

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.

```
ex_1.js
1 exports.S = function () {
2   var x=123,
3     y=321,
4     z=x+y;
5   return z;
6 };
7 exports.A = function () {
8   var x=123,
9     y=321,
10    z=(x+y)/2;
11  return z;
12 };
13
14 var http = require('http');
15 var d = require('./ex_1');
16
17 http.createServer(function (req, res) {
18   res.writeHead(200, {'Content-Type': 'text/html'});
19   res.write("sum: " + d.S() + '<br>' + "AVG: " + d.A());
20   res.end();
21 }).listen(8080);
22
```

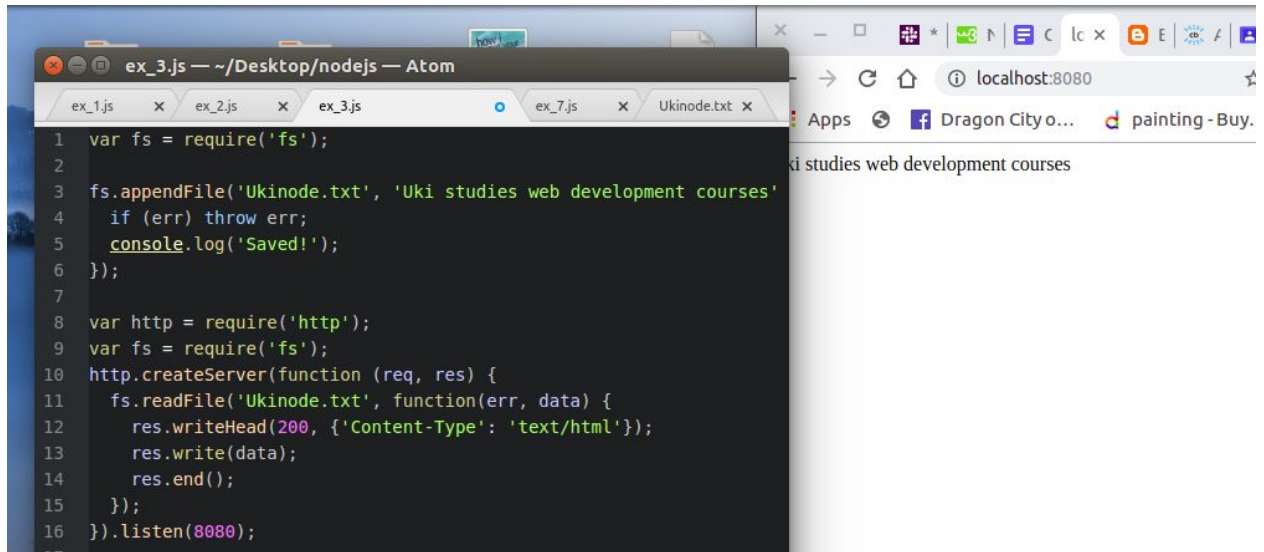
```
ukistu03@ukipc03: ~/Desktop/nodejs
TypeError: First argument must be a string or Buffer
    at ServerResponse.OutgoingMessage.write (_http_outgoing.js:470:11)
    at ReadFileContext.callback (/home/ukistu03/Desktop/nodejs/ex_3.js:1
    at FSReqWrap.readFileAfterOpen [as oncomplete] (fs.js:367:13)
[nodemon] app crashed - waiting for file changes before starting...
[nodemon] restarting due to changes...
[nodemon] starting node ex_1.js
Saved!
^Cukistu03@ukipc03:~/Desktop/nodejs$ ^C
ukistu03@ukipc03:~/Desktop/nodejs$ nodemon ex_1.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting node ex_1.js
[nodemon] restarting due to changes...
[nodemon] starting node ex_1.js
[nodemon] restarting due to changes...
[nodemon] starting node ex_1.js
ukistu03@ukipc03:~/Desktop/nodejs$ nodemon ex_1.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting node ex_1.js
```

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)

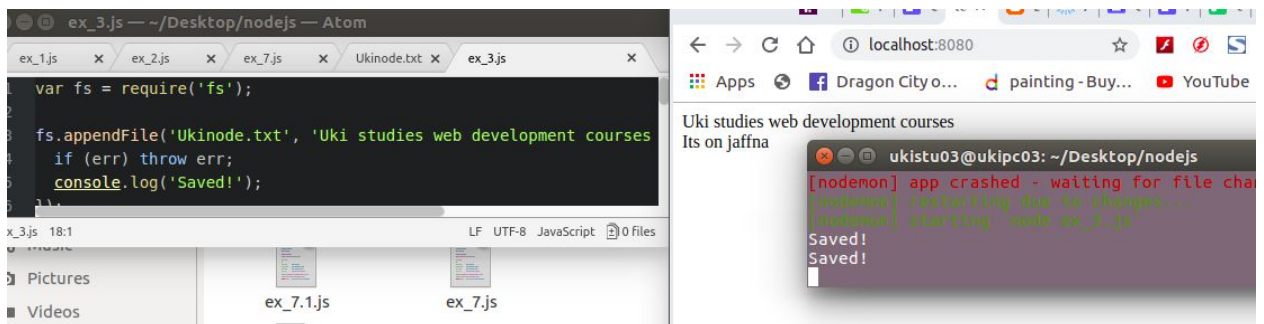
```
ex_2.js
1 var http = require('http');
2 http.createServer(function (req, res) {
3   res.writeHead(200, {'Content-Type': 'text/html'});
4   res.write('Welcome to Uki.I am Thivagar.');
```

```
ukistu03@ukipc03: ~/Desktop/nodejs
ukistu03@ukipc03:~/Desktop/nodejs$ nodemon ex_2.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting node ex_2.js
```

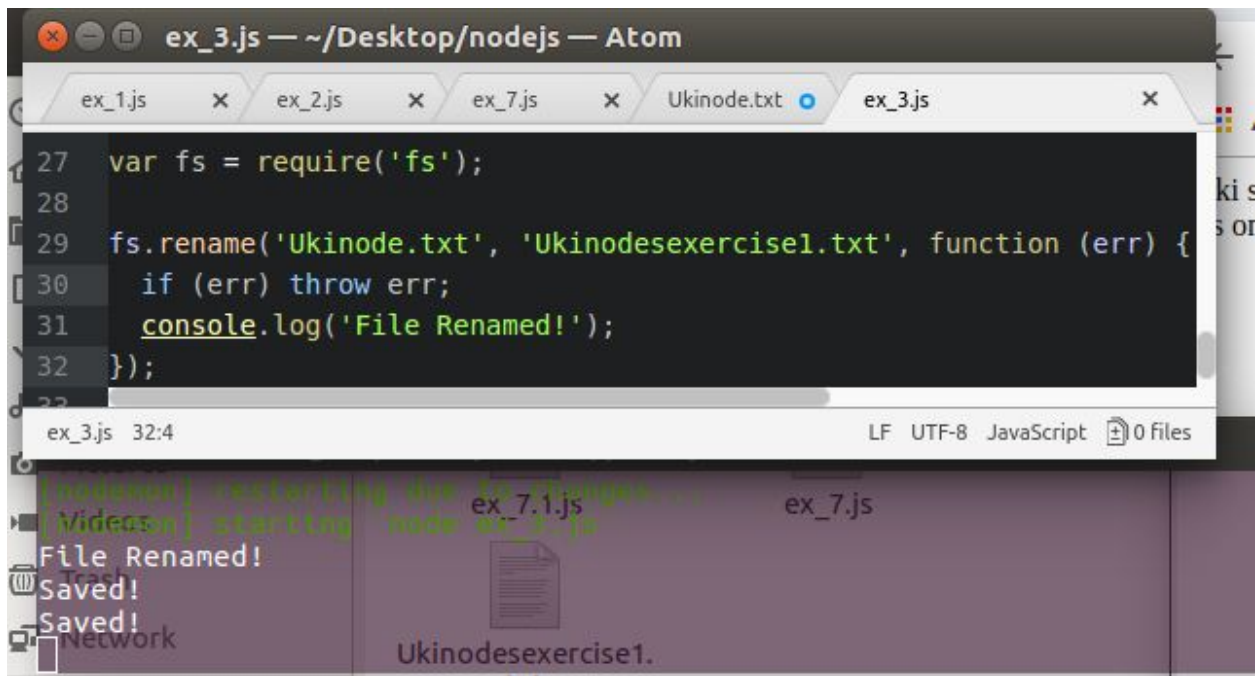
3. Using the file system module create a new file called ukinode.txt
 - 3.1 Write a paragraph about Uki into that file
 - 3.2 Serve that file to the client (Read File) over your server



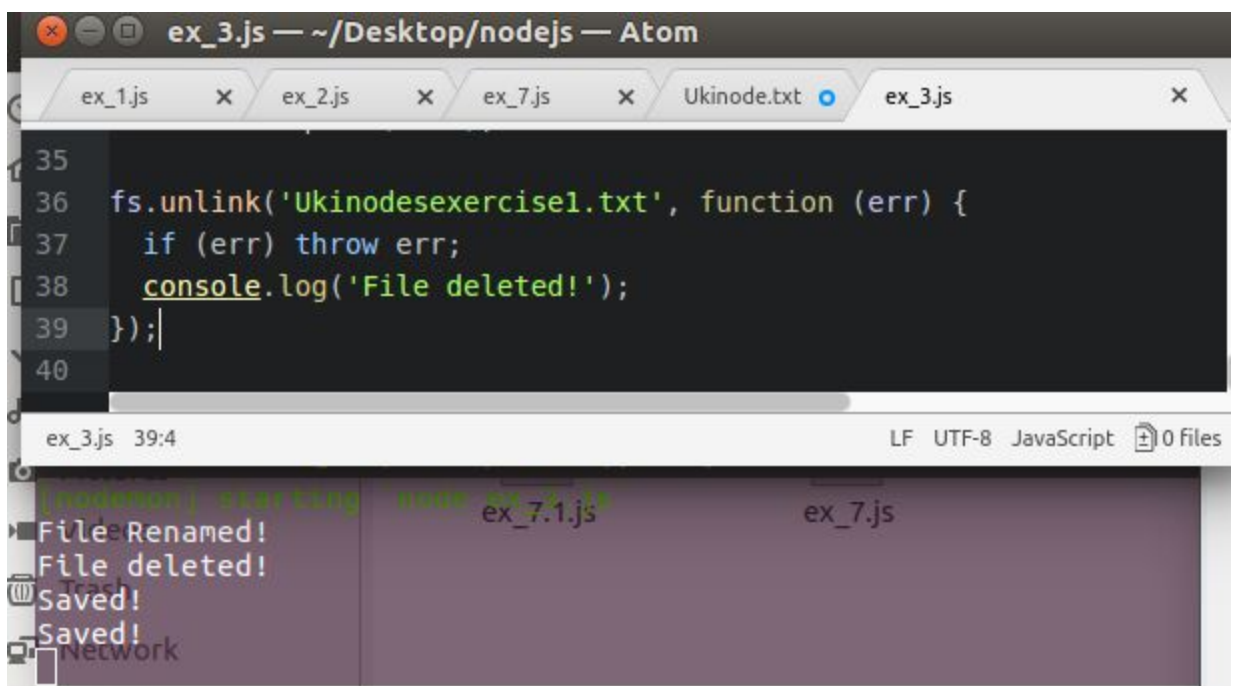
- 3.3 Append another paragraph about Uki and now serve the new file



- 3.4 Rename the file as ukinodejsexercise1.txt



3.5 Delete the file you created

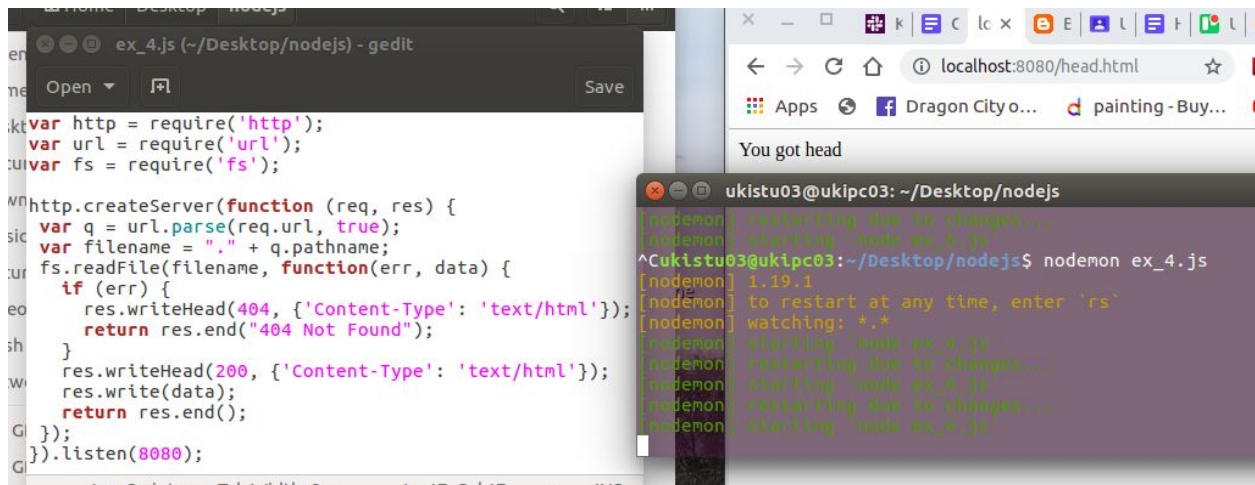


4. Create two html files called head.html which is a web page which says 'you have got head' and tail.html which is a web page which says 'you have got tail' and save them in the same folder as your node.js files. Create a Node.js file that opens the requested file and returns the content to the client.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

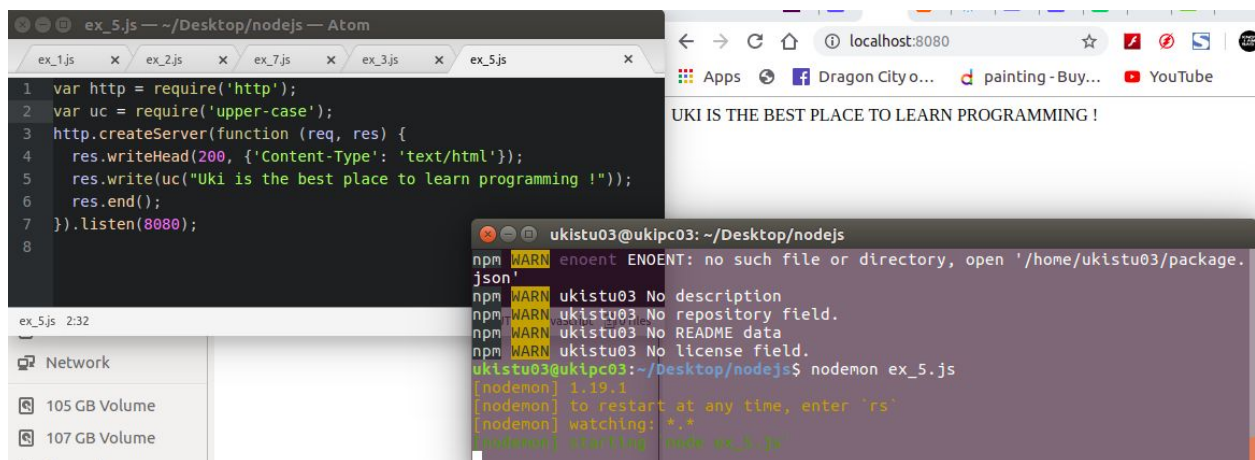
If you have followed the correct steps you should see two different results when opening these two addresses:

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

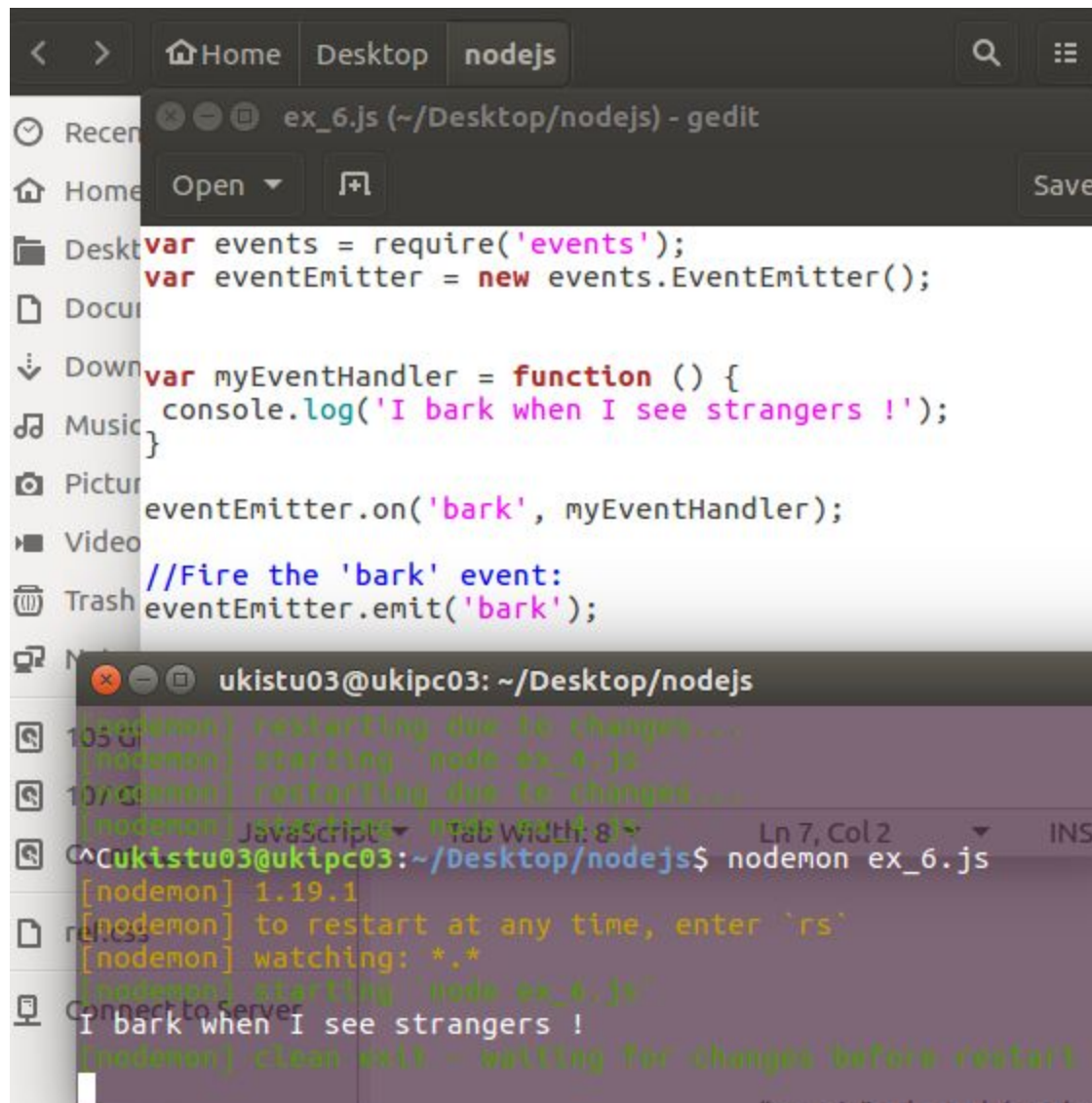
http://localhost:8080/tail.html -> You have got tail



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.



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



The image shows a code editor window titled 'ex_6.js (~/Desktop/nodejs) - gedit' and a terminal window below it. The code in the editor defines an event emitter and a handler for the 'bark' event. The terminal shows the command 'nodemon ex_6.js' being executed, which starts the application and outputs the message 'I bark when I see strangers !'.

```
var events = require('events');
var eventEmitter = new events.EventEmitter();

var myEventHandler = function () {
  console.log('I bark when I see strangers !');
}

eventEmitter.on('bark', myEventHandler);

//Fire the 'bark' event:
eventEmitter.emit('bark');
```

```
ukistu03@ukipc03: ~/Desktop/nodejs
[nodemon] restarting due to changes...
[nodemon] starting node ex_4.js
[nodemon] restarting due to changes...
[nodemon] starting node ex_4.js
ukistu03@ukipc03:~/Desktop/nodejs$ nodemon ex_6.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting node ex_6.js
I bark when I see strangers !
[nodemon] clean exit - waiting for changes before restart
```

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

```
ukistu03@ukipc03:~/Desktop/nodejs$ npm install formidable
/home/ukistu03
└─ formidable@1.2.1
  You have downloaded the Formidable module, you can include the module in
  any application:
  var formidable = require('formidable');
  ...
  form.parse(req, function(err, fields, files) {
    ...
  });

npm WARN ukistu03 No description
npm WARN ukistu03 No repository field
npm WARN ukistu03 No README data
npm WARN ukistu03 No license field
ukistu03@ukipc03:~/Desktop/nodejs$
```

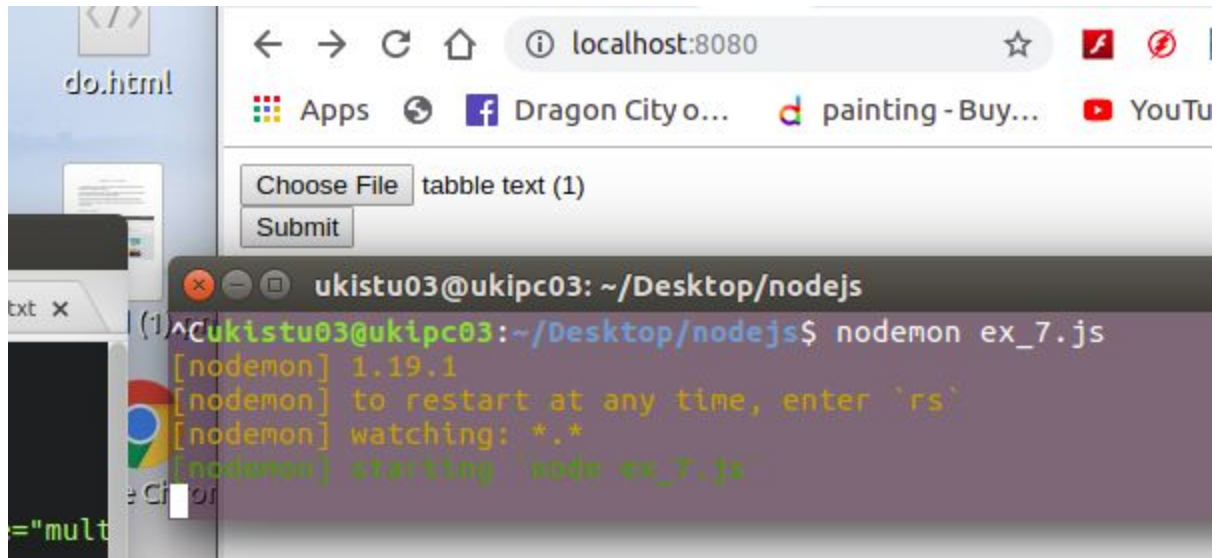
7.1 Save that uploaded file into your Documents directory.

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

http.createServer(function (req, res) {
  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();
})

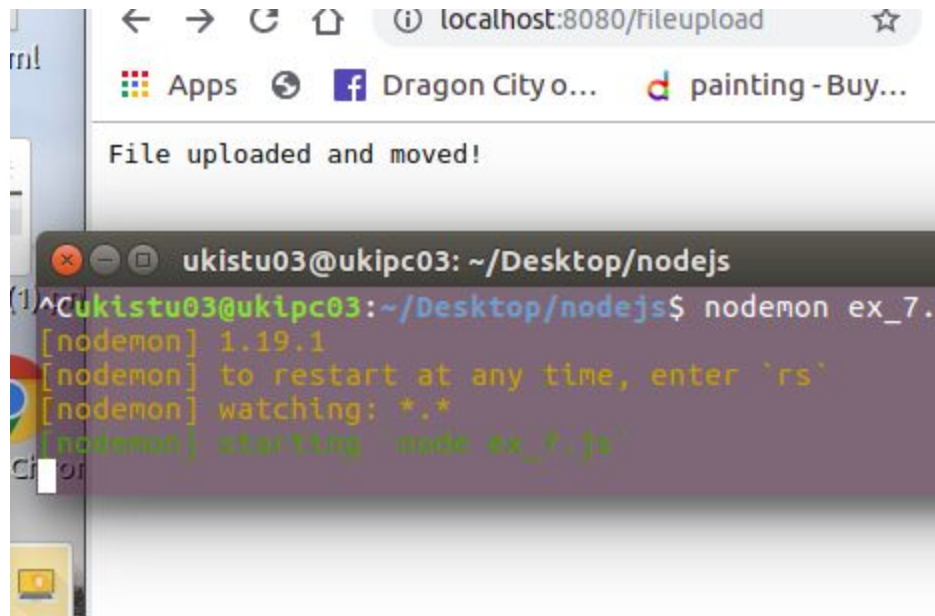
var http = require('http');
var formidable = require('formidable');

http.createServer(function (req, res) {
  if (req.url == '/fileupload') {
    var form = new formidable.IncomingForm();
    form.parse(req, function (err, fields, files) {
      res.write('File uploaded');
      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();
  }
})
```




```
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.fileupload.path;
      var newpath = 'C:/Users/Your Name/' + files.fileupload.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="fileupload"><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 !"


```
Open ▾ 
var nodemailer = require('nodemailer');

var transporter = nodemailer.createTransport({
  service: 'gmail',
  auth: {
    user: 'youremail@gmail.com',
    pass: 'yourpassword'
  }
});

var mailOptions = {
  from: 'thivagar1998@email@gmail.com',
  to: 'keerthynages@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);
  }
});
```

8.1 Now instead of text send a basic html formatted mail.