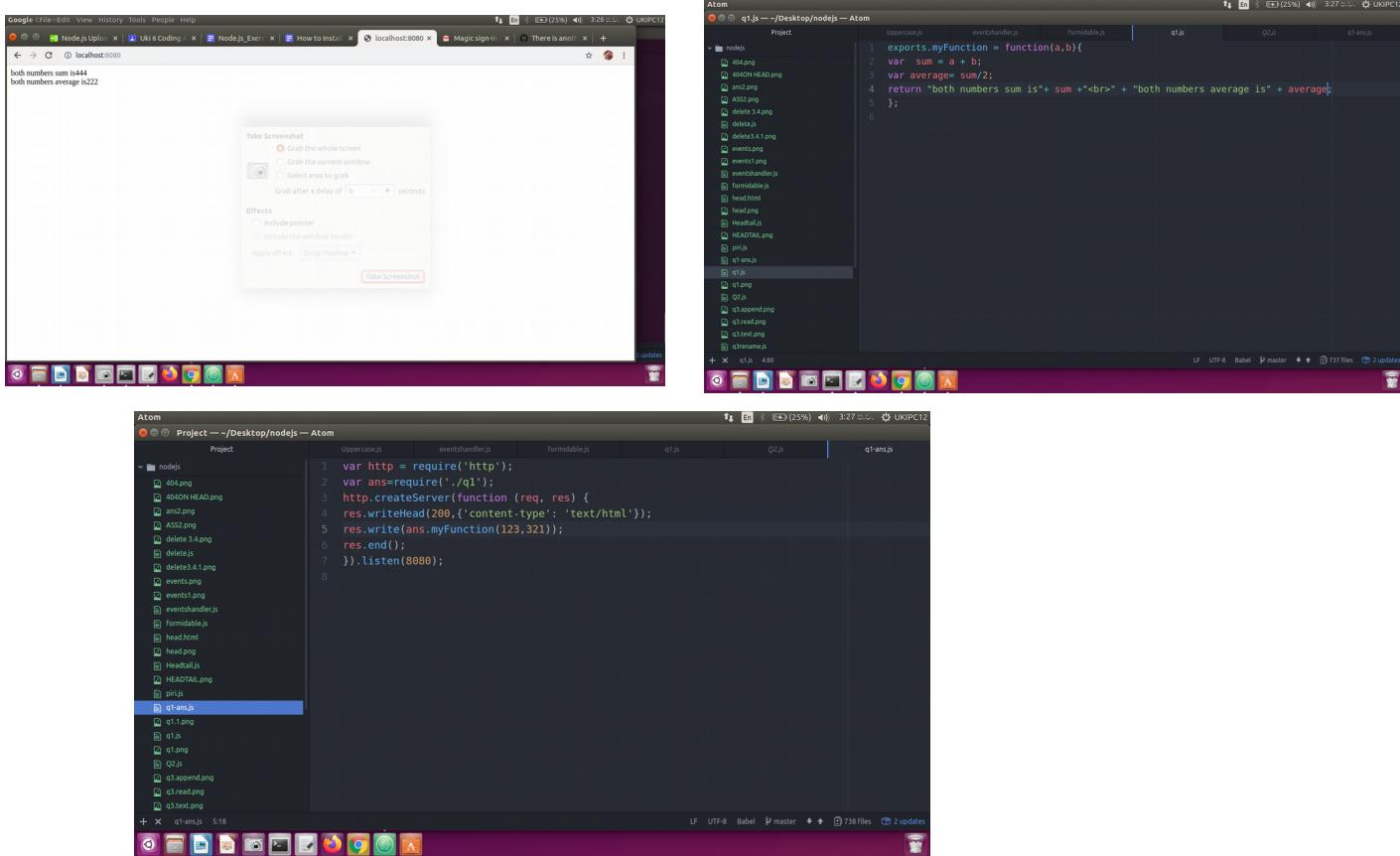
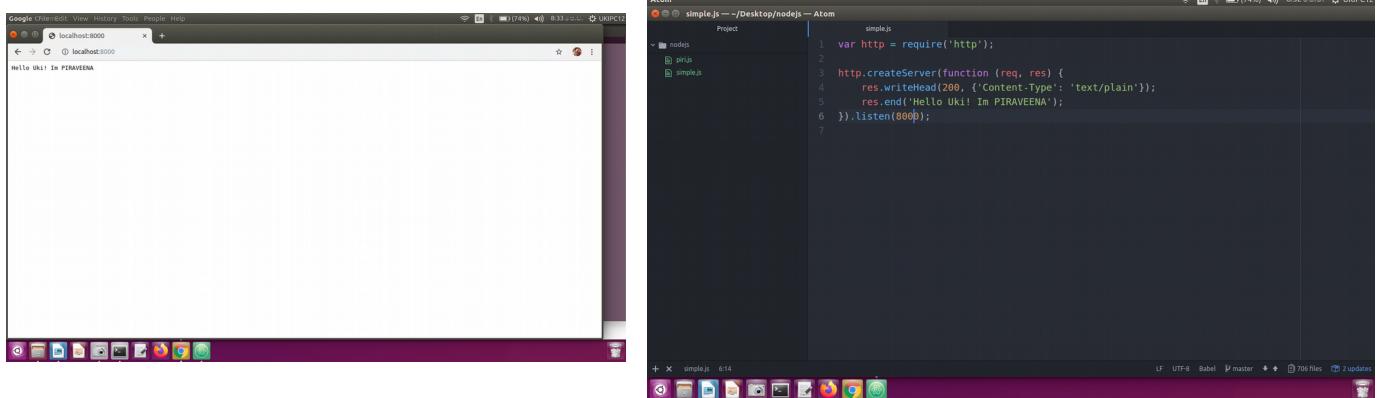


1. Create a custom module which returns the sum and average of any two numbers passed into it. Require the module and run the server by passing 123 and 321 so that the server prints out the sum and average.



```
q1.js — ~/Desktop/nodejs — Atom
Project
Uppercase.js eventhandler.js Formidable.js
q1.js Q2.js q1-ans.js
1 exports.myFunction = function(a,b){  
2     var sum = a + b;  
3     var average= sum/2;  
4     return "both numbers sum is"+ sum +"  
5     <br> "+ "both numbers average is "+ average;  
6  
q1-ans.js — ~/Desktop/nodejs — Atom
Project
q1.js  
1 var http = require('http');  
2 var ans=require('./q1');  
3 http.createServer(function (req, res) {  
4     res.writeHead(200, {'Content-Type': 'text/html'});  
5     res.write(ans.myFunction(123,321));  
6     res.end();  
7 }).listen(8080);  
8
```

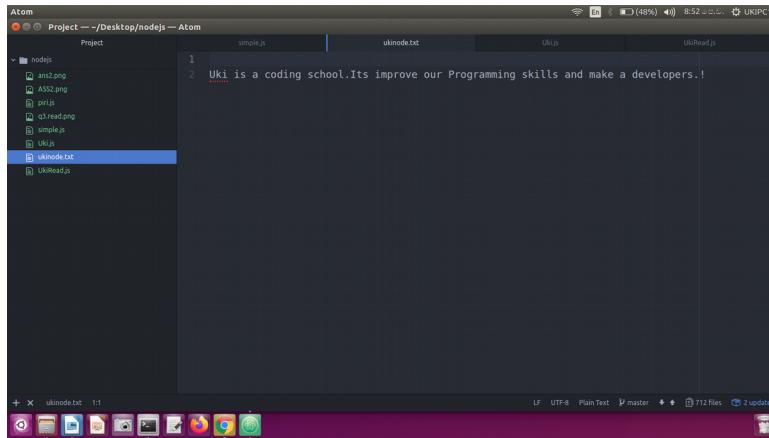
2. Create a simple http server and print “Welcome to Uki. I am **yourname**” when a request is sent to your server via the port 8000. (Note - Change different port numbers and check)



```
simple.js — ~/Desktop/nodejs — Atom
Project
simple.js
1 var http = require('http');  
2  
3 http.createServer(function (req, res) {  
4     res.writeHead(200, {'Content-Type': 'text/plain'});  
5     res.end('Hello Uki! Im PIRAVEENA');  
6 }).listen(8000);  
7
```

Using the file system module create a new file called ukinode.txt

3.1 Write a paragraph about Uki into that file

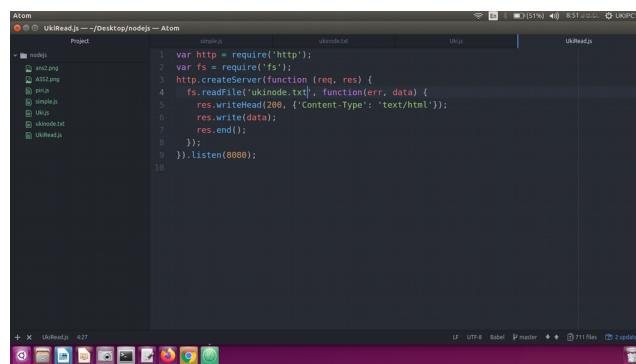


```
Atom
Project — ~/Desktop/nodejs — Atom
  Project
    nodejs
      an2.png
      A5S2.png
      p1t.png
      q3-read.png
      simple.js
      Uki.js
      Ukihead.js
      ukinode.txt
      ukinode.js

  simple.js          ukinode.txt          Uki.js          Ukihead.js
1  Uki is a coding school. Its improve our Programming skills and make a developers.!
2

LF  UTF-8  Plain Text  master  +  712 files  2 updates
```

### 3.2 Serve that file to the client (Read File) over your server

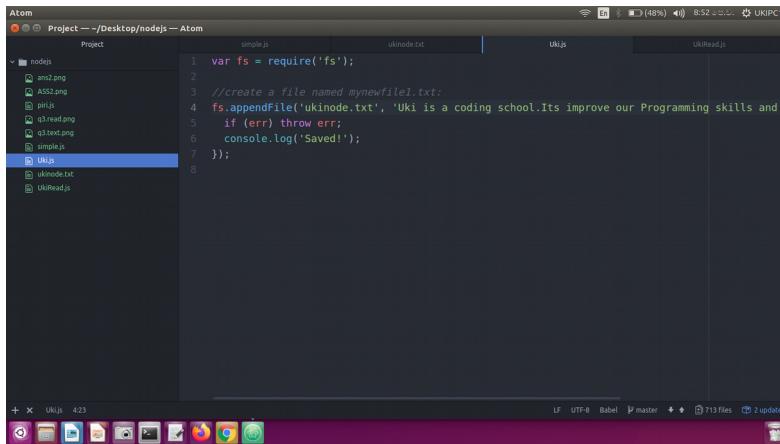


```
Atom
Project — ~/Desktop/nodejs — Atom
  Project
    nodejs
      an2.png
      A5S2.png
      p1t.png
      q3.append.png
      q3.read.png
      q3.text.png
      simple.js
      Uki.js
      Ukihead.js
      UkiRead.js

  simple.js          ukinode.txt          Uki.js          Ukihead.js
1  var http = require('http');
2  var fs = require('fs');
3  http.createServer(function (req, res) {
4    fs.readFile('ukinode.txt', function(err, data) {
5      res.writeHead(200, {'Content-Type': 'text/html'});
6      res.write(data);
7      res.end();
8    });
9  }).listen(8080);
10

LF  UTF-8  Babel  master  +  712 files  2 updates
```

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

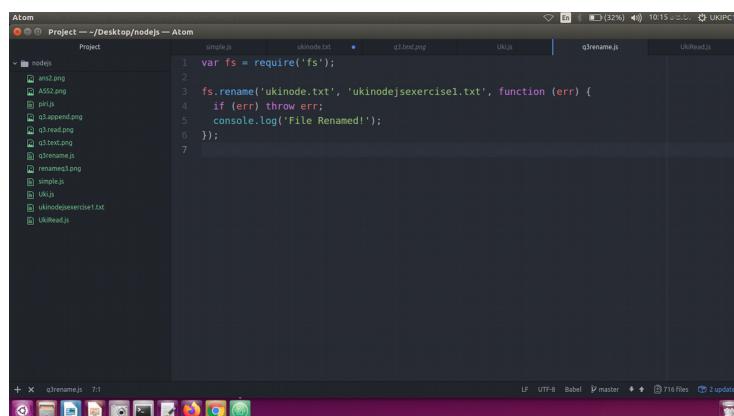


```
Atom
Project — ~/Desktop/nodejs — Atom
  Project
    nodejs
      an2.png
      A5S2.png
      p1t.png
      q3.append.png
      q3.read.png
      q3.text.png
      simple.js
      Uki.js
      Ukihead.js
      ukinode.txt
      UkiRead.js

  simple.js          ukinode.txt          Uki.js          Ukihead.js
1  var fs = require('fs');
2
3  //create a file named mynewfile1.txt:
4  fs.appendFile('ukinode.txt', 'Uki is a coding school. Its improve our Programming skills and make a developers.!');
5  if (err) throw err;
6  console.log('Saved!');
7};

LF  UTF-8  Babel  master  +  712 files  2 updates
```

### 3.4 Rename the file as ukinodejsexercise1.txt



```
Atom
Project — ~/Desktop/nodejs — Atom
  Project
    nodejs
      an2.png
      A5S2.png
      p1t.png
      q3.append.png
      q3.read.png
      q3.text.png
      simple.js
      Uki.js
      Ukihead.js
      ukinode.txt
      UkiRead.js
      q3rename.js

  simple.js          ukinode.txt          q3.rename.png          Uki.js          q3rename.js          UkiRead.js
1  var fs = require('fs');
2
3  fs.rename('ukinode.txt', 'ukinodejsexercise1.txt', function (err) {
4    if (err) throw err;
5    console.log('File Renamed!');
6  });
7;

LF  UTF-8  Babel  master  +  712 files  2 updates
```

### 3.5 Delete the file you created

```

Terminal
ukistu12@ukipc12:~/Desktop/nodejs$ node delete.js
File deleted!

```

```

Atom
Project
delete.js
1 var fs = require('fs');
2
3 fs.unlink('exercisel.txt', function (err) {
4   if (err) throw err;
5   console.log('file deleted!');
6 });

```

#### 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.

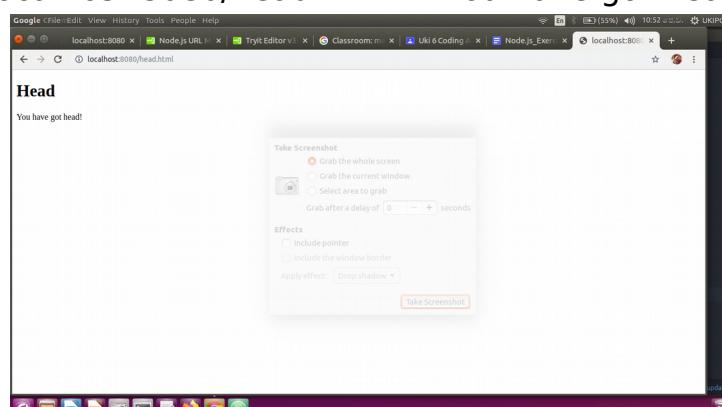
```

Atom
Headtail.js — /Desktop/nodejs — Atom
Project
1 var http = require('http');
2 var url = require('url');
3 var fs = require('fs');
4
5 http.createServer(function (req, res) {
6   var q = url.parse(req.url, true);
7   var filename = __dirname + q.pathname;
8   fs.readFile(filename, function(err, data) {
9     if (err) {
10       res.writeHead(404, {'Content-Type': 'text/html'});
11       return res.end("404 Not Found");
12     }
13     res.writeHead(200, {'Content-Type': 'text/html'});
14     res.write(data);
15     return res.end();
16   });
17 }).listen(8080);

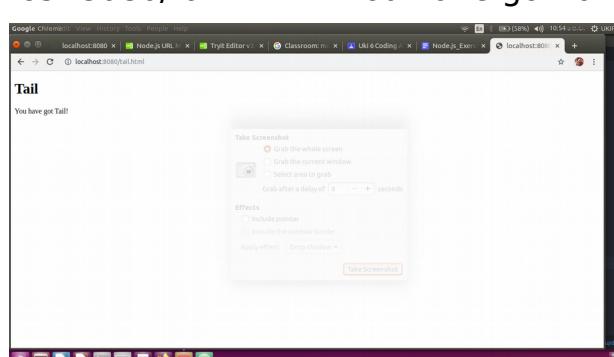
```

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

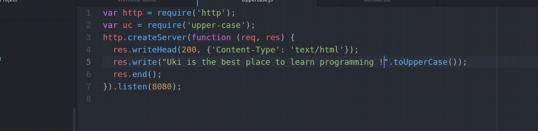
<http://localhost:8080/head.html> -> You have got head



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



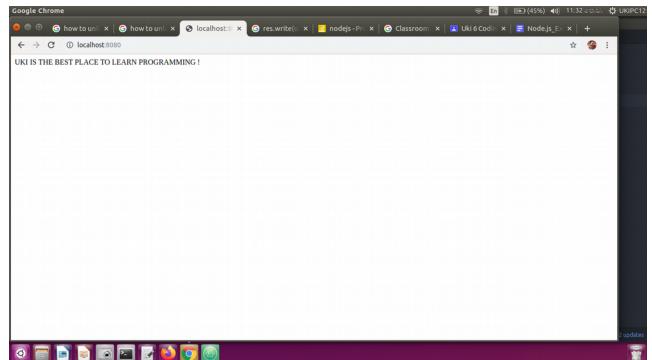
5. Install the package “upper-case” using NPM and create a Node.js file that will convert the output "Uki is the best place to learn programming !" into upper-case letters.



The screenshot shows the Atom code editor with the following details:

- Project Path:** Uppercase.js — ~/Desktop/nodejs — Atom
- File:** Uppercase.js
- Code Content:**

```
1 var http = require('http');
2 var uc = require('upper-case');
3 http.createServer(function (req, res) {
4   res.writeHead(200, {'Content-Type': 'text/html'});
5   res.write("Uki is the best place to learn programming !".toUpperCase());
6   res.end();
7 }).listen(8088);
```
- Bottom Status Bar:** UTF-8, label, 19 minutes, 44, 11:32, UKPCN



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

```
Terminal eventhandler.js ~/eventhandler.js 13:24
Project
  nodejs
    404.png
    40404N HEAD.png
    an2z.png
    ASST.png
    delete_3.png
    delete.js
    delete3.41.png
    eventhandler.js
    head.html
    head.png
    header.png
    header41.png
    HEATML.png
    index.js
    index.html
    index.png
    index_repeated.png
    loading.png
    tall.html.png
    tall.html.js
    tallimage.png
    recommended.png
    simple.js
    tail.html
    tail.png
    UI.js

ukistul2@ukipc12:~/Desktop/nodejs$ node Uppercase.js
/home/ukistul2/Desktop/nodejs/Uppercase.js:5
  res.write(uc("Uki is the best place to learn programming!"));
          ^
TypeError: uc is not a function
    at Server.<anonymous> (/home/ukistul2/Desktop/nodejs/Uppercase.js:5:13)
    at Server.emit (events.js:182:13)
    at parserOnIncoming (_http_server.js:652:12)
    at HTTPParser.parserOnHeadersComplete (_http_common.js:109:17)
ukistul2@ukipc12:~/Desktop/nodejs$ node eventhandler.js
I bark when I see strangers !
ukistul2@ukipc12:~/Desktop/nodejs$ █
```



```
eventsHandler.js:1
  var events = require('events');
  ^

eventsHandler.js:2
  var eventEmitter = new events.EventEmitter();
  ^

eventsHandler.js:3
  ^

eventsHandler.js:4
  //Create an event handler:
  ^

eventsHandler.js:5
  var myEventHandler = function () {
  ^

eventsHandler.js:6
    console.log("I bark when I see strangers !");
  ^

eventsHandler.js:7
  ^

eventsHandler.js:8
  ^

eventsHandler.js:9
  //Assign the event handler to an event:
  ^

eventsHandler.js:10
  eventEmitter.on('bark', myEventHandler);
  ^

eventsHandler.js:11
  ^

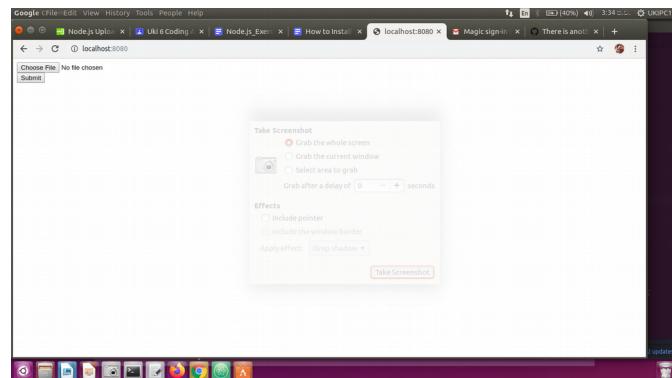
eventsHandler.js:12
  //Fire the 'scream' event:
  ^

eventsHandler.js:13
  eventEmitter.emit('bark');
  ^

eventsHandler.js:14
  ^
```

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

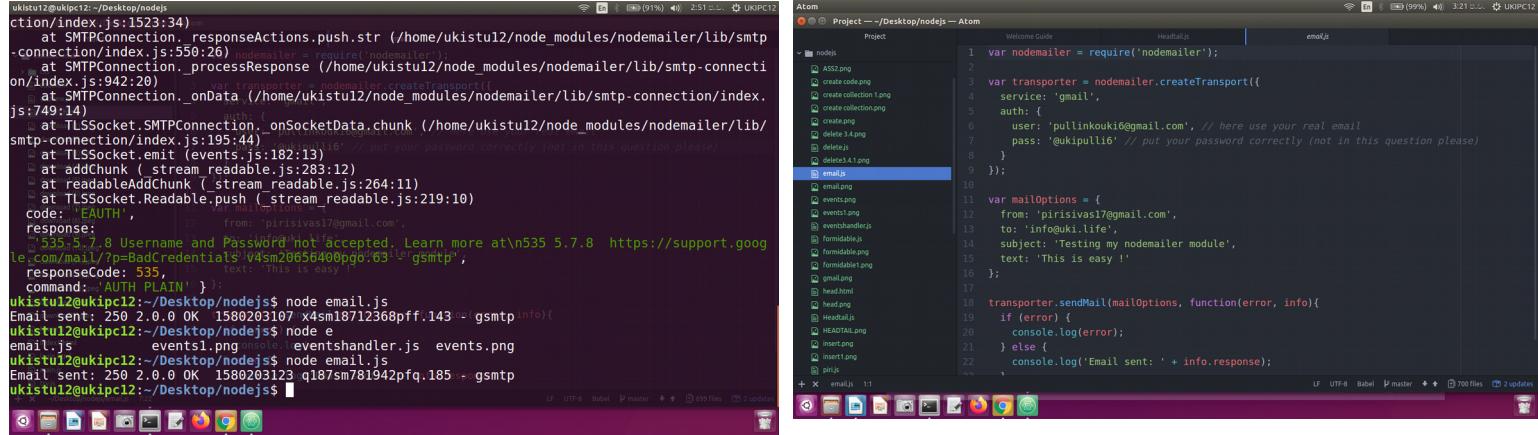
7.1 Save that uploaded file into your Documents directory.



## 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.



The screenshot shows a Linux desktop environment with a terminal window and an Atom code editor window. The terminal window on the left displays the command-line output of a Node.js application. The Atom editor on the right shows a file named 'email.js' containing the code for sending an email using the nodemailer module.

```
ukistu12@ukipc12:~/Desktop/nodejs$ node index.js
var nodemailer = require('nodemailer');
var transporter = nodemailer.createTransport({
  service: 'gmail',
  auth: {
    user: 'pullinkouki@gmail.com', // here use your real email
    pass: 'ukipulli6' // put your password correctly (not in this question please)
  }
});
var mailOptions = {
  from: 'pirisivas17@gmail.com',
  to: 'info@uki.life',
  subject: 'Testing my nodemailer module',
  text: 'This is easy !'
};
transporter.sendMail(mailOptions, function(error, info){
  if (error) {
    console.log(error);
  } else {
    console.log('Email sent: ' + info.response);
  }
});

ukistu12@ukipc12:~/Desktop/nodejs$
```

```
1 var nodemailer = require('nodemailer');
2
3 var transporter = nodemailer.createTransport({
4   service: 'gmail',
5   auth: {
6     user: 'pullinkouki@gmail.com', // here use your real email
7     pass: 'ukipulli6' // put your password correctly (not in this question please)
8   }
9 });
10
11 var mailOptions = {
12   from: 'pirisivas17@gmail.com',
13   to: 'info@uki.life',
14   subject: 'Testing my nodemailer module',
15   text: 'This is easy !'
16 };
17
18 transporter.sendMail(mailOptions, function(error, info){
19   if (error) {
20     console.log(error);
21   } else {
22     console.log('Email sent: ' + info.response);
23   }
24 });

ukistu12@ukipc12:~/Desktop/nodejs$
```