

## Node.js Exercise 1

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.

```
4/programming/nodejs — Atom
exercise1.js x exer1.js x
1 var http = require('http');
2 var sample = require('./exer1.js');
3
4 http.createServer(function (req, res) {
5   res.writeHead(200, {'Content-Type': 'text/plain'});
6   res.write("Sum Number = " + sample.sumNum(123 , 321));
7   res.write(" Average Number is = " + sample.average(123,321));
8   res.end();
9 }).listen(8080);
10
```

```
programming/nodejs — Atom
exercise1.js x exer1.js
1 exports.sumNum = function (a,b){
2   return a+b ;
3 }
4
5 exports.average = function (a,b){
6   return (a+b)/2 ;
7 }
8
```

```
← → ↻ ⓘ localhost:8080
Sum Number = 444 Average Number is = 222
```

```
ukistu11@ukipc11: ~/Documents/uki4/programming/nodejs
ukistu11@ukipc11: ~/Documents/uki4/programming/nodejs
ukistu11@ukipc11:~/Documents/uki4/programming/nodejs$ nodemon exercise1.js
[nodemon] 1.17.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node exercise1.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)

```
uki4/programming/nodejs — Atom
firstnode.js
1 var http = require('http');
2
3
4 http.createServer(function (req, res) {
5   res.writeHead(200, {'Content-Type': 'text/html'});
6   res.write(" Welcome to Uki, I'm Vinoliya");
7   res.end();
8 }).listen(8000);
9
```



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

#### 3.1 Write a paragraph about Uki into that file

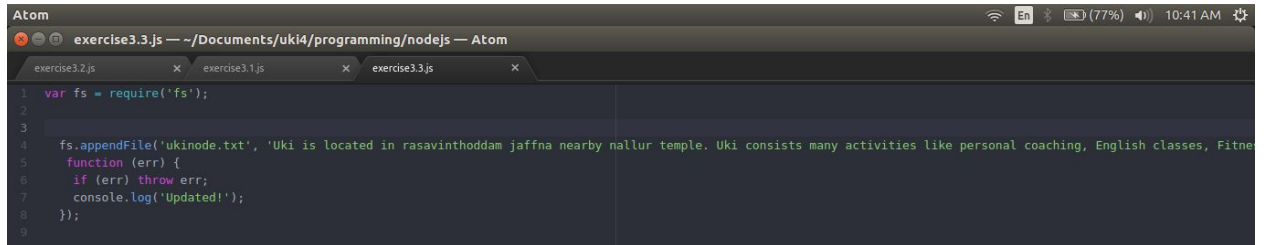
```
uki4/programming/nodejs — Atom
firstnode.js
1 var fs = require('fs');
2
3 fs.appendFile('ukinode.txt', 'It is a full scholarship based Accelerator program for Coding in Jaffna, Sri Lanka targeting students
4 function (err){
5   if (err) throw err;
6   console.log("finished");
7 });
8
```

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

```
Atom
exercise3.2.js — ~/Documents/uki4/programming/nodejs — Atom
exercise3.1.js x ukinode.txt x exercise3.2.js x
1 var http = require('http');
2 var fs = require('fs');
3
4 http.createServer(function (req, res){
5   fs.readFile('ukinode.txt', function(err, data) {
6     res.writeHead(200, {'Content-Type': 'text/html'});
7     res.write("Welcome to Uki, I'm Vinoliya ");
8     res.write(data);
9     res.end();
10   });
11 }).listen(8000);
12
```

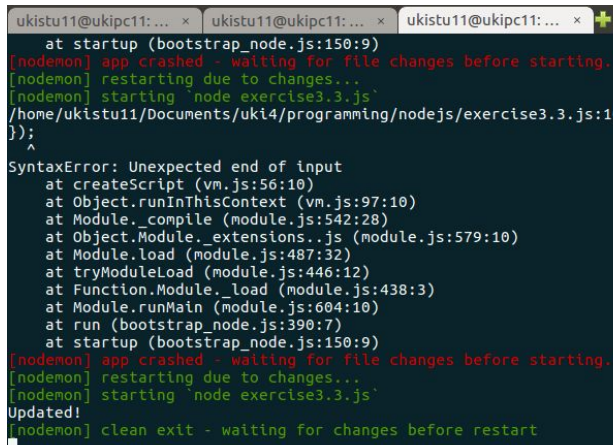


#### 3.3 Append another paragraph about Uki and now serve the new file



The screenshot shows the Atom editor with a file named `exercise3.3.js` open. The code in the file is as follows:

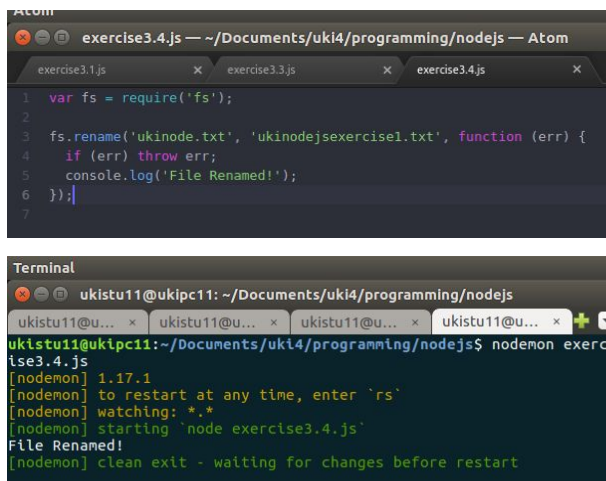
```
1 var fs = require('fs');
2
3
4 fs.appendFile('ukinode.txt', 'Uki is located in rasavinthoddam jaffna nearby nallur temple. Uki consists many activities like personal coaching, English classes, Fitne
5 function (err) {
6   if (err) throw err;
7   console.log('Updated!');
8 }
9
```



The screenshot shows a terminal window with the following output:

```
at startup (bootstrap_node.js:150:9)
[nodemon] app crashed - waiting for file changes before starting.
[nodemon] restarting due to changes...
[nodemon] starting 'node exercise3.3.js'
/home/ukistu1/Documents/uki4/programming/nodejs/exercise3.3.js:1
});
^
SyntaxError: Unexpected end of input
    at createScript (vm.js:56:10)
    at Object.runInThisContext (vm.js:97:10)
    at Module._compile (module.js:542:28)
    at Object.Module._extensions..js (module.js:579:10)
    at Module.load (module.js:487:32)
    at tryModuleLoad (module.js:446:12)
    at Function.Module._load (module.js:438:3)
    at Module.runMain (module.js:604:10)
    at run (bootstrap_node.js:390:7)
    at startup (bootstrap_node.js:150:9)
[nodemon] app crashed - waiting for file changes before starting.
[nodemon] restarting due to changes...
[nodemon] starting 'node exercise3.3.js'
Updated!
[nodemon] clean exit - waiting for changes before restart
```

### 3.4 Rename the file as ukinodejsexercise1.txt



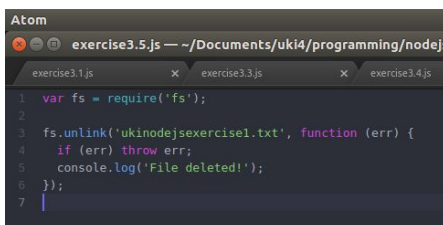
The top part of the screenshot shows the Atom editor with a file named `exercise3.4.js` open. The code in the file is as follows:

```
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
```

The bottom part of the screenshot shows a terminal window with the following output:

```
ukistu11@ukipc11: ~/Documents/uki4/programming/nodejs
ukistu11@ukipc11:~/Documents/uki4/programming/nodejs$ nodemon exercise3.4.js
[nodemon] 1.17.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node exercise3.4.js'
File Renamed!
[nodemon] clean exit - waiting for changes before restart
```

### 3.5 Delete the file you created



The screenshot shows the Atom editor with a file named `exercise3.5.js` open. The code in the file is as follows:

```
1 var fs = require('fs');
2
3 fs.unlink('ukinodejsexercise1.txt', function (err) {
4   if (err) throw err;
5   console.log('File deleted!');
6 });
7
```

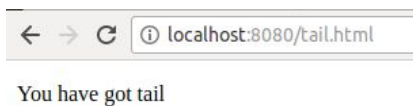
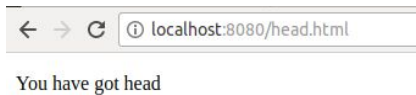
```
Terminal
ukistu11@ukipc11: ~/Documents/uki4/programming/nodejs
ukistu11@ukipc11:~/Documents/uki4/programming/nodejs$ nodemon exercise3.5.js
[nodemon] 1.17.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting 'node exercise3.5.js'
File deleted!
[nodemon] clean exit - waiting for changes before restart
```

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

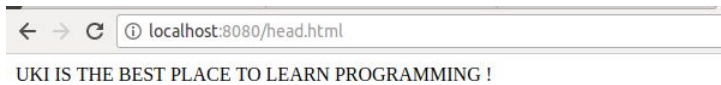


```
Atom
tail.html -- ~/Documents/uki4/programming/nodejs -- Atom
exercise4.js x head.html x tail.html x
1 <!DOCTYPE html>
2 <html lang="en" dir="ltr">
3 <head>
4 <meta charset="utf-8">
5 <title></title>
6 </head>
7 <body>
8 <p>You have got tail</p>
9 </body>
10 </html>
11
```

```
Atom
head.html -- ~/Documents/uki4/programming/nodejs -- Atom
exercise4.js x head.html x tail.html x
1 <!DOCTYPE html>
2 <html lang="en" dir="ltr">
3 <head>
4 <meta charset="utf-8">
5 <title></title>
6 </head>
7 <body>
8 <p>You have got head</p>
9 </body>
10 </html>
11
```

```
Atom
exercise4.js -- ~/Documents/uki4/programming/nodejs -- Atom
exercise4.js x head.html x tail.html x
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 = "." + 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);
18
```

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.



```
Atom
exercise5.js — ~/Documents/uki4/programming/nodejs — Atom
exercise5.js
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(uc("Uki is the best place to learn programming !"));
6   res.end();
7 }).listen(8080);
8
```

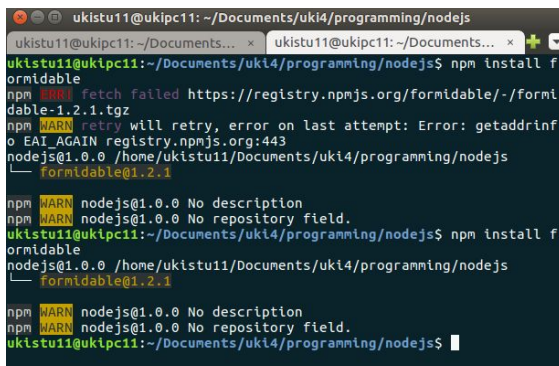
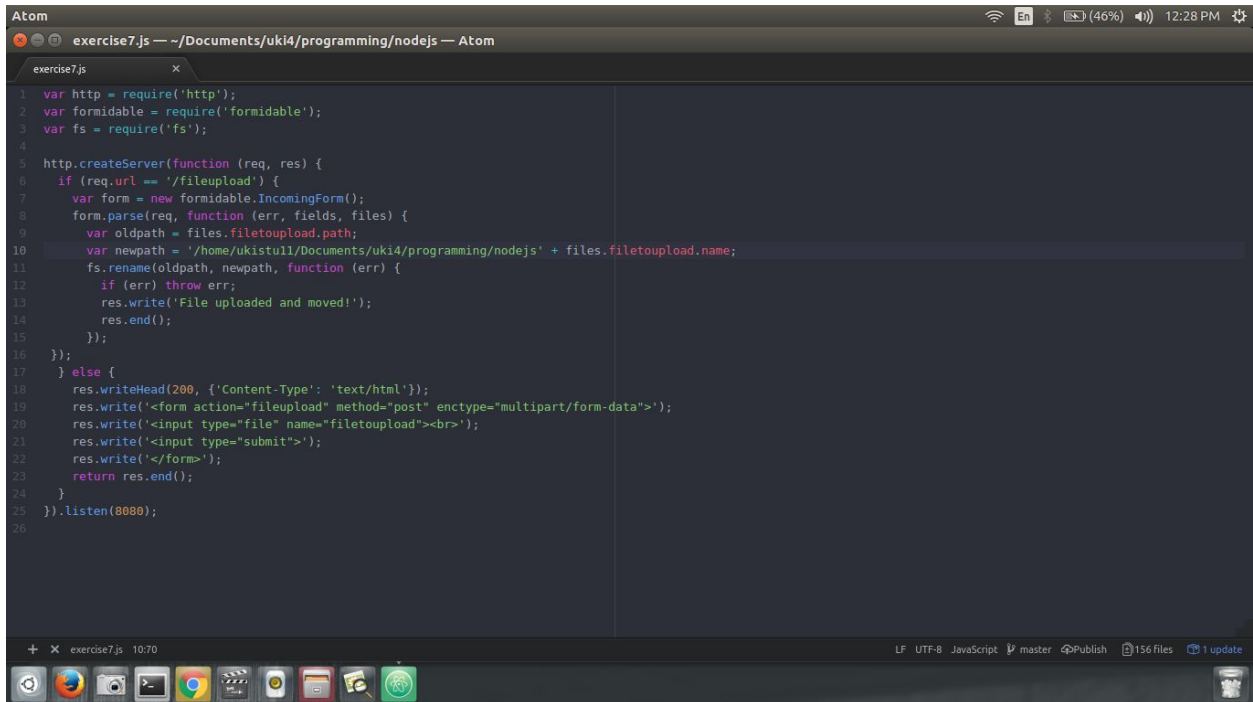
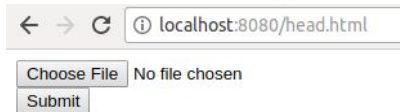
6. Create an event handler function that will say “I bark when I see strangers !” when a "bark" event is fired.

```
Atom
exercise6.js — ~/Documents/uki4/programming/nodejs — Atom
exercise5.js exercise6.js
1 var events = require('events');
2 var eventEmitter = new events.EventEmitter();
3
4
5 var myEventHandler = function () {
6   console.log('I bark when I see strangers !');
7 }
8
9
10 eventEmitter.on('bark', myEventHandler);
11
12
13 eventEmitter.emit('bark');
14
```

```
Terminal
ukistu11@ukipc11: ~/Documents/uki4/programming/nodejs
ukistu11@ukipc11:~/Documents/uki4/programming/nodejs$ nodemon exercise6.js
[nodemon] 1.17.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node exercise6.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.

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.



Atom

exercise8.js — ~/Documents/uki4/programming/nodejs — Atom

```
1 var nodemailer = require('nodemailer');
2
3 var transporter = nodemailer.createTransport({
4   service: 'gmail',
5   auth: {
6     user: 'vinoliyakavignan1997.com',
7     pass: '*****'
8   }
9 });
10
11 var mailOptions = {
12   from: 'vinoliyakavignana1997@gmail.com',
13   to: 'info@uki.life',
14   subject: 'Testing my nodemailer module',
15   text: 'That was 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 });
25
```

exercise8.js 7:19

LF UTF-8 JavaScript master Publish 195 files 1 update

ukistu11@ukipc11: ~/Documents/uki4/programming/nodejs

```
at Object.exports._errnoException (util.js:1018:11)
at exports._exceptionWithHostPort (util.js:1041:20)
at TCPConnectWrap.afterConnect [as oncomplete] (net.js:1086:14)
code: 'ECONNECTION',
errno: 'ENETUNREACH',
syscall: 'connect',
address: '2404:6800:4003:c03::6c',
port: 465,
command: 'CONN' }
[nodemon] clean exit - waiting for changes before restart
[nodemon] restarting due to changes...
[nodemon] starting 'node exercise8.js'
{ Error: getaddrinfo EAI_AGAIN smtp.gmail.com:465
  at Object.exports._errnoException (util.js:1018:11)
  at errnoException (dns.js:33:15)
  at GetAddrInfoReqWrap.onlookup [as oncomplete] (dns.js:76:26)
  code: 'ECONNECTION',
  errno: 'EAI_AGAIN',
  syscall: 'getaddrinfo',
  hostname: 'smtp.gmail.com',
  host: 'smtp.gmail.com',
  port: 465,
  command: 'CONN' }
[nodemon] clean exit - waiting for changes before restart
[nodemon] restarting due to changes...
[nodemon] starting 'node exercise8.js'
{ Error: getaddrinfo EAI_AGAIN smtp.gmail.com:465
  at Object.exports._errnoException (util.js:1018:11)
  at errnoException (dns.js:33:15)
  at GetAddrInfoReqWrap.onlookup [as oncomplete] (dns.js:76:26)
  code: 'ECONNECTION',
  errno: 'EAI_AGAIN',
  syscall: 'getaddrinfo',
  hostname: 'smtp.gmail.com',
  host: 'smtp.gmail.com',
  port: 465,
  command: 'CONN' }
[nodemon] clean exit - waiting for changes before restart
```