

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

```
1  var http = require('http');
2  var dt = require('./myfirstmodule');
3
4  http.createServer(function (req, res) {
5      res.writeHead(200, {'Content-Type': 'text/html'});
6      res.write("sum of two numbers is : " + dt.addNum(123,321) + "<br> "
7      | + "Average of two numbers: " + dt.avg(123,321));
8
9      res.end();
10 }).listen(8080);
11
```

```
1  exports.addNum = function (a,b) {
2      return a + b ;
3  };
4  exports.avg = function (a,b) {
5      return (a + b) / 2 ;
6  };
7
```

← ⓘ | localhost:8080

sum of two numbers is : 444
Average of two numbers: 222

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)

```
1  var http = require('http');
2  http.createServer(function (req, res) {
3      res.writeHead(200, {'Content-Type': 'text/html'});
4      res.end('Welcome to Uki. I am Sriram');
5  }).listen(8000);
6
```

```
ukistu05@UKIPC05:~/Documents/uki2/coding/nodejs$ node ex-q-1
```

← ⓘ | localhost:8000

Welcome to Uki. I am Sriram

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

3.1 Write a paragraph about Uki into that file

```
1 var fs = require('fs');
2
3 fs.appendFile('Ukinode.txt', '
4 This is a joint initiative by Yarl IT Hub and SERVE Foundation. Yarl IT Hub has been functioning for the last five years
5 as a not for profit social enterprise focusing on technology entrepreneurship in northern Sri Lanka and
6 SERVE Foundation is a four year old foundation focusing on using technology to uplift secondary education
7 standards in rural schools in northern Sri Lanka.
8 ');
9
10 console.log('Saved!');
```

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

```
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);
```

Welcome ... ex-q-3.2.js ex-q-8.js ex-q-3.1.js ex-q-3.5.js head.html tail.html open.js main3.js append.js update.js

This is a joint initiative by Yarl IT Hub and SERVE Foundation. Yarl IT Hub has been functioning for the last five years as a not for profit social enterprise focusing on technology entrepreneurship in northern Sri Lanka and SERVE Foundation is a four year old foundation focusing on using technology to uplift secondary education standards in rural schools in northern Sri Lanka.

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

```
1 var fs = require('fs');
2
3 fs.appendFile('Ukinode.txt', ' This Coding Accelerator program is an intensive training on full-
4 stack development with server-side Node.js and modern, client-side, front-end libraries and
5 frameworks. Throughout this intensive course the student will study professional software development
6 techniques and practices as he/she advances his/her skills on JavaScript.
7 ');
8
9 console.log('Updated!');
```

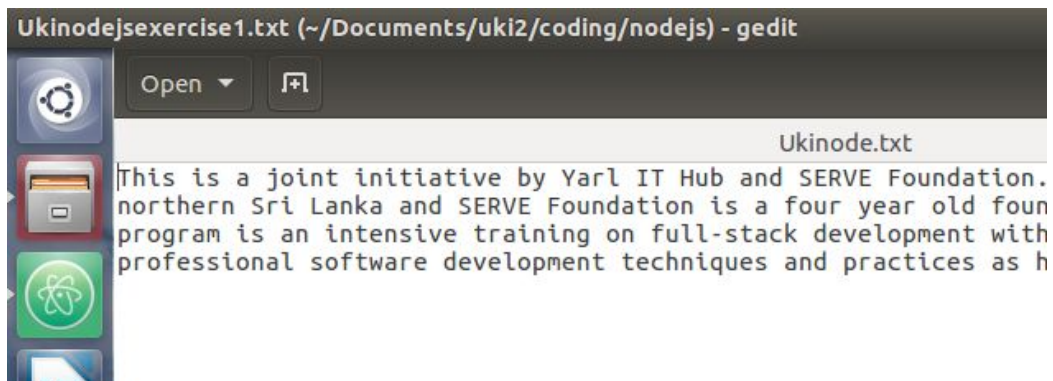
```
ukistu05@UKIPC05:~/Documents/uki2/coding/nodejs$ node ex-q-3.3
Updated!
ukistu05@UKIPC05:~/Documents/uki2/coding/nodejs$
```

This is a joint initiative by Yarl IT Hub and SERVE Foundation. Yarl IT Hub has been functioning for the last five years as a not for profit social enterprise focusing on technology entrepreneurship in northern Sri Lanka and SERVE Foundation is a four year old foundation focusing on using technology to uplift secondary education standards in rural schools in northern Sri Lanka. This Coding Accelerator program is an intensive training on full-stack development with server-side Node.js and modern, client-side, front-end libraries and frameworks. Throughout this intensive course the student will study professional software development techniques and practices as he/she advances his/her skills on JavaScript.

3.4 Rename the file as ukinodejsexercise1.txt

```
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
```

```
ukistu05@UKIPC05:~/Documents/uki2/coding/nodejs$ node ex-q-3.4
File Renamed!
ukistu05@UKIPC05:~/Documents/uki2/coding/nodejs$
```



3.5 Delete the file you created

```
ukistu05@UKIPC05:~/Documents/uki2/coding/nodejs$ node ex-q-3.5
File deleted!
ukistu05@UKIPC05:~/Documents/uki2/coding/nodejs$
```

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.

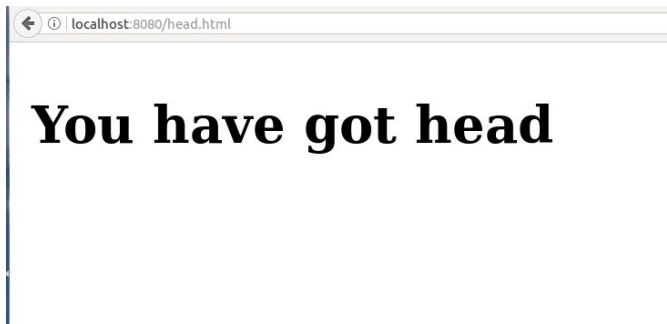
If you have followed the correct steps you should see two different results when opening these two addresses:

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

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

```
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 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);
```



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.

```
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
```



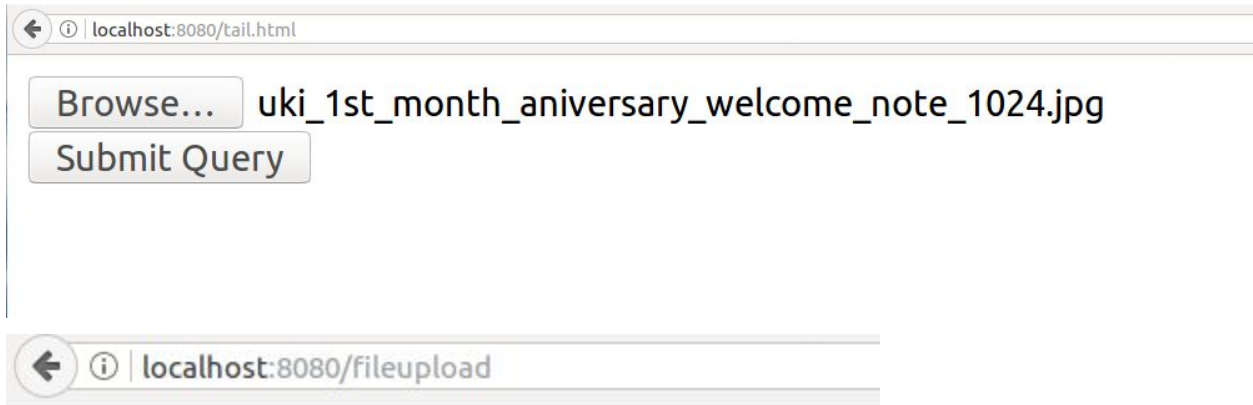
6. Create an event handler function that will say "I bark when I see strangers !" when a "bark" event is fired.

```
1  var events = require('events');
2  var EventEmitter = new events.EventEmitter();
3
4  //Create an event handler:
5  var myEventHandler = function () {
6      console.log('I bark when I see strangers !');
7  }
8
9  //Assign the eventhandler to an event:
10  EventEmitter.on('bark', myEventHandler);
11
12  //Fire the 'scream' event:
13  EventEmitter.emit('bark');
14
```

```
ukistu05@UKIPC05:~/Documents/uki2/coding/nodejs$ node ex-q-6
I bark when I see strangers !
ukistu05@UKIPC05:~/Documents/uki2/coding/nodejs$
```

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.



The image shows two browser screenshots. The top screenshot is of a page at `localhost:8080/tail.html` with a file upload form. It contains a "Browse..." button, a text input field with the filename `uki_1st_month_aniversary_welcome_note_1024.jpg`, and a "Submit Query" button. The bottom screenshot is of a page at `localhost:8080/fileupload` displaying the message "File uploaded and moved!" in a large, bold, black monospace font.

File uploaded and moved!

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.

```
1  var nodemailer = require('nodemailer');
2
3  var transporter = nodemailer.createTransport({
4    service: 'gmail',
5    auth: {
6      user: 'jegaatheesansriram@gmail.com',
7      pass: 'srisriram'
8    }
9  });
10 var mailOptions = {
11   from: 'jegaatheesansriram@gmail.com',
12   to: 'info@uki.life',
13   subject: 'Testing my nodemailer module',
14   text: 'This is easy!'
15 };
16
17 transporter.sendMail(mailOptions, function(error, info){
18   if (error) {
19     console.log(error);
20   } else {
21     console.log('Email sent: ' + info.response);
22   }
23 });
24
```