**Muhammad Abdullah**

**SE(6A) | 19F-0916**

Web Engineering Lab

Lab9: NodeJs EMAIL, FS,URl

TASK 1

NodeJS Code:

var http = require ('http');

var fs = require ('fs');

http.createServer(function (req, res) {

    fs.readFile('Task1.html', function(err, data) {

      res.writeHead(200, {'Content-Type': 'text/html'});

      res.write(data);

      return res.end();

    });

  }).listen(8080);

Output:

A screenshot of a computer

Description automatically generated with medium confidence

TASK 2

NodeJS Code:

var http = require ('http');

var fs = require ('fs');

http.createServer(function (req, res) {

    fs.readFile('Task2.txt', function(err, data) {

      res.writeHead(200, {'Content-Type': 'text/html'});

      res.write(data);

      return res.end();

    });

  }).listen(8080);

Output:

A picture containing logo

Description automatically generated

Note: File is provided in RAR

TASK 3

NodeJS Code:

var http = require ('http');

var fs = require ('fs');

fs.appendFile('mytext.txt', 'Hi, I am newly text added, happy to be in notepad file.', function (err) {

    if (err) throw err;

    console.log('Saved!');

  });

http.createServer(function (req, res) {

    fs.readFile('mytext.txt', function(err, data) {

      res.writeHead(200, {'Content-Type': 'text/html'});

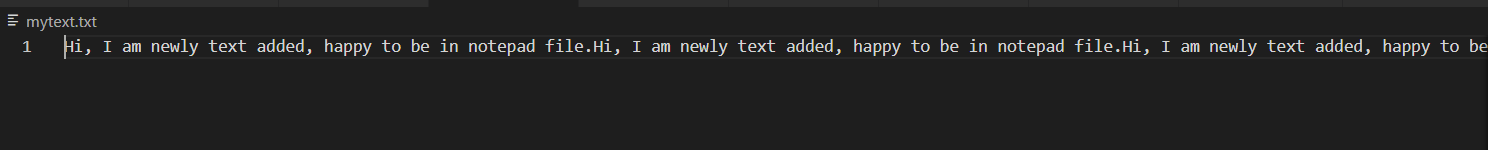
      res.write(data);

      return res.end();

    });

  }).listen(8080);

Output:



TASK 4

NodeJS Code:

var fs = require ('fs');

fs.rename('mytask4.txt', 'mynewTask4.txt', function (err) {

    if (err) throw err;

    console.log('File Renamed! PS: Check File Name Before Trying to Rename it Again!');

  });

TASK 5

var http = require('http');

var fs = require('fs');

var url = require('url');

http.createServer(function (req, res) {

  var q = url.parse(req.url, true);

  var filename = "." + q.pathname;

  fs.readFile(filename, function(err, data) {

    if (err) {

      res.writeHead(404, {'Content-Type': 'text/html'});

      return res.end("404 Not Found");

    }

    res.writeHead(200, {'Content-Type': 'text/html'});

    res.write(data);

    return res.end();

  });

}).listen(8080);

Output:

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

TASK 6

NodeJS Code:

var http = require ('http');

var string = 'hello world!'

http.createServer(function (req, res) {

    res.writeHead(200, {'Content-Type': 'text/html'});

      res.write(string);

      res.write('   After Converting String Into Upper Case : ');

      res.write(string.toUpperCase());

      return res.end();

  }).listen(8080);

Output:

Graphical user interface, text, application

Description automatically generated

TASK 7

NodeJS Code:

var http = require('http');

var fs = require('fs');

var formidable = require('formidable');

http.createServer(function (req, res) {

  if (req.url == '/fileupload') {

    var form = new formidable.IncomingForm();

    form.parse(req, function (err, fields, files) {

      var oldpath = files.filetoupload.filepath;

      var newpath = 'E:\FAST-NU\6th Semester\Web Engineering\Lab\Assignments\6' + files.filetoupload.originalFilename;

      fs.rename(oldpath, newpath, function (err) {

        if (err) throw err;

        res.write('File uploaded and moved!');

        res.end();

      });

 });

  } else {

    res.writeHead(200, {'Content-Type': 'text/html'});

    res.write('<form action="fileupload" method="post" enctype="multipart/form-data">');

    res.write('<input type="file" name="filetoupload"><br>');

    res.write('<input type="submit">');

    res.write('</form>');

    return res.end();

  }

}).listen(8080);

Output:

File Saved at Destination!



TASK Bonus

NodeJS Code:

var nodemailer = require('nodemailer');

var transporter = nodemailer.createTransport({

  service: 'gmail',

  auth: {

    user: 'f190916@nu.edu.pk',

    pass: 'phantom1424'

  }

});

var mailOptions = {

  from: 'f190916@nu.edu.pk',

  to: 'usman.azhar@nu.edu.pk',

  subject: '19F-0916 Bonus Task',

  text: 'Arabic Corpus'

};

transporter.sendMail(mailOptions, function(error, info){

  if (error) {

    console.log(error);

  } else {

    console.log('Email sent: ' + info.response);

  }

});

Output:

File Send to Sir Usman Azhar



TASK 8

NodeJS Code:

var http = require('http');

var formidable = require('formidable');

var nodemailer = require('nodemailer');

http.createServer(function (req, res) {

  if (req.url == '/sendEmail') {

    var form = new formidable.IncomingForm();

    form.parse(req, function (err, fields, files) {

        var transporter = nodemailer.createTransport({

            service: 'gmail',

            auth: {

            user: fields.semail,

           pass: fields.spass

         }

       });

           var mailOptions = {

           from: fields.semail,

            to: fields.remail,

           subject: fields.subject,

           text: fields.content

       };

           transporter.sendMail(mailOptions, function(error, info){

           if (error) {

                console.log(error);

           } else {

               console.log('Email sent: ' + info.response);

           }

           if (err) throw err;

        res.write('Email Send !');

        res.end();

       });

    });

  } else {

    res.writeHead(200, {'Content-Type': 'text/html'});

    res.write('<form action="sendEmail" method="post" enctype="multipart/form-data">');

    res.write('<label for="semail">Enter Sender Email:</label>')

    res.write('<input type="text" name="Semail" id="semail"><br><br>')

    res.write('<label for="spass">Enter Sender Password:</label>')

    res.write('<input type="password" name="Spass" id="spass"><br><br>')

    res.write('<label for="remail">Enter Receiver Email:</label>')

    res.write('<input type="text" name="Remail" id="remail"><br><br>')

    res.write('<label for="subject">Enter Subject:</label>')

    res.write('<input type="text" name="Subject" id="subject"><br><br>')

    res.write('<label for="content">Enter Content:</label>')

    res.write('<input type="text" name="Content" id="content"><br><br>')

    res.write('<input type="submit" name="Send Details" id="details">');

    res.write('</form>');

    return res.end();

  }

}).listen(8080);

Output:

Before Sending:

Graphical user interface, text

Description automatically generated

After Sending:

Graphical user interface, text, application

Description automatically generated