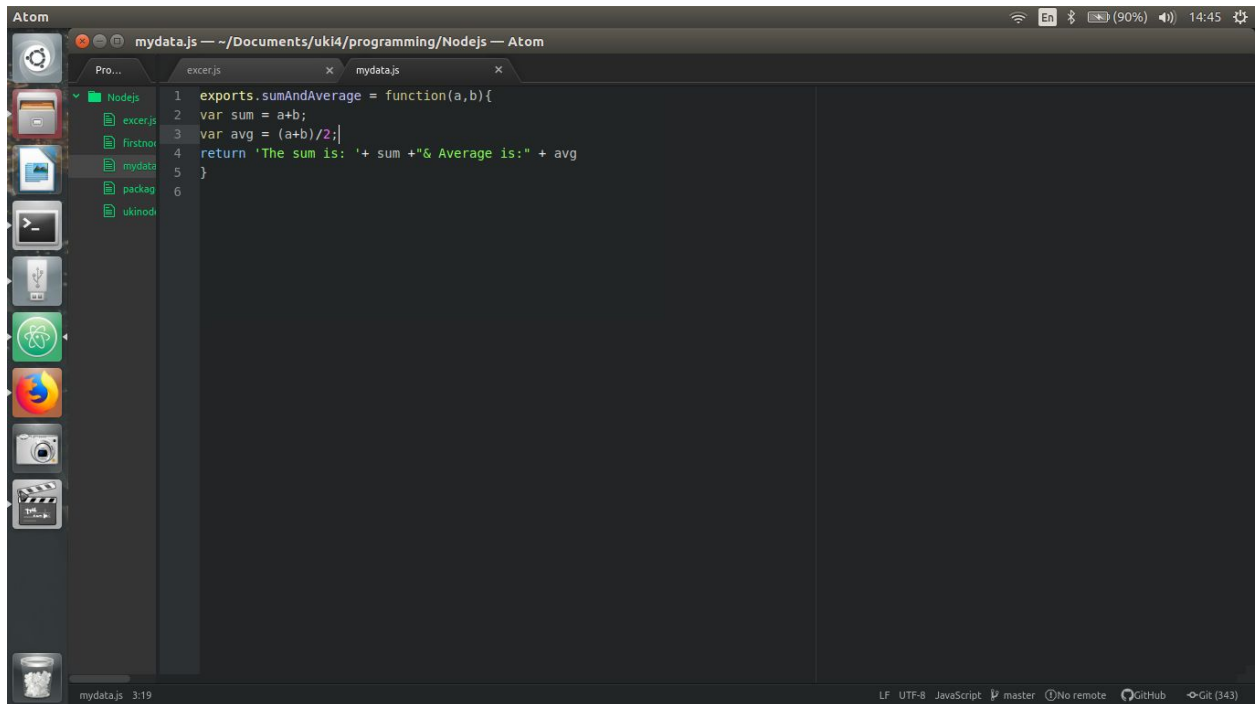


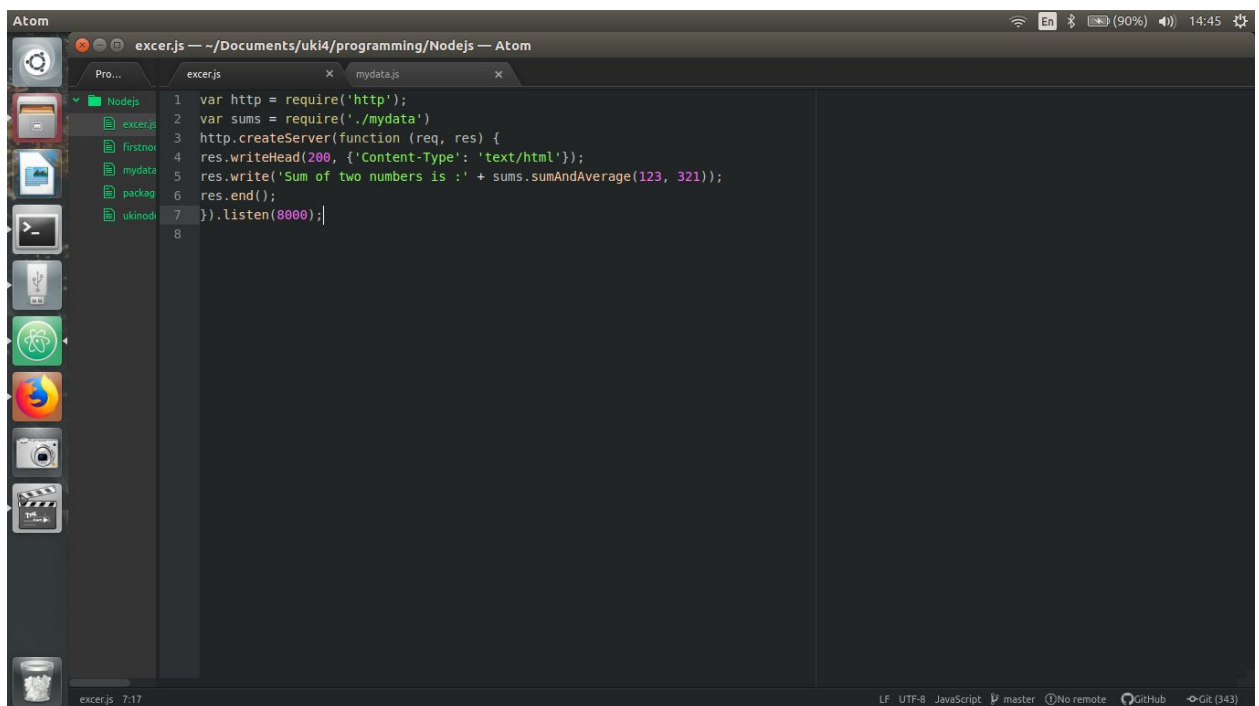
1. Create a custom module which returns the sum and average of any two numbers passed into it. Require the module and run the server by passing 123 and 321 so that the server prints out the sum and average.



A screenshot of the Atom text editor. The title bar shows 'Atom' and the file path 'mydata.js -- ~/Documents/uki4/programming/Nodejs -- Atom'. The left sidebar shows a file explorer with a tree view containing 'Nodejs', 'excer.js', 'firstnox', 'mydata', 'packag', and 'ukinodi'. The main editor area displays the content of 'mydata.js' with the following code:

```
1 exports.sumAndAverage = function(a,b){
2   var sum = a+b;
3   var avg = (a+b)/2;
4   return 'The sum is: ' + sum + '& Average is:' + avg
5 }
6
```

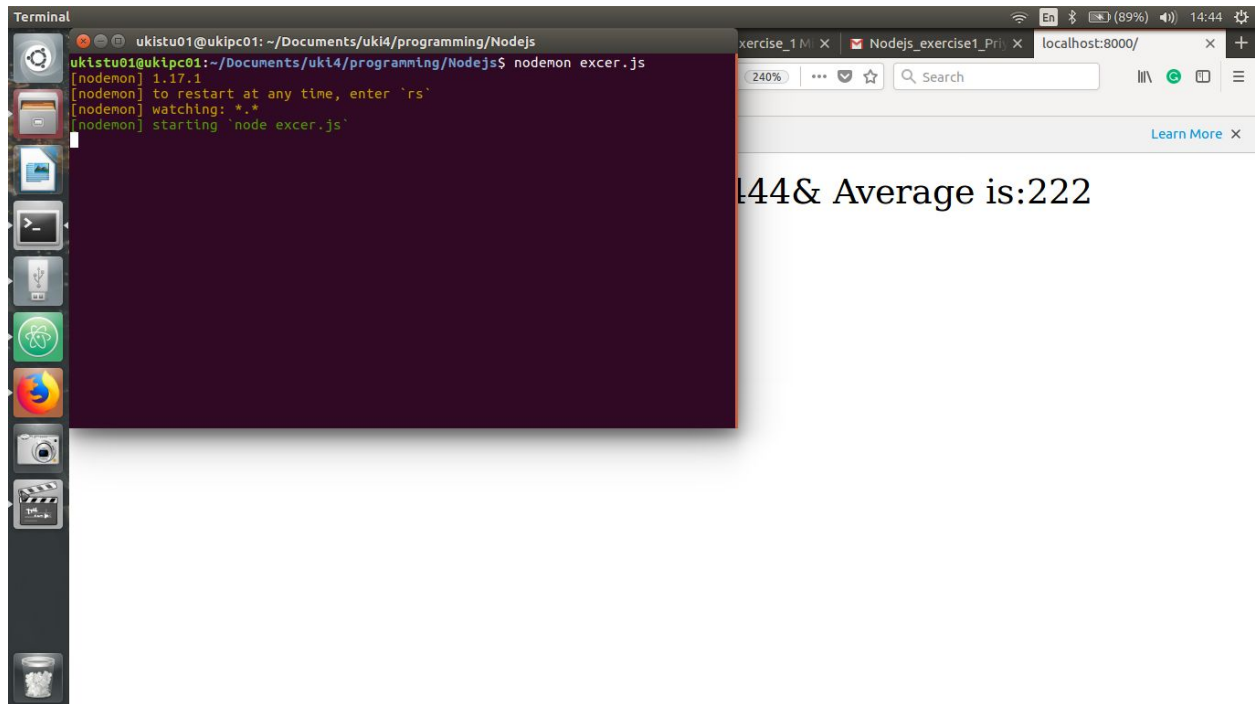
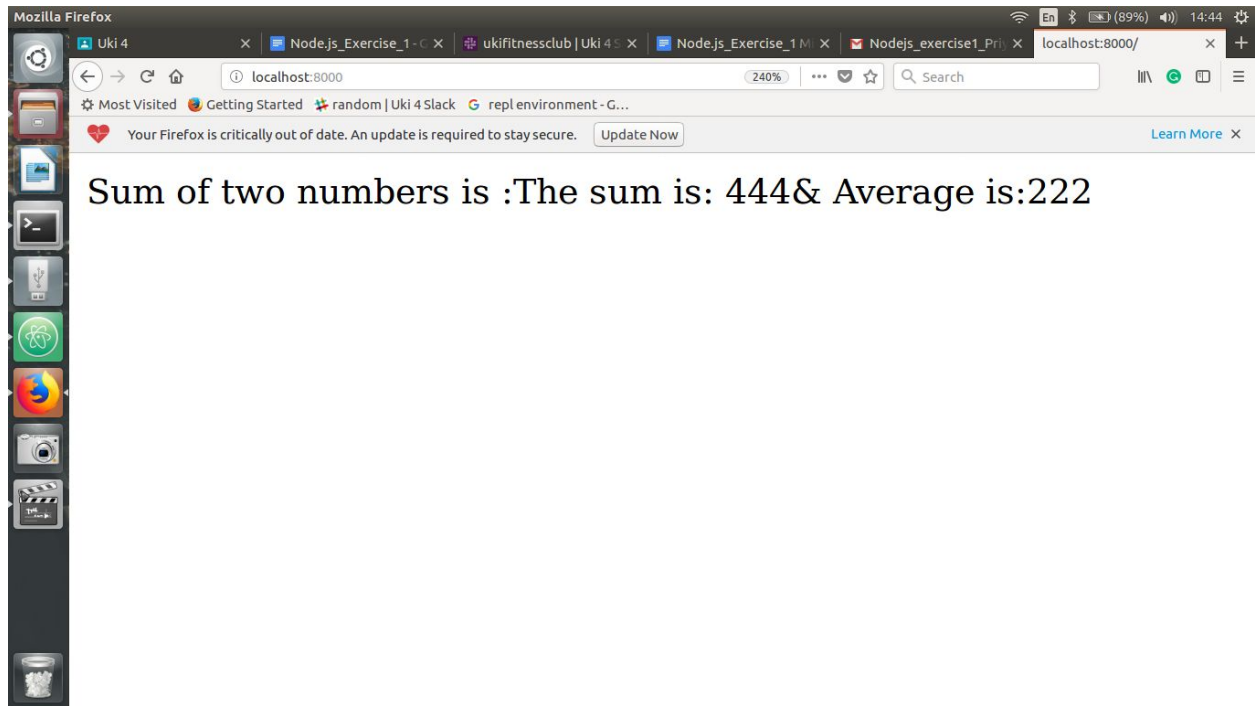
The status bar at the bottom indicates 'mydata.js 3:19', 'LF UTF-8 JavaScript', and 'master' with a 'No remote' status.



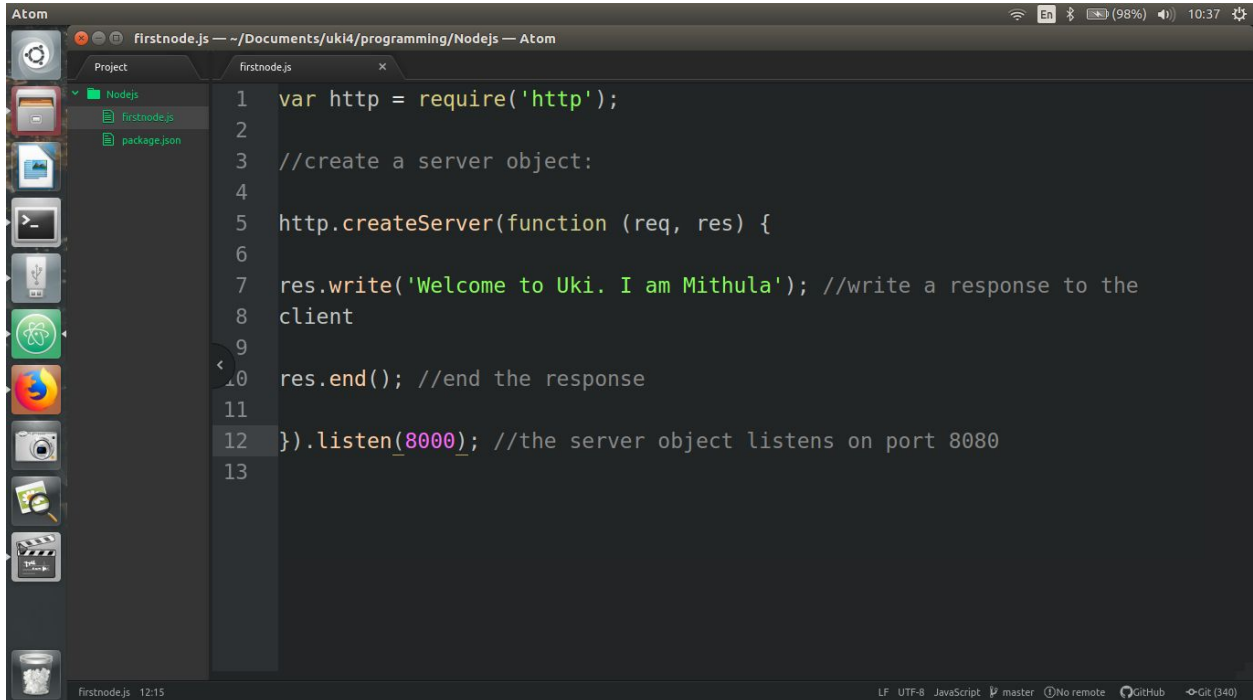
A screenshot of the Atom text editor. The title bar shows 'Atom' and the file path 'excer.js -- ~/Documents/uki4/programming/Nodejs -- Atom'. The left sidebar shows a file explorer with a tree view containing 'Nodejs', 'excer.js', 'firstnox', 'mydata', 'packag', and 'ukinodi'. The main editor area displays the content of 'excer.js' with the following code:

```
1 var http = require('http');
2 var sums = require('./mydata')
3 http.createServer(function (req, res) {
4   res.writeHead(200, {'Content-Type': 'text/html'});
5   res.write('Sum of two numbers is : ' + sums.sumAndAverage(123, 321));
6   res.end();
7 }).listen(8000);
8
```

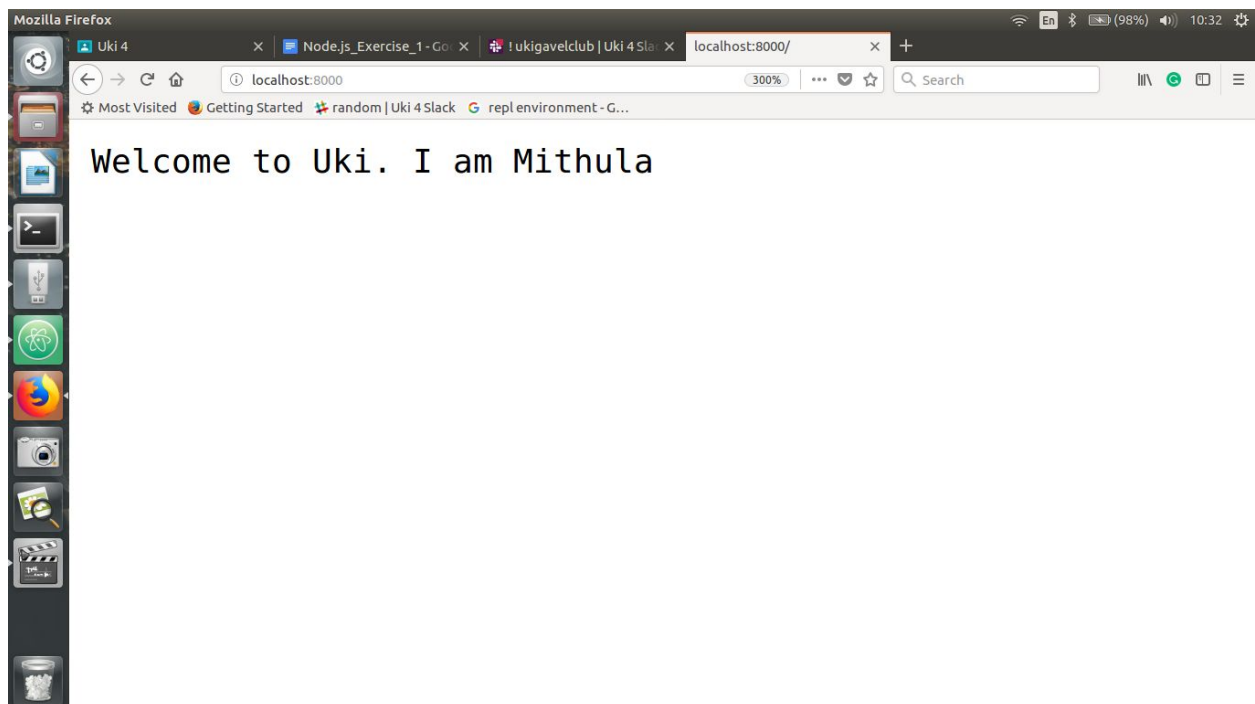
The status bar at the bottom indicates 'excer.js 7:17', 'LF UTF-8 JavaScript', and 'master' with a 'No remote' status.



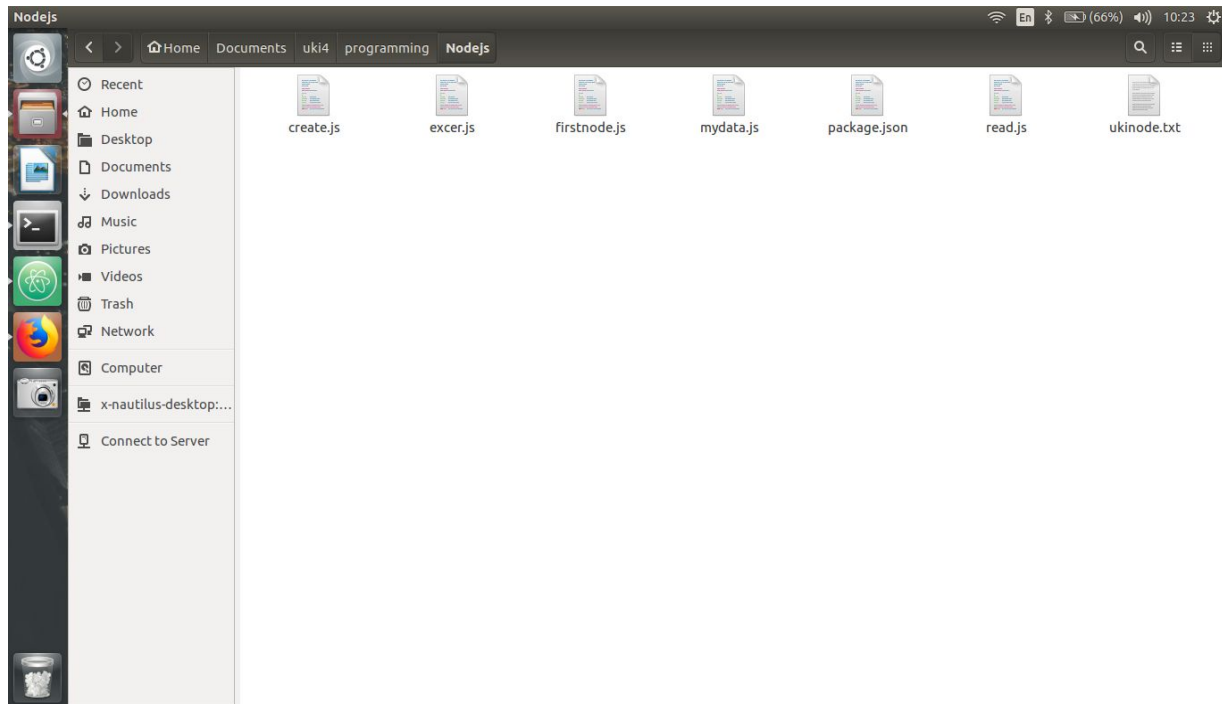
2. Create a simple http server and print “Welcome to Uki. I am **yourname**” when a request is sent to your server via the port 8000. (Note - Change different port numbers and check)



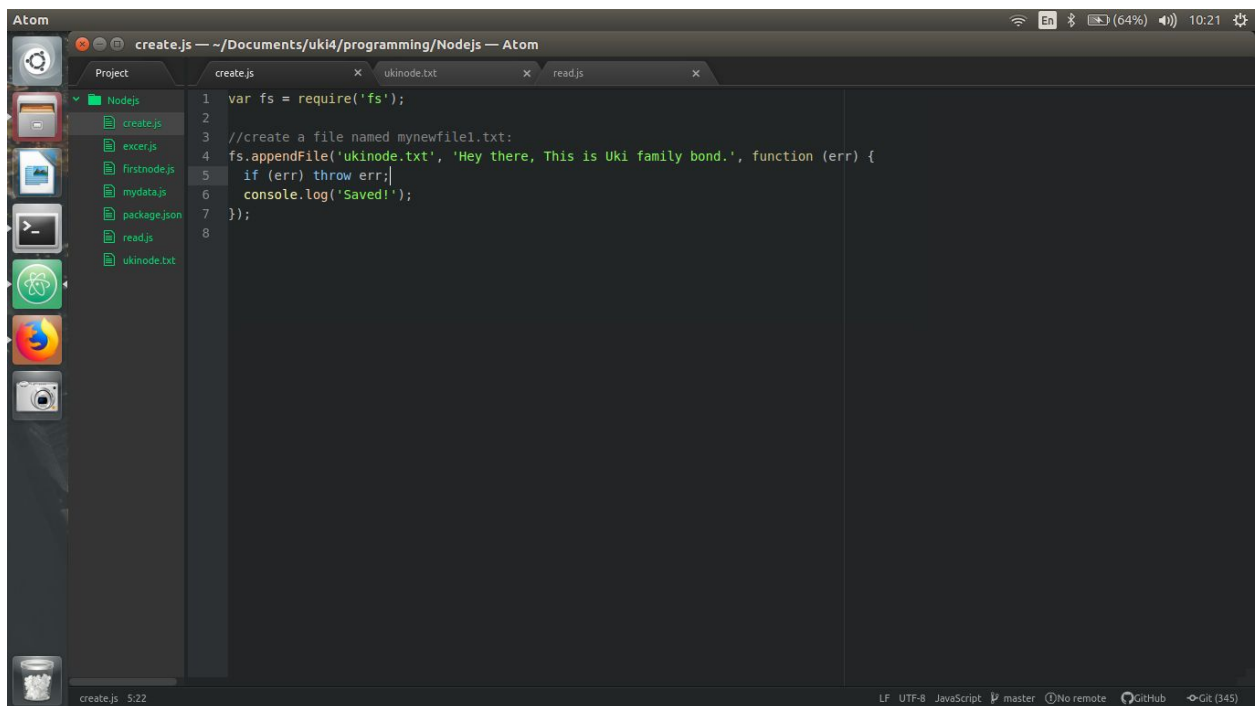
```
1 var http = require('http');
2
3 //create a server object:
4
5 http.createServer(function (req, res) {
6
7   res.write('Welcome to Uki. I am Mithula'); //write a response to the
8   client
9
10  res.end(); //end the response
11
12 }).listen(8080); //the server object listens on port 8080
13
```



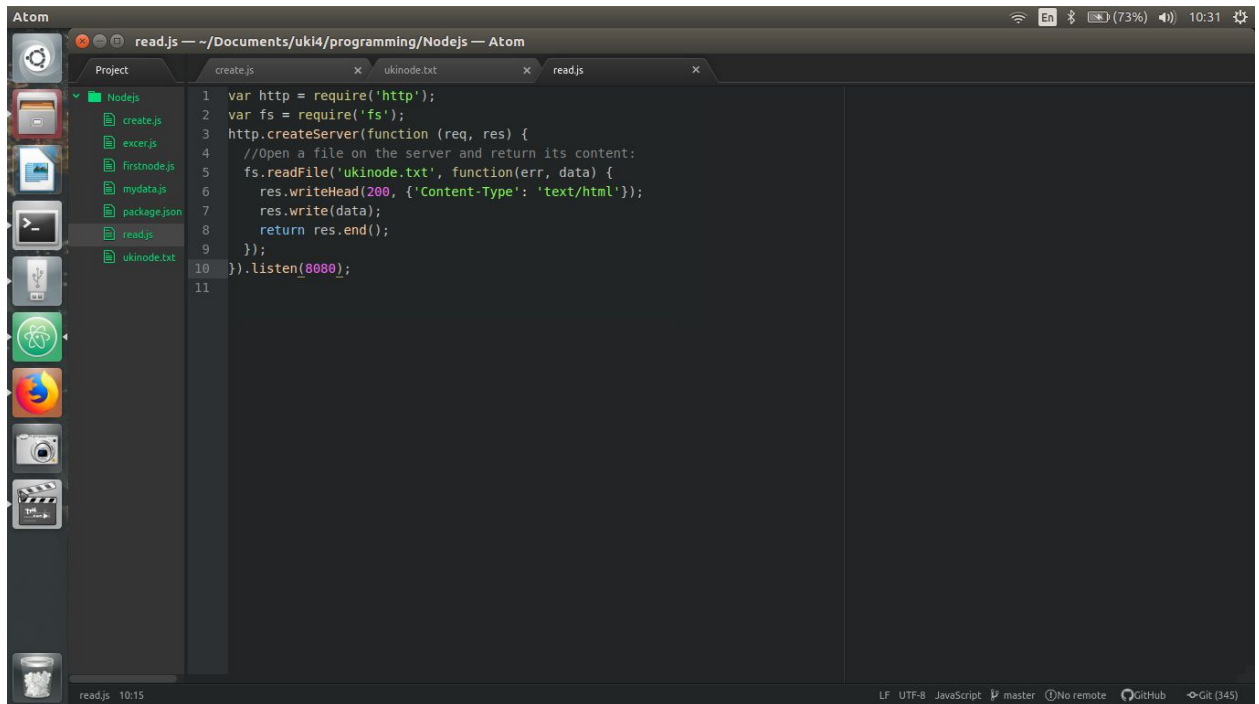
3. Using the file system module create a new file called ukinode.txt



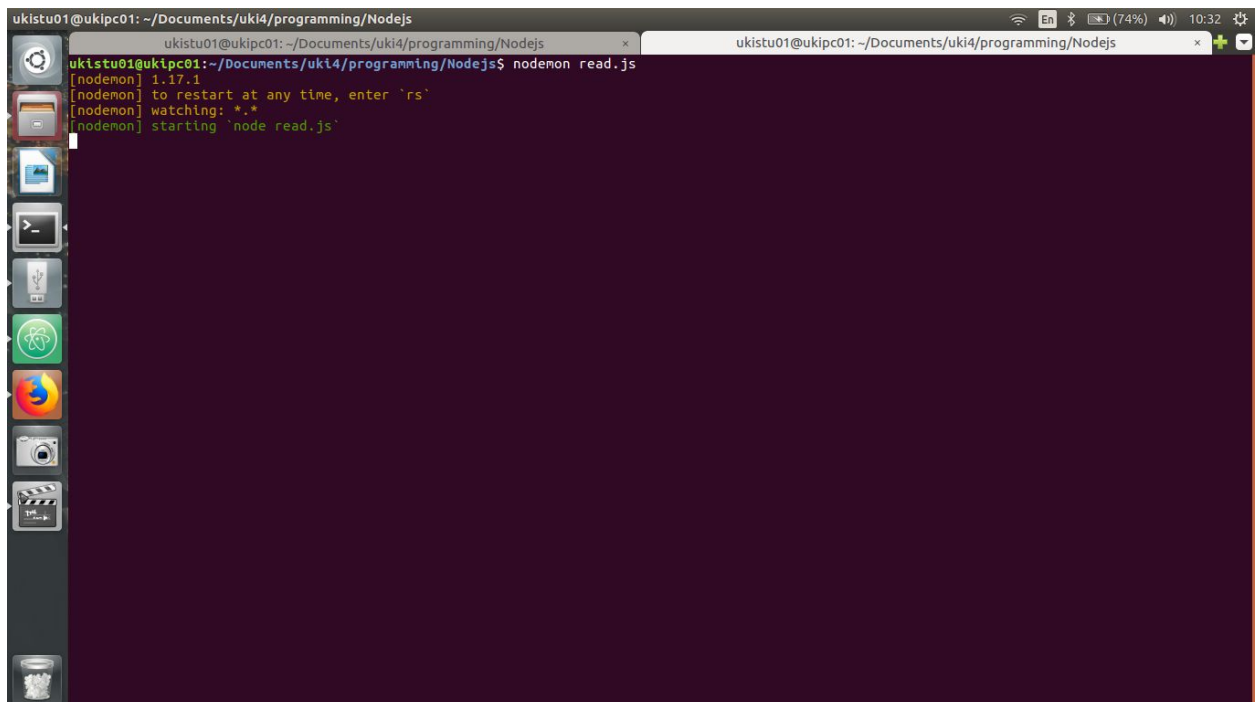
3.1 Write a paragraph about Uki into that file



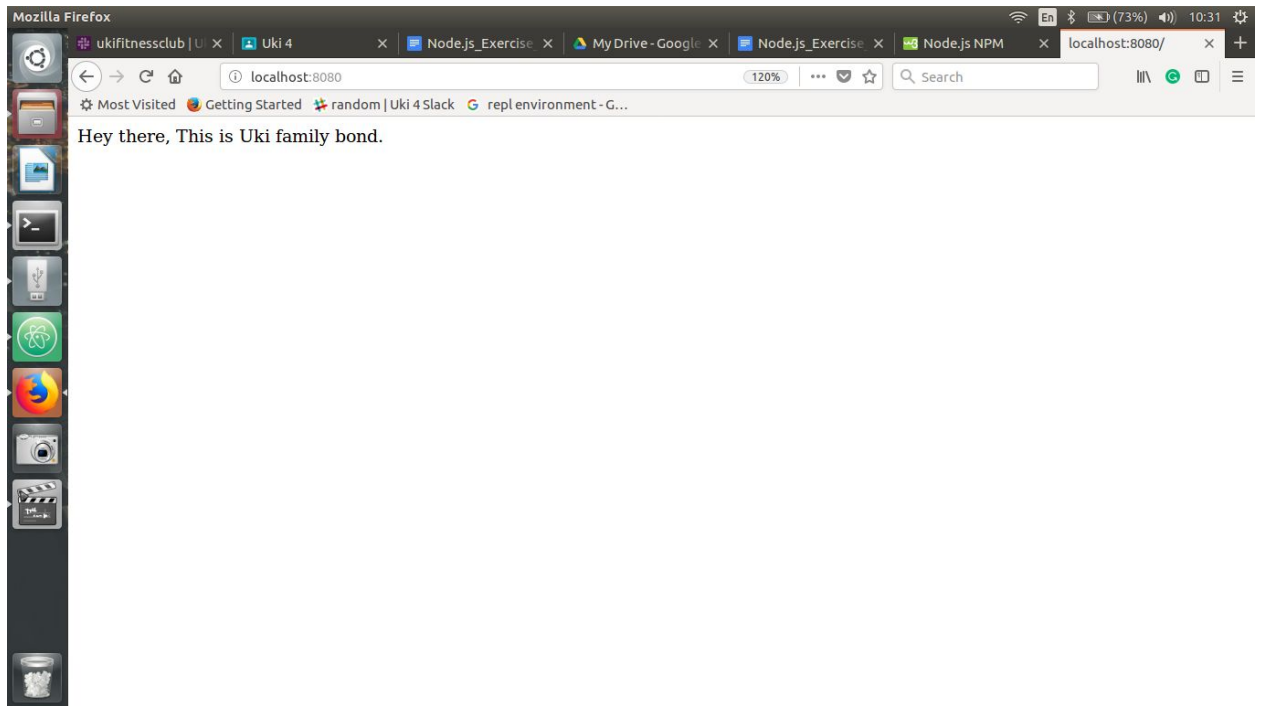
3.2 Serve that file to the client (Read File) over your server



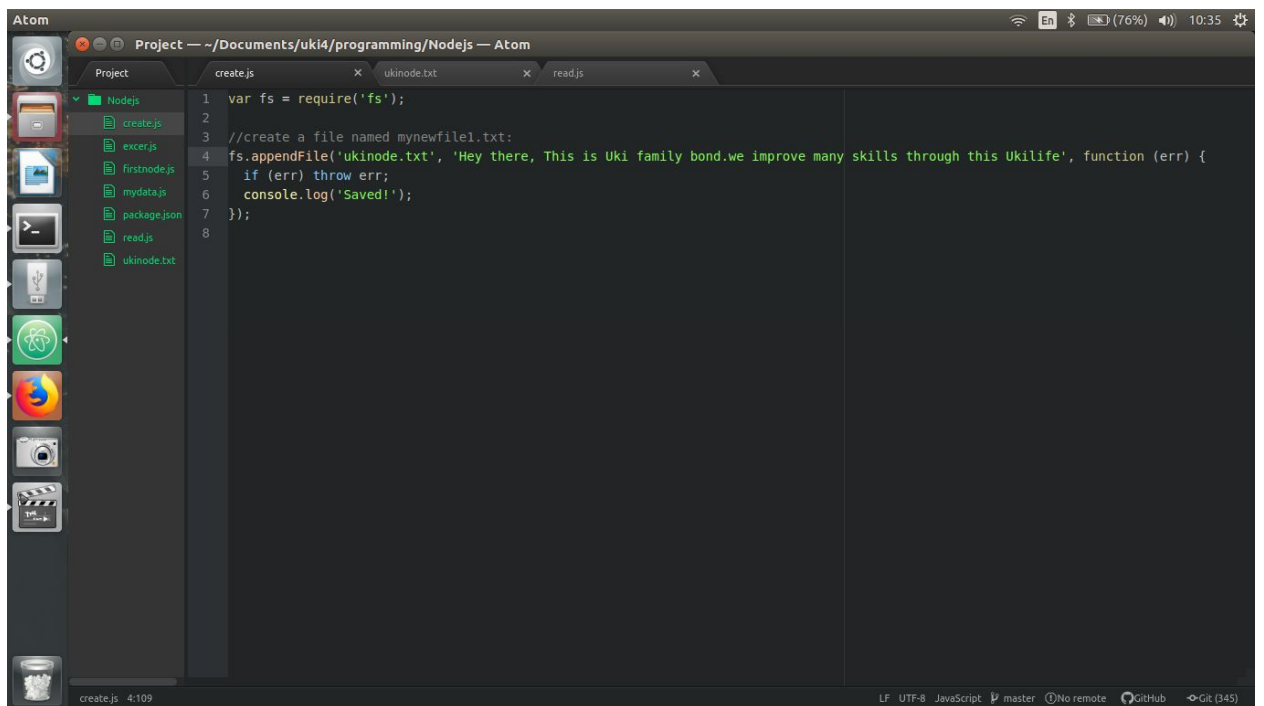
```
1 var http = require('http');
2 var fs = require('fs');
3 http.createServer(function (req, res) {
4 //Open a file on the server and return its content:
5 fs.readFile('ukinode.txt', function(err, data) {
6 res.writeHead(200, {'Content-Type': 'text/html'});
7 res.write(data);
8 return res.end();
9 });
10 }).listen(8080);
11
```

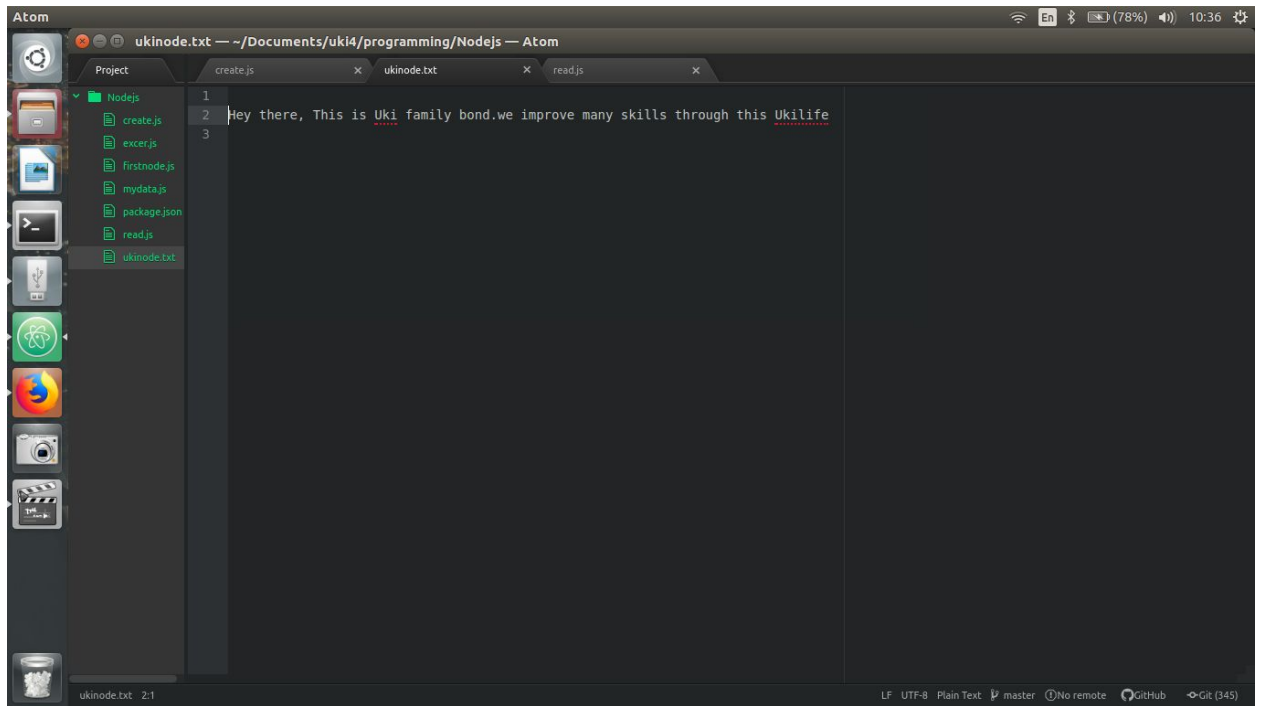


```
ukistu01@ukipc01: ~/Documents/uki4/programming/Nodejs
ukistu01@ukipc01:~/Documents/uki4/programming/Nodejs$ nodemon read.js
[nodemon] 1.17.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting 'node read.js'
```

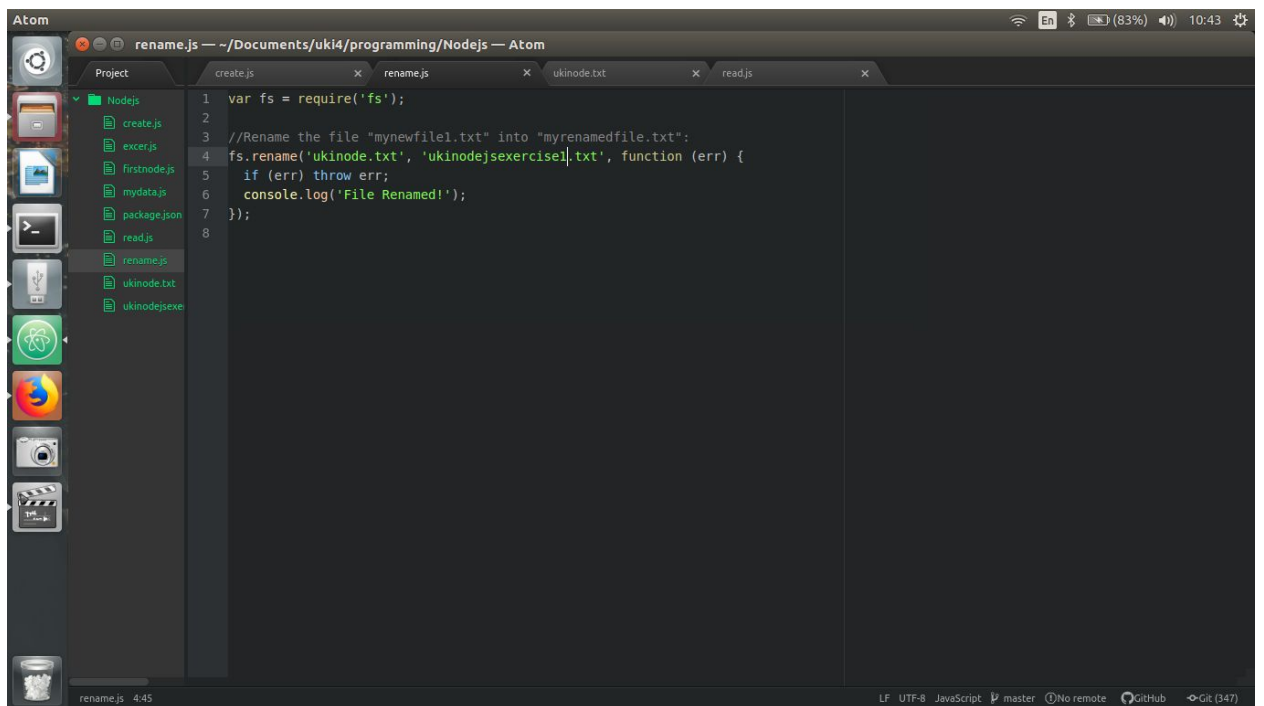


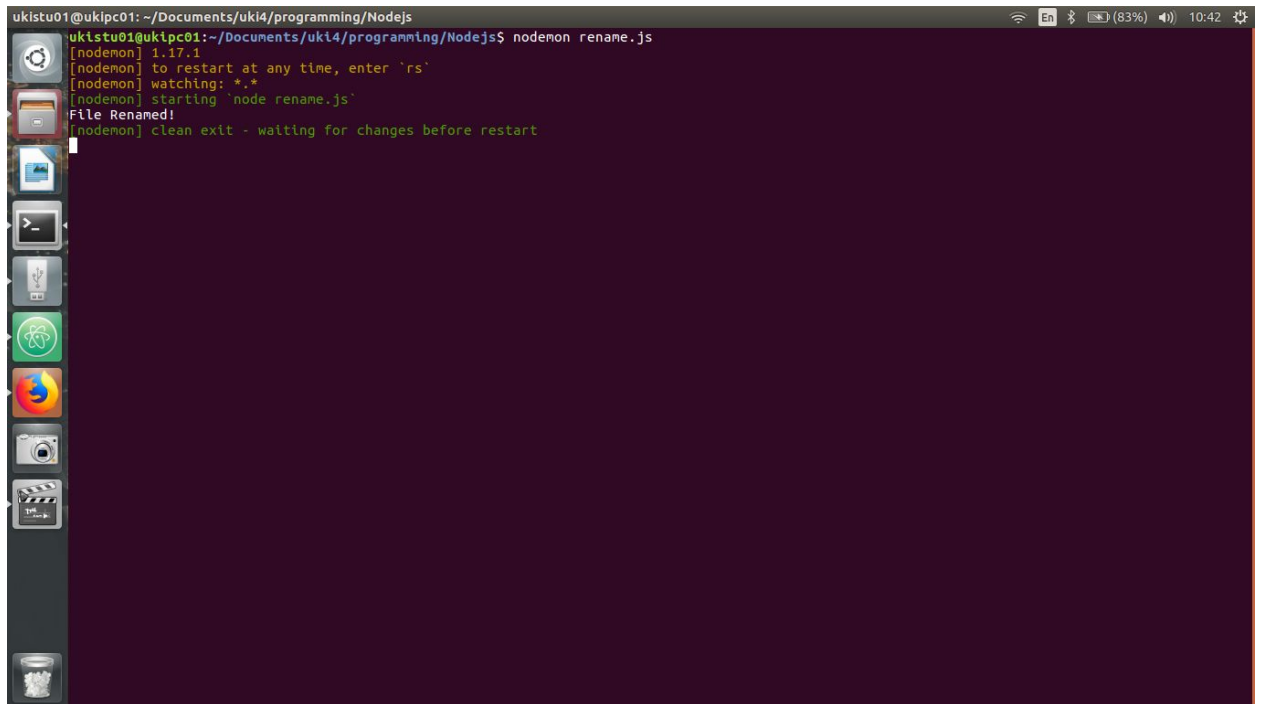
3.3 Append another paragraph about Uki and now serve the new file





3.4 Rename the file as ukinodejsexercise1.txt





A terminal window on a Linux system. The prompt is `ukistu01@ukipc01: ~/Documents/uki4/programming/Nodejs`. The user has run `nodemon rename.js`. The output shows nodemon version 1.17.1, instructions to restart with 'rs', the file being watched (*.*), and the file being renamed. The message "File Renamed!" is displayed in green. The terminal ends with "nodemon] clean exit - waiting for changes before restart".

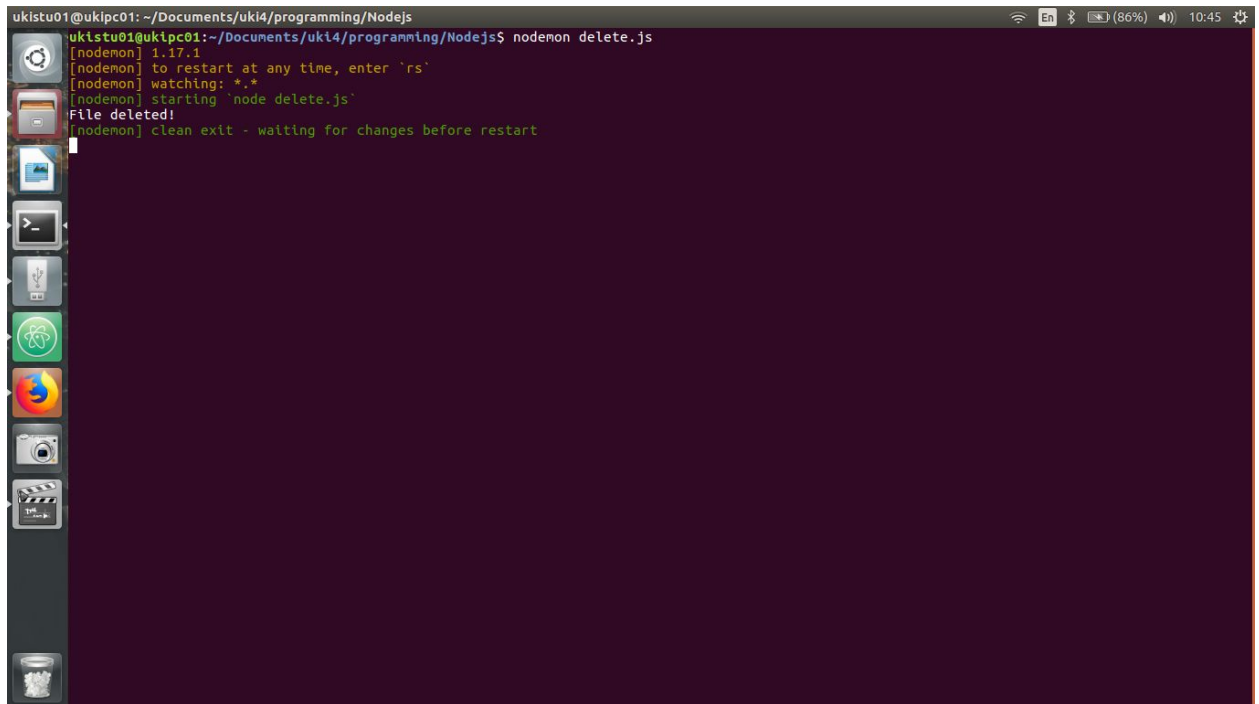
```
ukistu01@ukipc01: ~/Documents/uki4/programming/Nodejs$ nodemon rename.js
[nodemon] 1.17.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node rename.js'
File Renamed!
[nodemon] clean exit - waiting for changes before restart
```

3.5 Delete the file you created



The Atom text editor interface. The active file is `delete.js` in the project `~/Documents/uki4/programming/Nodejs`. The file explorer on the left shows a folder `Nodejs` containing `node_module`, `create.js`, `delete.js`, `excer.js`, `firstnode.js`, `mjh.js`, and `mydata.js`. The code in `delete.js` is as follows:

```
1 var fs = require('fs');
2
3 //Delete the file mynewfile2.txt:
4 fs.unlink('ukinodejsexercisel.txt', function (err) {
5   if (err) throw err;
6   console.log('File deleted!');
7 });
8
```

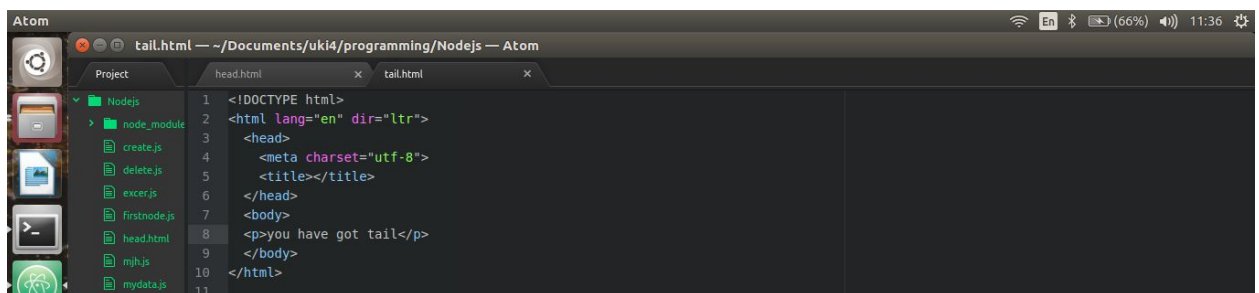
```
ukistu01@uklpc01: ~/Documents/uki4/programming/Nodejs
ukistu01@uklpc01:~/Documents/uki4/programming/Nodejs$ nodemon delete.js
[nodemon] 1.17.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node delete.js'
File deleted!
[nodemon] clean exit - waiting for 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:

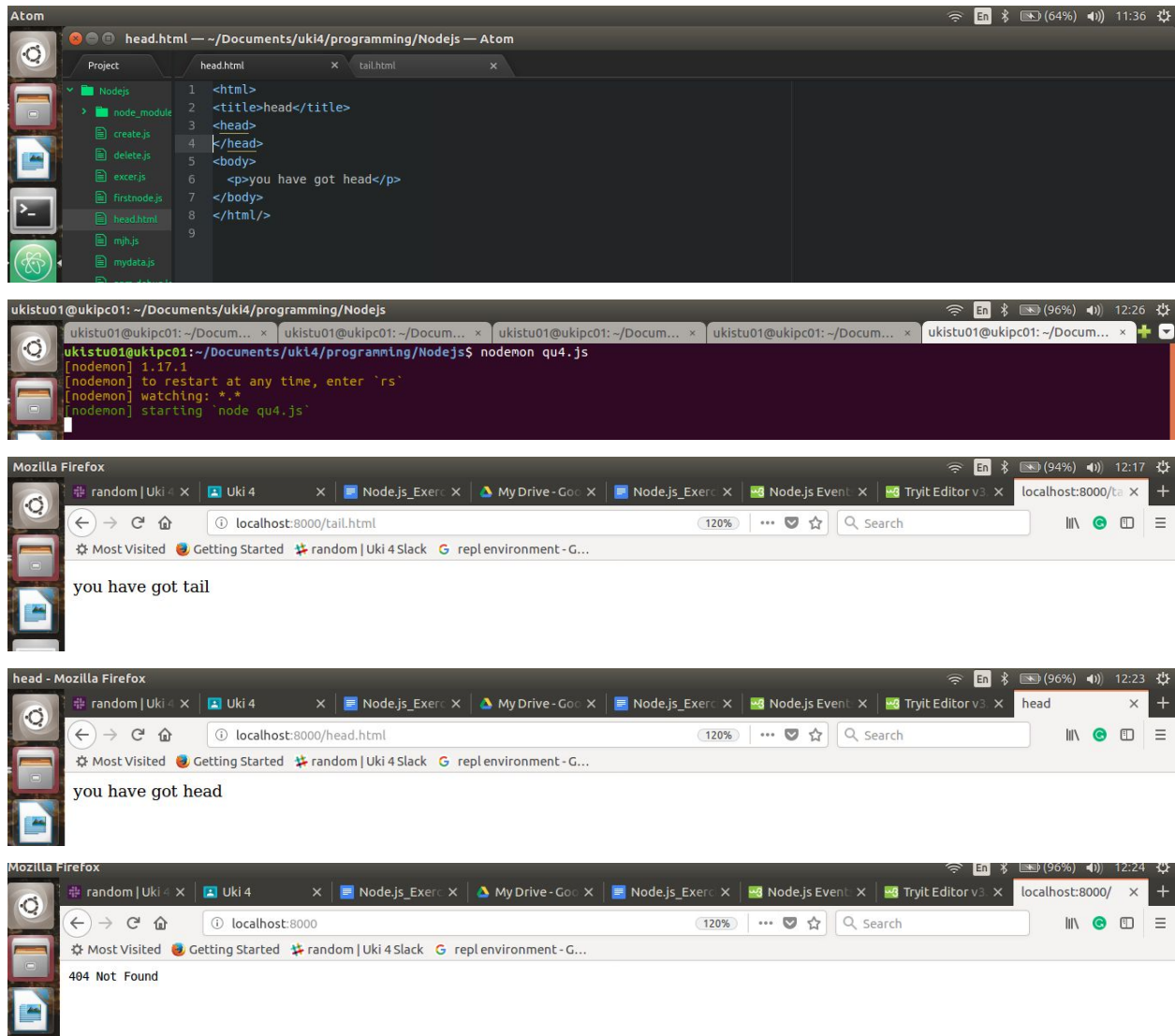
<http://localhost:8080/head.html> - > You have got head

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



```
Atom
tail.html — ~/Documents/uki4/programming/Nodejs — Atom
Project
  Nodejs
    node_module
      create.js
      delete.js
      excer.js
      firstnode.js
      head.html
      mjh.js
      mydata.js
  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
```

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.

```
ukistu01@ukipc01: ~/Documents/uki4/programming/Nodejs
ukistu01@ukipc01:~/Documents/uki4/programming/Nodejs$ npm install upper-case
mithu@1.0.0 /home/ukistu01/Documents/uki4/programming/Nodejs
└─ upper-case@1.1.3

npm WARN mithu@1.0.0 No repository field.
ukistu01@ukipc01:~/Documents/uki4/programming/Nodejs$ npm install formidable -g
/home/ukistu01/.npm/versions/node/v6.16.0/lib
└─ formidable@1.2.1

ukistu01@ukipc01:~/Documents/uki4/programming/Nodejs$ npm install nodemailer -g
npm ERR! Linux 4.8.0-49-generic
npm ERR! argv "/home/ukistu01/.npm/versions/node/v6.16.0/bin/node" "/home/ukistu01/.npm/versions/node/v6.16.0/bin/npm" "install" "nodemailer" "-g"
npm ERR! node v6.16.0
npm ERR! npm v3.10.10
npm ERR! code ENETUNREACH
npm ERR! errno ENETUNREACH
npm ERR! syscall connect
npm ERR! connect ENETUNREACH 2606:4700::6810:1a23:80
npm ERR! If you need help, you may report this error at:
npm ERR! <https://github.com/npm/npm/issues>

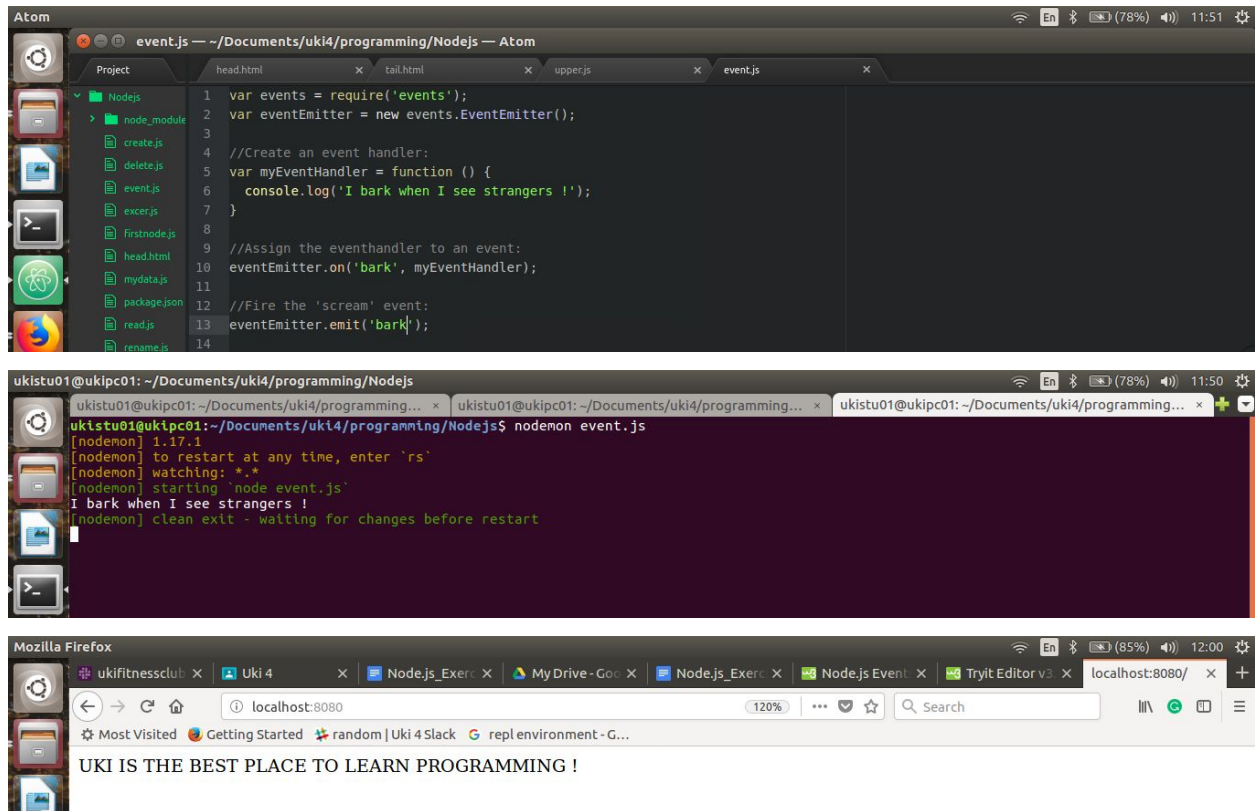
npm ERR! Please include the following file with any support request:
npm ERR! /home/ukistu01/Documents/uki4/programming/Nodejs/npm-debug.log
ukistu01@ukipc01:~/Documents/uki4/programming/Nodejs$
ukistu01@ukipc01:~/Documents/uki4/programming/Nodejs$
```

```
Atom
upper.js -- ~/Documents/uki4/programming/Nodejs -- Atom
Project: head.html x tail.html x upper.js x
Nodejs
├─ node_module
├─ create.js
├─ delete.js
├─ excers.js
├─ firstnode.js
├─ head.html
├─ mydata.js
├─ npm-debug.log
└─ package.json
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   /*Use our upper-case module to upper case a string:*/
6   res.write(uc("Uki is the best place to learn programming !"));
7   res.end();
8 }).listen(8080);
9
```

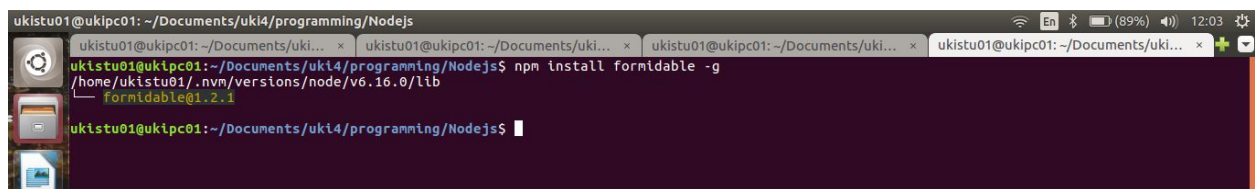
```
ukistu01@ukipc01: ~/Documents/uki4/programming/Nodejs
ukistu01@ukipc01:~/Documents/uki4/programming/Nodejs$ nodemon upper.js
[nodemon] 1.17.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node upper.js'
```

```
Mozilla Firefox
ukifitnessclub | Uki 4 | Node.js_Exercise | My Drive - Google | Node.js_Exercise | Node.js Events | localhost:8080/
localhost:8080
120%
UKI IS THE BEST PLACE TO LEARN PROGRAMMING !
```

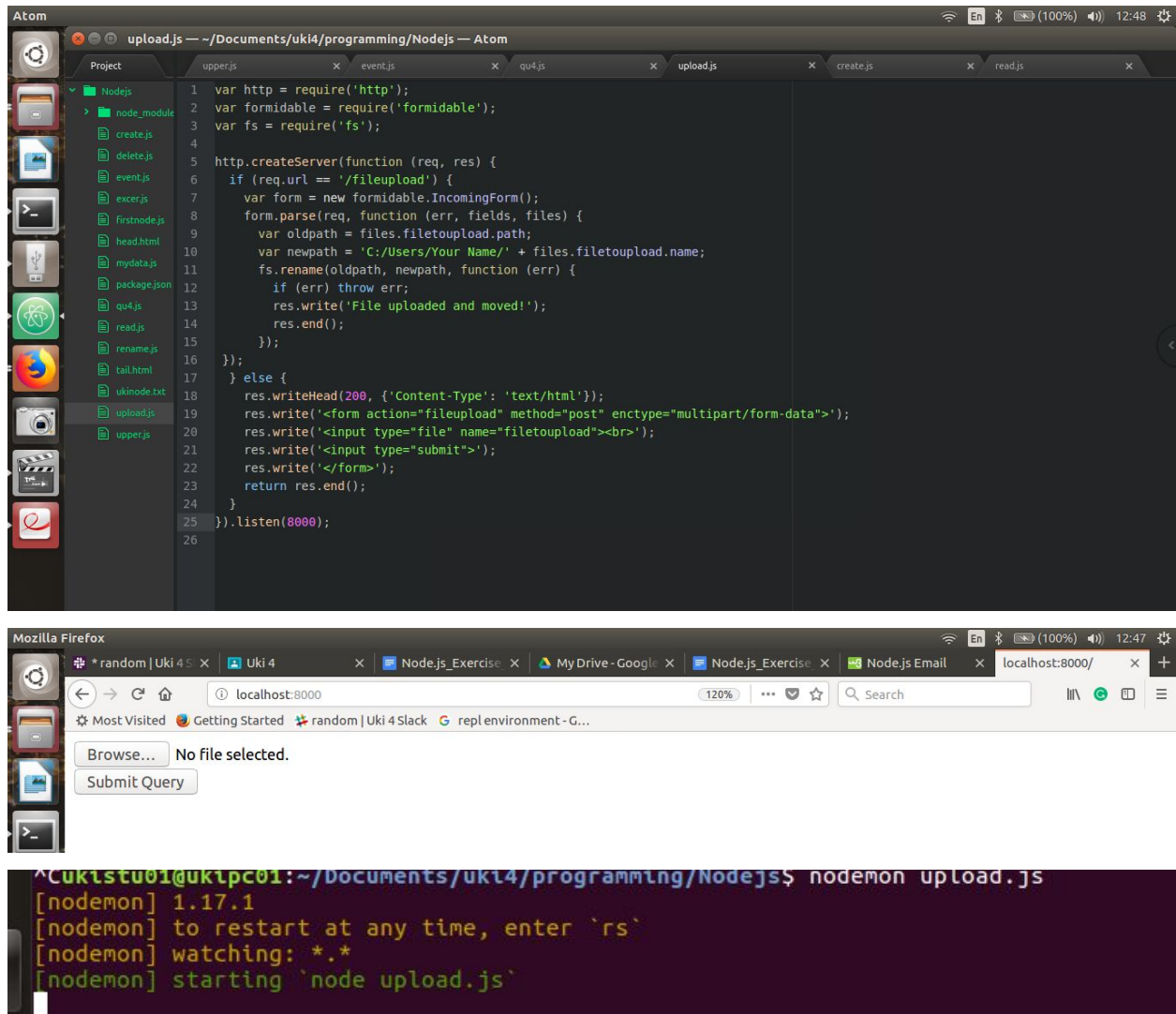
6. Create an event handler function that will say "I bark when I see strangers !" when a "bark" event is fired.



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 !”

