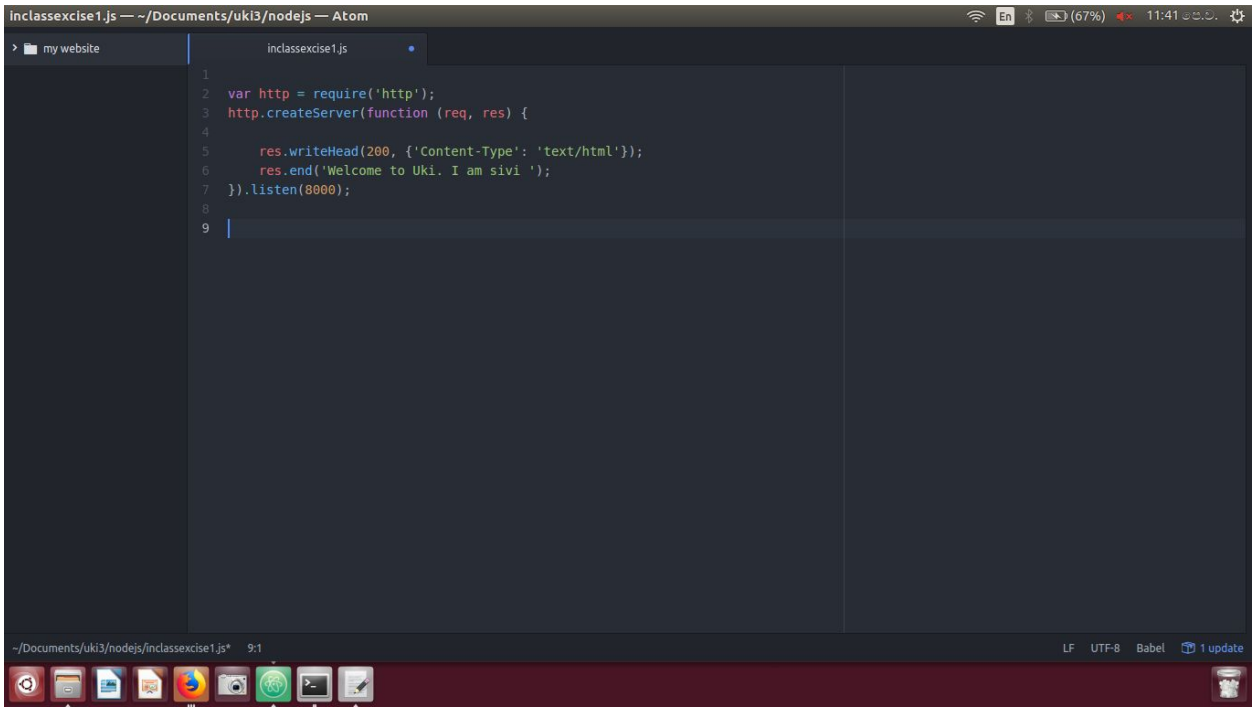
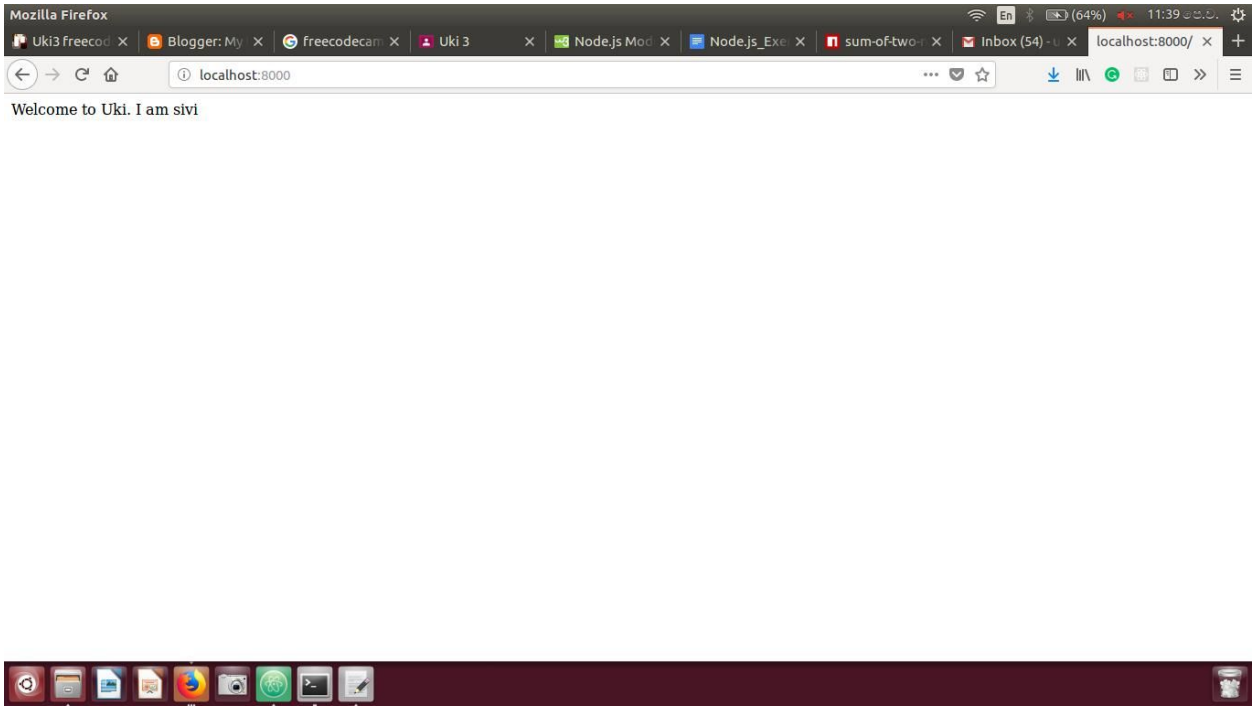
Node.js Exercise 1 Answers

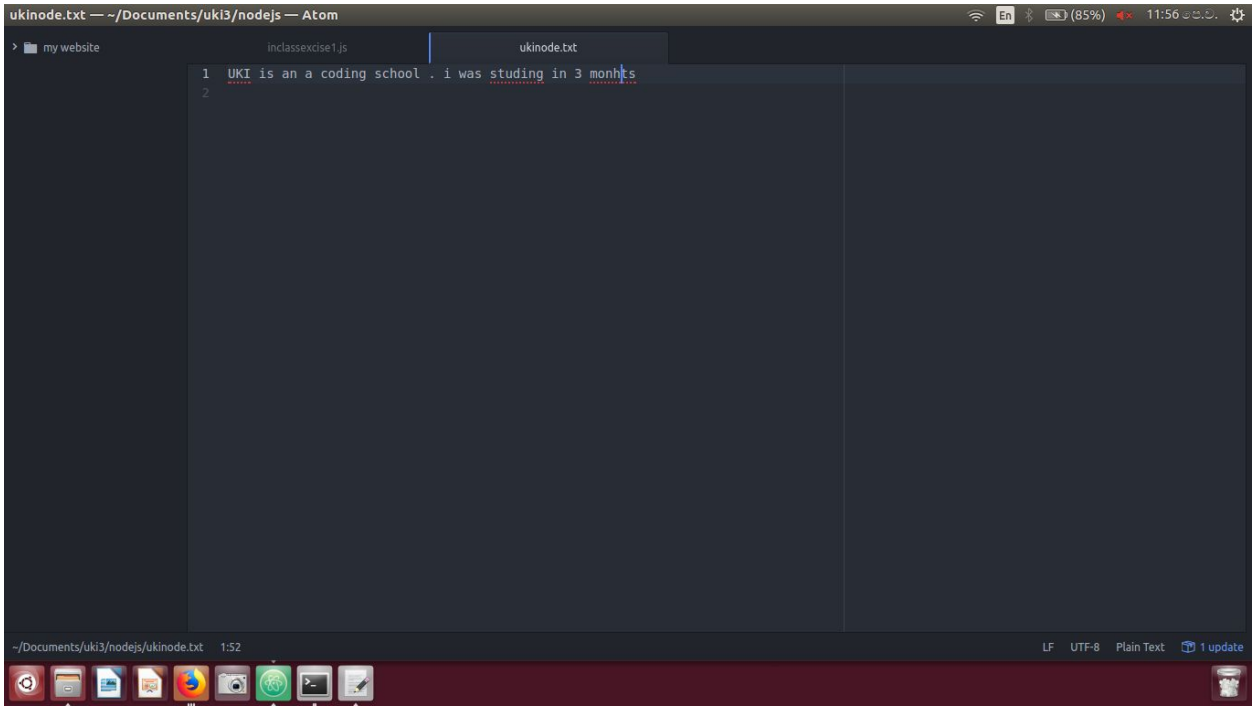
1. **Create a custom module which returns the sum and average of any two numbers passed into it. Require the module and run the server by passing 123 and 321 so that the server prints out the sum and average.**

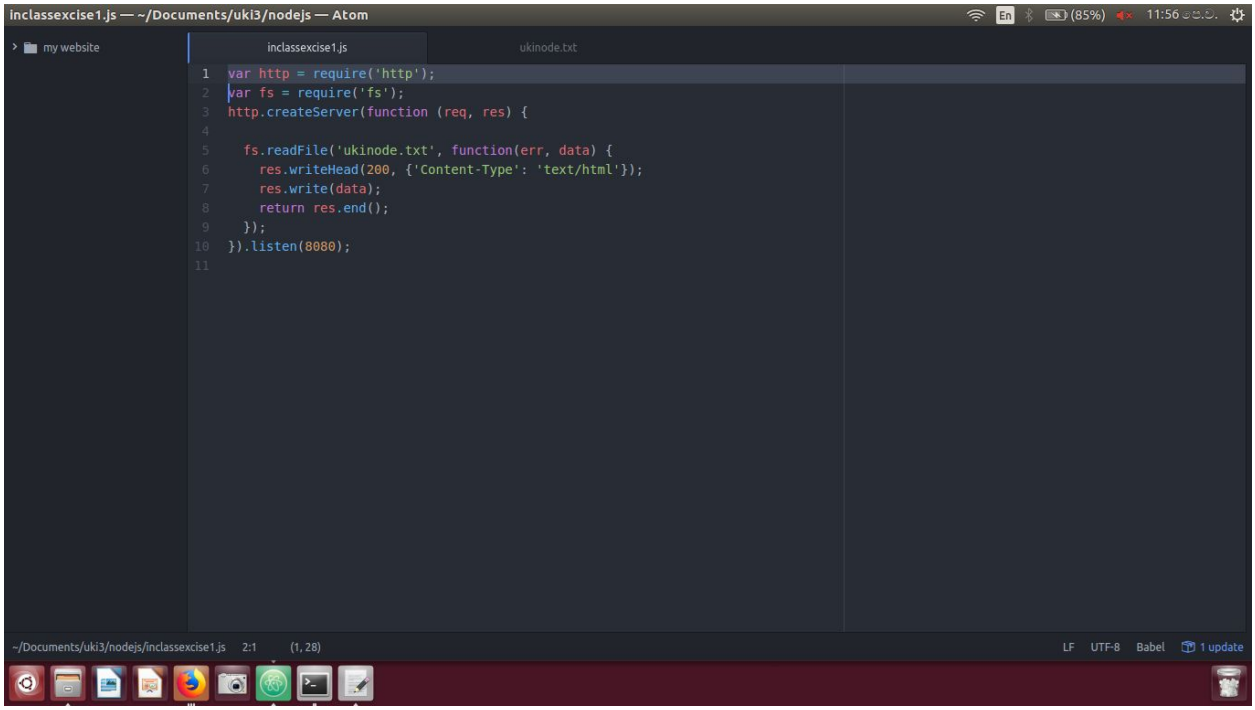
2. **Create a simple http server and print “Welcome to Uki. I am yourname” when a request is sent to your server via the port 8000. (Note - Change different port numbers and check)**

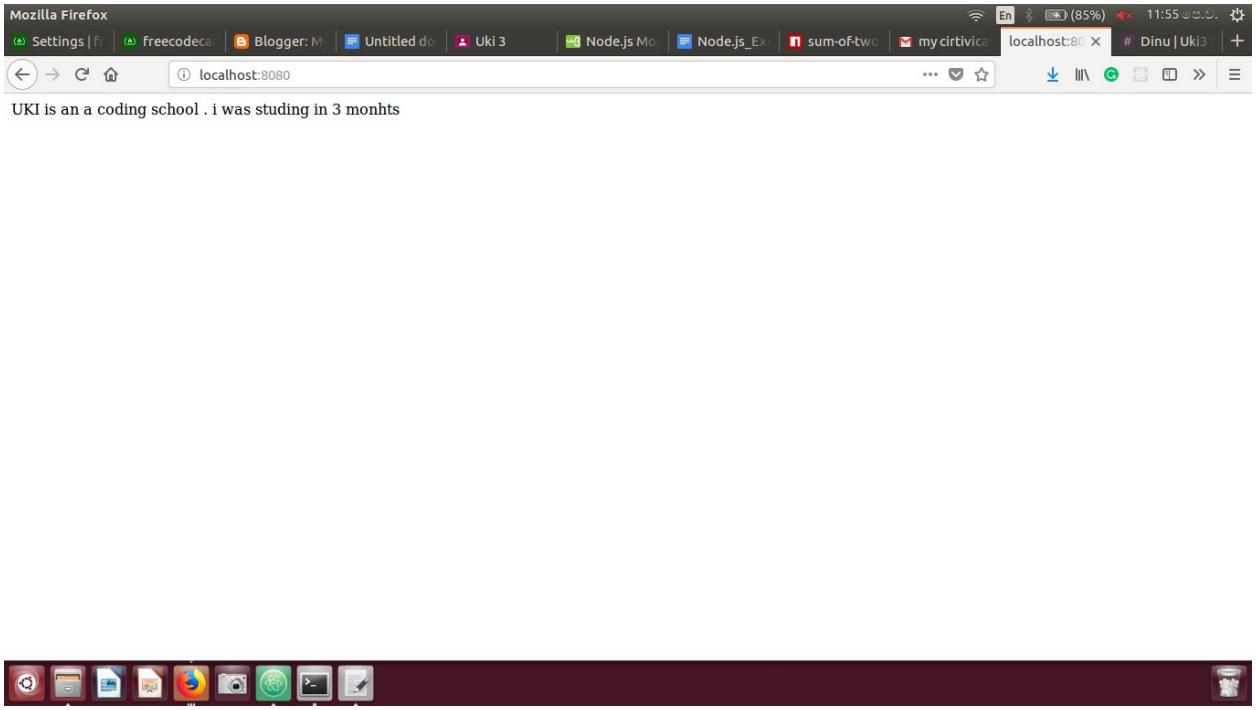




3. **Using the file system module create a new file called ukinode.txt**
   - **3.1 Write a paragraph about Uki into that file**
   - **3.2 Serve that file to the client (Read File) over your server**
   - **3.3 Append another paragraph about Uki and now serve the new file**
   - **3.4 Rename the file as ukinodejsexercise1.txt**
   - **3.5 Delete the file you created**


ukinode.txt — ~/Documents/uki3/nodejs — Atom
my website
inclassexcise1.js
ukinode.txt
1 UKI is an a coding school . i was studing in 3 monhts
2
~/Documents/uki3/nodejs/ukinode.txt 1:52
LF UTF-8 Plain Text 1 update


inclassexcise1.js — ~/Documents/uki3/nodejs — Atom
my website
inclassexcise1.js
ukinode.txt
1 var http = require('http');
2 var fs = require('fs');
3 http.createServer(function (req, res) {
4
5 fs.readFile('ukinode.txt', function(err, data) {
6 res.writeHead(200, {'Content-Type': 'text/html'});
7 res.write(data);
8 return res.end();
9 });
10 }).listen(8080);
11
~/Documents/uki3/nodejs/inclassexcise1.js 2:1 (1, 28)
LF UTF-8 Babel 1 update

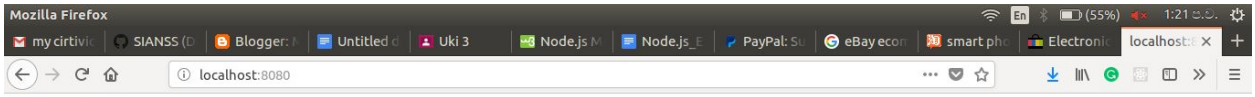


4. Create two html files called head.html which is a web page which says 'you have got head ' and tail.html which is a web page which says 'you have got tail' and save them in the same folder as your node.js files. Create a Node.js file that opens the requested file and returns the content to the client. If anything goes wrong, throw a 404 error.

If you have followed the correct steps you should see two different results when opening these two addresses:

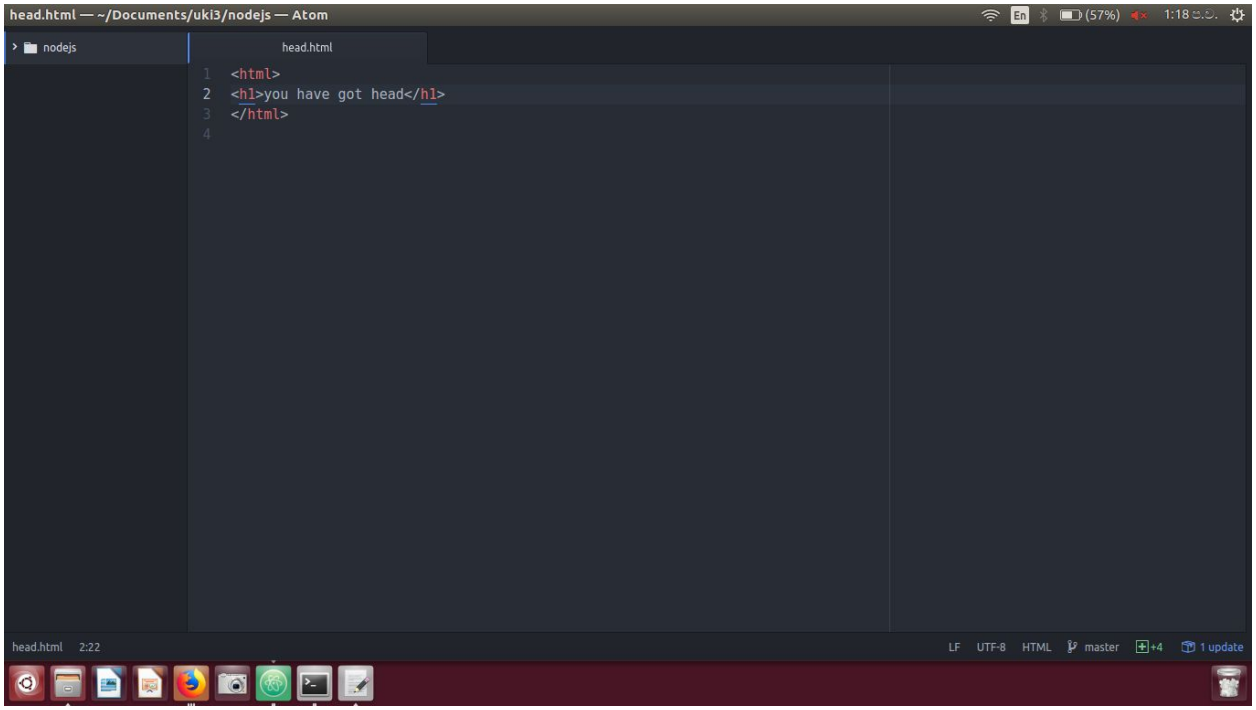
http://localhost:8080/head.html - > You have got head

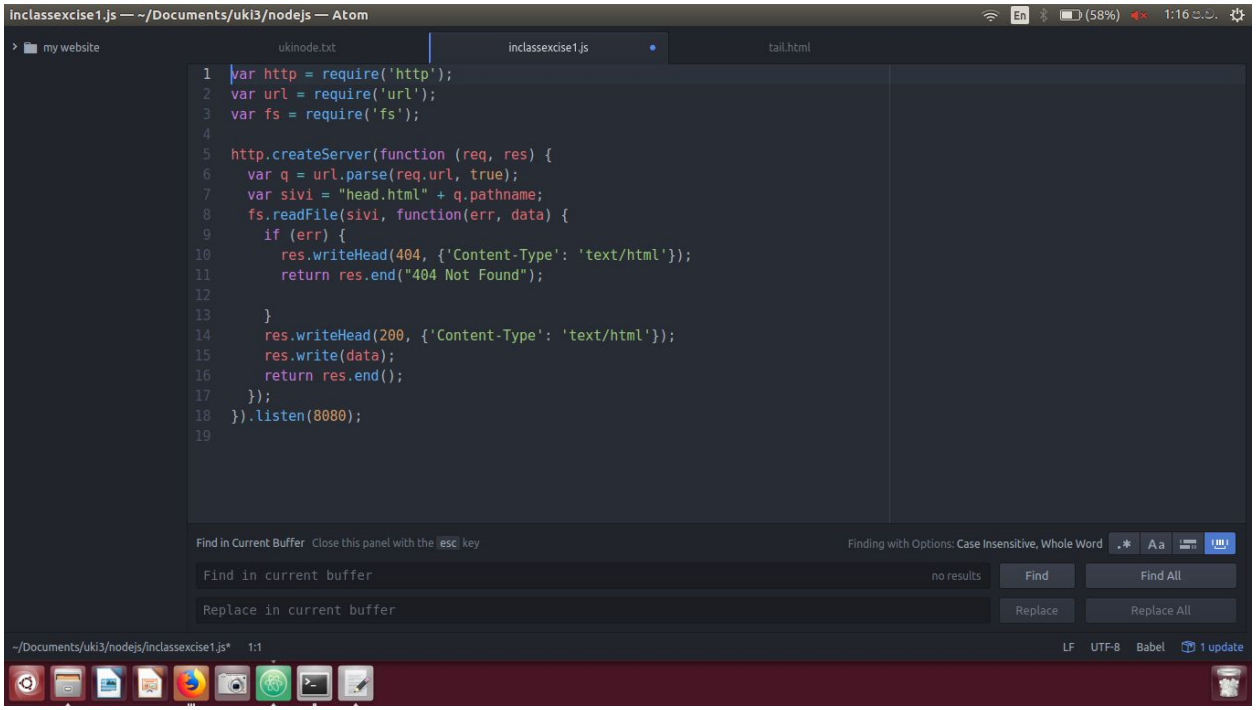
http://localhost:8080/tail.html -> You have got tail



404 Not Found

404 Not Found


head.html — ~/Documents/uki3/nodejs — Atom
nodejs
head.html
<html>
<h1>you have got head</h1>
</html>
head.html 2:22
LF UTF-8 HTML master +4 1 update


inclassexcise1.js — ~/Documents/uki3/nodejs — Atom
my website
ukinode.txt
inclassexcise1.js
tail.html
var http = require('http');
var url = require('url');
var fs = require('fs');
http.createServer(function (req, res) {
var q = url.parse(req.url, true);
var sivi = "head.html" + q.pathname;
fs.readFile(sivi, 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);
Find in Current Buffer Close this panel with the esc key
Finding with Options: Case Insensitive, Whole Word
Find in current buffer
no results
Find
Find All
Replace in current buffer
Replace
Replace All
~/Documents/uki3/nodejs/inclassexcise1.js* 1:1
LF UTF-8 Babel 1 update

5. Install the package “upper-case” using NPM and create a Node.js file that will convert the output "Uki is the best place to learn programming !" into upper-case letters.

```
var http = require('http');
var sivi = require('upper-case');

http.createServer(function ( res, req){
  res.writeHead(200,{'Content-Type':'text/html'});
  res.writeHead(sivi("Uki is the best place to learn programming !"))
  res.end();
}).listen(2000);
```

6. Create an event handler function that will say “I bark when I see strangers !” when a "bark" event is fired.

```
var evnt = require('events');
var evntemtr = new evnt.EventEmitter();
var animalhandler = function(){
  console.log("I bark when i see strangers !");
} evntemtr.on('bark', animalhandler);
evntemtr.emit('bark');
```

7. Install “formidable” module using npm and make a web page in Node.js that lets the user upload files to your computer.

7.1 Save that uploaded file into your Documents directory.

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

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.path;
        var newpath = ' file full  path   ' + files.filetoupload.name;

        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);
```

8. Using the Nodemailer module create a server and send a mail to info@uki.life with the subject : “Testing my nodemailer module” , text: “This is easy !”
8.1 Now instead of text send a basic html formatted mail.

```
var nodemailer = require('nodemailer');
var transporter = nodemailer.createTransport({
  service: 'gmail',
  auth: {
```

```
        user: 'acsian7@gmail.com',
        pass: 'sicar1996'
  }
});
var mailOptions = {
  from: 'acsian7@gmail.com',
  to: 'acsian7@gmail.com',
  subject: 'Sending Email using Node.js',
  text: 'That was easy!'
};
transporter.sendMail(mailOptions, function(error, info){
  if (error) {
        console.log(error);
  } else {
        console.log('Email sent: ' + info.response);
  }
});
```