoClient = require('mongodb').MongoClient;
var url = "mongodb://localhost:27017/Patient_Details";

MongoClient.connect(url, {useUnifiedTopology: true }, function(err, db) {
  if (err) throw err;
  console.log("Database Connected");
  var dbObject = db.db("Patient_Details");
  var myobj = {
                Name:'murali',
                age:15,
                ID:126,
                gender:'Male',
                address:'Erode',
                marital_status:'single',
                DateOfVisit:Date()
            };
    dbObject.collection("patients").insertOne(myobj, function(err, res) {
        if (err) throw err;
        console.log("inserted");
        dbObject.collection('patients').find().toArray(function(err,res){
            if(err) throw err;
            console.log(res);
            dbObject.collection('patients').deleteOne({Name:'murali'},function
(err,res){
                if(err) throw err
                console.log('Deleted murali');
                dbObject.collection('patients').find().toArray(function(err,re
s){
                    if(err) throw err;
                    console.log(res);
                    var upd_url = { Name:"Raj" };
                    var upd_values = { $set: {Name: "Raj Kumar", address: "Che
nnai" } };
```

```
                    dbObject.collection("patients").updateOne(upd_url, upd_val
ues, function(err, res) {
                        if (err) throw err;
                        console.log("updated Raj");
                        dbObject.collection('patients').find().toArray(functio
n(err,res){
                            if(err) throw err;
                            console.log(res);
                            db.close();
                        });
                    });
                });
            });
        });
    });
});
```

MongoDB queries

```
use Patient_Details

db.createCollection('patients')

db.patients.insertOne(
    {
        Name:'Raj',
        age:25,
        ID:123,
        gender:'Male',
        address:'Coimbatore',
        marital_status:'single',
        DateOfVisit:Date()
    }
)
```

OUTPUT:

```
Database Connected
inserted
[
  {
    _id: 608e5d2af10ecb0f7ee2427b,
    Name: 'Raj',
    age: 25,
    ID: 123,
    gender: 'Male',
    address: 'Coimbatore',
    marital_status: 'single',
    DateOfVisit: 'Sun May 02 2021 13:34:58 GMT+0530 (India Standard Time)'
  },
  {
    _id: 608e7c5b5a9d302148bf0d82,
    Name: 'murali',
    age: 15,
    ID: 126,
    gender: 'Male',
    address: 'Erode',
    marital_status: 'single',
    DateOfVisit: 'Sun May 02 2021 15:48:03 GMT+0530 (India Standard Time)'
  }
]
Deleted murali
[
  {
    _id: 608e5d2af10ecb0f7ee2427b,
    Name: 'Raj',
    age: 25,
    ID: 123,
    gender: 'Male',
    address: 'Coimbatore',
    marital_status: 'single',
    DateOfVisit: 'Sun May 02 2021 13:34:58 GMT+0530 (India Standard Time)'
  }
]
updated Raj
[
  {
    _id: 608e5d2af10ecb0f7ee2427b,
    Name: 'Raj Kumar',
    age: 25,
    ID: 123,
    gender: 'Male',
    address: 'Chennai',
    marital_status: 'single',
    DateOfVisit: 'Sun May 02 2021 13:34:58 GMT+0530 (India Standard Time)'
  }
]
PS F:\SEM6\IPlab\NODEJS\assignment8\D>
```

Learning Experience:

- Learnt how to run a node program.
- Learnt the use of readline library.
- Learnt how to read from file and from console.
- Learnt the use of url library
- Learnt how to get data from url.
- Learnt about mongodb and its CRUD.
- Learnt to create a web server.